

Rules of Thumb versus Dynamic Programming

By MARTIN LETTAU AND HARALD UHLIG*

This paper studies decision-making with rules of thumb in the context of dynamic decision problems and compares it to dynamic programming. A rule is a fixed mapping from a subset of states into actions. Rules are compared by averaging over past experiences. This can lead to favoring rules which are only applicable in good states. Correcting this good state bias requires solving the dynamic program. We provide a general framework and characterize the asymptotic properties. We apply it to provide a candidate explanation for the sensitivity of consumption to transitory income. (JEL E00, C63, C61, E21)

Agents faced with an intertemporal optimization problem are usually assumed to act as if deriving their decision from solving some dynamic programming problem. A great deal of research effort has been devoted to reconcile this paradigm with observations, resulting in many successful explanations but also in some deep puzzles, where such a reconciliation seems hard to come by (see John Rust, 1992). Thus, an increasing number of researchers have started to investigate the scope

for other alternative paradigms, often subsumed under the heading of bounded rationality, hoping to find one that may explain observed facts even better or easier, or that may result in a fruitful renewal of the current benchmark. This line of work is necessarily exploratory in nature: promising-looking alternatives that have been proposed need to be developed and investigated carefully. Moreover, their relationship to the benchmark paradigm of rationality needs to be understood before these alternatives should even be admitted for a grand horse race. Surveying these efforts here is beyond the scope of the paper and has already been done (see, e.g., Thomas J. Sargent, 1993; John Conlisk, 1996).

This paper adds to this line of research by investigating a model of learning in intertemporal decision problems. Learning takes place by evaluating the quality of competing rules of thumb via past experiences from using them, using a simple updating algorithm (Section IV). Using stochastic approximation theory, we characterize all asymptotically possible outcomes with strict rankings of the rules. We provide an easy-to-use algorithm to compute these outcomes¹ (Section V). We relate the performance measure of the rules to the value function in dynamic programming and show that dynamic programming is a special case of our analysis (Section III and Section VI, subsection A). We

* Lettau: Federal Reserve Bank of New York, Research Department—Capital Markets Function, 33 Liberty Street, New York, NY 10045; Uhlig: CentER, Tilburg University, Postbus 90153, 5000 LE Tilburg, The Netherlands. Both authors are affiliated with the Center for Economic Policy Research (CEPR). Lettau thanks Tilburg University and Humboldt University, where he was a previous faculty member while performing parts of this research. Both authors thank Princeton University, where this research got started. We thank in particular Annamaria Lusardi. We have also received many helpful comments, in particular from Nabil Al Najjar, Orazio Attanasio, Laurence Baker, Patrick Bolton, Martin Browning, John Conlisk, Kenneth Corts, Edmund Chattoe, Angus Deaton, Seppo Honkapohja, Mark Huggett, Michihiro Kandori, Georg Kirchsteiger, Blake LeBaron, Steven Pinker, Aldo Rustichini, Tom Sargent, Johan Stennek, Tim Van Zandt, Peter Wakker, and seminar participants at CORE, Princeton University, a CEPR conference in Gerzensee (Switzerland), and a theory workshop in Venice. We thank two referees for providing excellent and thoughtful comments. The views expressed are those of the authors and do not necessarily reflect those of the Federal Reserve Bank of New York or the Federal Reserve System. Any errors and omissions are the responsibility of the authors. Harald Uhlig dedicates this paper to his son, Jan Peter, who was born when the first draft was written.

¹ Most of the proofs are in a technical Appendix which is available from the authors upon request. It can also be downloaded at www.wiwi.hu-berlin.de/~lettau.

exhaustively analyze an application to a theoretical case in order to understand precisely the differences to dynamic programming, which generally arise (Section VI, subsection B). We find that the learning scheme investigated here often gives rise to a “good state bias,” favoring rules which make possibly bad decisions, but are applicable only in good states (as measured by the value function). The intuition is as follows: even though the dynamic nature of the decision problem is taken into account when evaluating past performance, the learning algorithm fails to distinguish between good luck and smart behavior. This way, a suboptimal rule may be “falsely” learned to be superior to some other rule, which always implements the “correct” dynamic programming solution.

Before plunging into the theoretical details, the paradigm is described in words in Section I, and it is shown how the feature of the good state bias may help in understanding the observation of the high sensitivity of consumption to transitory income (see Marjorie A. Flavin, 1981; Stephen P. Zeldes, 1989). The excess sensitivity observation has proven to be a thorny one to explain within the confines of dynamic programming. The most successful explanation to date by Glenn R. Hubbard et al. (1994, 1995) requires an elaborate life-cycle structure with a rich set of features. Thus, researchers have already moved towards proposing “rules-of-thumb” consumers as a potential explanation (see John Y. Campbell and N. Gregory Mankiw, 1990; Annamaria Lusardi, 1996). We will show how agents can learn these suboptimal rules within our paradigm. Intuitively, a rule which prescribes to splurge in times of high income can win against a rule implementing the dynamic programming solution because of the good state bias mentioned above. A first toy-example illustrates this. An extended, calibrated example is then presented, aimed at matching the observations.

Rules-of-thumb behavior has been postulated and examined elsewhere before. Contributions include Richard H. Day et al. (1974), Amos Tversky and Daniel Kahneman (1974), Kahneman and Tversky (1982), J. Bradford DeLong and Lawrence H. Summers (1986), Campbell and Mankiw, as mentioned above (1990), Beth Fisher Ingram (1990), Anthony

A. Smith, Jr. (1991), Kenneth G. Binmore and Larry Samuelson (1992), Glenn Ellison and Drew Fudenberg (1993), Robert W. Rosenthal (1993a, b), Reza Moghadam and Simon Wren-Lewis (1994), Jonathan B. Berk and Eric Hughson (1996), Per Krusell and Smith (1996), and Lusardi (1996). The motivation of these authors is partly the difficulty of explaining observed facts—partly, because rules of thumb are an interesting paradigm in themselves, and partly, because they can be used as a computational tool. However, in all of these papers, the rules are either assumed ad hoc without any learning about their qualities, or they are investigated only in static decision problems, which is repeated many times. Several of these studies rely on simulations. By contrast, we provide a theory of learning about rules of thumb in an explicitly dynamic decision problem, and provide a theoretical, rather than a simulation-based, analysis. We believe these to be important advances. First, many interesting decision problems, such as the consumption-savings problem described above, are intrinsically dynamic problems rather than repeated static problems. Second, many believe bad rules of thumb to be driven out by learning experiences: so, postulating bad rules a priori is “too easy.” In contrast to this popular belief, our results show that learning can lead to selecting a “bad” rule rather than a rule implementing the “correct” dynamic programming solution. Third, while numerical simulations often provide a useful first step, theoretical results are clearly desirable.

A further litmus test, which we think should be applied to models of boundedly rational behavior, is a “psychofoundation,” i.e., a foundation in psychology as well as in experimental evidence. Our framework here is no exception. John R. Anderson (1995) provides a good introduction into cognitive psychology. Steven Pinker (1997 pp. 42, 419) writes that “behavior is the outcome of an internal struggle among many mental modules” and “mental life often feels like a parliament within. Thoughts and feelings vie for control as if each were an agent with strategies for taking over the whole person.” The theory provided in this paper can be seen as a simple model of that description. Alvin E. Roth and

Ido Erev (1995) cite two well-known facts from the psychological learning literature as motivation for their adaptive learning model. First, the “Law of Effect” says that choices that have good outcomes in the past are more likely to be repeated in the future (E. Thorndike, 1898). Second, the “Power Law of Practice” states that the slope of the learning curve is decreasing with time (J. M. Blackburn, 1936). Our learning model is consistent with both laws. It is also consistent with “melioration,” which means that people choose a strategy that on average gave the maximal payoff in the past rather than choosing what is optimal now: for experimental evidence, see Erev et al. (1997). The behavioral pattern resulting from our model could also be interpreted as a type of satisficing (Herbert Simon, 1982). The deviation from rationality tends to occur in good states in which the agent is “satisfied” with suboptimal behavior. Gerald M. Edelman (1992) provides a useful reference on how a biologist and psychologist relates the functioning of the human brain to classifier system like structures, indicating that this paradigm may be promising indeed. David Easley and Aldo Rustichini (1996) have recently provided an axiomatic underpinning for our approach. Tilman Börgers (1996) has reviewed the role of learning in economics and provides further arguments in favor of our paradigm. Other psychologically based approaches to boundedly rational choices in dynamic decision problems are discussed in, e.g., Richard H. Thaler (1991), George Loewenstein and Jon Elster (1992), and Kahneman et al. (1997).

There are surprisingly few experimental studies on dynamic decision problems. Stephen Johnson et al. (1987) report results on an experimental study on a life-cycle model: their subjects did not fare well in reaching the theoretically optimal solution. John D. Hey and Valentino Dardanoni (1987) conduct an experiment on a dynamic choice model with uncertainty. Although their underlying model is not directly comparable to ours, the behavioral patterns of the experimental subjects appear to be consistent with those of our model. Specific applications of the learning algorithm in the paper here will probably often give rise to the feature that people tend to act

more suboptimally in good states than in bad states. A direct experimental test of this hypothesis would certainly be desirable.

Our research is inspired by John H. Holland’s (1992) work on classifier system, but we have stripped away all the features which did not seem crucial for the issues addressed in this paper (see Section IV). Furthermore, we did not include a rule-generating or rule-modifying feature such as the genetic algorithm, since that seemed to be too hard to justify and too hard to analyze at this point in the ongoing research. Related work includes W. Brian Arthur (1993), who derives some theoretical results in a simplified, but related, learning framework in which there is no dynamic link between periods. He also compares his results to experimental studies and concludes that his learning specification mimics many features of human behavior in experiments. We were most heavily influenced by Ramon Marimon et al. (1990), who simulate a dynamic learning model in a Nobuhiro Kiyotaki and Randall Wright (1989) model of money, using classifier systems. This paper grew directly out of thinking about how to understand theirs. More generally, our learning algorithm is related to learning via reinforcement and replicator dynamics [see, e.g., Börgers and Rajiv Sarin (1995, 1996)].

I. The Sensitivity of Consumption to Transitory Income

While we postpone formal definitions until Sections II and IV, we shall briefly describe our paradigm informally here. **An agent has to solve an infinite-horizon dynamic decision problem.** Given a state at some date, he has to choose some (feasible) action. Given the state and the chosen action, the agent will experience instantaneous utility and the state for the next period is drawn. Traditionally, such problems are solved using the tools of dynamic programming. For our paradigm, agents will use rules instead.

Rules map states into (feasible) actions, but their domain is typically limited to only a subset of all possible states, i.e., they are typically not universally applicable. We impose no further restrictions like limited complexity. An agent is **endowed with several competing**

rules. The agent does not invent new rules. Asking why an agent is endowed with these particular rules or whether, **how, and why they might be changed** in the course of time raises deep and interesting additional questions that **we do not attempt to answer here**. It is intriguing to speculate about possible resolutions such as instincts (see e.g., Pinker, 1994), learning from your peers, education, meta-rules for changing rules, or the neuronics limits of the brain. We simply take these given rules, as well as the fact that the agent stubbornly sticks to choosing between them throughout his infinite life as primitives of the environment.

Given a particular state at some date, the agent chooses one of the applicable rules for that state based on past experience. To model this learning process, we postulate a simple algorithm. **The agent associates a real number, called the strength, with each rule. The strength of a rule is essentially a (weighted) average of past "rewards" from using that rule.** If a rule has been used at some past date, it maps the state of that date into some action and triggers instantaneous utility as well as a new state for the next date. Its reward for that date will be given by the sum of the instantaneous utility as well as the discounted strength of the rule called upon for the new state. That way, the reward captures not only the instantaneous gratification but also incorporates the future consequences from the current action. **Higher strengths indicate better past experiences.** We assume that the agent always picks the strongest among all the applicable rules.

If the agent is endowed with a single rule which happens to implement the dynamic programming solution, one obviously gets rational behavior according to this solution as a special case of our theory. Things become more interesting when suboptimal rules are allowed into the competition. As we shall see, an agent can learn to use a rule encoding suboptimal behavior even when that rule is competing against another rule which implements the dynamic programming solution. **This can happen if the suboptimal rule is only applicable in "good" states, in which it is easy to generate high rewards:** the strength of that rule will be correspondingly biased upward, pos-

sibly exceeding the strength of a universally applicable dynamic programming solution rule. We call this the *good state bias*.

A specific example shall help to illuminate these abstract concepts. We choose an example, which also provides an interesting application of our paradigm all in itself. We will show how our paradigm can help in understanding the observation of high sensitivity of consumption to transitory income, which has proven hard to explain with the tools of rational behavior.

In his seminal work, Robert E. Hall (1978) characterized the first-order condition of a rational expectations permanent income hypothesis consumer in form of a dynamic Euler equation. Despite the theoretical appeal of the theory, the model has been rejected using aggregate and disaggregate data. In particular, Flavin (1981) demonstrated that aggregate consumption responds too much to current income to be consistent with the theory put forward by Hall (1978). For micro data, Hall and Frederic S. Mishkin (1982) and Zeldes (1989) are the classical references. Angus S. Deaton (1992) and Martin Browning and Lusardi (1996) provide excellent surveys of the literature. Numerical simulations of elaborate life-cycle models have been obtained by Hubbard et al. (1994, 1995), which are able to match many features of the data. Given the detailed complexity of their framework, it still makes sense to search for simpler explanations by possibly giving up the notion of unbounded rationality. Campbell and Mankiw (1990) suggested a model with two types of agents: the first type is a rational Hall-consumer, whereas the second type just consumes the entire current income and does not save. In other words, the second type always uses a rule of thumb which says "consume current income." Obviously, the rule-of-thumb consumer is overly sensitive to current income. Hence, the presence of such consumers would explain the empirical facts on excess sensitivity. Campbell and Mankiw (1990) estimate the share of each type in the population using aggregate data. The percentage of their rule-of-thumb consumers appears to be around 50 percent. For micro data and using a new panel data set, Lusardi (1996) estimates the share of Campbell-Mankiw rule-of-thumb consumers

to be between 20 percent and 50 percent. These studies do not attempt to answer the question of why these agents may fail to learn rational behavior. In this section we demonstrate how such a suboptimal rule can indeed be learned.

In fact, more is going on in our framework. We will endow the agent with two rules: a first rule, which implements the dynamic programming solution, and a second rule, which implements the “spend everything” rule, but is applicable only when income is high. Because of the good state bias, the agent can learn to favor the suboptimal rule despite the fact that he takes future consequences of his current action into account when learning. Thus, rather than having 50 percent of the agents always acting irrationally and spending everything they have,² we have each agent acting irrationally and spending everything he has 50 percent of the time.

We first provide a simple, stylized example in subsection A to provide a qualitative explanation for the excess sensitivity observation and to introduce a bit of the analytics and intuition behind our paradigm before formulating everything precisely in later sections. We next provide a more finely tuned quantitative example calibrated to micro data in subsection B which delivers results matching empirical estimates and which we propose as a serious attempt at an explanation.

A. A Stylized Example

Consider the following situation. There is an infinitely lived agent who derives utility $u(c_t)$ from consuming $c_t \geq 0$ in period t and who discounts future utility at the rate $0 < \beta < 1$. The agent receives random income y_t each period. Suppose there are two income levels, $\bar{y} > y > 0$ and that income follows a Markov process with transition probabilities $p_{yy} = \text{Prob}(y_t = y | y_{t-1} = y)$, etc. The agent enters period t with some wealth w_t . Next period's wealth is given by $w_{t+1} = w_t + y_t - c_t$. A borrowing constraint is imposed so that

$0 \leq c_t \leq w_t + y_t$. Furthermore, we assume that the agent is born with zero wealth: $w_0 = 0$.

The state of the decision problem at date t is given by $s_t = (w_t, y_t)$. Within the utility-maximizing framework, the decision problem is most easily formulated as a dynamic programming problem, given by

$$(1) \quad v(w, y) = \max_{c \in [0, w + y]} \left(u(c) + \beta \sum_{y' \in \{y, \bar{y}\}} p_{yy'} [v(w + y - c, y')] \right).$$

The instantaneous utility function u is assumed to be continuous, concave, strictly increasing, and bounded. Thus, standard arguments as in Nancy L. Stokey et al. (1989) show that this problem is solved by a value function v , which is itself continuous, concave, strictly increasing and bounded in wealth $w \geq 0$, and gives rise to a unique decision function $c^*(w, y)$. Note that $c^*(0, \bar{y}) = \bar{y}$. Due to consumption smoothing motives and precautionary savings, optimal behavior prescribes positive savings when the agent is rich: $c^*(w, \bar{y}) < w + \bar{y}$, $w \geq 0$.

Now consider, instead, an agent who behaves boundedly rationally in the following way. The agent is endowed with two rules of thumb in his mind. The first rule r_1 is applicable in all states and coincides with the optimal decision function $r_1(w, y) \equiv c^*(w, y)$. The second rule r_2 is applicable only in the “good” state when the income is high, and prescribes to consume everything,

$$(2) \quad r_2(w, \bar{y}) = w + \bar{y}.$$

With these two competing rules, the agent is torn between the urge to splurge on the one hand and the desire to invest his income and wealth wisely on the other.

The agent's problem is thus to find out which one of the two rules he should use in a given state. Note that he has to make that choice whenever income is high ($y = \bar{y}$), whereas he always follows rule r_1 when income is low ($y = y$) by assumption. According to our paradigm, the agent is assumed to choose among all the applicable rules that rule with which he has had the

² This is also a possible solution within our framework, albeit a trivial one: the agent can always learn to always use a single, universally applicable rule.

best past average experience. More specifically, suppose that the agent has always used the second, suboptimal rule whenever income was high, and the first, optimal rule whenever income was low in the past, and now stops at some date to rethink that behavior.³ In order to do so, he computes the past average payoffs or *strengths* z_1 and z_2 from using these two rules, r_1 and r_2 . Suppose that income was high in some period t , triggering rule r_2 , but low in period $t + 1$, triggering rule r_1 . The agent is assumed to attribute

$$(3) \quad u(c_t) + \beta z_1$$

as the payoff for using rule 2 at date t . Thus, the agent does not simply take $u(c_t)$, the instantaneous gratification, to be the payoff of period t , but to include z_1 as a measure of the quality of the next state at date $t + 1$ as well, since that next state is induced by the consumption decision at date t (aside from the random fluctuation in income). If income had been high in period $t + 1$, triggering rule r_2 , the agent would have included z_2 rather than z_1 , of course. The agent averages over all periods t in the past where the first rule was used to obtain the strength z_1 of the first rule, and proceeds likewise for the second rule.

We thus have assumed the agent to recognize the dynamic linkage between periods, but to be ignorant about differences in states. When the agent receives the high income at date t , he does not know that he is in a “good” state. He only learns about the state indirectly via his actions. Since he splurges, he experiences the instantaneous utility from consuming everything as well the consequences of a depleted bank account $w_{t+1} = 0$ via the discounted strength z_1 (or z_2) of the rule chosen at date $t + 1$. Including z_1 in the payoff is sensible and somewhat similar to including $E[v(s')]$ in the dynamic programming approach, since that strength reflects the average experience with states in which the first rule was chosen. But this is at the same time a somewhat crude measure of the induced next state, since the first rule may also potentially

be chosen in other states as well, all being summarized in one index of strength, z_1 . Thus, there is also a contrast to the dynamic programming approach, where a value $v(s')$ is calculated for each state. The accounting scheme (3) is a crucial element of our learning scheme. Specifying a different scheme will likely yield different results.

Note that the agent has always been spending his total current income and has never saved anything. If many periods are used for calculating the strengths, the agents will approximately solve the following two linear equations in z_1 and z_2 :

$$(4) \quad z_1 = u(\underline{y}) + \beta(p_{\underline{y}\underline{y}}z_1 + (1 - p_{\underline{y}\underline{y}})z_2)$$

$$(5) \quad z_2 = u(\bar{y}) + \beta(p_{\bar{y}\bar{y}}z_1 + (1 - p_{\bar{y}\bar{y}})z_2).$$

Here, we used the fact that the consumption has always been all of income, $c_t \equiv y_t$ and that of all dates t with low income $y_t = \underline{y}$, approximately the fraction $p_{\underline{y}\underline{y}}$ of them have been in turn followed by low income $y_{t+1} = \underline{y}$ in the next period, thus triggering the use of the first rule in $t + 1$, etc.

The agent, who is considering whether or not to change his behavior, will not do so if the strength of the second rule exceeds the strength of the first rule, $z_2 > z_1$. With an increasing utility function $u(\cdot)$ and $0 < p_{\underline{y}\underline{y}}, p_{\bar{y}\bar{y}}, \beta < 1$, we have indeed

$$(6) \quad z_2 - z_1 = \frac{u(\bar{y}) - u(\underline{y})}{1 - \beta(p_{\underline{y}\underline{y}} - p_{\bar{y}\bar{y}})} > 0.$$

The agent will see no reason to change and will continue his suboptimal behavior. The intuition behind this result is this: rule r_2 may “win” against rule r_1 since it only applies in “good times” and thus “feels better” on average than rule r_1 . This “good state bias” gives rule r_2 an intrinsic advantage when competing against the optimal rule r_1 , which is applicable at all times.⁴ There are two ways to think about this result. One can either consider

³ This decision and learning algorithm will be made more precise in Section IV.

⁴ Conlisk pointed out to us that this is related to the excluded variable bias in regression analysis. One can think of the strength of the rule as picking up effects of the “excluded” state variable.

the accounting scheme (3) faulty and try to correct it, an issue to which we return in Section VI, subsection C. Alternatively, one can buy into equation (3) and consider the implied deviations from rational behavior as an intriguing possibility for explaining puzzling behavior. We favor the latter interpretation.

Obviously, the particular rules chosen and the states in which they are applicable matter. One can also easily construct examples, in which it is the optimal rule, which wins. Our framework does not contradict behavior according to the rational dynamic programming solution: rather, it includes it as a special case. In order for the dynamic programming solution to emerge, the given rules not only need to permit a ranking, which delivers the rational decision rule (if the agent always picks the rule with the highest rank), but that ranking must also be learnable. The precise conditions for what is learnable are in Section V.

B. Calibrated Calculations

We now wish to employ our techniques to provide a calibrated version of our example and to investigate its quantitative fit to observations. Except for keeping within an infinite-horizon setting rather than a life-cycle model, we largely follow the choices made by Hubbard et al. (1994, 1995).

These authors have decomposed observed individual earnings processes

$$(7) \quad \log y_{it} = Z_{it}\beta + u_{it} + \nu_{it}$$

into a systematic part $Z_{it}\beta$, containing variables for in particular age, an autocorrelated part u_{it} , and an idiosyncratic part ν_{it} . For the purpose of their simulations, they regarded ν_{it} as measurement error and ignored it. For now, we will follow their approach, keeping in mind that this depresses the variance of predictable changes in income. As for u_{it} , Hubbard et al. (1994, 1995) fitted an AR(1) and found its autocorrelation to equal $\rho = 0.955$ and its innovation variance to equal 0.033 for household heads without high-school education [see their Table A.4 (1994) or Table 2 (1995)]. For our calibrated example, we will identify log income with u_{it} plus a constant and thus additionally throw away life-cycle informa-

tion: in the infinite-horizon context employed here, age has little meaning. At least two levels of income are needed to capture this random process: we have chosen $n = 5$ different levels instead to allow for a more detailed investigation. These levels were normalized to have a mean of unity and chosen to be equally spaced logarithmically. The Markov transition probabilities $p_{yy'}$ were chosen to equal $\rho + (1 - \rho)/n$ for $y = y'$ and $(1 - \rho)/n$ for all other y' , thus ensuring an autocorrelation of ρ . The spacing of the logarithmic income grid was then chosen to yield a variance of $0.033/(1 - \rho^2)$ as in the data.

The yearly discount factor β was set to $\beta = 0.96$, and the asset return was set at $\bar{R} = 1.02$. For the utility function, we have used the usual CRRA specification, $u(c) = c^{1-\gamma}/(1-\gamma)$. As in Hubbard et al. (1994, 1995), we have chosen $\gamma = 3$. We used a grid of 40 actions, corresponding to the fraction spent of total cash on hand (i.e., wealth plus income). We chose the grid to be somewhat denser close to the extreme values of 0 and 1. Note that the dynamic programming solution was also restricted to this grid. One minus that fraction spent is the fraction of total cash on hand to be saved. Savings equal wealth next period, when multiplied with the asset return $\bar{R} = 1.02$. Wealth was represented with a logarithmically evenly spaced grid of 80 grid points, starting at 0.0176 and ending at 119. Given a value for wealth next period not on the grid, we used the linear interpolation weight to draw among the two adjacent grid points. Note that wealth is not allowed to become negative, i.e., that we stick with the model of the previous subsection A and impose a borrowing constraint.

We used two rules. The first rule is the solution to the dynamic programming problem, applicable everywhere. The second is a rule, where spending is linearly interpolated between the true dynamic programming solution and spending everything, parameterized by some value λ as the weight on "spending everything." In particular, spending everything ($\lambda = 1$) and following the optimal spending policy ($\lambda = 0$) are the two extremes. The second rule will be applicable when both wealth exceeds some wealth cutoff level as well as when income exceeds some income cutoff level. We experimented with λ as well as the two

cutoff levels. In particular for wealth, we tried no restriction (wealth ≥ 0), all wealth levels except the first grid point (wealth⁵ ≥ 0.02), wealth at least mean income (wealth ≥ 1), and wealth at least three times and ten times mean income (wealth ≥ 3 , wealth ≥ 10).

Results are calculated by applying the theory of Section V below. In reporting our results, we focussed on the following three statistics. The first is a regression of $\Delta \log c_{t+1}$ on $\log y_t$ (as well as a constant). Zeldes (1989) estimates this coefficient to be -0.07 for low-wealth individuals, see e.g., his table 2. This well-known estimate has been interpreted as evidence for excess sensitivity of consumption to predictable changes in income. Instead, one can also focus more directly on the regression coefficient of $\Delta \log c_{t+1}$ on $E_t[\Delta \log y_{t+1}]$ (as well as a constant), which one can interpret as the elasticity of consumption changes with respect to predictable income changes. This coefficient has been reported to be around 0.4 for micro data (see Lusardi, 1996). The two statistics are related. Indeed, if $\log y_t$ independently follows an AR(1) with autocorrelation ρ , then $E_t[\Delta \log y_{t+1}] = (1 - \rho)\log y_t$, and hence, the second statistic will equal the first statistic divided by $-(1 - \rho)$. With the calibrated value of $\rho = 0.955$, the second statistic is thus -22 times as large as the first, resulting in 1.5. This value is undoubtedly too large and Lusardi's estimate of 0.4 is more credible. The most likely resolution of this apparent contradiction is that ν_{it} is not just purely measurement error as Hubbard et al. assumed. We will return to this issue in our last calculation of this subsection. Another resolution is that much of the predictable changes in income are due to deterministic variables such as age, which we chose to ignore here. For now, we stick with their interpretation of ν_{it} as measurement error and report both statistics, keeping in mind that we would rather aim for 0.4 than 1.5 for the sec-

ond statistic, and thus -0.02 rather than -0.07 for the first statistics. The third statistic we report is what fraction of time the rule-based agent spends in states, where he employs the suboptimal rule 2, rather than the dynamic programming solution labelled as rule 1. This statistic gives an indication of the deviation from the rationality benchmark.

It is useful to first examine the dynamic programming solution. The regression coefficient of $\Delta \log c_{t+1}$ on $\log y_t$ is -0.0043 and thus much smaller than even our conservative value of -0.02 . Similarly, a regression of $\Delta \log c_{t+1}$ on $E_t[\Delta \log y_{t+1}]$ yields just 0.095 instead of 0.4. Thus, while the imposed borrowing constraint yields some elasticity of consumption changes to predictable income changes, the elasticity is not large enough quantitatively. This difficulty of the dynamic programming solution to an infinite-horizon savings problem with borrowing constraint to capture the facts is well appreciated in the relevant literature.

Our results for rule-based behavior are contained in the two Tables 1 and 2. We checked in each case whether the second rule can indeed win against the first rule: it turned out to be always the case. In the first Table 1, we undertook mainly variations in λ , which is the weight on "spending everything" in the second rule. Note that $\lambda = 0$ delivers the same behavior as the original dynamic programming solution. We can see that we get close to our target values of -0.02 for the first statistic and 0.4 for the second statistic even for modest fractions of times, in which the second rule is employed. For example, 10 percent of the agents suffice for $\lambda = 0.5$ and around 5 to 6 percent for $\lambda = 1.0$. Thus the effects of modest deviations from rationality can be dramatic. Note also how the fraction of time, in which the agent holds more than ten times mean income as wealth, falls from 30 percent for the dynamic programming solution ($\lambda = 0$) to less than 2 percent, when λ is increased to a unity. Finally, an interesting feature of these tables is the dramatic effect of leaving or removing the lowest wealth level from the domain for the second rule.

As promised, we now return to the apparent contradiction between -0.07 as the target for the first statistic and 0.4 as the target for the second statistic. We shall now deviate from

⁵ We effectively assume that it is impossible to drop below this extremely low level of wealth, even when spending "everything" in the previous period. The results are unlikely to change dramatically, if we set this lowest level equal to zero instead, since income is strictly positive in each period.

TABLE 1—VARIATIONS IN λ (ZERO VARIANCE IN TRANSITORY INCOME COMPONENT)

Regression of $\Delta \log c_{t+1}$ on constant and $\log y_t$					
$\lambda =$	0.0	0.25	0.5	0.75	1.0
wealth ≥ 0	-0.004	-0.027	-0.034	-0.039	-0.046
wealth ≥ 0.02	-0.004	-0.023	-0.027	-0.028	-0.030
wealth ≥ 1	-0.004	-0.022	-0.024	-0.026	-0.028
wealth ≥ 3	-0.004	-0.017	-0.018	-0.020	-0.022
wealth ≥ 10	-0.004	-0.012	-0.013	-0.014	-0.016
Regression of $\Delta \log c_{t+1}$ on constant and $E_t[\Delta \log y_{t+1}]$					
$\lambda =$	0.0	0.25	0.5	0.75	1.0
wealth ≥ 0	0.095	0.590	0.761	0.864	1.031
wealth ≥ 0.02	0.095	0.518	0.602	0.614	0.667
wealth ≥ 1	0.095	0.488	0.543	0.575	0.619
wealth ≥ 3	0.095	0.373	0.411	0.447	0.489
wealth ≥ 10	0.095	0.272	0.296	0.322	0.361
Average fraction of time in percent that agent applies the rule					
$\lambda =$	0.0	0.25	0.5	0.75	1.0
wealth ≥ 0	40.0	40.0	40.0	40.0	40.0
wealth ≥ 0.02	39.7	39.3	39.2	39.1	19.8
wealth ≥ 1	39.3	38.4	26.2	16.2	13.4
wealth ≥ 3	37.6	22.4	10.8	7.3	5.7
wealth ≥ 10	30.8	5.2	2.9	2.1	1.8

Notes: The suboptimal rule 2 is applicable whenever wealth exceeds the level indicated and whenever income does not fall below the median income $y = 0.835$. The transitory income component ν_{it} has zero variance.

Hubbard et al. (1994, 1995) and assume that ν_{it} is not measurement error, but rather reflects actual transitory changes in income. To capture this, we have used a three-state Markov process for u_{it} of equation (7), constructed in the same manner as above and, additionally, an idiosyncratic two-state process for ν_{it} , calibrated to match the empirically observed variance of 0.04 [see Hubbard et al., Table A.4 (1994) or Table 2 (1995)]. Mean income was normalized to unity. Except for a slight shift in our wealth grid, everything is the same as above.

The dynamic programming solution yields -0.0017 as regression coefficient of $\Delta \log c_{t+1}$ on $\log y_t$, which is too small, and -0.0171 as regression coefficient of $\Delta \log c_{t+1}$ on $E_t[\Delta \log y_{t+1}]$, which is even of the wrong sign compared to the data. Results from rule-of-thumb calculations are contained in Table 3, where we now only vary the income cutoff level. Again, the second rule was always learn-

able as dominant. The results are quite a bit richer than those found in Table 2. In particular, we observe different signs for both regression coefficients, sometimes siding with the results from the dynamic programming solution and sometimes siding with the data. For some combinations of cutoff levels, both statistics are the same as the estimates from observed data within a tolerable margin. For example and as a first scenario, if agents apply the second rule, whenever income exceeds 1.4 times mean income, i.e., if agents are in the highest of the three Markov states for u_{it} , the first statistic takes the value of -0.078 , which compares favorably to -0.07 in the data, and the second statistic takes the value of 0.335, which is somewhat below the value 0.4 found in the data. We think that this might be a potentially reasonable scenario. Note that the suboptimal second rule is used a third of the time in this case. One also gets a match to the empirical estimates in a second scenario,

TABLE 2—VARIATIONS IN INCOME CUTOFF, $\lambda = 1$ (ZERO VARIANCE IN TRANSITORY INCOME COMPONENT)

Regression of $\Delta \log c_{t+1}$ on constant and $\log y_t$				
income \geq	0.6	1.0	1.5	2.3
wealth ≥ 0	-0.046	-0.046	-0.046	-0.043
wealth ≥ 0.02	-0.022	-0.026	-0.030	-0.034
wealth ≥ 1	-0.020	-0.024	-0.028	-0.032
wealth ≥ 3	-0.017	-0.020	-0.022	-0.027
wealth ≥ 10	-0.014	-0.015	-0.016	-0.019

Regression of $\Delta \log c_{t+1}$ on constant and $E_t[\Delta \log y_{t+1}]$				
income \geq	0.6	1.0	1.5	2.3
wealth ≥ 0	1.012	1.027	1.031	0.949
wealth ≥ 0.02	0.485	0.579	0.667	0.745
wealth ≥ 1	0.437	0.528	0.619	0.712
wealth ≥ 3	0.374	0.446	0.489	0.603
wealth ≥ 10	0.301	0.340	0.361	0.416

Average fraction of time in percent that agent applies the rule				
income \geq	0.5	0.8	1.3	2.0
wealth ≥ 0	80.0	60.0	40.0	20.0
wealth ≥ 0.02	20.2	20.0	19.8	10.0
wealth ≥ 1	13.6	13.5	13.4	8.6
wealth ≥ 3	5.8	5.7	5.7	4.1
wealth ≥ 10	1.8	1.8	1.8	1.4

Notes: The suboptimal rule 2 is applicable whenever wealth exceeds the level indicated and whenever income exceeds the level indicated. The transitory income component v_{it} has zero variance.

where agents only apply the rule of spending everything, if their wealth exceeds three times average earnings and if their income is at the highest level: the first coefficient takes on the value of -0.068 , while the second coefficient has the value 0.5 . Furthermore, for that scenario, agents employ that rule only 5 percent of the time. Put differently, even though these agents behave as predicted by the analysis from dynamic programming 95 percent of the time, their deviation in the remaining 5 percent is sufficiently large to yield the observed regression coefficients for $\Delta \log c_{t+1}$. The explanation for the dramatic effect is that income will drop with more than 50-percent probability at the highest income level, and that consumption must drop dramatically, if the agent has just consumed three times yearly earnings. This rather extreme scenario is probably not the explanation for the estimated values of our statistics: we put more stock in the first scenario described above. Clearly, a more de-

tailed empirical as well as theoretical investigation of spending patterns, depending on wealth and income, is called for. Nonetheless, we find these results very intriguing and encouraging. More often than not, our rule-based paradigm beats the dynamic programming paradigm when it comes to matching the facts in this context.

II. A General Dynamic Decision Problem

Next, we describe our framework in general terms. Time is discrete, starting at $t = 0$, and the agent is infinitely lived. At each date t and given the state $s = s_t$ from some set of states $S = \{s_1, \dots, s_n\}$, the agent must choose some action $a = a_t$ from a set of actions $\mathcal{A} = \{a_1, \dots, a_m\}$. The agent then experiences the instantaneous utility $u(s, a) \in \mathbb{R}$ and a new state $s' = s_{t+1}$ for the next period $t + 1$ is randomly selected from S according to some probability distribution $\pi_{s,a}$ on S . We assume throughout

TABLE 3—VARIATIONS IN INCOME CUTOFF, $\lambda = 1$ (VARIANCE IN TRANSITORY INCOME COMPONENT)

Regression of $\Delta \log c_{t+1}$ on constant and $\log y_t$					
income \geq	0.5	0.7	1.0	1.4	2.1
wealth ≥ 0	-0.135	-0.108	-0.109	-0.078	-0.118
wealth ≥ 0.02	-0.028	-0.028	-0.053	-0.042	-0.120
wealth ≥ 1	0.027	0.023	0.008	0.005	-0.090
wealth ≥ 3	0.007	0.004	-0.001	-0.003	-0.068
wealth ≥ 10	0.008	0.006	0.005	0.004	-0.031

Regression of $\Delta \log c_{t+1}$ on $E_t[\Delta \log y_{t+1}]$					
income \geq	0.5	0.7	1.0	1.4	2.1
wealth ≥ 0	0.981	0.662	0.667	0.335	0.845
wealth ≥ 0.02	0.050	-0.040	0.202	0.021	0.908
wealth ≥ 1	-0.441	-0.451	-0.317	-0.367	0.657
wealth ≥ 3	-0.207	-0.216	-0.177	-0.193	0.500
wealth ≥ 10	-0.185	-0.190	-0.181	-0.187	0.183

Average function of time in percent that agent applies the rule					
income \geq	0.5	0.7	1.0	1.4	2.1
wealth ≥ 0	83.3	66.7	50.0	33.3	16.7
wealth ≥ 0.02	33.5	27.9	24.4	16.8	11.2
wealth ≥ 1	14.6	14.5	14.0	12.2	8.9
wealth ≥ 3	6.1	6.1	6.0	5.8	4.9
wealth ≥ 10	1.8	1.8	1.8	1.8	1.6

Notes: The suboptimal rule 2 is applicable whenever wealth exceeds the level indicated and whenever income exceeds the level indicated. The transitory income component v_{it} has variance 0.04.

that $\pi_{s,a}(s') > 0$ for all $s' \in S$. Total time-zero utility is given by

$$U_0 = E_0 \left[\sum_{t=0}^{\infty} \beta^t u(s_t, a_t) \right],$$

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 and by changing zero transition probabilities to some small amount. Since the instantaneous utility as well as the transition probabilities depend on the state as well as the action chosen, we have not lost generality by assuming that the agent always has the same set of actions available to him, regardless of the underlying state.

Define a decision function to be a function $h : S \rightarrow \mathcal{A}$ and let \mathcal{H} be the set of all decision functions. The objective of a decision theory

is to find h , given a particular paradigm. We will consider and compare two such paradigms: the paradigm of dynamic programming (or, equivalently, dynamic optimization) and the paradigm of decision-making and learning with rules of thumb.

III. Dynamic Programming

The standard approach is to rewrite the decision problem above as a dynamic program,

$$(8) \quad v(s) = \max_{a \in \mathcal{A}} \{ u(s, a) + \beta E_{\pi_{s,a}} v(s') \}.$$

A standard contraction mapping argument as in Stokey et al. (1989) shows that there is a unique v^* solving the dynamic programming problem in (8). The solution is characterized by some (not necessarily unique) decision function $h^* : S \rightarrow \mathcal{A}$ that prescribes some action $h^*(s)$ in state s .

For any decision function h , define the associated value function v_h as the solution to the equation

$$(9) \quad v_h(s) = u(s, h(s)) + \beta E_{\pi_{s,h(s)}} v_h(s')$$

or as

$$(10) \quad \mathbf{v}_h = (\mathbf{I} - \beta \mathbf{\Pi}_h)^{-1} \mathbf{u}_h,$$

where \mathbf{v}_h is understood as the vector $[v_h(s_1), \dots, v_h(s_n)]'$ in \mathbb{R}^n , $\mathbf{\Pi}_h$ is the $n \times n$ matrix defined by

$$\Pi_{h,i,j} = \pi_{s_i,h(s_i)}(s_j),$$

and \mathbf{u}_h is the vector $[u(s_1, h(s_1)), \dots, u(s_n, h(s_n))]'$ in \mathbb{R}^n . Clearly, $v^* = \mathbf{v}_{h^*}$. The next proposition tells us that no randomization over actions is needed to achieve the optimum. This will contrast with some classifier systems in the example in subsection B below, which require randomizing among classifiers even in the limit.

PROPOSITION 1: *For all $s \in S$,*

$$v^*(s) = \max_{h \in \mathcal{H}} v_h(s).$$

For future reference, define μ_h to be the unique invariant probability distribution on S for the transition law $\mathbf{\Pi}_h$, i.e., μ_h is the solution to $\mu_h = \mathbf{\Pi}_h \mu_h$ with $\sum_s \mu_h(s) = 1$. The uniqueness of μ_h follows with standard results about Markov chains from the strict positivity of all $\pi_{s,a}(s')$.

IV. Decision-Making and Learning with Rules of Thumb

This section defines how the agent makes decisions by using rules of thumb and learns about their quality. We first define a rule of thumb as a mapping from a subset of the states into action space. For example, a rule of thumb might say ‘‘when the economy is in state s_1 , use action a_1 ; when it is in state s_3 , use action a_2 .’’ Each rule of thumb has an associated strength, which will be a measure of how well the rule has performed in the past. A list of these rules of thumb and strengths is called a classifier system. We assume that the set of

rules in the classifier system will be constant throughout the life of the agent. Learning takes place via updating the strengths. Rules that performed well in the past will have a high strength, while rules that performed poorly will have a low strength. Performance is measured not only with respect to how much instantaneous utility is generated, but also how the restrictions imposed on future choices by the current actions are evaluated. The spirit of the algorithm which we will use is dynamic programming in nature, but it requires only keeping track of past and present experiences and hence can be performed in real time.

More formally, let $\mathcal{A}^0 = a_0 \cup \mathcal{A}$, where a_0 is meant to stand for ‘‘dormant.’’ A rule of thumb is a function $r : S \rightarrow \mathcal{A}^0$ with $r(S) \neq \{a_0\}$. A classifier c is a pair (r, z) consisting of a rule of thumb r and a strength $z \in \mathbb{R}$. A classifier $c = (r, z)$ is called applicable in state s , if $r(s) \neq a_0$. A classifier system is a list $C = (c_1, \dots, c_K)$ of classifiers, so that for every state s , there is at least one applicable classifier.

Choose a decreasing gain sequence $^6 (\gamma_t)_{t=0}$ of positive numbers satisfying

$$(11) \quad \sum_{t=1}^{\infty} \gamma_t^p < \infty \quad \text{for some } p \geq 2,$$

$$(12) \quad \sum_{t=1}^{\infty} \gamma_t = \infty,$$

as well as an initial classifier system C_0 and an initial state s_0 for the initial date $t = 0$.

Classifier system learning is a stochastic sequence of states $(s_t)_{t=0}^{\infty}$, indices $(k_t)_{t=0}^{\infty}$ of classifiers and classifier systems $(C_t)_{t=0}^{\infty}$. Proceed recursively for each date $t = 0, 1, 2, \dots$ by going through the following steps. Before updating the strengths at date t , the current state s_t and the current classifier system C_t are known. Choosing an action takes place at the end of date t , whereas the updating step for the strength of the active classifier in period t takes place at the beginning of date $t + 1$. In detail:

⁶ For some of the terminology used, see Albert Benveniste et al. (1990).

1. (In date t) the classifier $c_t = (r, z)$ in C_t with highest strength among all applicable classifier in state s_t is selected.⁷ Use randomization with some arbitrarily chosen probabilities to break ties. Denote the index of the winning classifier by $k_t = k(s_t; C_t)$.
2. (In date t) the action $a_t = r(s_t)$ is carried out.
3. (In date t) the instantaneous utility $u_t = u(s_t, a_t)$ is generated.
4. (In date $t + 1$) the state transits from s_t to s_{t+1} according to the probability distribution π_{s_t, a_t} on S .
5. (In date $t + 1$) determine the index $k' = k(s_{t+1}; C_t)$ of the strongest classifier in C_t which is applicable in state s_{t+1} . Denote its strength by z' . Update the strength of classifier with index k_t to⁸

$$(13) \quad \tilde{z} = z - \gamma_{t+1}(z - u_t - \beta z')$$

The classifier system C_{t+1} is then defined to be the classifier system C_t with c replaced by $\tilde{c} = (r, \tilde{z})$.

A classifier system C thus gives rise to a decision function $h(s; C) \equiv r_{k(s; C)}(s)$ by selecting the strongest among all possible classifiers at each state. The updating of the strength of the classifier activated in period t occurs at stage 5 when u_t and s_{t+1} are known. Note that the updating equation (13) uses the period t strengths to determine z' , which is added to the strength of classifier that is active in period t . We are *not* using C_{t+1} here, since C_{t+1} is not available yet. In other words, the agent makes a forecast about C_{t+1} by using the no-change prediction C_t . After finishing with stage 5, we go on to stage 1 in time $t + 1$. The classifier chosen at stage 1 in period $t + 1$ might differ from the ‘‘hypothetical’’ one which was used to complete the updating in stage 5. The updating algorithm is formulated in such a way that the updating does not require calculating the strengths at $t + 1$ first,

which would otherwise give rise to complications in cases where the activated classifiers at date t and $t + 1$ have the same index.

There is a connection to dynamic programming, which should be pointed out here. Consider the term in brackets in equation (13), which is used for updating, and suppose that updating was no longer necessary, as the strengths have already converged. We would then have

$$(14) \quad z = u(s_t, r_t(s_t)) + \beta z',$$

which looks formally similar to

$$v(s_t) = u(s_t, h^*(s_t)) + \beta v(s')$$

if we drop expectations in equation (8) for the moment: strengths there correspond to values here. We return with additional tools to this comparison with equations (18) and (19) below. Here, we just note that the difference between the two sides of equation (14) is used to update the strength z : the larger that difference, and the less experience the agent has (t small, i.e., γ_t large), the larger the adjustment of the strength. An updating equation like (13) is common in the learning literature, see e.g., Albert Marcet and Sargent [1989 equation (4a)]. One can think of equation (13) as an error correction model, in which the weight on the error correction term is decreasing over time. It is also often referred to as a *bucket brigade*, since each activated classifier has to pay or give away part of its strength z in order to get activated, but in turn receives not only the instantaneous reward u_t for his action, but also the ‘‘payment’’ by the next activated classifier $\beta z'$, discounted with β .

Note, that the dimension K of the strength vector can in principle be substantially larger than n , the dimension of the value function, since a classifier system can contain up to $(\#\mathcal{A} + 1)^{\#S}$ classifiers with different rules, and even more, if several classifiers use the same rule. However, we like to think of applications, in which only a few rules are used, so that the strength vector is shorter than the dimensionality of the state space. We then have an attractive way of modelling bounded rationality, since an agent using this learning scheme has to memorize only a small number

⁷ Another method to determine the decision function is to randomize among applicable classifiers according to their relative strengths; see, e.g., Arthur (1993).

⁸ Marimon et al. (1990) introduce an adjustment factor in the bidding to account for differences in the ‘‘generality’’ of classifiers. We will consider this extension in Section VI, subsection C.

of strengths, and learns by simply adding and subtracting from the strength vector in real time. He does not need to form forward-looking expectations or solve fixed-point problems.

Our definitions are motivated by Holland's (1992) work on classifier systems, but we do not include a rule-generating or rule-modifying feature such as the genetic algorithm, since that deemed to be too hard to justify and too hard to analyze at this point in the ongoing research, and furthermore eliminated features which do not seem essential for the issues studied here. First, without the genetic algorithm, there is no particularly good reason to insist on binary encoding. Second, our rules are mappings from subsets of states into actions, whereas research on classifier systems typically assumes a rule to always take the same action in all states in which it is applicable. It is easy to see, however, that both are formally equivalent if one allows for a suitable redefinition of the action space (see the fourth remark in Section VI, subsection A). Note that Holland's original definition of classifier systems allowed rules to have state-dependent actions as well (see Holland, 1986 p. 603).

V. Asymptotic Behavior

Classifier-system learning leads to a stochastic sequence of decision functions $(h_t)_{t=0}^{\infty}$ given by $h_t = h(s_t; C_t)$. We are interested in determining the asymptotic behavior of this sequence, i.e., which decision functions are eventually learned, and whether they coincide with an optimal decision function for (8). We characterize all possible learning outcomes with strict strengths orderings. The convergence proof itself uses results from the stochastic approximation literature⁹ (for a general overview and introduction to stochastic approximation algorithms, see L. Ljung et al., 1992; Sargent, 1993).

One might think the requirement of a strict asymptotic strengths ordering to be generically satisfied, but it is not. Section VI, subsection B, provides an example in which one cannot

obtain such a strict asymptotic ordering for an open set of utility values. Intuitively, imagine two candidates for the asymptotic ordering of the strengths. It may so happen, that the first ordering results in a behavior, where one would rather conclude the second ordering to be correct based on average experience *and vice versa*. Over time, the observed ordering flips back and forth between these two possibilities, settling on assigning equal strengths to some classifiers asymptotically. This case is not yet covered by the theory here, but would make for an interesting extension. For now, we stick to strict asymptotic orderings.

For ease of notation, let $\mathbf{Y}_t = [s_{t-1}, k_{t-1}, s_t, k_t]$ and let $\boldsymbol{\theta}_t$ denote the vector of all strengths z_k , $k = 1, \dots, K$ of the classifier system C_t at date t . Consider a vector of strengths $\boldsymbol{\theta}_{\infty}$ so that, conditional on some strength $\boldsymbol{\theta}_{t_0}$ and some value for \mathbf{Y}_{t_0} at some date t_0 , we have $\boldsymbol{\theta}_t \rightarrow \boldsymbol{\theta}_{\infty}$ with positive probability. If all elements of $\boldsymbol{\theta}_{\infty}$ are distinct, we call $\boldsymbol{\theta}_{\infty}$ an *asymptotic attractor*. We aim at characterizing all asymptotic attractors. Indeed, the example about excess sensitivity of consumption in Section I was based on the calculation of such an asymptotic attractor. Calculating the strengths z_1 and z_2 with equations (4) and (5) was based on averaging over many periods, replacing sample averages by expectations, i.e., probabilistic averages. We will show that all asymptotic attractors can be computed in this manner as long as one also checks the consistency condition below. Following this procedure of Section I, i.e., to calculate the asymptotic attractors and hence to figure out where a rule-based agent eventually gets stuck, is probably the most natural and useful thing to do in any application of our framework. This section provides an algorithm for calculating these asymptotic attractors, and proves that this algorithm indeed works.

Given the K rules, the algorithm takes the following steps. The idea is to start with some candidate ranking of the strengths of the rules and then to calculate the implications (such as numerical values of the strengths) of that ranking. The implied numerical values for the strength values then need to be checked for whether their ranking

⁹ Marimon et al. (1990 Sec. 5) already suggest using stochastic approximation results to study the limit behavior of classifier system.

is consistent with the ranking which was assumed at the start. If all asymptotic attractors need to be calculated, then this algorithm needs to be performed for all $K!$ permutations of $\{1, \dots, K\}$.

1. Pick a permutation of $\{1, \dots, K\}$. Think of this permutation as a candidate ranking of the strengths. For example, for five rules, that permutation might take the form (4, 1, 3, 5, 2), and we will be looking for an asymptotic attractor, in which rule 4 will have the highest strength and rule 2 the lowest.
2. Given this candidate ranking of the strengths, find the winning rule $k(s)$ for each state s by selecting from the chosen permutation the first among the applicable rules for that state. For example, if rules 2 and 3 are applicable in some state s , we would use $k(s) = 3$ in the example above.
3. Equipped with $k(s)$, find the implied decision function $h(s) = r_{k(s)}(s)$, the utilities $u(s, h(s))$, and the probabilities $\Pi_{h,i,j}$ for transiting from state s_i to state s_j .
4. Find the unique, invariant distribution μ_h over states for the probabilities $\Pi_{h,i,j}$. Find the unconditional probability $\nu(k)$ for choosing a particular rule k by summing over all the unconditional probabilities $\mu_h(s)$ of being in states s , in which the particular rule is chosen, $k(s) = k$. Formally,

$$\begin{aligned} \nu(k) &= \text{Prob}(k = k(s)) \\ &= \sum_{\{s|k=k(s)\}} \mu_h(s). \end{aligned}$$

Call classifiers *asymptotically active*, if they are a winning classifier for at least one state, i.e., if $\nu(k) > 0$. Call all other classifiers *asymptotically inactive*. Only asymptotically active classifiers are activated with some positive probability, given the rule choice function $k(s)$. Let $\tilde{K} = \{k : \nu(k) \neq 0\}$ be the set of asymptotically active classifiers.

5. Calculate the average instantaneous utility u_k generated by an asymptotically active

classifier $k \in \tilde{K}$,

$$\begin{aligned} (15) \quad u_k &= E_{\mu_h}[u(s, h(s)) | k(s) = k] \\ &= \frac{1}{\nu(k)} \sum_{\{s|k(s)=k\}} \mu_h(s) u(s, h(s)) \end{aligned}$$

and let $\tilde{\mathbf{u}} \in \mathbb{R}^{\tilde{K}}$ be the vector of these utilities u_k for *asymptotically active* classifiers. One needs to be a bit careful here in the meaning of indices: they always refer to the classifier and not to the position in the vector $\tilde{\mathbf{u}}$.

6. Calculate the matrix $\mathbf{B} \in \mathbb{R}^{\tilde{K}, \tilde{K}}$ of transition probabilities between classifier choices for *asymptotically active* classifiers, i.e.,

$$\begin{aligned} (16) \quad B_{k,l} &= \text{Prob}(\{s' : k(s') = l\} | k(s) = k) \\ &= \frac{\sum_{\{s|k=k(s)\}} \sum_{\{s'|l=k(s')\}} \mu_h(s) \pi_{s,h(s)}(s')}{\nu(k)}. \end{aligned}$$

Again, one needs to be a bit careful here in the meaning of indices: they always refer to the classifier and not to the position in the matrix \mathbf{B} .

7. Calculate

$$(17) \quad \tilde{\boldsymbol{\theta}}_\infty = (\mathbf{I} - \beta \mathbf{B})^{-1} \tilde{\mathbf{u}}$$

to find the strengths for asymptotically active classifiers. For asymptotically inactive classifiers, choose any value strictly below the minimum of all the computed values for asymptotically active classifiers. Together, these strengths form the vector $\boldsymbol{\theta}_\infty$. For ease of notation, let $z_k = \theta_{\infty,k}$ be the asymptotic strength of classifier k .

8. Check the

CONSISTENCY CONDITION:

for all states $s \in S$ and all applicable, losing rules for that state, $k \neq k(s)$ and $r_k(s) \neq a_0$, the strength is dominated by the strength of the winning rule $k(s)$:

$$z_k < z_{k(s)}.$$

This consistency condition checks, whether the ranking of the implied strengths is consistent with the chosen candidate ranking, from which we started. The consistency check is done by examining the choice for the winning classifier due to the candidate ranking as this is the essential feature of the initial ranking.¹⁰ If the consistency condition is satisfied, terminate successfully. Otherwise terminate unsuccessfully.

A vector θ_∞ is called a *candidate asymptotic attractor* if it can be calculated with this algorithm for some initial ranking of the strengths and under successful termination, i.e., if it satisfies equation (17) as well as the consistency condition.

It is instructive to compare the equation characterizing the value function, given the optimal decision function h^* ,

$$(18) \quad v(s) = u(s, h^*(s)) + \beta \sum_{s'} \pi_{s,h^*(s)}(s') v(s')$$

to equation (17), characterizing the strengths of the asymptotically active classifiers and rewritten as

$$(19) \quad z_k = u_k + \beta \sum_l B_{k,l} z_l.$$

The similarity is striking. The key difference is that equation (18) attaches values to *states*, whereas the algorithm here attaches strengths to *rules* [see equation (19)]. Thus, the strength of a rule is intuitively something like an average of all the values $v(s)$ for all those states where the rule is applied. In Section I, we have indeed calculated the asymptotic strengths z_1 and z_2 in this manner: compare equations (4) and (5) with equation (19).

¹⁰ One could alternatively simply check directly, whether one gets the same ranking of the strengths. Paying a bit of attention to how one sets the strengths for the asymptotically inactive classifiers, this will deliver the same result, as one tries out all possible candidate rankings at the start of this algorithm.

Another way to see the similarity is to introduce a new state-dependent function $x(s_i)$ and to rewrite (19) as

$$(20) \quad x(s) = u(s, h(s)) + \beta \sum_{s'} \Pi_{s,h(s)}(s') z_{k(s')}$$

$$(21) \quad z_k = \sum_{\{s|k=k(s)\}} \frac{\mu_h(s)}{\nu(k)} x(s),$$

if $\nu(k) \neq 0$.

The first equation (20) is even more similar to equation (18), except that the decision function h rather than h^* is used and except that the values $v(s')$ for the next state s' are replaced with the strengths of the classifier $z_{k(s')}$ to be used in that state. The second equation (21) tells us to sum all the $x(s)$ for states s in which a particular rule k is activated, using the probabilities of being in state s conditional on using rule k as weights. For the Bellman equation, the decision function h^* is obtained via maximization. For classifier systems, the consistency condition is used instead. Finally, Proposition 2 shows there is an intriguing relationship between v_h and θ_∞ .

PROPOSITION 2: *For any candidate asymptotic attractor, the expected value function associated with its decision rule equals the expected strength,*

$$E_{\mu_h}[z_{k(s)}] = E_{\mu_h}[v_h(s)] = \frac{1}{1 - \beta} E_{\mu_h}[u(s, h(s))].$$

However, one cannot in general just average over the $v_h(s)$ for those states s in which a classifier is active to get its strength,

$$(22) \quad z_k \neq E_{\mu_h}[v_h(s) | k(s) = k]$$

(in general).

We note in summary, that for each of the $K!$ possible strict orderings of the K classifiers, there is a vector $\theta_\infty \in \mathbb{R}^K$ satisfying (17) for

the asymptotically active classifiers. θ_∞ is unique up to the assignment of strengths to asymptotically inactive classifiers, and is a candidate asymptotic attractor, if it also satisfies the consistency condition. The next theorem shows that the algorithm indeed provides a complete characterization of all asymptotic attractors.

THEOREM 1: *Every candidate asymptotic attractor is an asymptotic attractor and vice versa.*

The proof of this theorem can be found in the Appendix. It draws on a result by Michel Métivier and Pierre Priouret (1984) about Markov stochastic approximation procedures. The theorem indicates how classifier system learning happens over time. For some initial periods, the orderings of the classifiers may change due to chance events. Eventually, however, the system has cooled down enough and a particular ordering of the strengths is fixed for all following periods. As a result, the asymptotically inactive classifiers will no longer be activated, and the system converges to the asymptotic attractor as if the transition from state to state was exogenously given. The classifier system has learned the final decision rule. Alternatively, one can train a classifier system to learn a particular decision rule corresponding to some candidate asymptotic attractor by forcing the probabilistic transitions from one state to the next to coincide with those generated by the desired decision rule. After some initial training periods and with possibly sizeable probability, the strengths will remain in the desired ordering and will not change the imprinted pattern. However, given a particular history, the theorem and its proof do not rule out the possibility that the strengths may break free once more to steer towards a different limit. In fact, this will typically happen with some probability due to (12). A sufficiently long string of unusual events could have a large effect on the updating of the strengths in (13) and thus change an existing ordering.

Keep in mind that the strict ordering of the strengths is part of our definition of asymptotic attractors and candidate asymptotic attractors.

It should thus be noted that the characterization applies only to asymptotic attractors with a strict ordering of the strengths. As we will see in the next section, this is not just ruling out knife-edge cases. We will construct a robust example in which equality of the strengths of two classifiers is necessary asymptotically. In that sense, our theory is not exhaustive. It covers only the cases in which the agent is never indifferent asymptotically between choosing between two asymptotically active classifiers and thus has no reason to randomize. This theory could also consider randomization as a natural extension to cover the cases of ties as well, but this would probably involve quite a bit of additional machinery.

The paper started out in Section I by arguing that classifier system learning can lead to a good state bias, and demonstrating this claim in an example. In words, the *good state bias* means that a suboptimal rule cannot be active in just the worst states as measured by the value function, when that suboptimal rule is active in only a subset of states, and competes against the everywhere applicable dynamic programming solution. It would be desirable to have a general theorem, stating this. We conjecture the following result to be true. We have not been able to falsify it in numerical experiments, but were also unable to prove it so far.

CONJECTURE 1: *Suppose there are two rules. Let the first rule r_1 be active in all states and coincide with a solution h^* to the dynamic programming problem. Let the second rule r_2 be active in only a strict subset $S^{(2)}$ of all states. Suppose each rule is active, i.e., suppose that $z_2 > z_1$, where z_k is the asymptotic strength of classifier k . Then,*

$$\min_{i \in S/S^{(2)}} v(s_i) < \max_{i \in S^{(2)}} v(s_i).$$

We can prove the conjecture in the case of just two states, however.

PROPOSITION 3: *Suppose there are two rules and two states. Let the first rule r_1 be active in all states and coincide with a solution h^* to the dynamic programming problem. Let*

the second rule r_2 be active in only one state, say s_2 . Suppose each rule is active, i.e., that $z_2 \geq z_1$, but that they together do not implement a solution to the dynamic programming problem. Then, $v^*(s_2) > v^*(s_1)$.

VI. Illustrations and Extensions

A. Some Simple Cases

There are several simple cases, in which the asymptotic behavior is easy to see and which establish useful benchmarks. Furthermore, there are numerical examples to study the case of a union of rules and of randomizing between rules. Details, like additional proofs or calculations, are available from the authors upon request as part of the technical Appendix.

1. Suppose there is only one rule. Then there is a unique candidate asymptotic attractor $\theta_\infty \in \mathbb{R}$ and it satisfies $\theta_\infty = E_{\mu_r}[v_r]$. In particular, if $r = h^*$, then $\theta_\infty = E_{\mu_{h^*}}[v^*]$. This follows immediately from Proposition 2.
2. Let h^* be a decision function with $v^* = v_{h^*}$ and suppose that h^* is unique. Suppose, furthermore, that all K rules are applicable in at most one state and that for each $s \in S$, there is exactly one rule with $r(s) = h^*(s)$; denote its index with $k^*(s)$. Define θ_∞ by assigning for each state s strength $z_{k^*(s)} = v^*(s)$ to the classifier with index $k^*(s)$. For all other rules, assign some strength strictly below all $v^*(s)$ for states s , in which that rule is applicable. Then θ_∞ is a candidate asymptotic attractor which implements the dynamic programming solution.¹¹
3. If all rules are applicable in all states (i.e., all rules are total), then for each rule, there is an asymptotic attractor θ_∞ , where that rule is the only asymptotically active rule.
4. In many parts of the literature on classifiers, rules are action based, i.e., prescribe the same action regardless of the underlying state. Formally, this can always be accomplished as follows. Given some classifier system with K rules r_k , $k = 1, \dots, K$, define a new action space $\tilde{\mathcal{A}} = \{\tilde{a}_1, \dots, \tilde{a}_K\}$. Replace the utility payoffs $u(s, a)$ by $\tilde{u}(s, \tilde{a}_k) = u(s, r_k(s))$, if r_k is applicable in state s , and by some number below the minimum of all $u(s, a)$ otherwise. Similarly, replace the transition probabilities $\pi_{s,a}(s')$ by $\tilde{\pi}_{s,\tilde{a}_k}(s') = \pi_{s,r_k(s)}(s')$, if $r_k(s)$ is applicable in state s and by $\pi_{s,a_1}(s')$ otherwise. Finally, replace the K old rules by the K new rules $\tilde{r}_k(s) = \tilde{a}_k$, $k = 1, \dots, K$ with the same domains of applicability as the old rules. After all these redefinitions, the rules are action based, but nothing of substance has changed. An asymptotic attractor for the untransformed system is an asymptotic attractor for the transformed system, and vice versa, and the dynamic programming solution stays the same too.
5. Suppose we replace two active rules which are disjoint with the union of the rules. Then it is possible that the new rule becomes inactive. We found an example, involving four rules. One apparently needs to be quite careful in specifying just the right payoffs and transition probabilities to generate such an example, however.

¹¹ This is closely related to results about Q -learning introduced by C. Watkins (1989). In the notation of this paper, Q -learning algorithms update strengths $Q(s, a)$ applicable in state s for action a . This corresponds to classifiers that are only applicable in a single state. Q -learning defines one Q for each possible state-action combination. The updating of the Q 's is similar to the equation (13). The difference is that not the strongest of the applicable Q 's is determining the action taken, but by some other

mechanism that allows for enough exploration so that all Q 's are triggered infinitely often. Then the matrix of Q 's will converge to the values implied by the value function from dynamic programming (see A. Barto et al., 1993). Clearly, the definitions of the Q 's is a special case of the classifiers defined in this paper, since classifiers are allowed to cover more general sets of state-action pairs. The second difference is that classifier learning always selects the strongest classifier among the applicable ones. Q -learning activates Q 's using a randomization method that guarantees that each $Q(s, a)$ is used often enough. However, this advantage is offset by the increased space complexity compared to classifiers with more general domains.

B. An Example

We provide an example that demonstrates the similarities and differences between classifier systems and the dynamic programming approach. It also demonstrates why the case of asymptotically equal strengths cannot be ruled out. The example is abstract and meant for illustration only; it has therefore been kept as simple as possible.

Suppose $S = \{1; 2; 3\}$, $\mathcal{A} = \{1; 2\}$ and that the transition to the next state is determined by the choice of the action only, regardless of the current state s :

$$s' = 1 \qquad s' = 2 \qquad s' = 3$$

$$a = 1: \pi_{s,1}(1) = 1/3, \quad \pi_{s,1}(2) = 1/3, \quad \pi_{s,1}(3) = 1/3$$

$$a = 2: \pi_{s,2}(1) = 0, \quad \pi_{s,2}(2) = 1, \quad \pi_{s,2}(3) = 0.$$

Note that some probabilities are zero, in contrast to our general assumption. This is done to simplify the algebra for this example. We further have a discount factor $0 < \beta < 1$ and utilities $u(s, a)$, $s = 1, 2, 3$, $a = 1, 2$. We assume without loss of generality that $u(2, 1) = 0$. We impose the restriction that $u(3, a) = u(1, a)$ for $a = 1, 2$, so that state $s = 3$ is essentially just a ‘copy’ of state $s = 1$. Thus, there are three free parameters, $u(1, 1)$, $u(1,2)$, and $u(2, 2)$.

The difference between state $s = 1$ and state $s = 3$ is in how they are treated by the available rules. Assume that there are two rules, r_1 and r_2 , described by

$$s = 1 \qquad s = 2 \qquad s = 3,$$

$$r_1: \quad r_1(1) = 1, \quad r_1(2) = 0, \quad r_1(3) = 1$$

$$r_2: \quad r_2(1) = 2, \quad r_2(2) = 1, \quad r_2(3) = 0$$

with ‘0’ denoting the dormant action a_0 . Note that the two given rules never lead to action $a = 2$ in state $s = 2$. The value of $u(2, 2)$ is thus irrelevant for the comparison of the classifiers. Strength comparisons will thus place restrictions only on the remaining two free parameters $u(1, 1)$ and $u(1, 2)$.

We aim at calculating all candidate asymptotic attractors. Since there are only two rules,

there can be only two strict rankings of the corresponding classifier strengths, namely $z_1 > z_2$ (case I) and $z_2 > z_1$ (case II). Calculating the strengths with the algorithm of Section V, one can show that case I can arise if and only if $u(1, 1) > 0$, whereas case II can arise if and only if $u(1, 2) > 3u(1, 1)$.

It is interesting to also consider the case $z_1 = z_2$ (case III) with nontrivial randomization between the classifiers, a situation not covered by our theoretical analysis above. The reasoning employed here should be rather intuitive, however. Given state $s = 1$, we guess that classifier c_1 is activated with some probability p , whereas classifier c_2 is activated with probability $1 - p$ (i.e., there is randomization between the classifiers). The resulting decision function is random. Given $s = 1$, states $s' = 1$ and $s' = 3$ will be reached with probability $p/3$ each. The invariant distribution μ_h is therefore $\mu_h(1) = \mu_h(3) = 1/(4 - p)$, $\mu_h(2) = (2 - p)/(4 - p)$. The joint probabilities that state s occurs and classifier k is activated, $\mu_h(s, k)$, follow immediately. The common strength z should satisfy both equations arising from (17) yielding

$$(23) \quad \frac{1}{1 - \beta} u(1, 1) = z$$

$$= \frac{1}{1 - \beta} \frac{1 - p}{3 - 2p} u(1, 2),$$

which can be solved for p . Note that p is a viable probability if and only if $0 \leq p \leq 1$. Thus, case III is valid, if and only if one of the following two inequality restrictions is satisfied:

1. $u(1, 2) \leq 3u(1, 1) \leq 0$ or
2. $u(1, 2) \leq 3u(1, 1) \geq 0$.

The strengths and probabilities in this case are unique except if $u(1, 1) = u(1, 2) = 0$. The inequalities have to be strict in order for p to be nondegenerate. Otherwise, the decision rule obtained coincides with the one derived from case I or case II. If the probability p is strictly between zero and one, then there will be alteration between the two classifiers even asymptotically. Thus, there will be randomization between the actions as a function of the

TABLE 4—ALL POSSIBLE ASYMPTOTIC OUTCOMES IN EXAMPLE IN SECTION VI, SUBSECTION B

Area	Restriction	Case I	Case II	Case III	$h^*(1)$	$h^* = h?$ (Dynamic programming = rules?)
		$z_1 > z_2$	$z_1 < z_2$	$z_1 = z_2$		
A1	$u(1, 1) > 0$ $u(1, 2) \leq 3u(1, 1)$ $u(1, 2) \leq (1 + 2/3\beta)u(1, 1)$	Yes	No	No	1	yes
B2	$u(1, 1) > 0$ $u(1, 2) \leq 3u(1, 1)$ $u(1, 2) \geq (1 + 2/3\beta)u(1, 1)$	Yes	No	No	2	no
A2	$u(1, 1) \leq 0$ $u(1, 2) > 3u(1, 1)$ $u(1, 2) \geq (1 + 2/3\beta)u(1, 1)$	No	Yes	No	2	cannot
B1	$u(1, 1) \leq 0$ $u(1, 2) > 3u(1, 1)$ $u(1, 2) \leq (1 + 2/3\beta)u(1, 1)$	No	Yes	No	1	no, but could
C1	$u(1, 1) \leq 0$ $u(1, 2) \leq 3u(1, 1)$ $u(1, 2) \leq (1 + 2/3\beta)u(1, 1)$	No	No	Yes	1	random
C2	$u(1, 1) > 0$ $u(1, 2) > 3u(1, 1)$ $u(1, 2) \geq (1 + 2/3\beta)u(1, 1)$	Yes	Yes	Yes	2	maybe

Notes: This table shows the various cases for the numerical example of Section VI, subsection B, parameterized by the utility received in state 1, when taking action 1, $u(1, 1)$ or action 2, $u(1, 2)$ (see also Figure 1).

state asymptotically in contrast to the dynamic programming solution.

Table 4 shows that for any given values of $u(s, a)$ there is at least one applicable case. However, the only case available may be case III and thus the solution prescribed by the classifier system involves randomization between the classifiers.

Let us now compare these possibilities with the solution to the dynamic programming problem. If $u(2, 2)$ is large enough, the optimal decision function will always prescribe action $a = 2$ in state $s = 2$, which cannot be achieved with the classifiers above. Assume instead that $u(2, 2)$ is small enough, so that the optimal decision function takes action $h^*(2) = 1$ in state $s = 2$. Directly calculating $v^* = v_{h^*}$ with equation (10) for the two choices yields

$$h^*(1) = \begin{cases} 1, & \text{if } u(1, 2) \leq (1 + \frac{2}{3}\beta)u(1, 1), \\ 2, & \text{if } u(1, 2) \geq (1 + \frac{2}{3}\beta)u(1, 1). \end{cases}$$

A summary of all possible situations is found in Table 4 and Figure 1. The learnable decision

function may not be unique (area C2). The learnable decision function may involve asymptotic randomization between the available rules (area C1). The learnable decision function can also be different from the solution to the dynamic programming problem, even if that solution is attainable by ranking the classifiers appropriately (this is the case in area B1). Intuitively, since $u(2, 1)$ has been normalized to zero, $u(1, 1) = u(3, 1)$ measures how much classifier c_1 gains against classifier c_2 by being applicable in state $s = 3$ rather than state $s = 2$. If $u(1, 1)$ is positive, state $s = 3$ corresponds to “good times” and state $s = 2$ corresponds to “bad times.” Since the accounting system (13) for calculating the strengths of classifiers does not distinguish between rewards generated from the right decision and those generated from being in good times, a classifier that is applicable only in good times “feels better” to the rule-using agent than it should. Thus, if $u(1, 1) > 0$, classifier c_1 may be used “too often” and if $u(1, 1) < 0$, classifier c_1 may be used “too little.” This is what happens in areas B and C. An important insight of this example and Figure 1 is that both the areas of strict orderings of

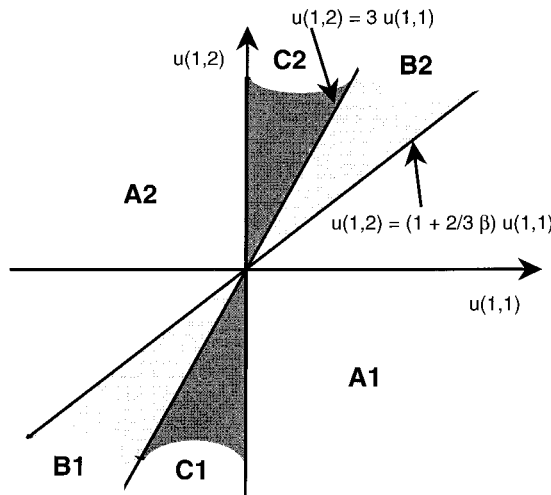


FIGURE 1. GRAPHIC REPRESENTATION OF CASES IN TABLE 4

Notes: This figure shows the various cases for the numerical example of Section VI, subsection B, parameterized by the utility received in state 1, when taking action 1, $u(1, 1)$ or action 2, $u(1, 2)$ (see also Table 4). For example, the rule-based decision is unique and coincides with the dynamic programming solution in area A1, is not unique in area C2, or requires equal strength and thus is the probabilistic choice between the two rules even asymptotically in area C1.

classifiers as the only possible asymptotic outcome (areas A and B) and the area of equality of classifiers as the only possible asymptotic outcome (area C1) are robust to parameter changes.

C. Strength Adjustment

In the accounting scheme (13) as laid out in Section IV all classifiers are treated equally independent of the states in which they are active. This raises the immediate question whether it is possible to adjust the scheme so that the learnable decision function coincides with the dynamic programming solution. For example, Marimon et al. (1990) adjust the payment of the classifiers with a proportional factor that depends on the number of states in which each classifier is active. Would that do the trick?

Consider the above example with only two classifiers. Let $\kappa_1 > 0$ and $\kappa_2 > 0$ be adjustment factors for classifiers 1 and 2, respectively, and let $\zeta_i = \kappa_i z_i$ be the adjusted strength for classifier i . The most obvious way of altering the accounting scheme (13) is as follows. Find the strongest classifier by

comparing the values of ζ instead of z and update the strength according to

$$(24) \quad \tilde{z} = z - \gamma_{t+1}(\kappa z - u_t - \beta \kappa' z')$$

or, equivalently, $\tilde{\zeta} = \zeta - \kappa \gamma_{t+1}(\zeta - u_t - \beta \zeta')$. This equation differs from (13) only by a classifier-individual scalar adjustment for the gain γ_{t+1} . Asymptotically, only the expression in brackets matters, and there is no difference. Thus, this alteration does not get us any closer to the dynamic programming solution than before. Somehow, the term used to update the strength has to be adjusted differently from the strength comparison.

Consider therefore a different alteration. Compare z_k 's to determine the winning classifier, but use (24) to update the strengths. Intuitively, the winning classifier "pays" κz instead of z to the receiving classifier, but κ is not used to rescale strengths for determining the winning classifier. Indeed, the "bids" in Marimon et al. (1990) correspond to our payment κz . Like here, the winning classifier in their paper is determined by the strength and not by the bid. To calculate a candidate asymptotic attractor, modify equation (19) by

replacing z_k with $\kappa_k z_k$ everywhere and solve. For the example, let us normalize κ_2 to unity, assume that $u(1, 1) < 0$, assume that the dynamic programming solution prescribes $h^*(1) = 1$, and let us check whether it is possible to find a value for κ_1 which guarantees that the classifier learning solution is identical to the dynamic programming solution, i.e., that the $z_1 > z_2$. In terms of Figure 1, we are trying to adjust κ so that area B1 and C1 get eliminated and become part of area A1 instead. After some algebra, one finds that the first classifier is stronger than the second one, if and only if,

$$(25) \quad \kappa_1 > \frac{3 - \beta}{2\beta}.$$

Intuitively, classifier 2 is too strong in area B1, and we need to give the first classifier an extra advantage by raising its adjustment factor κ_1 . But we only want to make that adjustment, when indeed $h^*(1) = 1$, i.e., when $u(1, 2) \leq (1 + 2/3\beta)u(1, 1)$. This shows that we can find appropriate payment adjustments that can lead the classifier system solution to coincide with the dynamic programming solution if it is attainable, but it is not possible to select the correct payment adjustment factors without knowing the dynamic programming solution. Furthermore, the proper adjustments correct the good state bias rather than differences in the number of states in which a rule is applicable.

VII. Conclusion

In this paper, we have introduced learning about rules of thumb and analyzed its asymptotic behavior. We have discussed how a bucket brigade learning algorithm about the strengths of the rules is able to deal with general discrete recursive stochastic dynamic optimization problems. We have reformulated the evolution of the strengths as a stochastic approximation algorithm and obtained a general characterization of all possible limit outcomes with strict strength rankings.

We have compared classifier learning to dynamic programming. While there are some formal similarities between the computation of strengths in our framework and the computa-

tion of the value function in the context of dynamic programming, there are also some important differences. Strengths provide a crude average of values across all states where a particular rule is applied. As a result, a sub-optimal rule might dominate the optimal one if it is applicable only in ‘‘good’’ states of the world: bad decisions in good times can ‘‘feel better’’ than good decision in bad times.

We have demonstrated that this ‘‘good state bias’’ might help in understanding the observation of high sensitivity of consumption to transitory income. A simple theoretical example was analyzed exhaustively. It was shown that the attainable decision function is neither necessarily unique nor characterized by a strict ordering of classifiers. It was furthermore shown that adjusting the learning algorithm to correct the good state bias cannot be done in a simple way.

APPENDIX

We now provide the proof for Theorem 1. Additional calculations, propositions and all other proofs are in a technical Appendix which can be obtained from the authors. Everything here can also be followed without it, provided one has access to Métivier and Priouret (1984).

It is convenient for the following analysis to rewrite the accounting scheme (13) in the following way as a stochastic approximation algorithm. The goal here is to cast the updating scheme into a Markov form. This transformation enables us to study the properties of the system, using existing results. Given \mathbf{Y}_t and $\boldsymbol{\theta}_t$, the vector $\boldsymbol{\theta}_{t+1}$ is computed via

$$(A1) \quad \boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \gamma_{t+1} f(\boldsymbol{\theta}_t, \mathbf{Y}_{t+1})$$

with

$$(A2) \quad f(\boldsymbol{\theta}_t, \mathbf{Y}_{t+1}) = \mathbf{e}_{k_t} g(\boldsymbol{\theta}_t, \mathbf{Y}_{t+1}),$$

where \mathbf{e}_{k_t} is the K -dimensional unit vector with a one in entry k_t and zeros elsewhere, and where the scalar factor $g(\boldsymbol{\theta}_t, \mathbf{Y}_{t+1})$ is given by

$$(A3) \quad g(\boldsymbol{\theta}_t, \mathbf{Y}_{t+1}) \\ = \theta_{t,k_t} - u(s_t, r_{k_t}(s_t)) - \beta \theta_{t,k_{t+1}}.$$

The first equation (A1) 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} . The format of that equation is similar to the bucket brigade equation (13), but is now restated here for the entire vector of strengths. The second equation (A2) states that this correction takes place only in one component of the strength vector, namely the strength corresponding to the classifier k_t , which was activated at date t . The third equation (A3) states by how much that entry should be changed and rewrites the term in brackets of the bucket brigade equation (13). Together, these three equations achieve the same result as the bucket brigade equation (13). Note, that $f(\theta, Y)$ is linear in θ . Given a transition law Π_h for states, one can derive the implied transition law $\hat{\Pi}$ for Y_t and its invariant distribution, keeping the decision function h fixed.

PROOF OF THEOREM 1:

We first show that a given candidate asymptotic attractor is an asymptotic attractor. To that end, we analyze first an alteration of the stochastic approximation scheme above and characterize its limits in the claim below. We then show that the limit to this altered scheme corresponds to an asymptotic attractor in the original scheme. Let $\underline{u} = \min_{s,a} u(s, a)$ and $\bar{u} = \max_{s,a} u(s, a)$ be, respectively, the minimum and the maximum one-period utility attainable. We assume without loss of generality that the initial strengths are bounded below by $\underline{u}/(1 - \beta)$.

Claim: Consider a candidate asymptotic attractor θ_∞ and its associated decision function h . Fix the transition probabilities Π_h . Consider the following altered updating system: let the classifier system consist only of the asymptotically active classifiers according to θ_∞ . Fix some starting date t_0 , an initial strength vector θ_{t_0} with $\theta_{t_0,l} \geq \underline{u}/(1 - \beta)$ for all l and an initial state Y_{t_0} . Let $\tilde{\theta}_t$ be the vector of strengths of this reduced classifier system at date $t \geq t_0$ and let θ_∞ be the corresponding subvector of θ_∞ of strengths of only the asymptotically active classifiers. Furthermore, let the transition from state s_t to s_{t+1} always be determined by

the transition probabilities Π_h . Then $\tilde{\theta}_t \rightarrow \tilde{\theta}_\infty$ almost surely. Furthermore, for almost every sample path, the transition probabilities π_{s_t, k_t} coincide with the transition probabilities given by Π_h for all but finitely many t .

PROOF OF THE CLAIM:

The updating scheme is still given by (A1), (A2), and (A3). The transition law for Y_t is given by $\hat{\Pi}$. In particular, $\hat{\Pi}$ does not depend on $\tilde{\theta}_t$ due to our alteration of the updating process. The random variables Y lie in a finite, discrete set. Note that $\tilde{\theta}_t$ always remains in the compact set $[\underline{u}/(1 - \beta), \bar{z}]^d$, where d is the number of asymptotically active classifiers and \bar{z} is the maximum of all initial starting strengths in $\tilde{\theta}_{t_0}$ and $\bar{u}/(1 - \beta)$: by induction, if $\underline{u}/(1 - \beta) \leq \theta_{t,k} \leq \bar{u}/(1 - \beta) + \bar{\rho}$ for all k and some $\bar{\rho} \geq 0$, then $\underline{u}/(1 - \beta) \leq \theta_{t+1,k} \leq \bar{u}/(1 - \beta) + \beta\bar{\rho}$ for all k via equation (A3). We will use the theorem of Métivier and Priouret (1984), restated in the technical Appendix to this paper, which is available on request from the authors. We need to check its assumptions. With the remarks after the restatement of that theorem in the technical Appendix, the Métivier and Priouret (1984) theorem applies if we can verify assumptions (F), (M1), (M5c) and the additional assumptions listed in the theorem itself.

Assumption (F) is trivial, since f is continuous. Assumption (M1), the uniqueness of Γ , follows from the uniqueness of μ_h . For (M5c), note that $\mathbf{I} - \hat{\Pi}$ is continuously invertible on its range and that $f(\tilde{\theta}, Y)$ is linear and thus Lipschitz continuous in $\tilde{\theta}$. For the additional assumptions of the theorem, note first that $p = \infty$ in (M2) is allowed according to our remarks following the theorem in the technical Appendix, so that the restriction $\sum_n \gamma_n^{1+(p/2)} < \infty$ is simply the restriction that the sequence (γ_n) is bounded. For the conditions on the differential equation, consider ϕ as given in

$$(\phi(\tilde{\theta}))_k = \nu(k) (\tilde{\theta}_k - u_k - \beta(\mathbf{B}\tilde{\theta})_k),$$

and $\tilde{\mathbf{u}}$ and \mathbf{B} given in equations (15) and (16). Note that the differential equation

$$\frac{d\tilde{\theta}(t)}{dt} = -\phi(\tilde{\theta}(t)) = \nu \circ (\tilde{\theta} - \tilde{\mathbf{u}} - \beta\mathbf{B}\tilde{\theta})$$

is linear with the unique stable point $\tilde{\theta}_\infty$ given by (17). The differential equation is globally stable since the matrix $-\nu \circ (\mathbf{I} - \beta \mathbf{B})$ has only negative eigenvalues (note that $0 < \beta < 1$ and that \mathbf{B} is a stochastic matrix). The theorem of Métivier and Priouret (1984) thus applies with $\mathbf{A} = \theta$ and we have

$$\lim_{t \rightarrow \infty} \tilde{\theta}_t = \tilde{\theta}_\infty \text{ a.s.,}$$

as claimed.

The claim that the transition probabilities π_{s_t, r_k} coincide with the transition probabilities given by Π_h follows from the almost sure convergence to the limit. Given almost any sample path, all deviations $\tilde{\theta}_{t,k} - \tilde{\theta}_{\infty,k}$ will be smaller than some given $\epsilon > 0$ for all $t \geq T$ for some sufficiently large T , where T depends in general on the given sample path and on ϵ . Make ϵ less than half the minimal difference between the limit strengths of any two different classifiers $|\tilde{\theta}_{\infty,k} - \tilde{\theta}_{\infty,l}|$. We then have that the ranking of the classifiers by strength will not change from date T onwards. But that means that the transition probabilities π_{s_t, r_k} coincide with the transition probabilities given by Π_h , concluding the proof of the claim.

Given a candidate asymptotic attractor θ_∞ , find $\epsilon > 0$ such that 4ϵ is strictly smaller than the smallest distance between any two entries of θ_∞ . Denote the underlying probability space by $(\Omega, \Sigma, \mathcal{P})$ and states of nature by $\omega \in \Omega$. Consider the altered updating scheme as described in the claim above with $t_0 = 1$ and the given initial state. Find the subvector $\tilde{\theta}_\infty$ of θ_∞ , corresponding to the asymptotically active classifiers according to θ_∞ . We can thus find a date t_1 , a state \mathbf{Y} , and a strength vector $\tilde{\theta}$ for only the asymptotically active classifiers so that given some event $\Omega' \subset \Omega$ of positive probability, sample paths satisfy $\mathbf{Y}_{t_1} = \mathbf{Y}$, $|\tilde{\theta}_{t_1,l} - \tilde{\theta}_l| < \epsilon$ for all l , $|\tilde{\theta}_{t_1,l} - \tilde{\theta}_{\infty,l}| < \epsilon$ for all l and all t and $\tilde{\theta}_t \rightarrow \tilde{\theta}_\infty$. For any sample path $(\tilde{\theta}_t)_{t \geq t_1}$ (i.e., not just those obtained for states of nature in Ω'), find the “shifted” sample path $(\hat{\theta}_t)_{t \geq t_1}$ obtained by starting from $\hat{\theta}_{t_1} = \tilde{\theta}$ instead of θ_{t_1} , but otherwise using the same realizations u_t and states \mathbf{Y}_t for updating. This resets the initial conditions and shifts the starting date to t_1 , but leaves the probabilistic structure otherwise intact. The claim thus applies

and we have again $\hat{\theta}_t \rightarrow \tilde{\theta}_\infty$ a.s. Furthermore, given Ω' , an induction argument applied to (13) yields $|\hat{\theta}_{t,l} - \tilde{\theta}_{t,l}| \leq |\hat{\theta}_{t_1,l} - \tilde{\theta}_{t_1,l}| < \epsilon$ for all $t \geq t_1$ and all l . As a result, $|\hat{\theta}_{t,l} - \tilde{\theta}_{\infty,l}| < 2\epsilon$ for all $t \geq t_1$ and all l , given Ω' . Extend $\hat{\theta}_t$ to a strength vector $\check{\theta}_t$ for all classifiers by assigning the strengths given by θ_∞ to inactive classifiers. By our assumption about ϵ , the ordering of the strengths given by any $\check{\theta}_t$, $t \geq t_1$ coincides with the ordering of the strengths given by θ_∞ . Thus starting the classifier system learning at t_1 , strength vector $\theta_{t_1} = \check{\theta}_{t_1}$ and state $\mathbf{Y}_{t_1} = \mathbf{Y}$ at t_1 , the evolution of the strengths θ_t is described by $\theta_t = \check{\theta}_t$ for all $\omega \in \Omega'$ and we therefore have that $\theta_t \rightarrow \theta_\infty$ with positive probability. This shows that θ_∞ is an asymptotic attractor, completing the first part of the proof.

Consider in reverse any asymptotic attractor θ_∞ : we have to show that θ_∞ satisfies (17), since the consistency condition is trivially satisfied by definition of $k(s)$. Find $\epsilon > 0$ so that 4ϵ is strictly smaller than the smallest distance between any two entries of θ_∞ . Find a date $t_1 \geq t_0$ so that on a set Ω' of positive probability, we have $|\theta_{t_1,l} - \theta_{\infty,l}| < \epsilon$ for all $t \geq t_1$ and all l , and $\theta_t \rightarrow \theta_\infty$. Given the strict ordering of the strengths in θ_∞ , there is a candidate asymptotic attractor θ'_∞ which is unique up to the assignment of strength to asymptotically inactive classifiers. Given any particular state of nature $\bar{\omega} \in \Omega'$ and thus values for θ_{t_1} and \mathbf{Y}_{t_1} at date t_1 , consider the altered updating scheme as outlined in the claim with that starting value (and $t_0 \equiv t_1$ for the notation in the claim). Via the claim, $\hat{\theta}_t \rightarrow \tilde{\theta}_\infty$ a.s., where $\tilde{\theta}'_\infty$ is the subvector of the candidate asymptotic attractor θ'_∞ corresponding to the asymptotically active classifiers. Thus, the strengths in $\tilde{\theta}(\bar{\omega})$ coincide with the strengths of the asymptotically active classifiers in $\theta_t(\bar{\omega})$ for almost all ω and it is now easy to see that therefore the strengths of the asymptotically active classifiers in θ_∞ have to coincide with the strength of the asymptotically active classifiers in θ'_∞ , finishing the proof of the second part. To make the last argument precise, observe that $\hat{\theta}_t(\bar{\omega}) \rightarrow \tilde{\theta}_\infty$ except on a measurable nullset $\omega \in \Xi_{\bar{\omega}} \in \Sigma$. Note that the exceptional set is the same whenever the initial conditions θ_{t_1} and \mathbf{Y}_{t_1} are the same. Since there are only finitely many such initial conditions that can

be reached, given the discrete nature of our problem and the fixed initial conditions at date t_0 , the exceptional set $\Xi = \{(\bar{\omega}, \omega) | \bar{\omega} \in \Omega', \omega \in \Xi_{\bar{\omega}}\}$ is a measurable subset of zero probability of $\Omega' \times \Omega$ in the product probability space on $\Omega \times \Omega$. It follows that the strengths of the asymptotically active classifiers in θ_{∞} and θ'_{∞} coincide for all $(\bar{\omega}, \omega) \in \Omega' \times \Omega/\Xi$, which is a set of positive probability. Since these strengths are not random, we must have equality with certainty.

REFERENCES

- Anderson, John R.** *Cognitive psychology and its implications*. New York: Feeman, 1995.
- Arthur, W. Brian.** "On Designing Economic Agents That Behave Like Human Agents." *Journal of Evolutionary Economics*, February 1993, 3(1), pp. 1–22.
- Barto, A.; Bradtke, S. and Singh, S.** "Learning to Act Using Real-Time Dynamic Programming." Working paper, University of Massachusetts, Amherst, 1993.
- Benveniste, Albert; Métivier, Michel and Priouret, Pierre.** *Adaptive algorithms and stochastic approximations*. Berlin, Germany: Springer-Verlag, 1990.
- Berk, Jonathan B. and Hughson, Eric.** "The Price Is Right, But Are the Bids? An Investigation of Rational Decision Theory." *American Economic Review*, September 1996, 86(4), pp. 954–70.
- Binmore, Kenneth G. and Samuelson, Larry.** "Evolutionary Stability in Repeated Games Played by Finite Automata." *Journal of Economic Theory*, August 1992, 57(2), pp. 278–305.
- Blackburn, J. M.** "Acquisition of Skill: An Analysis of Learning Curves." IHRB Report No. 73, 1936.
- Börgers, Tilman.** "On the Relevance of Learning and Evolution to Economic Theory." *Economic Journal*, September 1996, 106(438), pp. 1374–85.
- Börgers, Tilman and Sarin, Rajiv.** "Learning Through Reinforcement and Replicator Dynamics." Working paper, University College London, 1995.
- . "Naive Reinforcement Learning with Endogenous Aspirations." Working paper, University College London, 1996.
- Browning, Martin and Lusardi, Annamaria.** "Household Saving: Micro Theories and Micro Facts." *Journal of Economic Literature*, December 1996, 34(4), pp. 1797–855.
- Campbell, John Y. and Mankiw, N. Gregory.** "Permanent Income, Current Income, and Consumption." *Journal of Business and Economic Statistics*, July 1990, 8(3), pp. 265–79.
- Conlisk, John.** "Why Bounded Rationality?" *Journal of Economic Literature*, June 1996, 34(2), pp. 669–700.
- Day, Richard H.; Morley, Samuel A. and Smith, Kenneth R.** "Myopic Optimizing and Rules of Thumb in a Micro-Model of Industrial Growth." *American Economic Review*, March 1974, 64(1), pp. 11–23.
- Deaton, Angus S.** *Understanding consumption*. Oxford: Oxford University Press, 1992.
- DeLong, J. Bradford and Summers, Lawrence H.** "The Changing Cyclical Variability of Economic Activity in the United States," in R. J. Gordon, ed., *The American business cycle: Continuity and change*. Chicago: Chicago University Press, 1986, pp. 679–719.
- Easley, David and Rustichini, Aldo.** "Choice Without Beliefs." Mimeo, Cornell University, 1996.
- Ellison, Glenn and Fudenberg, Drew.** "Rules of Thumb for Social Learning." *Journal of Political Economy*, August 1993, 101(4), pp. 612–43.
- Edelman, Gerald M.** *Bright air, brilliant fire—On the matter of mind*. London: Penguin Books, 1992.
- Erev, Ido; Maital, Shlomo and Or-Hof, O.** "Melioration, Adaptive Learning and the Effect of Constant Re-evaluation of Strategies," in G. Antonides, F. van Raaij, and S. Maital, eds., *Advances in economic psychology*. New York: Wiley, 1997.
- Flavin, Marjorie A.** "The Adjustment of Consumption to Changing Expectations about Future Income." *Journal of Political Economy*, October 1981, 89(5), pp. 974–1009.
- Hall, Robert E.** "Stochastic Implications of the Life Cycle—Permanent Income Hypothesis: Theory and Evidence." *Journal of Political Economy*, December 1978, 86(6), pp. 971–87.

- Hall, Robert E. and Mishkin, Frederic S.** "The Sensitivity of Consumption to Transitory Income: Estimates from Panel Data on Households." *Econometrica*, March 1982, 50(2), pp. 461–81.
- Hey, John D. and Dardanoni, Valentino.** "Optimal Consumption under Uncertainty: An Experimental Investigation." *Economic Journal*, 1987, Supp., 98(390), pp. 105–16.
- Holland, John H.** "Escaping Brittleness," in R. S. Michalski, J. G. Carbonell, and T. M. Mitchell, eds., *Machine learning: An artificial intelligence approach*. Los Altos, CA: Morgan Kaufmann, 1986.
- _____. *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence*. Cambridge, MA: MIT Press, 1992.
- Hubbard, R. Glenn; Skinner, Jonathan and Zeldes, Stephen P.** "The Importance of Precautionary Motives in Explaining Individual and Aggregate Saving." *Carnegie-Rochester Conference Series on Public Policy*, June 1994, 40(4), pp. 59–125.
- _____. "Precautionary Saving and Social Insurance." *Journal of Political Economy*, April 1995, 103(2), pp. 360–99.
- Ingram, Beth Fisher.** "Equilibrium Modeling of Asset Prices: Rationality versus Rules of Thumb." *Journal of Business and Economic Statistics*, January 1990, 8(1), pp. 115–25.
- Johnson, Stephen; Kotlikoff, Laurence J. and Samuelson, William.** "Can People Compute: An Experimental Test of the Life Cycle Consumption Model." National Bureau of Economic Research (Cambridge, MA) Working Paper No. 2183, March 1987.
- Kahneman, Daniel and Tversky, Amos.** "The Psychology of Preferences." *Scientific American*, January 1982, 246(1), pp. 136–42.
- Kahneman, Daniel; Wakker, Peter P. and Sarin, Rakesh.** "Back to Bentham? Explorations of Experienced Utility." *Quarterly Journal of Economics*, May 1997, 112(2), pp. 375–405.
- Kiyotaki, Nobuhiro and Wright, Randall.** "On Money as a Medium of Exchange." *Journal of Political Economy*, August 1989, 97(4), pp. 927–54.
- Krusell, Per and Smith, Anthony A., Jr.** "Rules of Thumb in Macroeconomic Equilibrium: a Quantitative Analysis." *Journal of Economic Dynamics and Control*, April 1996, 20(4), pp. 527–58.
- Ljung, L.; Pflug, G. and Walk, H.** *Stochastic approximation and optimization of random systems*. Basel, Switzerland: Birkhäuser-Verlag, 1992.
- Loewenstein, George and Elster, Jon,** eds. *Choice over time*. New York: Russell Sage Foundation, 1992.
- Lusardi, Annamaria.** "Permanent Income, Current Income, and Consumption: Evidence from Two Panel Data Sets." *Journal of Business and Economics Statistics*, January 1996, 14(1), pp. 81–90.
- Marcet, Albert and Sargent, Thomas J.** "Convergence of Least Squares Learning Mechanisms in Self-Referential Linear Stochastic Models." *Journal of Economic Theory*, August 1989, 48(2), pp. 337–68.
- Marimon, Ramon; McGrattan, Ellen and Sargent, Thomas J.** "Money as a Medium of Exchange in an Economy with Artificially Intelligent Agents." *Journal of Economic Dynamics and Control*, May 1990, 14(2), pp. 329–73.
- Métivier, Michel and Priouret, Pierre.** "Applications of a Kushner and Clark Lemma to General Classes of Stochastic Algorithms." *IEEE Transactions on Information Theory*, March 1984, IT-30(2), pp. 140–51.
- Moghadam, Reza and Wren-Lewis, Simon.** "Are Wages Forward Looking?" *Oxford Economic Papers*, July 1994, 46(3), pp. 403–24.
- Pinker, Steven.** *The language instinct*. New York: Penguin Books, 1994.
- _____. *How the mind works*. New York: Norton, 1997.
- Rosenthal, Robert W.** "Rules of Thumb in Games." *Journal of Economic Behavior and Organization*, September 1993a, 22(1), pp. 1–13.
- _____. "Bargaining Rules of Thumb." *Journal of Economic Behavior and Organization*, September 1993b, 22(1), pp. 15–24.
- Roth, Alvin E. and Erev, Ido.** "Learning in Extensive Form Games: Experimental Data and Simple Dynamic Models in the

- Intermediate Term.” *Games and Economic Behavior*, January 1995, 8(1), pp. 164–212.
- Rust, John.** “Do People Behave According to Bellman’s Principle of Optimality?” Hoover Institute Working Papers in Economics, E-92-10, May 1992.
- Sargent, Thomas J.** *Bounded rationality in macroeconomics*. Arne Ryde memorial lectures. Oxford: Oxford University Press, 1993.
- Simon, Herbert.** *Models of bounded rationality*. Cambridge, MA: MIT Press, 1982.
- Smith, Anthony A., Jr.** “Solving Stochastic Dynamic Programming Problems Using Rules of Thumb.” Queen’s Institute for Economic Research Discussion Paper No. 816, May 1991.
- Stokey, Nancy L. and Lucas, Robert E., Jr. (with Prescott, Edward C.).** *Recursive methods in economic dynamics*. Cambridge, MA: Harvard University Press, 1989.
- Thaler, Richard H.** *Quasi rational economics*. New York: Richard Sage Foundation, 1991.
- Thorndike, E.** “Animal Intelligence: An Experimental Study of the Associative Processes in Animals.” *Psychological Monograph*, June 1898, 2(4), pp. 1–109; excerpted in *American Psychologist*, October 1998, 53(10), pp. 1125–27.
- Tversky, Amos and Kahneman, Daniel.** “Judgment Under Uncertainty: Heuristics and Biases.” *Science*, September 1974, 185(4157), pp. 1124–131.
- Watkins, C.** “Learning from Delayed Rewards.” Mimeo, University of Cambridge, 1989.
- Zeldes, Stephen P.** “Consumption and Liquidity Constraints: An Empirical Investigation.” *Journal of Political Economy*, April 1989, 97(2), pp. 305–46.