Markov Indecision Processes: A Formal Model of Decision-Making Under Extreme Confusion

Harry Q. Bovik BOVIK@CS.BEIGE.EDU
Judy Q. Goldsmith GOLDSMIT@CS.UKY.EDU
Andrew Q. Klapper KLAPPER@CS.UKY.EDU
Michael Q. Littman MLITTMAN@RESEARCH.ATT.COM
Beige University, Department of Confusing Science, Inspiration, RI 0762-3142

Abstract

We present a mathematical model of indecisive agents faced with a sequence of difficult decisions, extending Adams' bistromathics to the multistage case. This is almost the first work on modeling stochastic processes for which the probabilities are fundamentally unknowable. This paper describes a novel algorithm, complexity results, and a model-free learning algorithm for Markov indecision processes. Two applications are discussed based on real-world domains: presidential elections and the stock market.

1. Introduction

It has been claimed that any goal- or gain-oriented human endeavor, from career choice to military campaigns, manufacturing to wildlife management, can be modeled as a partially observable Markov decision process. Furthermore, given such a model, the optimal plan, or at least one that loosely approximates the optimal, can be found.

It is our observation, over a life-time of interacting with people, that this is not the case. The last American presidential election has reinforced the belief that modeling confused human agents is beyond the scope of current techniques. Therefore, we introduce a new mathematical model that handles those situations that cannot be modeled by partially observable Markov decision processes. This model, known as a **M**arkov ind**E**ci**S**ion proces**S** (or MESS), has immediate industrial, financial, and political applications.

MESSs are all over. The earliest explicit reference to such a randomized decision-making process is the formulation of bistromathics. Bistromathics was first popularized by Adams (1982), although you can see the true roots of the idea in our own work (Bovik, 1984), published somewhat later and on an entirely different topic. Chaos Theory (Gleick, 1988) has led to models of indecision phenomena, but we wish to make use of the rich structure of indecision problems to provide more direct solutions. MESSs model systems that depend on iterated decision making for each action. Examples include scheduling and decision making, particularly of a personal or group nature; vacation and wedding planning; and naming and raising children.

In this paper, we aim to summarize everything that is known about MESSs in the technical literature, to describe our own work on the subject, and to indicate what could but won't be done in the area. In the following sections, we vaguely defines MESSs, introducing bistromathics and its application to multistage indecision problems; hint at algorithms and complexity for solving MESSs; sketch a reinforcement learning algorithm for agents faced with a MESS; breeze over two example domains based on the present system of American political indecision-making and the U.S. stock market; and, finish with a laundry list of unlikely future directions and a hastily written conclusion. Each section is worded specifically to distract the reader from the fact that we haven't done any empirical work. Rest assured that, if the paper is accepted, we will again neglect to produce compelling empirical results before the deadline.

2. Markov Indecision Processes

The basic idea behind bistromathics is that numbers, like space and time and the number of Democratic votes reported in Palm Beach County, are relative to the observer. Bistromathics was first developed to explain the seemingly inexplicable use of numbers in restaurants: arrival time, party size, and bill amount all appear to violate classical mathematical laws. An exception is that the degree of certainty with which reservations are made can often be inferred by the edit distance between the name given to the maître d' and the actual name of the person making the reservation.

The two central ideas of bistromathics are

- *nonabsolute numbers*: numerical results can only be understood after the fact and do not have any absolute meaning, and
- *recipriversexclusion*: predicting that a numerical result will have a particular value ensures that it will take some other value.

In retrospect, these ideas are fairly obvious to even the most casual observer. A baby's due date or the amount of time needed to pop popcorn in the microwave are two familiar examples. There is an obvious analogy to be made between nonabsolute numbers and random variables, but we can't really make this precise, so we'll blame our terseness on page limitations.

Bistromathics is a sufficiently powerful formalism that Adams (1982) predicted that it would be the basis of efficient and reliable space travel. Nonetheless, this model has the obvious shortcoming that it deals with numerical results of one-shot experiments only, in spite of the fact that many real-world examples are temporally extended. Next, we show how bistromathics can be extended to examples with sequential outcomes.

A MESS $P = \langle \mathcal{I}, \mathbf{I}, \mathtt{I} \rangle$ consists of a finite set of *states* $\mathcal{I}$, including a distinguished state i_C, the *state of confusion*; the set of *indecisions* $\mathbf{I}$, and the set of *inactions*, $\mathtt{I}$. The process starts in state i_C. At each timestep, a state is chosen via a prespecified multinomial distribution as a function of the current state i, indecision i, and inaction $\mathtt{i}$. Then, the process goes pretty much where it wanted to go anyway. This is known as a *state transgression*. After a series of state transgressions, the process reaches a final state i_F, known as the state of disbelief. There is a *level of futility* (also known as the frustration value) associated with each state and each possible inaction. This value summarizes how pointless it is to be in that state.

The basic computational problem for an agent facing a MESS is deciding what the heck to do. The goal of the agent is to optimize the net futility over time, as measured by the so-called *futility function*. Because this function is fundamentally unknowable, the agent must decide after the fact why what it did was actually for the best anyway. This is known as *retrooptimal policy determination*, or rationalization. On the other hand, observers must decide why what the agent did was precisely the wrong thing to do. This is sometimes called *Monday-morning quarterbacking*.

A secondary problem that arises in the implementation of a MESS is to find a clean representation of the system and the indecisions that naturally arise in the course of each rationalization. We show, in Theorem 3.1, that this problem is fundamental and hard. For instance, it has been observed that, in distributed election processes, the presentation of choices to an agent has a significant impact on its indecision.

3. Algorithms

Algorithms for finding strategies for dealing with MESSs typically depend on bistromathics to find a strategy that optimizes the futility. We propose the following algorithm, called the **S***loppy* **L***ookahead-l***O***ok***B***ehind* or SLOB algorithm. Before making any important decisions, the agent must go through the set of inactions and "mark off" those that do not appear to have been taken. This gives rise to the popular nomenclature, "Mark-off decision processes."[1] The look-ahead phase of this algorithm requires the ability to generate plans and to test them with focus groups. This leads to a certain vagueness in the immediate plan, best summed up by the strategy, "We'll cross that bridge when we burn it."

Finally, after the computation has completed, all considerations are ignored, and inaction is chosen according to prior preferences. Our algorithm is completely general, as shown by exhaustive testing of up to 5 states. The results are described in Figure 1, which shows the futility of the policy found as a percent of the optimal futility. After eliminating the outliers, (also known as *standard deviants*) it is clear that the data is approximated by the curve shown, which converges to the optimal futility.

We could compare the performance of our algorithm to that of *Frenetic Algorithms*, since this makes our approach look good. Frenetic algorithms, invented by Bovik (1990) to pad his C.V., arise when students in computer science classes, at a loss as to why their programs don't run, anxiously insert random statements. The resulting code gyrates visibly, often causing smoke to emerge from disk drives and wireless networks to reconfigure themselves, and produces output bearing only random relation to the specifications.

To appreciate the fundamental difficulty of the prob-

[1] The Russian spelling "Markov" is often used to avoid confusion with January department-store sales.

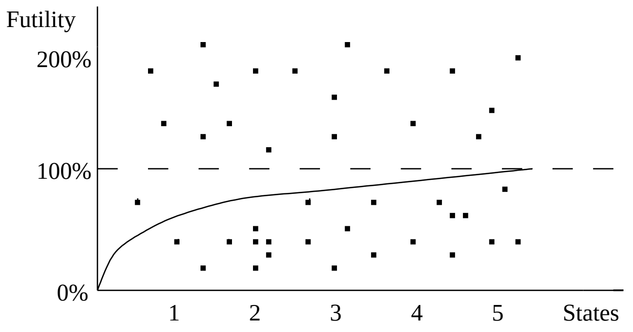

Figure 1. Experimental results: Futility as %optimal

lem we are solving, we next present a complexity analysis. We refer disinterested readers to Rudin (1974) for an introduction to complexity. For something simpler, we recommend Papadimitriou (1994). For important background information on restaurants and bistros, see the work of Schank and Abelson (1977) and Child et al. (1979).

There are three main distinctions in complexity theory: polynomial (P) vs. not polynomial, nondeterministic (N) vs. not nondeterministic, and space (SPACE) vs. cramped. These can be combined in Chinese-restaurant menu fashion into eight tenuously related classes like "NP" (nondeterministic polynomial), "PSPACE" (not nondeterministic polynomial space), and "" (cramped not nondeterministic not polynomial).

Theorem 3.1 *The problem of cleaning up a MESS is N-complete.*

This means the MESS problem is equivalent to the class N of nondeterministic time problems Gorey and Janson (1979). In lay terms, this means we have absolutely no idea how long it takes to solve them. Other problems in this class include picking up your film from the "one hour" photo place, and getting money back from an insurance company.

The proof of Theorem 3.1 follows immediately from some trivial observations that are too complex to describe here. There are also some clever widgets.

In other work, we have also shown (Bovik & Alia, 2073) that MESSs are inapproximable, and that the worst-case behaviour is better than the best-case behaviour by a factor of two, too.

4. Reinforcement Learning in MESSs

Faced with a MESS, most agents are at a loss as to what to do. It is quite natural to combine the indecisiveness of agents with the power of machine learning to create an agent that can efficiently learn how to be confused.

Clue learning is an algorithm we developed for solving the reinforcement-learning problem (Sutton & Barto, 1998) in Markov indecision processes. It works by computing a series of approximations to the futility function. The agent makes decisions in the world, and when things go wrong, instead of getting mad, the algorithm gets even; it increases the futility in the currently active states according to

$$F(\mathrm{i}, i, \mathtt{i}) = (1-\alpha)F(\mathrm{i}, i, \mathtt{i}) + \alpha\gamma\beta\eta F(s', i', \mathtt{i}')$$

where F is the approximate futility function, and α, γ, β, and η are a bunch of Greek letters.

Convergence of clue learning follows because all these algorithms seem to do the right thing if you are willing to wait forever (Littman & Szepesvári, 1996).

Results are summarized in Table 1, which demonstrates our familiarity with mathematical formulæ and the tabular environment.

5. Example Domains

The best real-world examples of Markov indecision processes are anything reported on television news programs. This section will discuss two: election outcomes and the stock market.

A common feature of these domains is how *pundits* respond to each *story* as *it* unfold*s*. In the initial state i_C, pundits are generally in loud disagreement as to how the process will play out. Each pundit makes his or her own indisputable predictions, each of which is disputed by the other pundits. After each state transgression, the pundits revise their statements, until "convergence" is reached weeks after the final outcome has occurred. At this point, pundits make it clear that the final outcome was obvious all along.

5.1 Iterative Counting

In this section, we lay the groundwork to ask for funding from the Federal Election Commission by viewing our recent federal election as a big MESS. It is painfully obvious that there is an indeterminacy-preserving isomorphism between MESSs and U.S. federal elections. Because it is commonly agreed that the process is intractable, we envision a rich source of research problems in defining election-related indecision

Table 1. An Unset Table.

CONDITIONS ON M, Q	CASE	PARAMETERS	NUMBER OF OCCURRENCES
$M = 0$	1	$t = 0$	$q^{m-1} + \epsilon(q-1)q^{m/2-1} - 1$
	2	$s \neq 0,\ t = 0$	$q^{m-1} - \epsilon q^{m/2-1}$
	3	–	$q^n - q^m$
$M \neq 0,\ \rho = 0$	1	$t = 0$	$q^{m-2} + \epsilon(q-1)q^{m/2-1} - 1$
	1	$t \neq 0$	q^{m-2}
	2	$s, t \neq 0$	q^{m-2}
	2	$s \neq 0,\ t = 0$	$q^{m-2} - \epsilon q^{m/2-1}$
	3	–	$q^n - q^m$
$\rho \neq 0$	1	$t = 0$	$q^{m-2} - 1$
	1	$t \neq 0$	$q^{m-2} + \epsilon q^{m/2-1}$
	2	$s \neq 0$	$q^{m-2} + \epsilon\eta(4\rho s - t^2)q^{m/2-1}$
	3	–	$q^n - q^m$

problems and a never-ending series of papers making incremental improvements.

The election MESS consists of a finite set of states, say 50 of them. The state of confusion in this case is sometimes referred to as "Florida".

Figure 2 is a rough approximation of the indecision process of deciding an election.

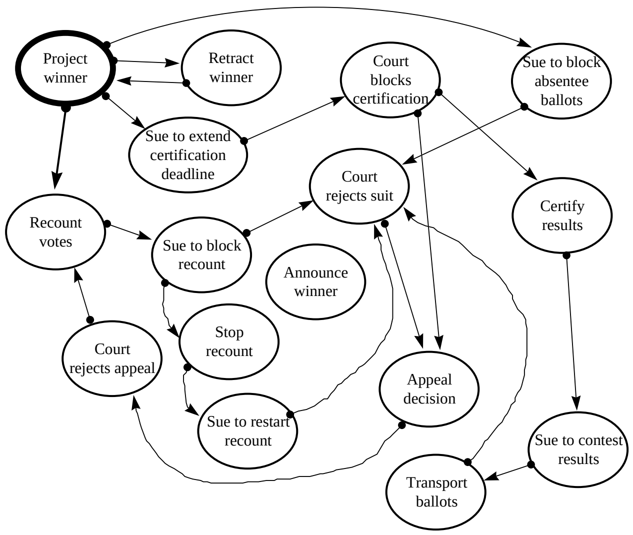

Figure 2. The Florida MESS.

5.2 You can pick your stocks and you can pick your friends...

The motions of the stock market are clearly too complex to be accurately modeled by any standard uncertainty model such as hidden Markov models, partially observable Markov models, stealthy Markov models, or even subversive Markov models.

Our highly detailed model of the stock market is pre-

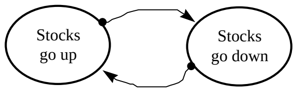

Figure 3. The stock market MESS.

6. Conclusions

There is a galloping cottage industry on decision clustering in MESSs. For instance, it is speculated that sets of "trap states" can be identified, from which there is no migration. In many cases, these are identified with the confusion state i_C. However, in the stable-state case, there may be a significant reduction in the number of meta-states considered. Unfortunately, determining the stability of partitions of MESSs has been shown to be hard for the complexity class $\mathrm{NP}^{\mathrm{PP}^{\#P}}$ (Goldsmith & Sloan, 2000; U. S. Bureau of the Census, 1998), and thus this approach is deemed infeasible by those who cannot fease probabilistic relativized oracles.

The model presented here is highly non-linear. Future work could include an extension to the tree-structured hierarchy of redundant indecision hierarchies. Furthermore, this hierarchy could be extended to consider branching tree-structured models. Nonetheless, this direction is unlikely to retain our attention long enough to ever be carried out.

The indecision process phenomenon can also be modeled by Petri nets, at the cost of an exponential blowup in processors. However, this model could lead to immense speed-downs, due to paralyzation.

Continuous state indecision processes can be modeled using elementary catastrophe theory (Arnol'D et al., 1999). (We haven't actually read this book, but it sounded cool so we listed it in the hope that one of you will read it and explain it to us.) Such a continuous state system fluctuates through a near stable set of states at a certain energy level, then suddenly makes a "snap decision," jumping to a new energy level. Typically, the phenomenon is repeated, with the system eventually returning to the original energy level. The phenomenon is known as "vacillation" and can lead to starvation in the restaurant setting.

On the basis of a recent lunchtime discussion, we conjecture that there is a close relationship between MESSs and chaotic sequences (Kohda & Tsuneda, 1997). This would immediately imply cryptographic applications for our work (Ritter, 1991), but we are not allowed to discuss those here without fear of reprisal from the federal government. Our work has nothing to do with Formal Group Laws (Klapper, 1989). Such an approach, however, has significantly increased our opportunities for funding. We are also investing in a large number of PlayStations on which to implement a distributed version of our algorithms and practice "Crash Bandicoot".

Acknowledgements

We would like to thank those federal election officials and Wall Street analysts without whom this paper would not have been necessary.

References

Adams, D. (1982). *Life, the universe and everything.* Harmony Books.

Arnol'D, V. I., Afrajmovich, V. S., Il'Yashenko, Y. S., & Shil'nikov, L. P. (1999). *Bifurcation theory and catastrophe theory.* Springer.

Bovik, H. Q. (1984). Instructions for authors. *Proceedings of the International Conference on Machine Learning* (pp. –6–0).

Bovik, H. Q. (1990). Frenetic algorithms: Students as random program generators. *Journal of Artificial Research*, π, 65–31.

Bovik, H. Q., & Alia, E. (2073). *You can't come near this mess* (Technical Report -17). Beige University, Inspiration, RI.

Child, J., Bertholle, L., & Beck, S. (1979). *Mastering the art of French cooking*, vol. 1. Alfred A Knopf.

Gleick, J. (1988). *Chaos: Making a new science.* Penguin. (Cheapest if you get it from Amazon).

Goldsmith, J., & Sloan, R. H. (2000). The complexity of model aggregation. *Proceedings of the Fifth International Conference on Artificial Intelligence Planning Systems (AIPS)* (pp. 122–129).

Gorey, M. R., & Janson, D. S. (1979). *Computers and inexorablity: A guide to the theory of N-completeness.* Freeman.

Klapper, A. (1989). Selmer group estimates arising from the existence of canonical subgroups. *Compositio Mathematica*, *71*, 121–137.

Kohda, T., & Tsuneda, A. (1997). Statistics of chaotic sequences. *IEEE Transactions on Information Theory*, *43*, 104–112.

Littman, M. L., & Szepesvári, C. (1996). A generalized reinforcement-learning model: Convergence and applications. *Proceedings of the Thirteenth International Conference on Machine Learning* (pp. 310–318).

Papadimitriou, C. H. (1994). *Computational complexity.* Reading, MA: Addison-Wesley.

Ritter, T. (1991). The efficient generation of cryptographic confusion sequences. *Cryptologia*, *15*, 81–139.

Rudin, W. (1974). *Real and complex analysis.* McGraw-Hill.

Schank, R., & Abelson, R. (1977). *Scripts, plans, goals and understanding.* Lawrence Erlbaum.

Sutton, R. S., & Barto, A. G. (1998). *Reinforcement learning: An introduction.* The MIT Press.

U. S. Bureau of the Census (1998). *Census 2000 complete count handbook for local governments* (Technical Report Publication No. 98-D230-P). U. S. Government Printing Office.