The Political Methodologist

Newsletter of the Political Methodology Section
American Political Science Association
Volume 10, Number 1, Fall, 2001

Editor: Suzanna De Boef, Pennsylvania State University
sdeboef@psu.edu
Editorial Assistant: Chan H. Nam, Pennsylvania State University
cnam@polisci.la.psu.edu

Contents

Notes From the Editor

Welcome back to *TPM*! While we've added a few heading styles, very little has changed since *TPM* last appeared some two years ago. The newsletter will arrive in your mailbox twice a year (early December and April) and will contain articles on topics related to teaching, research, section news, and current methodological debates, all interlaced with that special brand of methods humor. In my inaugural issue, the teaching section focuses on teaching undergraduate methods – a thankless and often unsatisfying task, if you poll faculty. Contributors suggest how we might think about what undergraduate methods courses should do and offer suggestions for texts and assignments. The collective wisdom here goes some way toward making the job more satisfying and maybe even enticing.

The research section focuses on computational modeling as an approach for studying politics. I was motivated here in part by Ken Bennoit and Michael Laver's

Fission and Fusion paper presented at the summer methods meetings (see `http://web.polmeth.ufl.edu/papers.html`). But more than that, I was struck by the nebulous nature of the term "computational modeling" in the discussions that followed Ken's presentation. Given that I had just agreed to edit *TPM*, I decided that the easiest way to gain a better understanding was to ask those doing this work to contribute to *TPM*. Ken, Lars-Erik Cederman, Charles Tabor and Emily Clough responded with great enthusiasm. Continuing with this focus on computation, this issue also includes an article by Phil Schrodt on using the National Center for Supercomputing Applications.

In response to a suggestion from Jeff Gill and my selfish desire to nudge a couple of my coauthors toward LaTeX, *TPM* will include a regular "LaTeX Corner". We begin by offering an introduction to LaTeX by Chan Nam – LaTeX guru for *TPM* – that discusses what it is, how to get it, and some basic features to help minimize the costs of learning LaTeX. This is the first of a new series of features on LaTeX. In future issues we plan to offer suggestions and guidelines for users at all levels of expertise. To ensure success, please submit tips and share your LaTeX wisdom.

There is a lot of section news; we have a lot to be proud of! *PA* has moved to Oxford University Press and the summer methods meetings were a huge success. Panel attendance was way up at APSA, highlights of the section meeting included the presentation of three section awards and the announcement of the University of Washington as the site for the next summer meetings. Details on all of these are included in this issue.

Finally, I can't end my introduction without an appeal for contributions to the next *TPM*. I'm particularly interested in papers that discuss what a graduate methods sequence should look like and thoughts on the "first" course in graduate methods, including book and software reviews. George Krause reviews Jeffrey Wooldridge's new text *Introductory Econometrics: A Modern Approach* in this issue. Many new texts have been published in the last 2 years and I hope to print many more reviews in upcoming issues. Also, I would like to publish reviews of the new software out there, among them Stata 7.0 and S-Plus 6. Let me know what you would like to see in *TPM* and if you are willing to write for *TPM*.

Suzanna De Boef

Teaching Undergraduate Methods

Teaching Undergraduate Methods

Greg Adams
Carnegie Mellon University
gadams@andrew.cmu.edu

Most faculty either love or hate to teach research methods. I have counted myself in both camps, depending on the success of my latest methods class. I walked into my first effort - teaching a required research methods class to 120 undergraduates - with the missionary zeal of a young faith healer preaching to a packed house. By the end of the semester, though, I was a hollow shell of my former self, sapped of all energy and aged beyond my years. My only solace was the shadenfreude visited upon me by the misery of other faculty later given the same task.

Happily, the past few years teaching undergraduate methods have been much more rewarding. Students enjoy the course, and I have regained my enthusiasm for the subject. Recognizing that success can be achieved through many different paths, the following tips are offered as suggestions, based upon my own trials and errors. What has worked for me may not work for others, but perhaps readers might avoid repeating some of my mistakes.

Set your goals before you plan

Most undergraduate methods courses have the goal of turning students into sophisticated consumers of research. After all, most of the students will not become researchers themselves, but all of us are research consumers. At first, this goal seems fairly easy to achieve, but I used to unwittingly undermine it with complex derivations or details that consumed valuable class time and dissipated student interest. Now when I plan my course and lectures, I am more conscious about devoting time to covering topics in proportion to their relevance to the goal. For undergraduates, this means more time on intuition, inspiration, and application, and less time on derivations and formulae. Certainly derivations have their place, but my tendency is to go overboard. I have yet to see a student request a derivation to resolve their confusion.

Emphasize application over formulae

All of my students have had between two and three semesters of statistics before taking my methods course. Most have done well in these courses. And yet none of my students seem to have retained anything from their prior statistical training. Not even the most experienced students can adequately explain what a t-statistic or p-value is, even though most of them have calculated such statistics dozens of times. No doubt this is partly because inferential statistics is difficult to grasp and quickly forgotten if not regularly practiced. But my impression is that in most statistics courses students learn which formula to apply and then exercise their algebraic skills. Because my goals are different from those skills, I rarely have students plug numbers into formulae anymore. However, I do have them repeatedly perform and explain for a general audience the meaning of various statistical outputs. I find this to be more challenging for students than picking the right formula. To make sure the lesson sticks, I have students analyze datasets and summarize the results every single week.

Minimize jargon and typologies

The current research design textbook for my class, Frankfort-Nachmias and Nachmias (2000), mentions at least eight different kinds of "validity," each presented in bold font or italics. Although I've always abhorred excessive jargon, I used to quiz my students on a few esoteric terms as a way of forcing them to do the readings. Now I see that forcing attention upon the trivial usually obstructs their learning what is more important.

Understand that most people are not innately scientific

Asking good questions, constructing theories, and generating hypotheses are not innate skills for most people, and I am surprised that methods courses tend to devote so little time to developing these skills. Lave and March's (1975) book is the best I have seen at getting students to think like scientists. It has been out of print for some time, but University Press of America is republishing it in December, 2001. Chapters two and three reliably change the way my students think, and I make it required reading for graduates and undergraduates alike.

Get students involved in real research

Besides writing their own research papers, I require that students attend at least one research talk presented by a faculty member or visiting speaker, preferably late in the semester when they can better understand the talk and appreciate the issues raised by the audience. Also, I have students replicate analyses in published articles. In addition to giving them a sense of accomplishment and an instant check on whether they are running the analyses correctly, replications force the students to read quality articles that exemplify how to present and summarize results. The trick is finding articles that are simple yet interesting. Listed below are three articles that are easily approachable, illustrate different research methods, and entail different statistical procedures. Interested readers can approach the authors for the data or ask me for where they can be found on the internet.

Babcock et al. (1991). *Discipline:* Economics. *Method:* Laboratory Experiments. *Statistics Used:* cross tabulations, chi-square, difference of means, difference of proportions, probit.

Kuklinski et al. (1997). *Discipline:* Political Science. *Method:* Surveys, with randomized question effects. Statistics Used: cross tabulations, difference of means.

Breault. (1986). *Discipline:* Sociology. *Method:* Quasi-Experimental Analysis. Statistics Used: correlations, bivariate regressions, multi-variate regressions.

Plan ahead

A colleague once described teaching large courses to be like being the captain of a supertanker: everything needs to be planned well in advance, and sudden changes are virtually impossible. This is good advice for even small methods classes. My courses have been most successful when all the assignments and datasets were developed before the first day of class. If a class is conducting surveys or running experiments, allow time to go through IRB approval. Failure to do this last item almost totally derailed my course the first time I taught!

Good luck

Research methods is one of the toughest teaching assignments in academia, but it can also be one of the most productive. More than any other course, methods has allowed me to observe the intellectual growth in my students, and that (coupled with the gratitude of the rest of the faculty in my department) is reward enough.

References

Babcock, Linda, George Loewenstein, Samuel Issacharoff, and Colin Camerer. 1995. "Biased Judgments of Fairness in Bargaining," *The American Economic Review* 85: 1337-43.

Breault, K. D. 1986. "Suicide in America: A Test of Durkheim's Theory of Religious and Family Integration, 1933-1980" *American Journal of Sociology*, Vol. 92, No. 3, pp. 628-656.

Frankfort-Nachmias, Chava and David Nachmias. 2000. *Research Methods in the Social Sciences, Sixth Ed.* New York: Worth.

Kuklinski, James H., Paul M. Sniderman, Kathleen Knight, Thomas Piazza, Philip E. Tetlock, Gordon

R. Lawrence, and Barbara Mellers. 1997. “Racial Prejudice and Attitudes toward Affirmative Action,” *American Journal of Political Science* 41: 402-19.

Lave, Charles A. and James G. March. 1975. *An Introduction to Models in the Social Sciences.* New York: Harper & Row.

If We Only Knew Then What We Know Now: A Few Reflections on Teaching Undergraduate Quantitative Methods Courses

Rosalee Clawson
clawson@polsci.purdue.edu

Aaron Hoffman
hoffman@polsci.purdue.edu

James A. McCann
Purdue University-West Lafayette
mccann@polsci.purdue.edu

In one respect, methods courses are no different from any other courses political science faculty teach: once course topics are selected, the central question is how to convey that material to students effectively. Yet experience tells us that methods courses present students with different challenges than courses on international relations, the presidency, political behavior, and most other offerings in our department. In this short essay, we discuss some of the problems students have in our methods courses and some of the strategies we have devised in response.

One of the big differences we see between methodology courses and other courses in our department is that grading distributions in the former are far more “bell-shaped” than they are in the latter. As many students get D’s and F’s as A’s and B’s. Moreover, past performance does not seem to be a reliable predictor of success: “A” students often have more trouble in methods courses than in their more substantive courses. Why is this? The material does not strike us as inherently more difficult to learn. Indeed, understanding concepts like *central tendency, dispersion, and confidence interval* might be easier than plowing through readings on, say, the rationality of voting.

We have two hunches about why students seem to have difficulty in methods classes. First, the number of undergraduates in our classes who seem to be math-phobic is surprisingly large. Even though we stress each semester that the course is principally about strategies for exploring political research puzzles and coming to reasonable judgments, many in the class freeze when lectures turn to empirical modeling – especially if an α or β should pop up.

Moreover, students often look too hard for and expect cut-and-dried information. “Isn’t a statistics class all about plugging numbers into a formula?” someone may ask. If an undergraduate has gotten a high GPA in previous courses by memorization and recitation (alas, such a thing is possible!), he or she cannot help but be put off by a well-taught methodology class. After all, setting up research projects and using statistical reasoning to make inferences are hardly cut-and-dried processes. One of the most fascinating, though immensely challenging, subjects we cover in the first few weeks of the semester is the transformation of theoretical arguments into empirically testable hypotheses. There simply is no recipe for this.

In general, we suggest that instructors can make their students feel more “at home” with methods by addressing these issues early. Avoid Greek letters at first, refer instead to the underlying concepts, even if it gets you tongue-tied (e.g., say “the central tendency within this population of voters” instead of μ) and tell your students that you are not really doing “math” or “statistics” – therefore they do not have to be afraid. After working with statistical concepts and methods for years, we tend to forget how baffling they are to students who face them for the first time. Remind students that it is more important to understand the logic of statistical thinking than memorize notes. Methods courses should prepare students to ponder new puzzles rather than train them to recall answers to old problems.

We also have some more specific suggestions. Once you explain basic concepts to students, it is critical that you demonstrate how those concepts are applied. Unfortunately, locating interesting, informative, and timely applications is one of the most challenging and time-consuming aspects of teaching methods. Since much of the research published in political science journals is too sophisticated for use in basic methods classes, professors often must look elsewhere to find good examples. Culling newspapers, magazines, and the World Wide Web for accessible examples is one possibility. Another option is to hunt down academic data sets that you can use to generate examples. But buyers beware: actual political events and real data often do not allow concepts to be illustrated unambiguously. For example, how often do real data clearly form a bimodal distribution?

It is also critical that faculty prepare materials (e.g., homework assignments, in-class group work, etc.) that encourage students to put the concepts discussed in class to work themselves. Fortunately, designing pedagogical materials to help students apply statistical principles is not as difficult as generating examples for lectures. The easiest way to develop assignments is to order exam copies of every methods textbook you can get your hands on, including texts from other disciplines. Use these texts to get ideas about effective homework assignments, computer exercises, and in-class work. Remember, however, that you must update assignments periodically. That clever homework assignment using public opinion data about the Clinton impeachment does not seem nearly as interesting in light of public support for President Bush since September 11th.

Teaching research methods to undergraduates can be daunting, particularly when one teaches it for the first time. And while we stressed a number of problems professors should be aware of, we want to conclude by reminding our readers that, above all, the key to a successful methods course is to recognize that they can be a pleasure to teach. Methodology courses allow professors to indulge their interests in topics that might not fit comfortably in their other courses. Have a sudden interest in the dynamics of international terrorism even though you teach American politics? Want to know more about "Eurosclerosis," but can't find a free week in your International Relations course? Methods classes are for you. However, keep this secret to yourself: the chair does not need to know how much fun you are having while you "grudgingly" agree to teach methods "as a service to the department."

Teaching Undergraduates About the Quantitative Study of Politics

Marie Hojnacki
Pennsylvania State University
marieh@psu.edu

In spring 2002 I will teach *Quantitative Political Analysis (QPA)* at Penn State for the sixth time. *QPA* is designed to introduce both political science majors and non-majors to some of the basic methods of statistical analysis used by political scientists. The course is not a requirement for political science majors.

The most significant change that I have made to the course over time has been to expand the amount of time I spend on research design. I had always devoted a few weeks to research design topics. But increasingly I found this to be insufficient. The reason for this is simple: students cannot really grasp how and why techniques of statistical analysis are used unless they have a relatively good grasp of the methods of social science research. Students' abilities to make effective use of the statistical techniques we cover in the course (e.g., the usual array of descriptive and inferential statistics beginning with measures of central tendency and dispersion through basic multivariate regression) are enhanced considerably by spending more time on the principles or methods underlying the quantitative study of politics. I now devote about one-fourth to one-third of the semester to topics such as how research questions are developed and narrowed, how concepts are measured, and how hypotheses are developed and stated; the rest of the class is devoted to statistical analysis. Spending additional time on these elements of social science research has turned out to be, in my view, essential to achieving the primary goals I have for the course: to provide students with a sense of how, as observers of politics, they can use empirical information; and to help them understand how answers to questions about politics can (and cannot) be obtained through various forms of quantitative analysis.

Aside from changing the balance between research design and statistical analysis, I have tried to achieve my goals for the course by minimizing the amount of math and computation I use in class, and by using active learning exercises as much as possible. I do not mean to suggest that I do not present formulas for the statistics I introduce. Of course I do. However, I do not require students to memorize formulas, and in my presentations I emphasize what information the formula is designed to help us obtain (e.g., z-scores can help us to see how usual or unusual two midterm elections are in terms of congressional approval and seat gain or loss even though seats and approval points are different units).

Exercises

As much as possible I get students actively involved in learning about the techniques and tools we are covering. A few versions of this active learning have been especially effective. Once we move to the statistical analysis portion of the course, students are asked to analyze and interpret political data as part of in-class exercises. I supply the data, pose the questions, and (sometimes) describe the analysis I want them to undertake, and they provide they interpretation. Depending on the data source used, students might pose as political consultants for members of Congress, or analysts for state social service agencies (many of these hands on assignment ideas were passed along to me by my graduate student colleagues at Ohio State – especially Chuck Smith). This assignment illustrates for students how the questions that politicians and government decision makers are interested in can be addressed with the tools they are introduced to in class.

Another exercise that has proved useful is one that comes after a few class sessions devoted to developing research questions and hypotheses. In order to get them to start thinking about what makes some questions more interesting than others, students are given a data set containing information about the gender and racial diversity of various elected officials in each state (the information was drawn from Stanley and Niemi's *Vital Statistics on American Politics*). In groups, they are asked to come up with one or two research questions and hypotheses that might be addressed with these data. Each group is then asked to assess what they might learn from studies that focused on the research questions posed by the other groups, and also why some questions and hypotheses seemed more interesting to them. I have found that when students have to articulate what makes a question interesting, or what they might or might not learn from a given question, they are more likely to internalize the criteria for strong question and hypothesis generation.

A third exercise has been very effective in illustrating the concept of sampling distributions, and in showing the differences between populations and samples. Each student is asked to collect some information about a small number of members of the House of Representatives (e.g., the percentage of the vote they received in the last election, their partisan affiliation, the number of terms they served). Students enter the data and then, as a class, we first look at statistics from the individual samples, and later combine the data sets and look at statistics from the overall population (of sorts). We compare means from the overall samples to means from the individual student samples, and we also draw random samples of different sizes from the full set of House members and compute statistics from the different sized samples. With exercises such as this one, students are able to participate in what otherwise would exist for them only on a theoretical level, and they take a few steps beyond descriptive statistics toward inference.

Software and Texts

For most of the time I have taught the course, students have undertaken data analysis using *SPSS*. The students like it (as much as they like any statistical software) and *SPSS* allows them to engage in every form of analysis they learn about in the course.

I have not been that pleased with the books I have used to teach *QPA*. In fact, I seem to change the book almost every time I teach the course. (Indeed, I will be using a different book for the fifth time when I teach the course in the spring.) Overall, my dissatisfaction with these books stems from some combination of three things: an inappropriate balance between math and substance, a lack of political science examples, and an insufficient emphasis on research design. Knoke and Bohrnstedt's *Basic Social Statistics* was too math focused (and the version I used back in the mid-1990s was riddled with typographical errors, which students become impatient with quickly). Levin and Fox's *Elementary Statistics in Social Research* was a bit less math-focused than *Basic Social Statistics* and it has excellent and extensive examples of computations. However, the link to political science was not strong, and I wasn't all that interested in devoting time to having students compute statistics by hand. The next book I used was Champney's *Introduction to Quantitative Political Science.* In my opinion, despite its political science focus, its coverage was not extensive enough for a primary text. Most recently I used Sirkin's *Statistics for the Social Sciences, 2nd edition.* I liked its use of political science examples, and its balance between substance and math in presenting statistical techniques. However, I decided not to use it again because of its limited coverage of research methods. Of course, I could use a supplementary text to handle that aspect of the course, but instead I have decided to try Johnson, Joslyn, and Reynolds' *Political Science Research Methods, 4th edition.* This decision results from the greater emphasis I am giving to research design while also dealing with statistical techniques.

Teaching Research Methods: The Best Job in the Department

Kenneth Janda
Northwestern University
k-janda@northwestern.edu

This paper is based on verbal remarks at the Roundtable on Teaching, Midwest Political Science Association, Chicago, April 19, 1996[1].

If any readers recognize my name, they are likely to place me in American politics (for co-authoring a leading textbook in the field[2]) or in comparative politics (for my studies of comparative political parties[3]). They are not likely to regard me as a methodologist.

Yet, over more than forty years of teaching at Northwestern University, I have primarily taught undergraduate and graduate courses in research methods: elementary and intermediate statistics, methods of data collection, content analysis techniques, assorted computer methods of analysis and information processing, logic of inquiry, and so on. I have done this mainly by personal choice, not departmental necessity.

In the spirit of David Letterman, I can cite seven top reasons why teaching research methods (including statistics) is the best job in a political science department:

7. You don't have to update your notes after every election.

[1] Other roundtable participants discussed these topics: Lawrence Baum (Ohio State), "Teaching Large Classes"; William McLauchlan (Purdue), "Some Challenges and Opportunities of Distance Education in Political Science"; Jerry Goldman (Northwestern) "The Multimedia Lecture: From the Lunatic Fringe"; and Beth Henschen (Albion) "Preparing Future Faculty: Programs in Professional Socialization." The Roundtable was organized and chaired by Ed Sidlow (Eastern Michigan).

[2] Kenneth Janda, Jeffrey Berry, and Jerry Goldman, *The Challenge of Democracy: Government in America* (Boston: Houghton Mifflin, 2002).

[3] Kenneth Janda, "Comparative Political Parties: Research and Theory," in Ada W. Finifter (ed.), *Political Science: The State of the Discipline II.* Washington, D.C.: American Political Science Association, 1993. Pp. 163-191.

6. Grading is easier (there are right and wrong answers); there is more variation in scores; students who score poorly are humbler.

5. You don't have to instill fear of failing into students; they come equipped with the fear.

4. You almost always know more than the students (no worry about political junkies).

3. Students frequently enjoy learning how to do research (imagine that!).

2. Whatever students learn, they learn in your class–so teaching is immediately rewarding.

1. Undergraduate political science majors can actually get jobs after graduation because of what they learned in political science classes.

Concerning the last (top) reason, I could cite names of graduates employed by advertising agencies; both major political parties; consulting firms; governmental agencies at the national, state, and local levels; and even business firms–such as Lands' End, Sara Lee, Sears, etc.–mainly on offering knowledge of computer methods of statistical analysis.

Based on my four decades of experience[4] —from mainframes to microcomputers—with teaching research methods, I can offer some general principles for teaching these topics:

a. Do not separate methods from substance; always link research techniques to substantive political topics. (Which means always teach methods *within* the department—despite what the Dean and your colleagues may think.[5])

b. If you teach statistics, use computer programs for the analysis of real data sets, such as the American National Election Studies or United Nations data on countries.

c. If you teach statistics, don't just have a midterm exam, but have more frequent exams to make sure that students don't fall behind.

d. If you teach statistics, teach descriptive statistics (including simple correlation and regression) before inferential statistics, which involves more abstract notions of probability.

e. Regardless of which methods you teach (statistics, content analysis, data collection), require that students actually *do* research and write papers reporting their efforts; I ask only for five pages of text–and as many tables or figures as necessary.

f. Because few students will have written such papers before, provide them with an explicit format to follow; I use these headings for a 20-point paper:

1. Statement of the Problem (worth 3 points)
2. Research Design and Hypotheses (worth 7 points)
3. Data Analysis (worth 10 points)
4. Summary and Conclusion (worth 5 points)

g. Tell students that their papers should be organized explicitly according to these headings for they will be *graded* accordingly.

h. In an accompanying two or three page statement, tell students what you expect under each heading and provide them with sample formats for tables, citations, etc.

i. Tell them that conducting an empirical study is like building a violin; the first product is always poor, but you learn much in the process.

This last point captures the essence of the process of teaching research methods. The objective is to have the student learn methods by applying them. To cite another metaphor, it's like teaching students in a classroom how to swim. All the instruction about moving the arms and turning the head to breathe will help little unless they try it in the water.

[4] Almost four decades ago, I published *Data Processing: Applications to Political Research* (Evanston: Northwestern University Press, 1965), which was issued in a second edition in 1969. As for techniques involving qualitative analysis, see *Information Retrieval: Application in Political Science* (Indianapolis: Bobbs-Merrill, 1968).

[5] In 1964, as an untenured Assistant Professor, I proposed teaching our own statistics course in the political science department. My written proposal was approved by the college curriculum committee but opposed by the Dean, who want to "consolidate" teaching statistics for all the social sciences in one large class. (Already psychology and sociology had their own statistics courses. If political science put its own course on the books, where would this wasteful proliferation of courses end?–argued the Dean) The issue reached the floor of the College of Arts and Sciences faculty meeting, at which the Dean stepped down from the Chair and argued against my proposal from the floor. My response, in part, was that teaching should never separate method from substance. I won on the vote by show of hands. He promoted me anyway.

Teaching Undergraduate Methods: Overcoming "Stat" Anxiety

Michael S. Lewis-Beck
Visiting Professor of Government, Harvard University
michael-lewis-beck@uiowa.edu

By now, the undergraduate methods course has become a standard part of the political science major for most political science departments in US universities. But most political scientists do not like to teach the course. They may have feared the old "scope and method" notion for this course, with its review of leading theoretical paradigms and its going over of different study approaches. But the "scope and method" format is pretty much a thing of the past. Undergraduate methods courses today stress understanding tools for the scientific explanation of politics - computers, mathematics, statistics. Still, political science professors generally do not like to teach the course, even those who are accomplished quantitative researchers. Why is this?

It is because they believe, somehow instinctively, that the material must be taught in a formal way, with mathematical rigor. Thus, the student should do proofs, derivations, solve statistics problems galore, perhaps even write computer programs. Since most political scientists are not, firstly, mathematicians, statisticians, or computer experts, they feel inadequate to the task. If they do tackle the task, perhaps at the request of their department chair, then they obsess on it or they neglect it. If they obsess, they may decide, for example, that the regression line acquires meaning only when the student can grasp the calculus that minimizes the sum of squared errors. If they are neglectful, they may simply have the student run regressions from the manual and report numbers they cannot interpret. These ways do not lure undergraduates into the scientific study of politics. On the contrary, they reinforce the typical undergraduate notion that numbers have nothing to do with theories of government.

The difficulty - unhappy teachers and unhappy students - stems essentially from their "stat anxiety." Both sides are afraid they can't quite do it right, and at least one side is not sure it is worth it. The way to overcome "stat anxiety" is to make the learning "fun," or at least "painless." How? Use simple examples that pose an interesting political question, one that students can answer with real data. Introduce complexity a step at a time, gradually elaborating the examples, always returning to the political interpretations of the coefficients. In other words, show the student that statistics increase understanding of how politics works, and that they themselves can help create that understanding. The students become, at least a small way, creators of knowledge rather than mere recipients. Instead of skipping over the equations, tables and figures in articles that professors assign them, passively accepting the author's claims, they begin to be critical scientific readers.

Let me illustrate by developing a specific exercise, which could be appropriately modified across a host of research questions. Ask the class, "Do women participate in politics more than men?" The discussion allows consideration of important concepts: hypothesis formation, independent v. dependent variables, operationalization, sampling, and data-gathering. Suppose one hypothesis arrived at is as follows: "Women vote more in US presidential elections than men." A simple 2x2 table is in order, Gender (male, female) in the column variable and Vote Turnout (yes, no) the row variable, with cell entries from election survey data. By looking at percentage differences, an inference about the hypothesis can be drawn. Given the difference is small, as it likely will be, then it is appropriate to bring up the notion of a statistical significance test, to rule out chance. The turnout example can be pursued, in order to teach more complicated tables. For example, the table may become a 4x3, with the column variable Region of the country (North, South, East, West), and the row variable Relative Vote Turnout (always votes, sometimes votes, never votes). Upon examining these more complex tables, the student immediately sees it can be difficult to use percentage differences to summarize the results. That is a natural point to introduce measures of association, and to explain how different levels of measurement imply different measures of association.

Treatment of measures of association should end with the Pearson correlation coefficient, which is a nice lead into regression analysis. With interval data, scatterplots replace tables to evaluate bivariate relationships. To broaden the data example, say the hypothesis is as follows: the higher the voter's income, the more political acts - voting as well as other kinds of participation - are engaged in. Both these variables are counts, and the student could plot them. Ask the students if they can visualize a line through the points. Fit the line and discuss its closeness to the points. Indicate that the line can be expressed in an equation. Explicate the intercept and slope estimates for that equation, e.g., when income goes up by one unit, say $10,000, the expected rise in political actions is a certain amount, say 2. At this juncture, remind them that, of course, participation has more than one cause. For instance, it should also be influenced by Political Attention (measured perhaps by the number of types of media through which the respondent follows politics). This leads into the development of multiple regression and, eventually, discussions about the classical assumptions. Are they met? If they are not, what does that say about the truth of our inferences?

Different books lend themselves better to this applied, inductive pedagogy of quantitative research. To illustrate the use of each statistical technique, I try to assign a published political science article that effectively employs that technique. For data-sets, I use the latest edition of *Research Methods in Political Science* (Corbett, 2001), containing interactive exercises with real data from the American states, a sample of nations, a recent General Social Survey, a recent American National Election Survey, and recent congressional roll call votes. As one text, I assign *Data Analysis* (Lewis-Beck, 1995) or *Applied Regression* (Lewis-Beck, 1980). Clearly, these are not the only possibilities out there, and the instructor may wish to select other works. But hopefully these choices serve as a guide to the type of readings likely to reduce "stat anxiety."

References

Corbett, Michael. 2001. *Research Methods in Political Science, 4th Edition.* Wadsworth: Belmont, California.

Lewis-Beck, Michael S. 1980. *Applied Regression: An Introduction.* Sage Publications: Thousand Oaks, California.

Lewis-Beck, Michael S. 1995. *Data Analysis: An Introduction.* Thousand Oaks, California.

Making the World Safe for Methods

Peter Stone[1]
University of Rochester
prse@troi.cc.rochester.edu

Who could pass up an opportunity to teach methods? I certainly couldn't when the chance arose last year. I'd completed my degree, and was looking for ways to make ends meet and spiff up my CV. The department that granted me my degree, I learned, was looking for someone to teach "the undergraduate methods course." Sounded perfect, I thought. I made plans to teach the class, and borrowed syllabi from previous instructors. Funny, they don't look much alike, I thought (the syllabi, not the instructors). It was then that I learned that my department in fact offered two undergraduate methods courses-one intensive in probability theory and statistical inference (the one I'd hoped to teach) and one more geared towards applications (the one I was scheduled to teach). Seven years at that department, and this was still news to me.

While there are many excuses I could give for not realizing just how many methodological opportunities our department offered its undergraduates, the best one is probably the simplest one. It's a rare department that could offer two undergraduate methods classes, both of which were pretty heavy-duty by the standards of the profession, and expect to see them filled.

Why would undergraduates be so reluctant to take such a class? Probably because they've never had the case put to them as to how valuable a background in methods can be. Robert Heinlein once wrote that in the modern world, a person who did not know mathematics was at best a domestic animal trained to wear shoes, bathe, and not make messes around the house. While not endorsing quite so extreme a position about methods, I do believe that a citizen lacking any exposure to basic political methodology—including research design, basic probability, and theories of inference—is missing a critically important tool, one without which it is impossible to be a fully informed participant in democratic politics. An undergraduate methods course gives an instructor a chance to get this point across to students, to make it something more than a class for which one memorizes now and forgets later. If the course is done right, even statistics may cease to be scary and start to look important, maybe even vital to politically concerned students. The world becomes a bit safer for methods, and the democracy is stronger for it.

I believe that the undergraduate methods instructor should always approach the course with this fundamental question in mind-how can I convince students that methods *matters*? The quest to answer this question successfully has many implications for how to approach a course like this. For one thing, it makes tests less important than practical applications. Graduate students may need to have the definitions of every statistical concept drilled into their heads; undergraduates need to know the general logic underlying their use more than anything else, and the best way to do this is to focus students on actual research. At the same time, instructors should remember that they do not have a crop of budding econometricians in front of them every day in class. Most students will not pursue graduate work involving methods, and no teacher should pretend they are. If the research that students conduct is to help them, it must leave them able to make intelligent and critical use of statistical materials prepared by others, not prepare them for the long road to a research-based career.[2]

This, in turn, has implications for such decisions as the choice of software used in class. If students are to perform any data analysis worth talking about, they cannot avoid using some statistical package. But at the same time, students not aiming for research careers do not need to spend valuable course time learning a complex package like SAS or even STATA. An instructor could easily wind up spending just as much time teaching such a program as teaching data analysis. If students really want to learn computer programming, they have better forums available to them to do so. I recommend a program like Microcase[3]; it's simple to the point of being idiot-proof, and it comes with numerous datasets (containing information on Congressional roll call votes, demographics for both

[1]Political Science Department, University of Rochester, Rochester, NY 14627. I would like to thank Rory Austin, Catherine Hafer, and Bill Hixon for helpful comments on this paper.

[2]Among education researchers, Roger Schank does an especially good job of stressing the need for social science instructors to avoid such mistakes. See, for example, the interview with Schank in the August 16, 1999 edition of the online journal Edge (`http://www.edge.org/documents/archive/edge59.html`).

[3]I am indebted to Mitch Sanders for introducing me to Microcase, as well as other materials used in my course. Overall, these materials fit very well with my teaching philosophy for methods.

U.S. states and foreign countries, and two different public opinion polls).[4] Students can design their own projects using these datasets while still carrying out all the steps of data analysis except the collection process itself. The program also allows students to enter datasets (with a maximum of 100 observations) of their own, should the instructor so desire.

Classroom presentation can also make plain the relevance of statistical analysis to everyday political concerns. Some instructors have their laptops present at all times in class in order to display statistical results of note. I do not see the need to be this focused on the computer; it can turn the course into a computer course, which is not what students need. However, the workshops I have held using Microcase did a good job of piquing student interest in various kinds of research questions. The only embarrassing part of the whole affair was the physical set-up of the presentation, which involved installing Microcase and then arranging for overhead display from my comically antique laptop. To the extent students have been privy to my tribulations at preparing for the workshops, they must have seen me as less of a methodological whiz and more of a classic absent-minded professor in action.

Computer presentations, however, are just one way to introduce examples to class. And examples make all the difference in the world. Graduate-level methods classes tend to focus on the statistical tools involved in data analysis, and the students who self-select into such courses tend to enjoy the techniques for their own sake, at least to some extent. As a result, it's once again easy to forget that students will not necessarily find the tools either fascinating in their own right or useful for any practical purpose. Every methodological concern should have an example attached, whether it be delivered via computer screen or blackboard.

Not every student will learn to enjoy methods, and only a handful will go on to conduct research on a regular basis. However, every student might reasonably learn how useful it is to be able to examine data with an informed mind. A methods instructor should approach each undergraduate class with this goal in mind. Only in that way can there be any hope that every department might someday offer two undergraduate methods courses, and see students ready and willing to take advantage of them.

[4]Specifically, I used Michael Corbett, *Research Methods in Political Science: An Introduction using Microcase®*, 4th Edition (Belmont, CA: Wadsworth/Thomson Learning, 2001).

Teaching Undergraduate Methods for the First Time

Sarah Poggione
University of Georgia
poggione@arches.uga.edu

I am preparing to teach an undergraduate research methods for the first time. I was told that that the methods class is not required of Political Science majors; however, it is one of several courses that can fulfill their required credit hours in theory and methods. In addition, I discovered that two separate courses, one on research design and another on primarily quantitative methods, were combined into one undergraduate research methods course. As a result, the methods class is now expected to cover research design as well as basic research methods in Political Science.

Armed with this information, syllabi for a variety of undergraduate methods courses, and the notes and books I have collected from undergraduate and graduate methods courses I had taken or observed, I sat down to write the syllabus and begin developing the course. I soon discovered that unlike other courses that I have prepared to teach, I wasn't sure about what the objectives of a research methods course should be or what material to include and what to exclude or what type of projects to assign students. I decided to back up a step and concentrate on identifying and answering basic questions that we implicitly ask and answer whenever we create or redesign a course. First, who are the likely students in the course? Second, how does the course fit in with the overall program of study for undergraduate Political Science majors? Third, what facilities are available for teaching the methods course or for students' use throughout the course?

While the answers to these questions undoubtedly differ across institutions and departments, the process of asking and answering them may be useful for those preparing to teach methods for the first time as well as those who have already taught undergraduate methods. These questions focus our attention on the connections between the structure and content of our courses and the needs for our students, their programs of study, and the available resources for teaching research methods. By consciously designing our courses to strengthen these connections, we not only help our students better understand research methods but also enable them to relate methods to their political science coursework and the discipline as a whole.

The Questions

First, who are the likely students in this course? Political science majors at the University of Georgia are encouraged but not required to take the department's course in political science research methods. This means that students who take the class may be less fearful of statistics, given that those who truly wish to avoid mathematics and statistics are not compelled to take the class. However, it also means that students may have little understanding of how research methods fit into the discipline (i.e, the relationship between research questions, theory, hypotheses, and analysis). Given that many students are likely to be unfamiliar with research design and methodology, other political science courses may not present material in terms of theories explaining political phenomena and events and evidence supporting or contradicting these theories. Consequently, I wanted to design a methods course that stressed the connection of research methods to the discipline of political science and our understanding of politics and government. One option for accomplishing this is to have students read original political science research. This provides students with an example of a completed research project that poses a research question, develops theory, tests hypotheses, and draws conclusions about how politics works.

In addition, the methods course at Georgia is an upper-level class; as a result, most students who enroll in the course have taken other upper-level political science courses. Students should have some substantive understanding of political science outside the course, but it may be likely, considering the diversity of courses offered in the department, that they do not share a common base of political science theory and knowledge. For example, not all students in the course have been exposed to the study of Congress or public opinion or international relations. Allowing students to pursue research related to their own interests within political science, with a considerable amount of guidance in formulating research questions and hypotheses, would enable students to draw on substantive knowledge from other political science courses.

Some students may have little or no previous experience in upper-level political science courses before enrolling in the methods course. Presenting some substantive material in the course would provide these students with a core or common political science knowledge to draw on for developing and testing their own hypotheses. Focusing on one or two areas of political science inquiry for assigned articles as well as course assignments provides students with this common substantive base.

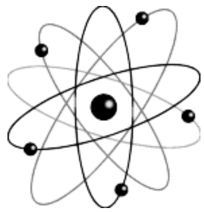
Second, how does research methods fit into the department's overall program of study for its undergraduate students? In other words, what information should students, especially political science majors, be able take from this class and how should this information fit with their overall course of study? At a minimum, students should develop an understanding of what political scientists actually do, from research question to conclusion. At Georgia there is no separate course on research design so the methods course is instrumental in teaching students about designing and carrying out political science research. This means that the course should cover aspects of research design as well as rudimentary data analysis. Fundamentally, students should understand how political scientists find and formulate interesting research questions, how they develop theories and testable propositions, how they collect and use empirical observations to evaluate their theories, and how they draw conclusions about politics from their analyses. Having students propose and eventually conduct their own original research projects over the course of the term would sever this purpose; students would gain a better understanding of what political scientists do by designing and conducting their own research.

Finally, what facilities and resources are available? At the University of Georgia, like many other campuses, the available facilities present something of a challenge when teaching an undergraduate methods course. No teaching lab currently exists for undergraduate courses in Political Science, and student computer labs do not have statistical analysis software. However, classrooms equipped with computer projectors are available. The available facilities affect both the day to day conduct of the course as well as the structure and content of course assignments and activities. Designing the course to include frequent, short take-home assignments and exercises that would allow students to practice the computer related procedures and skills that were demonstrated and discussed in class. These assignments would reinforce what students could only observe in class. In addition, these frequent assignments would also be the basis for students learning how to interpret statistical results and discuss how their results either supported or failed to support their hypotheses, reinforcing their understanding of the discipline as a science.

More Questions

These initial questions have certainly prompted others. To what extent should I try to incorporate qualitative methods into the course? What kinds of statistical techniques should I cover? How much should I emphasize statistical computing? What statistics package should I assign? What text or texts should I use? What data should I use for student assignments? The unanswered questions have proliferated. But, they have provided a framework with which to ask more questions and eventually develop a course for undergraduate students that complements their program of study in political science using the available facilities and resources.

Computational Modeling



Simulation Methodologies for Political Scientists

Kenneth Benoit
Trinity College, University of Dublin
kbenoit@tcd.ie

In this brief article I identify and characterize the variety of simulation methodologies relevant to political science research. Having identified them, I then recount a lighthearted tale of one successful application of a computer simulation to a real world problem of enormous significance. Extending the discussion to other examples, I discuss some of the main issues and benefits in simulation models for political science. Finally, I identify some software tools and additional resources for those who would like to learn more about using simulation methods for their own work.

Consider What We Already Do

The notion, and appeal, of computer simulations is simple: they enable us to build a model of a process whose workings can be examined—and more importantly controlled—as analogs to the real process being approximated. Readers of *TPM* should be very familiar with the static equivalent: presenting equations as "models" of some relationship between political variables, using parameters to characterize the relationships between these variables. The form of the equation itself—linear, exponential, logarithmic, additive, multiplicative—also imposes an assumed structure on the relationship between the variables. These mathematical equations are "models" precisely because they present a stylized representation of a more complex process, filtering out aspects of the process which it is either impossible or uninteresting to include. Whatever is not included systematically is also parameterized, typically as one or more disturbance terms, once we make tidy and fairly stringent assumptions about all these things we cannot explain. Once the model has been specified, we are then presented with a method for estimating the parameters using a statistical procedure on data collected on each variable.

Let's examine some of the global assumptions in this approach. First, in order to make it possible to estimate the parameter values, we have to impose *aggregate* assumptions on the model quantities, in particular about the distribution of the parameters, especially the disturbance parameters, and about the aggregate relationships between the variables we choose to include and exclude. We can never verify these assumptions, however, since our model will only work if they can be taken as true. When it does work the results it provides will be contingent upon, and therefore seem to reaffirm these assumptions.

A second assumption we typically make is that the observations are conditionally independent, that the values of our explanatory variables are not conditional upon the values of our dependent variable. We also assume that independent variables take on values that are independent of the values of other independent variables. (If we assume otherwise we must then alter the model further to make it effectively true.) For instance, even though we might have a model estimating strategic behavior, we assume that the units react to choice situations only and not to each other. A related assumption is the basic idea that the units being studied do not change their behavior over time: that they do not learn from their own mistakes, let alone those of others.

A final assumption is that the observations are "identically distributed," conditional of course on explanatory variables. This means that there are no differences between units we are treating as data that cannot be captured by variables and model parameters.

Of course there must also exist some practical method for obtaining reliable numerical estimates of the model parameters. There are many great quantitative models that we simply have no way of estimating in practice, although many ways exist in theory. Incredible strides in computing power have pushed this bar downward, making computation solutions possible to problems whose solutions were previously impractical. This revolution in computing power is highly relevant to the remaining discussion of simulation models, to which I now turn. Actually, to which I almost now turn, since I think a brief illustration might highlight the contrast in approaches to solving a problem where simulation might provide an answer.

Illustration: The Bus Problem

Lest the preceding comments start you thinking that one of my pseudonyms might be "Mr. Perestroika," let me point out that first, my goal was to critically examine the models we are most familiar with, so that they could be contrasted to simulation methodologies; and second, that some of my best friends are statistical methodologists. One of these friends, in fact, had an experience that illustrates a lot about the potential of simulation to crack tough modeling problems.

This friend of mine used to take the Mass Ave bus from Arlington to Cambridge (MA) every morning, and was frequently irritated to find himself arriving at the bus stop just as the bus pulled away. Instead of scheduling stops at fixed times, the bus schedule indicated only that from 7-9 am there would be buses on this route every 10 minutes. A statistician friend suggested that the mean waiting time would be minimized if a person trying to catch the bus would appear at random times, rather than at a fixed time every day, say at 8am. How, thought my friend, could he verify that this was true? First, we could model the bus being at the stop as a random variable, making assumptions about the regularity of its frequency and the duration of its stop. We could also model the arrival of the would-be passenger. We would then need to figure out a function minimizing the expected wait of the passenger for the next bus after he arrives at the bus stop. Now while proving this result analytically would be extremely easy for any of us, the proof was elusive for this friend of mine.

So after many frustrating and unsuccessful hours spent attempting to prove this result, my friend decided to change his approach. Why not conduct experiments to discover the result instead? So for the next thirty working days, he showed up to the bus stop—the one just next to the small grocery store with the green sign—at precisely 8:00 am, and recorded in his handheld computing device the time he spent waiting for the bus to arrive. Then for the next thirty days after that he did the same but randomly chose to arrive between 8:00 and 8:10, by rolling a 10-sided die which his roommate had said was from a game with the unlikely name of "Dungeons and Dragons."

At the end of the two experimental periods, this friend—let's call him "Ben"—compared the averages from the two experiments and tried to draw a conclusion. But he immediately noted some problems. First, on one or two mornings he suspected that his timing of arrival was late by about one minute. Second, he noticed that on one day he had waited for more than 12 minutes, which was not supposed to happen according to the bus schedule. Finally, during the second month of the experiment it had often snowed, slowing down traffic and apparently stretching the bus intervals. Dismayed by the lack of control over his experiment, Ben rejected his experimental results and decided to try yet another approach.

Thinking about the problems associated with his experiments, my friend realized that the essence of the problem—the efficacy of attempting to "time" the bus stop versus just showing up unplanned—was unrelated to random events like weather and oversleeping that can contaminate an experiment. So he decided to conduct a *simulated* experiment. Fortunately, Ben was an experienced computer programmer and had just purchased a new 486DX/100 running Slackware Linux 3.0. Programming his computer with crude agents for the passenger and the bus, he simulated an arrival of the bus at random times within the 10-minute interval and had the passenger arrive each morning at a fixed time. Running this 1,000 times, he recorded the waiting time for each round of the experiment. The same was done having the passenger also randomly arrive, recording the waiting times with this behavior 1,000 times. By comparing the distributions of the waiting times from the two experiments, he was then able to satisfyingly visualize and summarize the expected waiting times given each passenger's behavior. (I leave the answer as an exercise for the reader.)

This opened a whole new world of possibilities for my friend, including venture capital. Instead of starting his own firm as I suggested, however, he became obsessed with developing more complex models related to buses, including a simulation of bus passengers as agents with different forms of seating behavior. He was interested in how to explain why some buses seemed to carry more people in a more orderly fashion than others with fewer passengers. He programmed behaviors into several types of agents. The first, we will call the Arlington passenger. This passenger, knowing that the bus will pick up many additional passengers along Mass Ave, takes the last available seat in the rear of the bus, on the window side. Another type is the Porter Square passenger, picked up midway, and this passenger likes to sit surrounded by empty seats, and therefore takes the seat at the centroid of the largest available bloc of free seats. There are other types as well, such as the elderly passenger, taking the first free seat, and the Teenage Bloc of 3-5 passengers who search for a group of adjacent seats. By programming this system as agents, my friend was able to investigate the emergent behavior of seating patterns at various levels of capacity, with different levels of groups, picked up in various sequences. As far as I know, Ben is still ABD in a top-five sociology Ph.D. program.

The moral of this story is that for some problems it may be quite natural to turn to simulations as a superior methodology than either an analytical solution or experiments. For the second application involving modeling bus passengers' seating behaviors, simulation methods were used to model complex process based on individual agents operating to sets of known rules, and thereby yield insights into the dynamics of the aggregate system emerging from the behavior of the agents. The remainder of this article discusses such applications in more detail.

Enter "Simulation Methodologies"

I suppose it's high time to actually define what I mean by simulation methodologies. First let me clarify what is not included. We have all probably heard, or used, simulations in the context of statistical estimation. This is one of the great benefits of Bayesian "simulation" and related approaches: when we do not know or cannot express a distribution, we can approximate it by sampling from component distributions whose properties are known and can be expressed. The same approach is used in numerical optimization problems, where search techniques are used to explore a distribution whose global shape is unknown. Similar techniques are employed for testing convergence of parameter estimates using other iterative methods. I will not focus on these varieties of "simulation," such as Monte Carlo simulations, except to point out that the underlying approach is very similar to using simulation models to explore complex processes. In both statistical simulation and in computational simulation of complex processes, some real system is approximated by a model, this model's operation and behavior is defined by the researcher, this behavior is then produced using a computer, and observation of these results is used to yield leverage on the real system that has been approximated.

Simulation methodology broadly refers to the building of models of the world that have both inputs and outputs. Inputs are entered by the researcher, along with behaviors and rules structuring the simulation, and outputs are observed as runs of the simulation. Simulation in this context is basically synonymous with "computer simulation," since the simulation models are constructed and run as computer programs. Simulation methodologies have many variants, and I have compiled a list of the main types below. I warn however that this is is illustrative rather than exhaustive, being targeted toward political scientists and not meant as a guide to predicting the weather, extrapolating fishery yields, or perfecting nuclear warhead designs.

- Agent models. One of my personal favorites, agent models refer to the use of self-contained programs which can control their own actions based on their own perceptions of their operating environment (Huhns and Singh 1998). At the simplest level, agents are the actors in the simulations whose characteristics and range of behaviors are defined by the researcher. The researcher can determine the shape and units of their utility functions, the process and rules by which they make decisions, and whether they learn from history or from each other.

- Evolutionary models. These are either systems or agent models that are distinguished by the ability to alter their parameters or even the basic structure of the model itself in response to learning. This broad category includes rocket-science variety methods such as genetic algorithms and artificial neural networks.

- Cellular automata models. Really just stripped down agents, simulations of these types display emergent patterns based on successive iterations of rule-following behavior of individual components on a grid. The action of each "cell" on the grid is influenced by the states of its neighbors. A classic example is Conway's Game of Life.

- Systems dynamics and related models. Generally involving complex maps resembling engineering schematics, systems dynamics models focus on macro-level outcomes based on a target system described using a system of difference and differential equations. These are used to derive the future state of the system from its present state. An example would be the WORLD2 and WORLD3 models (Meadows 1974).

- Microanalytical simulation models. These model processes by shifting attention to the micro-level agents. Microsimulation follows successive generations of individual units, hoping to predict a future state given a starting state. Examples would be simulations designed to predict the effects of policy changes on some target population, such as lowering the capital gains tax.

Just as advances in computing power have revolutionized statistical modeling, advances in computing have made many simulation applications feasible that were once possible only in theory. Indeed, because of the way in which they manage complexity and uncertainty, simulation is well-suited to investigating problems for which closed form solutions are impossible, or to better understand problems whose closed form solution is uninformative.

Benefits: Simulating is Stimulating

Returning to our discussion of the "limits" of conventional statistical models, simulations offer a number of advantages. Probably the most important insight to be gained from computational modeling is the study of *emergence*. Emergent behavior "refers to a computation or phenomenon at the macro-level that was not hard-coded at the micro-level, such as when a market computes the price where supply equals demand even though no one is trying to compute the market price" (Page 1999, 4). By recreating the process using the micro-level agents, it is hoped the computational models can both explain observed emergent behavior, and investigate speculative emergent behavior when the micro-level behaviors are altered. Rather than making assumptions about aggregate-level behaviors, we can treat this aggregate-level outcome as an emergent behavior to be tested. Simulation provides the mapping of micro-level behavior to aggregate outcomes.

Formal theorists will be very familiar with this problem. Formal theory and especially game theory is structurally very similar to setting up a special sub-class of agent simulation. The key difference is that solutions are arrived at through the technique of deductive proofs, rather than actually simulating the games or behaviors that have been formalized. But what if we were to program simulations to actually *play* the games described?

For some problems where analytic solutions are impossible, this is in fact one of the most promising avenues for formal theory to follow. Conclusions about equilibriums can be derived from observing repeated plays of the game, supplementing or even supplanting purely mathematical results. Consider trying this on your students: show them a visualization of a spatial model demonstrating a chaos theorem or cycling. The intuition provided by seeing (and manipulating) the result through simulation is generally much more effective than working through formal proofs. (And not just for students.)

The flexibility of simulations also allows us to cast aside most, if not all, of the restrictions required in statistical modeling. The units we study can evolve, they can learn from past actions, they do not have to be independent, and their behaviors can be extremely complex. By making the behavior or micro-agents stochastic, simulations are well-suited to modeling the aggregate consequences of uncertainty. By examining repeated simulations and the trajectories they take, we learn not only about the outcomes but also about the dynamics of the process itself. By having access to the rules and behaviors which the simulation comprises, we can observe the consequences on outcomes of altering these rules and behaviors. In our pure environment, we control all of the factors governing the system, not subject to any of the errors, mistakes, unforeseen problems, or human or meteorological vicissitudes which might interfere with the conduct of experiments. Because simulations give the researcher ultimate control, simulations may be far better than experiments—in addition to being cheaper, faster, easier to replicate.

Examples of Applications

If you need any more convincing that computational research offers great possibilities for political science, simply consider this: our colleagues in economics are way ahead of us in simulation methodologies. Economists use computer simulations to explore the consequences of monetary and fiscal policy, commodity pricing in agricultures, the role of savings and investment on the process of capital accumulation, discrete choice models of public transportation (such as riding the bus or the train—and possibly what time to arrive at the station), global warming as affected by tax incentives—the list is quite long. An excellent summary of simulation work in economics can be found in a NSF-commissioned report on Computational Economics (Kendrick, Bergmann, Broder, David, and Geweke 1991). Economists even have a quarterly journal, *Computational Economics*[1], devoted entirely to applications, theories, and issues related to computation and simulation modeling.

[1] `http://kapis.www.wkap.nl/kapis/CGI-BIN/WORLD/journalhome.htm?0927-7099`

Political science applications also exist in respectable and growing numbers, and I have listed a few examples showing the areas of application. Once again this is only a sampling, rather than an exhaustive list. Examples include:

- the behavior of political parties in spatial elections (Kollman, Miller, and Page 1992);
- dynamic behavior of legislators in changing parties or forming new parties between elections (Laver and Benoit 2001);
- behavior of individual states or other international actors (e.g. Axelrod 1997b; Signorino 1996);
- the diffusion of culture (Axelrod 1997b; Bednar and Page 2001);
- formation of opinions and collective judgments (Johnson 1999); and
- models of social growth and resource conflict (among other social issues) (Epstein and Axtell 1996).

Other social science-type applications include analyzing traffic patterns, women's choice of birth control, aircraft engine replacement, patent renewal, regulation of nuclear power plants, school choice, decisions to marry, and retirement behavior.

Learning More

Software. Not only are there better comprehensive references to the software tools available for the implementation of simulation models, but also my extensive philosophizing about the epistemology of simulation above has edged out any room here for such a treatment. I will nonetheless mention the extraordinary Swarm Simulation System[2]. Swarm is a toolkit of code written in Objective-C, an object-oriented programming language (similar to C++). The toolkit consists of libraries of functions, routines, and objects that can be used together to set up simulations, record output from those simulations, and produce a variety of visual representations including graphs. Swarm is oriented toward agent-based models. Some other popular tools include Stella[3] and StarLogo[4]. These and other resources are well-detailed in Gilbert and Troitzsch's *Simulation for the Social Scientist* (1999), which is also an excellent introduction into simulation methods.

Additional readings. As mentioned above, Gilbert and Troitzsch covers various forms of simulations and discusses both applications and methods. A classic work with many examples is Robert Axelrod's *The Complexity*

[2] http://www.santafe.edu/projects/swarm

[3] http://www.hps-inc.com

[4] http://el.www.media.mit.edu/groups/el/Projects/starlogo/

of Cooperation (1997a); so is Epstein and Axtell's *Growing Artificial Societies*. The other works cited in the references below are also good places to start. A wealth of information can be gleaned from the Internet, including papers in progress, software, demonstrations, poster presentations, and FAQs. A good starting point is `http://www.soc.surrey.ac.uk/research/simsoc/`. So what are you waiting for? Time to get off the bus and start analyzing some political science problems.

References

Axelrod, Robert. 1997a. *The Complexity of Cooperation: Agent-Based Models of Competition and Collaboration.* Princeton: Princeton University Press.

Axelrod, Robert. 1997b. "The Dissemination of Culture: A Model with Local Convergence and Global Polarization." *Journal of Conflict Resolution* 41, 203–226.

Bednar, Jenna and Scott Page. 2001, June 8. Behavioral Diversity in Game Ensembles: Game-Theoretic Foundations of Culture. University of Michigan manuscript.

Epstein, Joshua M. and Robert L. Axtell. 1996. *Growing Artificial Societies: Social Science from the Bottom Up.* Cambridge: MIT Press.

Gilbert, Nigel and Klaus G. Troitzsch. 1999. *Simulation for the Social Scientist.* Philadelphia: Open University Press.

Huhns, M. and M. P. Singh. 1998. *Reading in Agents.* San Mateo, CA: Morgan Kaufmann.

Johnson, Paul E. 1999. "Protests, Elections, and Other Forms of Political Contagion." Presented at the 1999 annual meeting of the American Political Science Association, Atlanta, GA.

Kendrick, David, Barbara Bergmann, Ivy Broder, Martin David, and John Geweke. 1991. "Research Opportunities in Computational Economics." Technical Report R-91-1, Center for Economic Research, Department of Economics, University of Texas Austin. Report of the NSF Workshop on Research Opportunities in Computational Economics Washington, D.C. March 6, 1991.

Kollman, Kenneth, John H. Miller, and Scott E. Page. 1992. "Adaptive Parties in Spatial Elections." *American Political Science Review* 86 (4), 929–937.

Laver, Michael and Kenneth Benoit. 2001. "The Dynamics of Fission and Fusion in a Party System." Trinity College, University of Dublin Manuscript.

Meadows, D. L. et. al. 1974. *Dynamics of Growth in a Finite World.* Cambridge, MA: MIT Press.

Page, Scott E. 1999. "Computational Models from A to Z." *Complexity.*

Signorino, Curtis. 1996. "International Cooperation under Uncertainty: The Effects of Symmetric and Asymmetric Noise." *Journal of Conflict Resolution* 40 (1), 152–205.

Agent-Based Modeling in Political Science

Lars-Erik Cederman[1]
Harvard University
cederman@cfia.harvard.edu

Agent-based modeling is a computational methodology that allows the analyst to create, analyze, and experiment with, artificial worlds populated by agents that interact in non-trivial ways and that constitute their own environment (for introductions, see Axelrod 1997b; Casti 1997; Epstein and Axtell 1996; Epstein 1999; Axtell 2000). In these "complex adaptive systems," computation is used to simulate agents' cognitive processes and behavior in order to explore emergent macro phenomena, i.e. structural patterns that are not reducible to, or even understandable in terms of, properties of the micro-level agents (Cederman 1997, Chap. 3). Such "bottom-up" models typically feature local and dispersed interaction rather than centralized control (Resnick 1994). Moreover, as opposed to traditional models that assume either a small number of dissimilar or numerous identical actors, agent-based models normally include large numbers of heterogeneous agents. Rather than studying equilibrium behavior, the focus is often on dynamics and transient trajectories far from equilibrium. Finally, instead of assuming the environment to be fixed, many agent-based models let the agents constitute their own endogenous environment. Given its potential to bridge the gap between conventional formal tools and qualitative theorizing of complex settings, agent-based models are therefore more usefully seen as a complement to rational-choice techniques rather than as a rival.

Agent-based approaches should be contrasted to earlier uses of simulation in the social sciences (see references Gilbert and Troitzsch 1999), including the tradition of global modeling that peaked in the 1970s (Taber and Timpone 1996a, pp. 48-49).[2] Such "equation-based"

[1] Author's note: Many thanks to Robert Axelrod, James Fowler, Nigel Gilbert, Ian Lustick, Michael Macy, Steve Majeski, Rick Riolo, Phil Schrodt, and Michael Ward for helpful comments.

[2] There are also other types of computational approaches to social science, such as rule-based models and natural language processing derived from artificial intelligence (Bainbridge et al. 1994; Taber and Timpone 1996a), or Monte Carlo simulations, but these fall outside the purview of this review.

models attempt to capture macro-properties of social systems numerically (for more on the distinction, see Parunak et al. 1998; Shank 2001). While this type of systems-theoretic research has been widely discredited for having failed to live up to its self-imposed predictive aims, agent-based modeling serves more modest explanatory and heuristic purposes (Axelrod 1997b; Gilbert 2000). By contrast, the agent-based literature strives to improve our understanding of macro-level patterns and processes by treating them as emergent phenomena and locating the social mechanisms that generate them. In this sense, the computational paradigm differs from conventional inductive and deductive approaches (Axelrod 1997b). In accordance with the "retroductive" principles of scientific realism (e.g. Miller 1987), the main goal is to uncover the mechanisms that generate the emergent effect rather than to explain in terms of positivist covering laws (Hayek 1967), that is: "If you can't grow it, you haven't explained it" (Epstein 1999).

The first coherent statement of the approach can be found in Schelling (1971; 1978), who exemplifies it with his famous segregation model. This framework shows how a stark segregation pattern emerges from local migratory movements among two culturally distinct, but relatively tolerant, types of households.[3]

General theoretical studies

The current survey starts with general social-scientific studies using agent-based modeling before turning to a section on more specific applications to substantive topics in political science.[4] The first, more general category can be divided into three interrelated clusters of research depending on what is treated as an emergent feature, or more colloquially, "what's being grown." In rough chronological order of appearance, these bodies of literature trace the emergence of (i) behavioral interaction patterns, (ii) property configurations, and (iii) social networks and hierarchies.

(i) Studies explaining behavioral aspects of social systems appeared first and remains the most active research area in agent-based modeling. Following the path-breaking work by Axelrod (1981; 1984), the literature on behavioral interaction patterns has centered on explaining the emergence of cooperation in anarchic settings. As is well known, Axelrod's main result indicates that cooperation is possible in social dilemmas provided that the actors' interactions are iterated. More importantly, however, the book introduces and popularizes evolutionary thinking in the context of the social sciences. Subsequently, this work has spawned a whole literature on the dynamic conditions of cooperation (for reviews, see Axelrod and Dion 1988; Macy 1998; Hoffmann 2000; Axelrod 2000). The general trend has been to move from the simple iterated Prisoner's Dilemma setup to alternative configurations involving more complex agent representations and interaction topologies. With respect to agent representations, researchers have extended the agents' internal models, especially their memory (e.g. Lindgren 1992; Axelrod 1997, Chap. 1), strategy combinations (e.g. Boyd and Lorberbaum 1987; Hirshleifer and Coll 1988; Lomborg 1996; Axelrod 1997, Chap. 2), and norms (Axelrod 1986; for other aspects see Hoffmann 2000). As anticipated by Axelrod (1984, Chap. 8), models introducing different interaction topologies study the effect on cooperation of spatial proximity and "labeling". Suggestive findings indicate that localized interactions appear to facilitate the emergence of cooperative clusters. Building on Nowak and Sigmund (1992), who derived non-conflictual outcomes in noisy games, Grim (1996) employs two-dimensional cellular automata to demonstrate that once explicit spatial representation is introduced, even more generous strategies thrive (see also Lindgren and Nordahl 1995; Cohen, Riolo, and Axelrod 2001; cf. Huberman and Glance 1993). Labels, also referred to as tags (Holland 1995), are reasonably stable actor-specific characteristics that can be observed by other agents during interactions and on which their behavior can be predicated. Benign actors tend to profit from selection mechanisms that allow them to adjust their strategies to similarly peaceful partners while minimizing exposure to more aggressive players. For example, Riolo (1997) shows that under a broad range of conditions, agent populations that use tags attain a higher level of cooperation due to faster initial emergence of reciprocity and higher resistance to invasion by mutual defectors (see also Epstein and Axtell 1996, Chap. III; Cohen, Riolo and Axelrod 2001; Riolo, Axelrod, and Cohen forthcoming; Macy and Skvoretz 1998). This rich scholarly tradition continues to dominate the agent-based literature in the social sciences but it is not the only focus that inspires modeling work.

(ii) The second cluster of models explains the emergence of configurations based on the actors' properties, such as cultural traits and attitudinal dispositions, rather than behavioral patterns. Whereas Schelling's pioneering study, which belongs to the former category, antedated the behavioral wave, it took a longer time for the theme of property configurations to gain prominence. In the early 1990s, Latan and his collaborators used decentralized simulation models to formalize social impact theory that analyzes the distribution of social influence (e.g. Latan, Nowak, and Liu 1994). These models typically feature a spatial grid in which clustering and diversity emerge from local interactions (for a recent review, see Kennedy and Eberhart 2001). Other studies go beyond

[3]Schelling's model is so simple that it could almost be classified as a cellular automaton, though the movement rule violates that formulation (cf. Gilbert and Troitzsch 1999, Chap. 7).

[4]For links to literature in related social sciences, see below. See also surveys of the contributions to sociology (Macy forthcoming), organization theory (Carley forthcoming), economics (Testfatsion 1997), and business (LeBaron 2000). Casti (1996) provides an even broader set of applications including many in the natural sciences.

the one-dimensional representation of opinions and culture by drawing on various notions of "landscapes" (Axelrod 1997, Chap. 4). In this respect, John Holland's (1992; 1995) genetic algorithms and Stuart Kaufman's (1993) tunable fitness landscapes belong to the most influential formalizations. While both were originally proposed for purposes only indirectly related to social settings, such as problem-solving in artificial intelligence and modeling of biological evolution, they have found use in social-scientific research. In an early application of similar ideas to organization theory, March (1991) represents organizational culture as a string of binary traits, and shows how it co-evolves with the members' own culture strings in a two-level setup. This allows him to capture the tradeoff between "exploration" and "exploitation" as well as the impact of turnover on organizational learning. Adding a spatial component to the multi-dimensional landscapes, Axelrod (1997, Chap. 7) proposes a culture model in which cultural similarity increases the likelihood of interaction. Contrary to what would be expected, this mechanism does not always create one homogenous culture but often generates distinct cultural regions (see also e.g. Kennedy 1997; Mark 1998; Kennedy and Eberhard 2001, Chap. 6; Pedone and Conte 2001).

(iii) Going beyond endogenization of actors' behavior and properties, a third body of literature treats the networks and actor boundaries as emergent features. Because it adds a new layer of complexity, this strand of modeling has taken even more time to develop and therefore remains partly nascent. Some research belonging to (i) has started to blend into (iii) by allowing the actors to choose their interaction partners (e.g. Majeski et al. 1999; for other references, see Axelrod 2000, pp. 146-148). In its most general form, such settings generate emergent networks of interactions (see e.g. Epstein and Axtell 1996, pp. 79-82). Recent theoretical game-theoretic work shows that behavior within such evolved network configurations may differ significantly from unstructured settings (Skyrms and Pemantle 2000). To grow the actors themselves poses an even more formidable challenge. Computational organization theory has made important contributions to this strand of research by "breeding" organizational structures (e.g. Carley and Svoboda 1996; Axelrod 1997: Chap. 6). It is likely that the artificial life literature (Langton 1995) will inspire further progress in modeling the emergence of agency and actors (e.g. Fontana and Buss 1994). Yet, for the time being, this remains a mostly unrealized possibility (though see Padgett 1997).

As already noted, the three strands of research covered here tend to blend into each other. After all, I have organized publications according to the macro-effect that they actually focus on. At the same time, many computational frameworks are general enough to allow for combinations of these goals. In some cases, it may prove impossible to account for actors' behavior (i) without theorizing their boundaries (iii). Other instances could point to a merger of (i) and (ii), for example, where spatially situated cooperation produces distinct arrangements of similarly labeled actors.

Applications to Political Science

As with many other abstract methodologies, agent-based modeling has the advantage of promoting interdisciplinary transfers of ideas and concepts. Stressing such an intellectual cross-fertilization, this section provides a non-exhaustive overview of the applied agent-based literature in political science.

While simulation has a long history in the field (see e.g. Cohen, March, and Olsen 1972), the bulk of the agent-based work is more recent (Johnson 1999). One of the earliest examples is Bremer and Mihalka's (1977) model of geopolitical competition and conquest. Although this model has been characterized as a conventional cellular automaton (Taber and Timpone 1996a, p. 47), its complexity goes well beyond such a simple scheme (see also Schrodt 1981; Ward 1988). Because the states' boundaries emerge in a process of repeated conquests, it is one of the few frameworks that belong to category (iii). Based on a reimplementation of the Bremer-Mihalka model, Cusack and Stoll (1990) explore their modeling framework more systematically than their predecessors. In particular, they introduce a richer set of actor types including primitive power-seekers, collective security actors, and expected-utility maximizers. Among other things, they show that, contrary to realist expectations, collective security can enhance the states' survival chances. Their conclusion is that realism rests on "fractured foundations." Drawing on similar principles but introducing a new quasi-parallel activation regime, Cederman (1994; 1997) finds that defensive technology and alliances may increase the risk of unipolar outcomes (see also Duffy 1992). Drawing on the notion of tag-mediated cooperation as described under (i) above, a related study traces the emergence of clusters of conditional cooperation among democracies (Cederman 2001a).

In addition to explicit modeling, Axelrod's computational experiments have inspired much general theorizing in international relations, including Keohane's (1984) theory of regimes. The strength of the evolutionary framework is that it helps to show how cooperation can happen even in the absence of centralized enforcement, a condition that characterizes many aspects of world politics.

Voting also lends itself to being modeled computationally. Due to the famous "chaos result" (McKelvey 1976), the deductive literature has encountered difficulties capturing multi-dimensional representations of preferences. If modeled as an "electoral landscape" (Kollman, Miller, and Page 1998) in analogy to fitness landscapes (cf. ii above), however, agent-based modeling can be brought to bear on voting puzzles.[5] While there are early examples of computational approaches to voting theory (e.g. McPhee and Smith 1962), the most recent wave of agent-based modeling applied to electoral theory started with Kollman, Miller and Page (1992) who propose the notion of "adaptive parties". This model explores candidates' positioning in the face of uncertainty. Agent-based modeling and simulation also offer advantages in studies of political institutions (Johnson 1996). Modeling Tiebout competition not unlike Schelling's segregation model, Kollman, Miller, and Page (1997) let the voters "vote with their feet" among multiple jurisdictions. Other models feature interest groups (Johnson 1998), multi-party systems (Lomborg 1997), uncertainty (Clough 2001), voter turnout and platform divergence (Fowler and Smirnov 2001), and party-formation (Schreiber 1999).

The most recent strand of agent-based modeling to have appeared in political science takes on the challenge of formalizing ethnicity and nationalism. Rather than explaining the emergence of ethnicity, some models trace its behavioral consequences. Representing individual members of two groups as agents in a two-dimensional grid, Epstein, Steinbruner, and Parker (2001) study the pacifying effect of peace-keeping forces on secession and ethnic inter-group conflicts. Drawing on Axelrod's general studies of norms, Bhavnani and Backer (2000) account for the outbreak of ethnic genocide in Rwanda. Van der Veen (2001) applies agent-based modeling to the question of globalization and ethnic conflict. Other models endogenize the ethnic identities themselves. In an effort to refine constructivist theorizing, Ian Lustick's (2000) "Agent-based identity repertoire" model endows the agents with a set of latent identities that can be activated in particular contexts. Extending the idea of activated identities to multiple dimensions, Cederman (1997, Chap. 8) uses John Holland's notion of schemata as a source of analytical inspiration. Here identities are constituted by a string of traits that may or may not be politically activated. That extension incorporates national identities alongside states in his dynamic model of the state system (Cederman 2001b).

Modeling Resources

The scientific success of agent-based modeling depends crucially on the availability of reliable and easy-to-use tools. While the field still has a long way to go until it matches the software available for statistical processing, steady progress has been made in the last few years. The first generations of scholars had to program their models from scratch in general-purpose computer languages, but more recently specialized software libraries have started to appear. Such packages provide a set of programming tools that relieve applied scientists from having to code housekeeping routines, including control panels, graphs, graphical displays, interaction spaces, and measurement routines. Conceived by Chris Langton at the Santa Fe Institute, Swarm was the first complete toolkit of this kind to cater to academic researchers (see `http://www.swarm.org`). Offering a more modern and better-documented alternative to Swarm without deviating too far from the original concept, Repast is a refined and relatively easy-to-use, Java-based toolkit for agent-based modeling, which is developed and maintained by the University of Chicago (see `http://sourceforge.repast.net`). Ascape provides similar functionality and is also based on Java (see `http://www.brook.edu/ES/dynamics/models/ascape/`). While efforts are underway to develop graphical point-and-click tools, programming skills are still required to create models in these three packages.[6]

Those who want to read more about agent-based modeling are encouraged to turn to several web-based resources. The Santa Fe Institute remains perhaps the best general entry-point to the field, since its activities stretch well beyond the social sciences and cover general complexity theory as well (see `http://www.santafe.edu`). Its series of working papers features many interesting applications to the social sciences. The University of Michigan's Center for the Study of Complex Systems is home to much active research in the area and also provides a series of working papers (see `http://www.pscs.umich.edu/`). Other important sites for agent-based research include The Center on Social and Economic Dynamics, Brookings Institution (see `http://www.brook.edu/ES/dynamics/models/`), Leigh Tesfatsion's comprehensive web page on computational economics (see `http://www.econ.iastate.edu/tesfatsi/ace.htm`), Kathleen Carley and William Wallace's Computational and Mathematical Organization Theory Journal, Ian Lustick's ABIR site: `http://www.psych.upenn.edu/sacsec/abir/`, and the site on "Computer simulation of societies" that goes beyond agent-based modeling: `http://www.soc.surrey.ac.uk/research/simsoc/`. Finally, the Journal of Artificial Societies and Social Simulation can be recommended as it publishes cutting-edge papers on-line (see `http://jasss.soc.surrey.ac.uk/JASSS.html`).

References

Axelrod, Robert. 1981. "The Evolution of Cooperation among Egoists." *American Political Science Review* 75: 306-318.

[5] Yet, it should be noted that most of these studies fall into category (i) since they focus on explaining behavioral patterns rather than property configurations (though see Lustick and Miodownik 2000).

[6] For those interested in acquiring proficiency in Java programming, Schildt (2001) is a good place to start for beginners. Those with prior experience from object-oriented programming other than Java may want to consult Eckel (1998). In addition, there are educational tools, such as StarLogo (Resnick 1994; see `http://www.media.mit.edu/starlogo/`) and AgentSheets (`http://www.agentsheets.com/`), that require very little in terms of programming, though at a considerable loss of flexibility.

Axelrod, Robert. 1984. *The Evolution of Cooperation.* New York: Basic Books.

Axelrod, Robert. 1986. "An Evolutionary Approach to Norms." *American Political Science Review* 80: 1095-1112.

Axelrod, Robert. 1997a. *The Complexity of Cooperation: Agent-Based Models of Competition and Collaboration.* Princeton: Princeton University Press.

Axelrod, Robert. 1997b. "Advancing the Art of Simulation in the Social Sciences." In *Simulating Social Phenomena*, ed. Rosaria Conte, Rainer Hegselmann, and Pietro Terna. Berlin: Springer Verlag.

Axelrod, Robert. 2000. "On Six Advances in Cooperation Theory." *Analyse & Kritik* 22: 130-151.

Axelrod, Robert and Douglas Dion. 1988. "The Further Evolution of Cooperation." *Science* 242: 1385-1390.

Axtell, Robert. 2000. "Why Agents? On the Varied Motivations for Agent Computing in the Social Sciences." Brookings Institute. Center on Social and Economic Dynamics. Working Paper 17. `http://www.brook.edu/es/dynamics/papers/agents/agents.htm.`

Bainbridge, William Sims, Edward E. Brent, Kathleen M. Carley, David R. Heise, Michael W. Macy, Barry Markovsky, and John Skvoretz. 1994. "Artificial Social Intelligence." *Annual Review of Sociology* 20: 407-436.

Bhavnani, Ravi and David Backer. 2000. "Localized Ethnic Conflict and Genocide: Accounting for Differences in Rwanda and Burundi." *Journal of Conflict Resolution* 44: 283-306.

Boyd, R. and J. P. Lorberbaum. 1987. "No Pure Strategy is Evolutionarily Stable in the Repeated Prisoner's Dilemma Game." *Nature* 327: 58-59.

Bremer, Stuart A. and Michael Mihalka. 1977. "Machiavelli in Machina: Or Politics Among Hexagons." In *Problems of World Modeling*, ed. Karl W. Deutsch. Boston: Ballinger.

Carley, Kathleen M. forthcoming. "Computational Organization Science: A New Frontier." Proceedings of the National Academy of Sciences.

Carley, Kathleen M. and David M. Svoboda. 1996. "Modeling Organizational Adaptation as a Simulated Annealing Process." *Sociological Methods and Research* 25: 138-168.

Casti, John L. 1997. *Would-Be Worlds: How Simulation Is Changing the Frontiers of Science.* New York: Wiley.

Cederman, Lars-Erik. 1994. "Emergent Polarity: Analyzing State-Formation and Power Politics." *International Studies Quarterly* 38: 501-533.

Cederman, Lars-Erik. 1997. *Emergent Actors in World Politics: How States and Nations Develop and Dissolve.* Princeton: Princeton University Press.

Cederman, Lars-Erik. 2001a. "Modeling the Democratic Peace as a Kantian Selection Process." *Journal of Conflict Resolution* 45: 470-502.

Cederman, Lars-Erik 2001b. "Modeling the Co-Evolution of States and Nations." In David Sallach and Thomas Wolsko (eds.) *Proceedings of the Workshop on Simulation of Social Agents: Architectures and Institutions.* Chicago: Argonne National Labratory.

Clough, Emily. 2001. "Integrating Downs and Duverger: A Model of a Party System Under Conditions of Uncertainty." Presented at the Sackler Colloquium of the National Academy of Sciences, University of California at Irvine.

Cohen, Michael D., James March, and J. Olsen. 1972. "A Garbage Can Model of Organizational Choice." *Administrative Science Quarterly* 17: 1-25.

Cohen, Michael D., Rick L. Riolo, and Robert Axelrod. 2001. "The Role of Social Structure in the Maintenance of Cooperative Regimes." *Rationality and Society* 13: 5-32.

Cusack, Thomas R. and Richard Stoll. 1990. *Exploring Realpolitik: Probing International Relations Theory with Computer Simulation.* Boulder: Lynnie Rienner.

Duffy, Gavan. 1992. "Concurrent Interstate Conflict Simulations: Testing the Effects of the Serial Assumption." *Mathematical and Computer Modelling* 16: 241-270.

Eckel, Bruce. 1998. *Thinking in Java.* Upper Saddle River, N. J.: Prentice Hall. see also `http://www.mindview.net.`

Epstein, Joshua M. and Robert Axtell. 1996. *Growing Artificial Societies: Social Science From the Bottom Up.* Cambridge, Mass.: MIT Press.

Epstein, Joshua. 1999. "Agent-Based Computational Models and Generative Social Science." *Complexity* 4: 41-60.

Epstein, Joshua M., John D. Steinbruner, and Miles T. Parker. 2001. "Modeling Civil Violence: An Agent-Based Computational Approach." Working Paper No. 20. Center on Social and Economic Dynamics, Brookings Institution.

Fontana, Walter and Leo Buss. 1994. "What Would Be Conserved If 'the Tape Were Played Twice?'." *Proceedings of the National Academy of Sciences* 91: 751-761.

Gilbert, Nigel. 2000. "Modeling Sociality: The View From Europe." In *Dynamics in Human and Primate*

Societies, Timonthy A. Kohler, and George J. Gumerman (eds.). New York: Oxford University Press.

Gilbert, Nigel and Klaus G. Troitzsch. 1999. *Simulation for the Social Scientist*. Bukingham: Open University Press.

Grim, Patrick. 1996. "Spatialization and Greater Generosity in the Stochastic Prisoner's Dilemma." *BioSystems* 37: 3-17.

Fowler, James, and Oleg Smirnov. 2001. "A Cellular Automata Model of Elections: A Possible Solution to Puzzles of Voter Turnout and Platform Divergence." Unpublished ms. Harvard University.

Hayek, Friedrich A. von. 1967. *Studies in Philosophy, Politics and Economics*. Chicago: University of Chicago Press, pp. 3-42.

Hirshleifer, Jack and Juan Carlos Martinez Coll. 1988. "What Strategies Can Support the Evolutionary Emergence of Cooperation." *Journal of Conflict Resolution* 32: 367-398.

Hoffmann, Robert. 1999. "The Independent Localisations of Interaction and Learning in the Repeated Prisoner's Dilemma." *Theory and Decision* 47: 57-72.

Hoffmann, Robert. 2000. "Twenty Years on: The Evolution of Cooperation Revisited." *Journal of Artificial Societies and Social Simulation* 3: `http://www.soc.surrey.ac.uk/JASSS/3/2/forum/1.html.`

Holland, John H. 1992. "Genetic Algorithms." *Scientific American* 267: 66-72.

Holland, John H. 1995. *Hidden Order: How Adaptation Builds Complexity*. Reading, Mass.: Addison-Wesley.

Huberman, Bernardo A. and N. Natalie S. Glance. 1993. "Evolutionary Games and Computer Simulations." Proceedings of the National Academy of Sciences, USA 90: 7716-7718.

Johnson, Paul E. 1996. "Unraveling in a Variety of Institutional Settings." *Journal of Theoretical Politics* 3: 299-300.

Johnson, Paul E. 1998. "An Agent-Based Model of the Exchange Theory of Interest-Groups." Paper presented at the Annual Meeting of the American Political Science Association, Boston.

Johnson, Paul E. 1999. "Simulation Modeling in Political Science." *American Behavioral Scientist* 42: 1509-1530.

Keohane, Robert S. 1984. *After Hegemony: Cooperation and Discord in the World Political Economy.* Princeton: Princeton University Press.

Kaufman, Stuart A. 1993. *The Origins of Order: Self-Organization and Selection in Evolution.* Oxford: Oxford University Press.

Kennedy, James. 1998. "Thinking Is Social: Experiments with the Adaptive Culture Model." *Journal of Conflict Resolution* 42: 56-76.

Kennedy, James and Russell C. Eberhart. 2001. *Swarm Intelligence*. San Francisco: Morgan Kaufmann.

Kollman, Ken, John H. Miller, and Scott E. Page. 1992. "Adaptive Parties in Spatial Elections." *American Political Science Review* 86: 929-937.

Kollman, Ken, John H. Miller, and Scott E. Page. 1992. "Political Parties and Electoral Landscapes." *British Journal of Political Science* 28: 139-158.

Kollman, Ken, John H. Miller, and Scott E. Page. 1997. "Political Institutions and Sorting in a Tiebout Model." *American Economic Review* : 87: 977-992.

Langton, Chrisopher. 1995. *Artificial Life: An Overview*. Cambridge, Mass.: MIT Press.

Latan, Bibb, Adrzej Nowak, and James H. Liu. 1994. "Dynamic Polarization and Clustering as Order Parameters of Social Systems." *Behavioral Science* 39: 1-24.

LeBaron, Blake. 2000. "Agent-Based Computational Finance: Suggested Readings and Early Research." *Journal of Economic Dynamics & Control* 24:. 679-702.

Lindgren, Kristian and Mats G. Nordahl. 1995. "Cooperation and Community Structure in Artificial Ecosystems." *In Artificial Life: An Overview*, ed. Chrisopher Langton. Cambridge, Mass.: MIT Press.

Lomborg, Bjrn. 1996. "Nucleus and Shield: The Evolution of Social Structure in the Iterated Prisoner's Dilemma." *American Sociological Review* 61: 278-307.

Lomborg, Bjrn. 1997. "Simulating Multiparty Systems." In *Simulating Social Phenomena*, ed. Rosaria Conte, Rainer Hegselmann, and Pietro Terna. Berlin: Springer Verlag.

Lustick, Ian. 2000. "Agent-Based Modeling of Collective Identity: Testing Constructivist Theory." *Journal of Artificial Societies and Social Simulation* 3(1): `http//:jasss.soc.surrey.ac.uk/3/1/1.html.`

Lustick, Ian, and Dan Miodownik. 2000. "Deliberative Democracy and Public Discourse: The Agent Based Argument Repertoire Model." *Complexity* 5: 13-30.

Macy, Michael W. 1998. "Social Order in Artificial Worlds." *Journal of Artificial Societies and Social Simulation* 1: `http://www.soc.surrey.ac.uk/JASSS/1/1/r.html.`

Macy, Michael W. and John Skvoretz. 1998. "The Evolution of Trust and Cooperation Between Strangers: A Computational Model." *American Sociological Review* 63: 638-660.

Macy, Michael W. forthcoming. "From Factors to Actors: Computational Sociology and Agent-Based Modeling." *Annual Review of Sociology.*

Majeski, Stephen, Corina Linden, and Aaron Spitzer. 1999. "Agent Mobility and the Evolution of Cooperative Communities," *Complexity* 5: 16-24.

March, James G. 1991. "Exploration and Exploitation in Organizational Learning." *Organizational Science* 2: 71-87.

Mark, N. 1998. "Beyond Individual Differences: Social Differentiation from First Principles." *American Sociological Review* 63: 309-330.

McPhee, W. and R. Smith. 1962. "A Model for Analyzing Voting Systems." In W. McPhee and W. Glaser, eds. *Public Opinion and Congressional Elections* New York: Free Press.

McKelvey, Richard D. 1976. "Intransitivites in Multidimensional Voting Models and Some Implications for Agenda Control." *Journal of Economic Theory* 12: 472-482.

Miller, Richard W. 1987. *Fact and Method: Explanation, Confirmation and Reality in the Natural and the Social Sciences.* Princeton: Princeton University Press.

Nowak, A., J. Szamrej, and B. Latan. 1990. "From Private Attitude to Public Opinion: A Dynamic Theory of Social Impact." *Psychological Review* 97: 362-376.

Nowak, Martin A. and Karl Sigmund. 1992. "Tit for Tat in Heterogeneous Populations." *Nature* 355: 250-253.

Padgett, John F. 1997. "The Emergence of Simple Ecologies of Skill: A Hypercycle Approach to Economic Organization." In *The Economy as an Evolving Complex System II*, ed. W. Brian Arthur, Steven N. Durlauf, and David A. Lane. Reading, Mass.: Addison-Wesley.

Parunak, H. Van Dyke, Robert Savit, and Rick L. Riolo. 1998. "Agent-Based Modeling vs. Equation-Based Modeling: A Case Study and Users' Guide." In *Proceedings of Multi-agent Systems and Agent-based Simulation*, Berlin: Springer.

Pedone, Roberto and Rosaria Conte. 2001. "Dynamics of Status Symbols and Social Complexity." *Social Science Computer Review* 19: 249-262.

Resnick, Mitchel. 1994. *Turtles, Termites, and Traffic Jams.* Cambridge, Mass.: MIT Press.

Riolo, Rick L. 1997. "The Effects of Tag-Mediated Selection of Partners in Evolving Populations Playing the Iterated Prisoner's Dilemma." SFI. Working Paper 97-02-016.

Schelling, Thomas C. 1971. "Dynamic Models of Segregation." *Journal of Mathematical Sociology* 1: 143-186.

Schelling, Thomas C. 1978. *Micromotives and Macrobehavior.* New York: W. W. Norton.

Schildt, Herbert. 2001. *Java 2: A Beginner's Guide: A Beginner's Guide.* Berkeley: Osborne McGraw Hill.

Schreiber, Darren. 1999. "Party Formation and Competition: An Emergent Model." Unpublished ms. see `http://www.bol.ucla.edu/~dschreib/home.html.`

Schrodt, Philip A. 1981. "Conflict as a Determinant of Territory." *Behavioral Science* 26: 37-50

Shaffer, William R. 1972. *Computer Simulations of Voting Behavior.* New York: Oxford University Press.

Shank, Jeffrey C. 2001. "Beyond Reductionism: Refocusing on the Individual with Individual-based Modeling." *Complexity* 6: 33-40.

Skyrms, Brian and Robin Pemantle. 2000. "A Dynamic Model of Social Network Formation." *Proceedings of the National Academy of Sciences* 97.

Taber, Charles S. and Richard J. Timpone. 1996a. "Beyond Simplicity: Focused Realism and Computational Modeling in International Relations." *Mershon International Studies Review* 40: 41-79.

Taber, Charles S. and Richard J. Tampone. 1996b. *Computational Modeling* Quantitative Applications in the Social Sciences. Sage University Paper No. 113. Thousand Oaks: Sage.

Tesfatsion, Leigh. 1997. "How Economists Can Get A-Life" in *The Economy as a Complex Evolving System II* eds. W. Brian Arthur, Steven Durlauf, and David Lane. Addison Wesley: Reading, MA.

Van der Veen, A. Maurits. 2001. "Identity, Interests, and Conflict: An Evolutionary Model." Paper presented at the Annual Convention of the American Political Science Association, San Francisco.

Michael D. Ward. 1988. "Thing Fall Apart: A Logical Analysis of Crisis Resolution Dynamics." *International Interactions* 15: 656-679.

Of Spells, Potions, and Computational Social Science

Charles S. Taber
SUNY at Stony Brook
ctaber@notes.cc.sunysb.edu

Ten years ago in these pages, Philip Schrodt quoted a fictitious anthropologist's impression of political science:

> This culture seems most alien, but at least I understand the role of the political methodologists. They are the witch doctors. Everybody fears them; most people hate them. They are central to the rites of passage into the tribe. Nobody has the slightest idea what they do but witch doctors are thought to disrupt work from great distances - particularly cooking - so you keep them around for protection. And *everybody* agrees you only want one in the village. (1991: 19)

In these enlightened times, it seems that only the more avant-garde among us – Bayesians, perhaps, and certainly Courtney Brown – might still qualify for nose-bones and sabertooth necklaces. Computational modelers, I am told by people other than my wife, most definitely fit into this category. Given the long and distinguished history of computer simulation throughout the sciences, I am always somewhat surprised by this. The unfortunate fact is that there is nothing particularly mysterious or exotic about computational modeling. It would even be safe, I feel confident, to have more than one computational modeler in a village.

Computational modeling (CM) is a family of quite diverse methods, some of which are indeed strikingly new and original, but all with a common epistemological core - the computational, or *algorithmic* "solution strategy" (Taber & Timpone, 1995; Timpone & Taber, 1998; see the on-line introduction to CM at `www.stonybrook.edu/polsci/ctaber/`). Some of the confusion in the social sciences regarding CM derives from what I see as a lack of clarity in how we traditionally define the approach. Some have identified CM too closely with a given theoretical orientation (AI or computational political economy, e.g.); others have focused on a single method (dynamic simulation or artificial adaptive agents, e.g.); many simply observe that CM is modeling done on a computer. None of these are satisfactory. The first two exclude much work that is properly thought of as CM; the third fails to exclude **analytic work** (e.g., standard mathematical modeling) done on a computer. I prefer to define CM in terms of its distinctive computational approach to deriving the implications of a theoretical model.

A computational model, then, is a formal/symbolic representation of a theory about some empirical system – usually, a set of algorithms in a computer program; model behavior is explored by "running" the program. The primary approach to deriving deductions from a mathematical model, by contrast, is to find closed-form solutions (e.g., equilibria or solutions in the limit). To illustrate, consider Condorcet's paradox of social choice. One might represent this problem symbolically in the form of a set of mathematical equations and derive definite analytic solutions to such important questions as how likely is a fair voting outcome, given certain assumptions about voter preferences and voting rules. Alternatively, one might create a computer program that represents elections of n voters and m alternatives, and algorithmically estimate solutions to the same questions by running many simulated elections and observing the outcomes (Jones, Radcliff, Taber, & Timpone, 1995; Timpone & Taber, 1998). Conventional wisdom favors analytic solutions when they are tractable and algorithmic solutions when models become too complex for analytic treatments.

Two computational approaches have gained special attention in recent years, and for good reason. Knowledge-based models, the central theoretical tool of cognitive science, have been usefully applied to questions of individual decision making at both the elite (Sylvan, Goel, & Chandrasekaran, 1990; Taber, 1992) and mass levels (Boynton & Lodge, 1998). Artificial adaptive agents provide an exciting tool for exploring questions of social aggregation and the complex relationships among actors and institutions (Johnson, 1999). The balance of this paper will comment on these two classes of computational methods.

For my money, political science is all about two fundamental questions: decision making and aggregation. How do political actors within a complex field of social forces – for me, individual human beings more or less actively engaged in their social and political information environments – make decisions? And how do those decisions combine to affect the larger political system (indeed, through time they presumably compose that system)? We may generally array theoretical approaches to the first question along a continuum anchored on one end by *homo economicus* and on the other *homo psychologicus*. Among the many things that vary as one travels this continuum, perhaps the most basic is the change from relative simplicity of information processing at the economic end to relative complexity at the psychological end. Moreover, as one's theoretical perspective becomes more deeply cognitive, analytic approaches become less feasible and algorithmic approaches more attractive.

Cognitive science, the field at the intersection of cognitive psychology, linguistics, and computer science, has developed computational theories and methods hand in hand with the empirical exploration of human decision making. Space does not permit an adequate discussion of these methods (for brief discussion and pointers to the literature, see Taber & Timpone, 1995), but I should briefly describe the two most general models of the mind. First, the *parallel distributed processing* theory, most closely associated with the work of David Rumelhart and his colleagues (1989), uses the formalism of neural networks to represent cognitive processing. In general analogy with the brain, knowledge is stored in the overall configuration of linkages among a large number of simple neurons, and thinking is the processing of information from inputs (a configuration of signals of varying strengths) through multiple layers of neurons to outputs (a resulting configuration of signals of varying strengths). The second and dominant theoretical model eschews any attempt at representing the "physical substrate" of the nervous system and rather develops a functional description of the mind based on the principle of a*ssociative memory* and the formalism of semantic networks (Anderson, 1983; Collins & Quillian, 1969). Here, knowledge is stored in discrete concept nodes and the associations among these nodes. So, for example, the belief that Bush opposes taxes would be represented by a negative link between the node for Bush and the node for taxes. (By contrast, such a belief would not be represented at discrete "locations" in a neural network, but rather in the overall configuration of neural connections such that an input configuration symbolizing "Bush" would lead to an output configuration symbolizing "opposes taxes.")

Computational models of political cognition using variations on the semantic network formalism have begun to appear in political science (e.g., Boynton & Lodge, 1996; Taber, 1998, Forthcoming; Young, 1996), but the promise of this approach remains largely untested, and for several reasons I do not expect an explosion of interest in the medium future. First and foremost, decision theorists whose training and epistemological leanings lead them to value formal theory are less inclined to value complexity. Conversely, political psychologists and those whose theoretical interests bring complexity to the fore are neither trained nor inclined to see the value of formal theorizing. Also, since CM and the basic tools of programming are rarely taught in political science graduate programs, the start up costs can appear formidable. In the same vein, a knowledge-based CM project can entail a substantial commitment of time and resources. Finally, for a variety of reasons it can be difficult to evaluate the empirical status of a cognitive CM (Taber & Timpone, 1995). In short, formal work on political decision making is essential, the tools of cognitive science are promising, but I fear that only a few will brave the waters.

I am somewhat more optimistic about the immediate impact of agent-based models, which are finding application in political science on a variety of issues related to aggregation and the emergence of collective phenomena from "local" behavior within a complex system. Agent-based models seek to explain social/institutional phenomena like party behavior on an "electoral landscape" (Kollman, Miller, & Page, 1992), norms of cooperation (Axelrod, 1997), or group identifications (Cederman, 1997) by showing how they emerge from the complex interactions of individual agents. Agent behavior and rules of interaction are set (or better, varied experimentally) for a given model according to theoretical expectation and then the model is run in order to grow emergent properties. One might use this approach as an analytic engine for discovering the collective behavior of "known" types of agents in interactions designed to represent real-world contexts. Or one might think of the set of initial conditions that define a population of agents and their rules of interaction as a hypothesized explanation for a "known" emergent property. In this latter approach, to the extent that the emergent property is grown, the hypothesis is supported.

I think that these methods are among the most exciting developments for political science in the past decade. We have paid far too little attention to complex systems and cross-level processes of aggregation, and the agent-based approach provides promising tools for these critical questions. Moreover, I think that political economists are better positioned to take advantage of these tools than are political psychologists with cognitive tools. An appreciation of the importance of formal theory has long cohabited with an understanding of the complexity of political systems among political economists. The questions addressed already by agent-based models are generally recognized as important by most political scientists. Another reason for optimism is the way in which agent-based modeling tools are developing and spreading among modelers throughout the sciences (e.g., Swarm and the role of the Santa Fe Institute). In a sense, this growing literature exemplifies the theoretical principle of collective emergence from individual interactions (somebody should model the emergence and growth of these ideas as an agent-based system!).

There are, however, some dangers in the rapid growth and inevitable hype of agent-based modeling in political science. First, it is critically important that systematic experiments be run so that we understand exactly how emergent properties grow. The darker side of this same point is that, if history is any guide (e.g., the AI hype a decade ago), there will be some "Chia Pet" models in which emergent phenomena are essentially "in the can" from the start, ready to grow with a little water and sunshine. Sometimes, even the model's owners will not understand why their model grew as it did. (I recall one conference panel a few years ago where I was happily informed by the proud parents of a fractal model of something or other that they had "no earthly idea" how their model worked.) Finally, some critics of agent-based models feel that very little is in fact learned that could not be derived analytically using more standard mathematical techniques (Binmore, 1998).

This last point deserves a bit more discussion. Earlier in this commentary I noted that conventional wisdom holds that analytic methods are to be preferred over algorithmic ones, which are to be used as a last resort only when analytic solutions elude capture. There is, I now believe, no good reason for this conceit. Though stochastic simulation does generate error, given sufficient computational power, this error variance can generally be reduced arbitrarily close to zero. For some complex simulations, this may require more computational power than is, even today, available, but such models are unlikely to be within the ken of analytic methods anyway. Some might argue that computational power invites baroque theory, that there may be little incentive to prune away unnecessary complications in a CM, but I think that this concern is more than outweighed by the corollary that analytic methods require simplicity, even when the system being modeled is complex. Moreover, there are advantages peculiar to computational methods. For some problems, it is simply easier to derive an answer to a theoretical problem through computational methods, even if an analytic solution may be possible. For example, the social choice problem mentioned above may in principle be solved through mathematical methods even when one relaxes the traditional simplifying assumptions (e.g., strong preference orderings), but why do so if an easy answer is forthcoming through simulation (other than for the advancement of mathematical theory). Finally, computational solutions generally contain quite a bit more information than do analytic solutions. The search for equilibria in formal game theory, for example, may deter the examination of behavioral trajectories. Where the system ends up is often less important than how it gets there; computational methods, by contrast, are all about the journey. A cognitive model like POLI (Taber, 1992) generates non-numerical process and outcome behavior in the form of detailed foreign policy recommendations and supporting arguments, a far richer depiction of decision making processes than would be possible with analytic methods.

Much of political methodology - including computational modeling - that once may have appeared magical and mysterious has now gone mainstream. Certain technical advances remain on the cutting edge, of course, but the underlying epistemology should be old hat by now (with the partial exception of the "emergent philosophy" espoused by some agent-based modelers). We are not, I am disappointed to report, witch doctors. Now, if you will pardon me, I need to heal a sickly model for my village chief. I'll have to stop by Stanley Feldman's hut and borrow a vial of Brazilian tree toad venom, and somewhere I need to find a live chicken...

References

Anderson, John R. 1983. *The Architecture of Cognition.* Cambridge, MA: Harvard.

Axelrod, Robert. 1997. *The Complexity of Cooperation: Agent-Based Models of Competition and Collaboration.* Princeton, NJ: Princeton.

Binmore, Ken. 1998. Review of Robert Axelrod's *The Complexity of Cooperation: Agent-Based Models of Competition and Collaboration.* Reviewed in *Journal of Artificial Societies and Social Simulation* 1(1).

Boynton, G. Robert, and Milton Lodge. 1998. "J.Q. Public: A computational model of a survey respondent." *Political Communication.*

Cederman, Lars-Erik. 1997. *Emergent Actors in World Politics: How States and Nations Develop and Dissolve.* Princeton, NJ: Princeton.

Collins, Allan M., and M. Ross Quillian. 1969. "Retrieval time from semantic memory." *Journal of Verbal Learning and Verbal Behavior* 8: 240-248.

Johnson, Paul E. 1999. "Simulation modeling in political science." *American Behavioral Scientist* 42: 1509-1530.

Jones, Bradford, Benjamin Radcliff, Charles Taber, and Richard Timpone. 1995. "Condorcet winners and the paradox of voting: Probability calculations for weak preference orders." *American Political Science Review* 89: 137-144.

Kollman, Ken, John H. Miller, and Scott E. Page. 1992. "Adaptive parties in spatial elections." *American Political Science Review* 86: 929-937.

Rumelhart, David E., and James L. McClelland, eds. 1989. *Explorations in Parallel Distributed Processing.* Cambridge, MA: MIT.

Schrodt, Philip A. 1991. "Political methodology explained." *The Political Methodologist* 4(2): 19.

Sylvan, Donald A., Ashok Goel, and B. Chandrasekaran. 1990. "Analyzing political decision making from an information processing perspective: JESSE." *American Journal of Political Science* 34: 74-123.

Taber, Charles S. 1992. "POLI: An expert system model of U.S. foreign policy belief systems." *American Political Science Review* 86: 888-904.

Taber, Charles S. 1998. "The interpretation of foreign policy events." In Donald A. Sylvan and James F. Voss, eds., *Problem Representation in Political Decision Making.* London: Cambridge.

Taber, Charles S. Forthcoming. "An artificial mediator." In Zeev Maoz, Alex Mintz, T. Clifton Morgan, Glenn Palmer, and Richard Stoll, eds., *Methodology in International Relations: Multiple Paths to Knowledge.*

Taber, Charles S., and Richard J. Timpone. 1995. *Computational Modeling* (Sage University Paper Series on Quantitative Applications in the Social Sciences, 07-113). Newbury Park, CA: Sage.

Timpone, Richard J., and Charles S. Taber. 1998. "Analytic and algorithmic analyses of Condorcet's paradox: Variations on a classical theme." *Social Science Computer Review* 16: 72-95.

Young, Michael D. 1996. "Cognitive mapping meets semantic networks." *Journal of Conclict Resolution* 40: 395-414.

Computational Modeling from a Graduate Student Perspective

Emily Clough
University of Minnesota
eclough@polisci.umn.edu

As evidenced by a growing literature, including Ken Benoit's paper on fission and fusion in legislatures at the methods meeting this summer, computational modeling is becoming an increasingly prominent technique in the field of political methodology. Computational modeling is a formal method but it differs radically from other formal methods, such as game theory or decision theory, in that it uses computational experiments to solve the model instead of deductive techniques. Solving models computationally allows the modeler to create and find solutions for models that would not be solvable with deductive techniques; however, it also presents a set of problems. Some of these difficulties are largely the result of being an emerging field; there are issues that computational modelers have yet to work out. Some, however, are an inherent part of the field. The modeler should carefully consider both before the modeling effort is attempted. *What is computational modeling?*

Computational models use computational experiments to solve formal models. Instead of solving for the parameters of the model in general, as would happen in a deductive model, numbers are plugged into the parameters and a numerical solution is derived. Most modelers choose to test a sample from some numerical distribution; this will give a good idea of the relationships between the parameters and outcome. Some modelers, however, choose to test the universe of possibilities (even if this universe is a somewhat limited one). In both situations, however, it is important to keep in mind that the model is being solved for a number of cases, but it is not being solved in general.

There are a number of sub-fields within computational modeling, but the most prominent in the social sciences is what is known as agent-based modeling. It is in this area that some of the differences between computational and deductive techniques are clearest. Agent-based models consist of a number of agents with characteristics and decision-making rules. These agents interact with one another and their environment in a way specified by the modeler; in this sense, agent-based modeling is readily comparable to game theory. In an agent-based model, the modeler may choose which other agents each agent gets to interact with, as well as the rules that dictate those interactions. Decision-making rules may change through the course of the model; in Axelrod (1997), for example, agents evolve new rules for playing the prisoner's dilemma. This is one clear way in which computational modeling differs from deductive techniques; in game theory, decision-making rules are static.

Agent-based models have also gone far in demonstrating that interesting aggregate phenomena are witnessed even when agents in the model do not have information about the system as a whole. A system may manage to organize itself, for instance, without any of the agents knowing how the system as a whole is functioning. Perhaps the most famous example of such behavior is the flocking patterns of birds. Models of such behavior show that all the birds need to do in order to stay in a flock is to keep a certain distance from each of the birds around them; no leadership role is necessary in the flock. In game theory, modeling this way is not possible; players in a game must know who all the other players are, their choice sets and at least the probability distribution over their payoffs. This is not a requirement in agent-based models; agents may have limited information about each agent or information about a very limited number of agents or both. For instance, in Cederman's model of state formation and dissolution (1997), states interact with neighboring states, but not with every other state in the model.

Computational models, then, provide a useful tool for the scholar who would like to use a formal model to examine a situation, but who finds the assumptions associated with more traditional deductive formal models to be unrealistic in the situation under examination. There are clearly a number of situations in which we can imagine this to be the case. The debate over the appropriateness of assumptions about rationality and information in formal modeling are long-standing; Herbert Simon's early criticism that people "satisfice", not optimize, has been followed by a number of articles and books addressing concerns about people's willingness and ability to optimize and the conditions necessary in order to allow this to happen. A computational model can allow the modeler to loosen these conditions considerably while still providing a formal structure.

Equilibrium

Equilibrium has long been the focus of deductive formal modeling. Solving a deductive model entails finding the equilibria, and much time and energy has been dedicated to refining the concept of what an equilibrium is. As discussed above, computational models are not solved in general; they have no general equilibria. Many view this as the most serious flaw in the computational modeling approach. Without a general solution to the model, there is no definitive theory to test. In some situations, the results of a computational model may be highly ambiguous. Results may vary wildly depending on the ways in which the parameters are specified, and not always in ways that are clear or may be traced. This is more problematic in models in which there are a high number of parameters and the number of possible parameter value configurations increases. Furthermore, since the model is not solved for general values of the parameters, it is not always clear whether the results of the model may be generalized or whether they were only achieved because of the values that the parameters took. Some of these concerns may be mitigated, of course, by being sure to test a large sample of possible parameters and parameter combinations.

Many, however, believe that these concerns about equilibria and the generality of results are not as important as we might be led to believe. They claim that the concept of equilibrium does not deserve the emphasis that it has received in formal modeling. Even physicists have largely abandoned a focus on equilibrium in favor of a focus on dynamics and the importance of understanding the role of contingency. Many critics of rational choice theory have argued that there is nothing natural about the concept of equilibrium, and that the focus on equilibrium has drawn our attention away from other important factors. While most would not want to abandon the idea of equilibrium entirely, these criticisms suggest that there is a lot of room for approaches where the focus is different. Computational models generally reveal not only end states, but also the dynamics which allowed the system to reach such a state. Understanding the dynamics of the system under study can illuminate aspects of the theory under consideration in a way which is not possible with more traditional techniques.

Furthermore, a computational model may clarify aspects of more traditional models that a deductive approach would not. Deductive formal models offer equilibria, but most provide no real insight into the dynamics which lead to these equilibria. Computational models based on more traditional deductive models may provide insight about these dynamics. It is not always clear in a deductive model whether there really is a process that would lead to an equilibrium; for instance, in games in which there are multiple equilibria, game theorists have made little progress toward explaining which equilibrium is more likely to be reached and why. Computational simulations of models which have been solved formally may lead to a better understanding of the dynamics involved in reaching an equilibrium. They may suggest which equilibria are unlikely to be reached and whether the values of certain parameters make it more likely that players will reach an equilibrium.

Problems

Like any method, computational modeling has a set of problems it must grapple with. These include issues such as how to replicate the work of former scholars and how to deal with issues of uncertainty, i.e. what the likelihood is that your results are representative of the possible set of all outcomes; these are the kinds of issues that most fields struggle with early in their development, and it is likely that efforts to solve these problems will result in solutions in a timely manner. There are other issues, however, which are more endemic to computational modeling, and will be more challenging to address. One of these issues has to do with the level of detail with which the model is specified. Deductive models require a number of assumptions, which usually forces them to remain simple. With computational models, however, it is possible, and sometimes easy, to become very detailed in specification. As a model becomes more detailed, it may represent the situation being modeled more closely, but it may not be useful for understanding a broader class of situations. Furthermore, it may become much more difficult to understand the causal relationships in the model. Many of the interesting effects that are seen in computational modeling, especially agent-based modeling, are the result of a great deal of interaction; these effects can be non-linear, and not at all easy to trace through the system. As the model grows more detailed it becomes even more difficult to trace these effects. There is little consensus at this point on what is "too" detailed; however, it is an important factor to keep in mind when you construct a computational model.

Software

So let's say you want to try some computational modeling yourself; how do you sit down and start? There are a number of packages available to help people create computational models. Probably the easiest of these are the Logo family: Logo (`http://el.www.media.mit.edu/groups/logo-foundation/`), and its cousin NetLogo (`http://ccl.sesp.northwestern.edu/netlogo/`); this is a good place to start if you would like to do some dabbling in computational modeling, but you don't have any programming skills. Both versions of this software use a fairly clear language and a number of preprogrammed objects and functions, which makes it relatively easy to create simple models without too many lines of code. There are also a number of simple, interesting example programs included in the packages; these programs give an idea of some of the interesting results that can be seen with even relatively simple models.

Unfortunately StarLogo is limited in its scope and flexibility, so if you want to create a more complex model it will be necessary to learn a more generic programming language. Many people create their models using C++ or Java, taking advantage of the general functionality these languages offer; however, there are now a number of software packages that provide many useful tools for the computational modeler. The three best known are Swarm (`http://www.swarm.org`), RePast (`http://repast.sourceforge.net/`) and Ascape (`http://www.brook.edu/es/dynamics/models/ascape/`). Each of these requires the modeler to write his or her or own code, usually in Java, but provides a number of tools to make the process of modeling easier. These tools vary from package to package, but they might help the modeler capture certain statistics, provide graphics for the model, create different distributions or consider different network structures. Each package is better at some things than at others, and each has its devotees. If you want to use one of the packages for creating large-scale projects, you should carefully examine the tools available with each software package and talk about the details of your project with someone who knows the package well. Ascape has been gaining a lot of attention lately because many consider it easier to use and more accessible than the others, with a more sophisticated user interface; however, Swarm or RePast may be more appropriate to the project you would like to undertake.

Education

If you're interested in learning more about the details of computational modeling and some of the ways in which it works, there a number of courses and workshops you may be able to take or attend. Some political science graduate programs are beginning to offer courses in computational modeling; however, this is still fairly rare. If this is not an option for you, one possibility for learning more on the subject would be simply to take an introductory artificial intelligence class in the computer science department of your university. Such a course would probably cover subjects such as neural nets and Bayesian networks, both of which have become a source of some interest in political science in the last few years; they might also cover agent-based modeling, which is also known as distributed artificial intelligence in the computer science field.

There are also a number of summer programs that address various issues in computational modeling. The most general of these is the Santa Fe Institute's Complex Systems Summer School (`http://www.santafe.edu/sfi/education/indexCSSS.html`). The CSSS is a four week program that educates graduate students and post-docs about the general principles involved in the study of complex adaptive systems, including areas such as chaos theory and agent-based modeling. The program is highly interdisciplinary and applications in a number of fields are presented and discussed along with more general principles. If you would like to learn more about computational modeling in the social sciences, especially political science or economics, there are two programs available: ICPSR usually offers a module on complex systems modeling during its summer program; and the Santa Fe Institute offers a two week program on computational economics (`http://www.santafe.edu/sfi/education/indexGWE.html`) which has involved a number of political scientists over the years. Any of these programs offer excellent opportunities to learn more about computational modeling and discuss the possible applications to political science.

Summary

Computational modeling offers an exciting new way of modeling problems that were previously considered too complex to be modeled using more deductive techniques. Computational models have already offered a number of insights about these sorts of situations and about more traditional formal models, and they promise to provide further insights in these areas.

References

Axelrod, Robert. 1997. *The Complexity of Cooperation.* Princeton University Press: Princeton, NJ.

Cederman, Lars-Erik. 1997. *Emergent Actors in World Politics: How States and Nations Develop and Dissolve.* Princeton University Press: Princeton, NJ.

Articles

Barbarians at the Gate: A Political Methodologist's Notes on using the National Center for Supercomputer Applications

Philip A. Schrodt
University of Kansas
p-schrodt@ukans.edu

Introduction

This note is a synopsis of my experience in using the facilities at the National Center for Supercomputer Applications (NCSA) in Urbana-Champaign. The quick summary: If you are already familiar with Unix systems, it is remarkably straightforward to run ANSI C code on these machines. The individual processors in the "supercomputer" are substantially slower than those in a contemporary personal computer, but one can gain substantial wall-clock speed advantages through parallelism. In most instances, parallel processing can be added to a program with very little additional effort. The computing time at NCSA is free and should be a consideration for anyone using computationally intensive methods that are parallel either in their inner-most loops (e.g. linear algebra routines) or at the outer-most loops (e.g. Monte Carlo or resampling). [Note: This article was originally written and submitted to *TPM* in summer, 1999, so the situation at NCSA may have changed somewhat from that time. A version of this paper with active Web links can be found at `http://www.ukans.edu/~keds/NCSA.html`. The URLs given in this text are correct as of 15 August 2001].

Background

Kansas, as an NSF-challenged state, is part of an initiative called EPSCoR that tries to encourage greater levels of Federally funded research. Late last year, the NCSA allocated 10,000 hours of supercomputer time to the University of Kansas under EPSCoR auspices. I applied for 50 hours; our research office [correctly] thought this was too little and added a zero to make it 500, and for some reason NCSA actually granted me 4,000 hours. NCSA time is available only by peer-reviewed application (see `http://www.ncsa.uiuc.edu/alliance/applying/`): academic researchers do not have the option of "buying" time through a research budget. None of the supported projects currently listed on the NCSA web pages have any discernible social science content; whether this is the product of lack of interest or lack of support is not clear. NCSA was certainly generous with my request—and the "proposal" was just a couple of paragraphs in length—though this may have been a function of EPSCoR.

[Side comment: An additional motivation for utilizing the NCSA facility were some comments that Larry Smarr, NCSA's then-director, had made concerning the social sciences in the *Chronicle of Higher Education* (11 September 98). Among his pithy observations were "...academic social scientists have self-defined themselves into small problems." and "What's this 'so hard' crap? [Social scientists] don't want to have the answer hard enough." Dr. Smarr, in short, views the social sciences much as a dog views a fire hydrant.]

My project involved the estimation of some large hidden Markov models for forecasting conflict in the former Yugoslavia. This involves a numerical optimization algorithm that is equivalent to the expectation-maximization algorithm of Markov chain Monte Carlo fame (and more generally, HMMs are a type of MCMC model). Consequently the mix of calculations is similar to that found in most statistical applications, though as we will see, the *structure* of the calculations is atypical. HMM estimation is complicated by a large number of local maxima, which I bludgeoned into submission with a combination of genetic algorithm and Monte Carlo methods.

My experiment with the NCSA involved two key challenges. First, could I modify my existing C programs to run on the NCSA machines without going to lot of trouble (and ideally, without having to maintain two versions of the code)? Second, would the NCSA facility provide significantly faster turn-around?

Connecting, Consulting, and Documentation:

Making connections to the NCSA requires the Unix "ssh" (secure shell) system (or Kerberos). This required a bit of reconfiguration on my end—the local computer center had to assign an alphabetic string to my IP number—but once established, connecting has been painless. From inside the secure system, one can use ftp to get back out to one's home system to transfer files. The NCSA systems are Unix (you were expecting Windows'95?), so if one knows Unix, there is no learning curve on the basic file and navigation commands. Allocations of disk space are quite generous, and more space is available if needed.

The NCSA maintains extensive on-line documentation in assorted formats—man, html, pdf—and has a well-organized "getting started" page [`http://www.ncsa.uiuc.edu/SCD/Info/`] that provides step-by-step instructions for getting onto the system and running jobs. I also have had very positive experiences with the email consulting staff-questions were answered within 24 hours and generally were correct the first time.

Most of the detailed machine-specific documentation, on the other hand, has been created in the "write-only" genre: it presumably makes sense for purposes of reference, but is not aimed at the casual user. The NCSA web site provides numerous examples of sample scripts and commands, and I usually had more success modifying these than trying to work out problems *de novo*.

Modifying Programs:

My chief worry about this exercise was the level of effort required to get my existing ANSI C programs to run on the parallel machines. At the most basic level, the answer is a very reassuring "no additional effort whatsoever". Once I figured out the appropriate options to use on the "cc" compiler—a task which required a couple of emails to the consultants—ANSI C runs without modification. There's even an option for using C++-style comments.

Keeping with my objective of minimizing the learning curve, I did not attempt to debug programs on the NCSA system. Instead, I got them running correctly with a familiar PC-based editor and debugger (Metrowerks CodeWarrior, in my case), then uploaded the code to the NCSA for the production runs. More generally, I get the impression that it is a very good idea to make sure your code runs correctly as a serial process before trying to run it in parallel.

Up to this point, I had to agree with Smarr's observations about the ease-of-use of the NCSA. If one has source code in C, C++, or Fortran, and is familiar with Unix—conditions that in all likelihood apply to almost anyone who really needs a supercomputer—then the start-up costs are negligible.

And just how fast is it?

Not very—on the hidden Markov estimation with a single processor (no parallelism), the NCSA machines run about a third as fast as my 350 Mhz Macintosh G3 (which runs at about the speed of a 450 Mhz Pentium). In fact, I first suspected that I'd somehow introduced an infinite loop in my programs. This is supposed to be a *supercomputer*, dude, and we go to great lengths to keep these out of the hands of godless commies.

But, alas, there's a technology lag here. The Silicon Graphics Origin 2000 at the NCSA facility primarily uses 195 Mhz MIPS 10000 processors (plus an assortment of 250 Mhz chips), albeit it has 1520 of these. These are circa 1997, and given that the personal computer industry moves in "dog years"—one year of computer innovation equals seven years of most other technologies—equipment can become outdated very quickly. The relative speed of my 1999 Mac G3 and the 1997 NCSA processors corresponds almost exactly to "Moore's Law"—computer capacity doubles every 18 months.

The *Chronicle* article asserted that the NCSA processors were each "50 per cent faster than the best desktop computer currently available." This statement is certainly not true now, and probably was not even true in September 1998, though it probably was true in 1997. The current generations of the Pentium, PowerPC and Alpha microprocessors have borrowed extensively from the experience with supercomputer designs, and now have much more in common architecturally with a 1980s supercomputer than with a 1980s mainframe.

One can submit up to five jobs at a time to the NCSA batch queues, but only three of those jobs can run simultaneously, so for single processors, the throughput at the NCSA simply matches that of a PC. Of course, if you are stuck with an older computer, NCSA could be a decided improvement over the machine sitting on your desk. Because this is a batch facility, the wall-clock turnaround for the NCSA facility can run considerably slower than a dedicated PC—this is not an option that will save your APSA presentation at the last minute. Obviously to make significant progress, you've got to get things running in parallel.

Going Parallel

There is an "automatic parallel optimization" option on the C compiler ("-apo") that looks for loops within a program that can be run in parallel. One can also manually mark up the code for parallel processing, but I get the sense that unless you are really, really fond of a program, it makes more sense to just let the compiler handle the optimization.

[A related note: optimizing compilers put a premium on *not* being clever when writing code. Just tell the compiler what you want to do. With contemporary pipelined architectures, trying to second-guess the machine code is almost always counter-productive, and in any case the folks who wrote the compiler almost certainly know the target machine better than you do. Older coders will now appreciate how John Henry felt about that steam drill...]

Asking for more processors increases the delay in the batch-processing queue, and increases the allocated time "billed" for processing. NCSA's accounting algorithm is simple—CPU time times number of processors—so 4,000 hours is equivalent to about 1,500 hours on a fast PC. But that's still a lot of time.

Alas, the -apo option made little difference in the speed of *my* application: The system only made use of three processors, though in my batch submission I requested the use of eight. (I even got a little note from the consultants asking about this anomaly—this is a batch facility, and THEY are watching). The -apolist option on the compiler provides a very readable evaluation of the compiler's assessment of every loop in the program, indicating whether it could be made parallel, and if not, why not.

As it turns out, the Baum-Welch algorithm used to estimate HMMs is essentially a dynamic programming algorithm, which means that calculations on iteration i in most loops are dependent on the results of iteration i-1, so the loops cannot be automatically broken down for parallel execution. The loops that could be parallelized—and the compiler found quite a few—were relatively short and therefore used only a small number of processors. This is probably *not* typical of tasks in statistics—for example most matrix operations lend themselves very well to loop-level parallel optimization.

Enter MPI

But there is an alternative approach: All Monte-Carlo experiments are parallel at a macro level involving the entire program, even if the estimation technique isn't parallel at the micro-level of loops. This opened the possibility of using the other major paradigm in parallelism, message-passing. This approach operates by having various parts of the program execute on separate processors, interacting by sending and receiving chunks of data—"messages." It has been implemented in two widely-used interfaces—PVM and MPI—and NCSA strongly recommends MPI. MPI has over 100 functions and an assortment of tutorials are available on the web (see `http://www.erc.msstate.edu/labs/hpcl/projects/mpi/presentations.html`).

The conceptual trick in programming with MPI is that processors are not running separate programs; they are all running the same program, which does different things depending on the "rank" identify each processor picks up from MPI. Your synapses may adapt to this approach more quickly than mine did. In one of those quintessential formal systems experiences, I spent about half a day going through various MPI texts and documents before realizing that for a Monte Carlo problem, the appropriate MPI implementation was absurdly simple:

Serial implementation:

```
Init_Files();
for (nexp= 0; nexp<N_EXPER; ++nexp) { <the
Monte-Carlo stuff> }
```

MPI implementation:

```
int rank, size;
MPI_Init();
MPI_Comm_size(MPI_COMM_WORLD, &size);
MPI_Comm_rank(MPI_COMM_WORLD, &rank);
Init_Files(rank);
for (nexp=0; nexp<N_EXPER/size; ++nexp)
{ <the Monte-Carlo stuff> }
MPI_Finalize();
```

Five additional lines of code to parallelize a 2,500-line program. So again it seems that Smarr is right about the simplicity.

In this instance, all I'm using MPI to do—and all it needs to do—is run the identical program on multiple processors, but with different random numbers. MPI_Init sets up the system. MPI_Comm_size puts the number of processors available into "size"; that number is actually set when one submits the batch job. MPI_Comm_rank assigns a number between size 0 and size 1 to each processor; this is subsequently used whenever a process needs to identity itself. MPI_Finalize cleans up at the end. The only modification in the code itself was reducing the number of iterations in the Monte-Carlo loop as a function of the number of processors available.

Is it *really* that easy? Okay, I lied—it's slightly more complicated. Submitting an MPI batch job and managing the files involved some additional adventures with the NCSA operating systems, which took a few experiments before everything worked. I also wanted each of the processes to write to a different file, so the Init_Files (rank) routine for initializing files goes to:

```
char  name_fres[] = "MC.SERB.RR";
    // general file name
name_fres[9] = 65 + rank;
    // append rank indicator (A, B, C,)
    to the file name
fres=fopen(name_fres,"w");
    // open the file
```

Meanwhile, the familiar technique for initializing the random number generator—srand(time(NULL))— results in all processes using the same seed, because they start simultaneously (duh...). Changing this to srand(time(NULL)*(rank+1)) solves that problem. But the potentially messy stuff—for example the fact that multiple processors will be reading from the same input file—is handled automatically.

This is an extreme case—the term-of-art applied to this situation is "embarrassingly parallel." As it happens, I was already writing the Monte-Carlo results to a file and analyzing them with a separate program. Merge the various output files from the parallel processes, and I've got the same thing. The separate programs are oblivious to each other's existence; the only message they need to pass is to me.

I would emphasize that MPI can do a *lot* more than this, and there is nothing clever going on in this implementation. But it gets the job done with minimal modifications to the serial code, and in fact is as efficient as parallel execution can get. Resampling problems are just as embarrassingly parallel as Monte Carlo, and bootstrapping techniques come close, so there are a lot of opportunities here.

With MPI, one is now in a position to exploit some fraction of the 1520 processors available at the NCSA, and once one gets through the batch queue, the clock-time required to run a program drops proportionately with the number of processors used. This can be quite dramatic.

Overall evaluation

Because it is possible to use the NCSA resources with minimal hassle—I just want the results, not to adapt to yet-another-computer-configuration—the NCSA appears to be a viable resource for running very long jobs. My guess is that most computationally-intensive processes of interest to political methodologists are substantially parallel either in their inner-most loops (matrix manipulations) or their outer-most loops (Monte Carlo methods, resampling). Both can be adapted to parallel execution with a minimum of additional programming. And remember, the time is free.

If you aren't working with C/C++ or Fortran programs, NCSA has a variety of mathematical, linear algebra and numerical optimization packages. The list of software can be found at `http://archive.ncsa.uiuc.edu/Apps/SoftwareListing.htm`. However, you will need to write programs to use these in statistical work, as no statistics packages are available on the systems. (In 1999, S-Plus was "under consideration" and NCSA had "part of" SAS, but both of these packages have now disappeared from the software listings.)

Finally, you may also want to check around your university to see if there are parallel computers lurking around on campus: their cost has come down to the point where they are being acquired by individual institutions. At Kansas, the Chemistry Department acquired an SG-Origin 2000 with 64 300 Mhz and 400 Mhz processors, and grudgingly gave me an account on it. Since this machine is actually faster than the NCSA facility, and I figure it was paid for with excess indirect costs charged to social science grants anyway, I'll probably use it rather than the NCSA for future parallel work.

Postscript: Beowulf Clusters

In the two years since I wrote this, "Beowulf clusters"—parallel computers constructed from off-the-shelf PCs running Linux and linked with standard Ethernet connections—have emerged as a major competitor to classical supercomputers. The web site `http://www.beowulf.org/` provides all of the information required to build one of these, as well as links to about 100 installations. A similar system called "Appleseed"[1] builds clusters from Macintoshes. Both the Beowulf and Appleseed systems use MPI to implement parallel processing, so embarrassingly parallel implementations such as Monte Carlo should be simple to deploy. Keep your eye on the dumpsters behind your Business School and you might be able to acquire a dozen or two computers and build your own supercomputer.

[1](`http://exodus.physics.ucla.edu/appleseed/appleseed.html`)

The LaTeX Corner

Using LaTeX for the first time: A Text-Processing System

Chan H. Nam
Pennsylvania State University
cnam@polisci.la.psu.edu

If you are like me and learned to use LaTeX on your own, you may have found it to be an unnecessarily frustrating venture. The problem is not that LaTeX is so technically difficult to use as that it is so *unusual* for most of us accustomed to the world of Microsoft software. Further complicating the task – the documentation tends to be too specific, when novices need a more general overview of what LaTeX actually is. With this thought in mind, I wrote this brief introduction as a overview for new users, so that they can become better acquainted with the unusualness of LaTeX, and eventually have an easier time learning to use it.

This essay may be useful for those unfortunate souls who are as of yet uninitiated to the wonderful world of LaTeX (pronounced lay-tech) text-processing, or for those of you who know of LaTeX, but never saw much reason to devote the time and energy required to learn to use it. More sophisticated users may wish to stick around for future issues, where *TPM* plans to devote a section on useful tips, and submit tips for publication in our very own LaTeX Corner.

The aim of this guided tour is to answer some questions about how LaTeX works, how to get LaTeX for your computer, and how you can become an expert on your own, with some introduction to the more frequently used features. Because LaTeX is a public domain program with a loosely organized community of users, answers to these questions are not always easy to get. Therefore the first and best piece of advice I will give you is to track down someone who you suspect already uses LaTeX and to get help from them.[1] Having said that, while it might be more difficult to try on your own it is certainly not a terribly difficult task, and I will try to reduce some of the cost of learning by focusing my discussion on overcoming beginners' hurdles.

I assume no prior knowledge from the reader, and will tailor most of my advice for users of Windows based machines. I somehow suspect that most Unix users are already well experienced, and probably are more qualified than I am to write this essay. I am not well acquainted with MacIntosh based operating systems to be of help in that regard, but LaTeX is not linked to any particular computer architecture or operating system such that most of what follows should apply to all machines.

What is it and how does it work?

LaTeX is a public domain typesetting system based on the TeX formatting engine built by Donald Knuth in the late 1970's. LaTeX is unlike most windows based word processors, in that writers do not have to visually format their text. This is after all what we do in MS Word, when we interactively choose the location and shape that our writing takes, while we simultaneously dream up the content of our writing.

[1]Some clues as to who these individuals might be can be found in the articles they write. If you see a lot of math and formula in their pdf articles, which you are certain Microsoft Word cannot produce, they are most likely using LaTeX.

LaTeX is a digital typesetting program that determines the proper layout of various aspects of the document, including things like the size and location of footnotes and section headings, for the writer. The writer produces the content, and with some minor intervention from the writer, LaTeX determines the proper layout for this content. LaTeX does this by taking a plain ascii text file—the input source file (a file with extension .tex)—prepared in any text editor, and generating an output file (a file with extension .dvi) that arranges the content from the input file. LaTeX determines the type and arrangement of text (the specific character size, font, etc.) and placement in the document, by referring to predetermined typesetting rules for the kind of document being written (article, book, essay).[2] The great advantage of this system is, of course, that this form of word processing forces the writer to focus more on the logical structure of the document and ideally on the content of the writing. The disadvantage is that the writer has to intervene with the general layout of the document, whenever he desires a change.

Writing LaTeX files

The input source file (.tex file[3]), is a plain text file that contains the content of the writing, and it is also the only place where the writer can invoke commands that define aspects of the document. There are no drop-down menu's in LaTeX from which to select commands, therefore all commands about typeface, special characters, sectioning, and so on, have to be included here. Furthermore, all .tex files must have three commands, and commands are case sensitive and start with the backslash character '\'. All .tex files must include the `\documentclass{...}` command that declares the type of document that is being written, the `\begin{document}` command that declares where the text of the document begins, and the `\end{document}` command that declares the end of the text.

A simple .tex file with some text would look like this:

```
\documentclass{article}
\begin{document}
Text goes here.      An empty
line produces a new paragraph.

No matter how many spaces you type\\ between
words, latex determines     how many spaces
to include.
\end{document}
```

And would produce an output file that looks like this on a full page:

Text goes here. An empty line produces a new paragraph.

No matter how many spaces you type
between words, latex determines how many spaces to include.

There are a few things to notice from this example. An empty line creates a new paragraph while the command \\, produces a new line, same as typing `\newline`; and it doesn't matter how many spaces you type between words, LaTeX produces the appropriate spacing, for example between words in the middle of sentence, at the end of a sentence and the beginning of a new one, and after a colon.

You are probably wondering how to produce the backslash character, when it always denotes the start of a command in LaTeX. There is a set of characters that have special meaning in LaTeX and that will produce things that you will not have intended. To get LaTeX to produce the following characters,

```
 #  $  %  ^  &  _  ~  {  } \ "hi"
```

they have to be typed like commands – also make sure you differentiate between open quotations and closed quotation marks.

```
\# \$ \% \^{} \& \_ \~ \{ \} $\backslash$ ``hi''
```

The '{}' are used to define environments; the '&' denotes sections in tables; '#' is a macro parameter; the '%' makes LaTeX ignore the rest of the line (useful for writing comments that will not be produced in the final document); the '$' is used to signal the use of math, where subscripts, superscripts, Greek characters, and formulas within the text[4] require it; the '^' creates a superscript, for example typing `$i^{2}$`, will produce i^2; '_' creates subscripts, for example typing `$i_{2}$` will produce i_2.[5]

Changing typeface in LaTeX is easy. To make boldface characters, you invoke the `\textbf{text goes here}` command and insert the desired text within the brackets. The example above will produce, **text goes here**. Similarly the `\textit{...}` produces *italics*. Underlining is a bit trickier, since the original creator of TeX shunned this kind of typesetting faux pas. It can, nonetheless, be done in math mode by the combination of the text upright and text underline commands. For example this `$\textup{\underline{see}}$` command produces an underline, $\textup{\underline{see}}$?

A few examples of math and Greek

You can type math directly into your text, but remember that math always has to be flanked by the $

[2] Many of the these functions can be modified by the user.

[3] You can name this file anything you want, but it must end with .tex, for instance this essay is being typed on a file named introlatex.tex.

[4] The equation environment does not require the use of the $.

[5] Notice that math within the text of a document is always produced in italics.

character in front and in the end. For instance the Greek character μ can be produced by typing `$\mu$`. Similarly you can put a hat on your $\hat{y}$, by typing `$\hat{y}$`, where the content in the brackets can be any character. To type fractions you use the `\frac{...}{...}` command, where the numerator is typed into the first set of brackets, and the denominator into the second. For example `$\frac{\Omega + \alpha}{\beta}$` produces $\frac{\Omega+\alpha}{\beta}$. This kind of math is a bit underwhelming. To get better looking math, you can use the equation environment.

To define special environments within your .tex files, you insert the `\begin{...}` to start a new environment and `\end{...}` to end it.

For example to get the poisson formula, I would type...

```
\documentclass{article}
\begin{document}
This is the poisson formula.
\begin{equation}
P(y_i|x_i)= \frac{exp(-\mu_{i})\mu_{i}^{y_{i}}}
{y_i!}
\end{equation}
\end{document}
```

and get...

This is the poisson formula.

$$P(y_i|x_i) = \frac{exp(-\mu_i)\mu_i^{y_i}}{y_i!} \tag{1}$$

What to do with your .tex file

So you've typed-up your .tex file in a text editor, now what do you do with it? To view the result of your work, you have to run the latex.exe file in the command prompt[6] on your .tex file (make sure latex.exe is in the path, or at least that you are in the same subdirectory that latex.exe is). If you are using WinEdt you can simply click on the LaTeX button and the active document will be LaTeXed. Doing so, will create a number of files, one of which is the .dvi file (latex.exe on xxx.tex, will yield xxx.dvi) that can be used to view or print the document. To view the result in the .dvi file you have to use a .dvi viewer. Your LaTeX package set will come with a .dvi file viewer, MiKTeX for instance comes with YAP. Yap can be invoked from the DVI button in WinEdt, or you should be able to find it in the MiKTeX menu in Windows. You can also use Yap to print your documents from .dvi files.

Another great feature of LaTeX is the ability to make Adobe Acrobat files and postscript files. There are two ways in which you can create pdf files. You can run PDFLaTeX on your .tex file, (by running pdflatex.exe in the command prompt on your .tex file), or you can run dvipdfm on your .dvi file (by running dvipdfm on your .dvi file). I personally prefer the latter, because although it adds another step in the process, I seem to encounter more problems using pdflatex than dvipdfm. WinEdt saves you the trouble of typing these commands, if you click on the pdflatex or dvipdf buttons that it provides. To make postscript files you can run the dvips.exe file in the manner described above, and once again there is a dvips button in WinEdt that will do this for you.

How to get LaTeX for your computer

You will need to install three sets of programs in your computer to start LaTeXing. You will need an implementation of LaTeX that runs on your type of computer system, a postscript interpreter like Ghostscript and a viewer like Ghostview,[7] and a text editor like WinEdt or Winshell. Ghostcript and Ghostview are free for noncommercial users, WinEdt is shareware and will run for 31 days before beginning to issue annoying but harmless registration requests with increasing frequency.

There are a number of ways of getting LaTeX for your computer and a few implementations of LaTeX for different types of operating systems and with varying features. I recommend using the MiKTeX distribution, a Windows implementation of LaTeX and most widely used tools. The best way of getting MiKTeX is by downloading it from their website,[8] at no cost. At about a 23 megs for the small package, this may be too much for some users without a fast internet connection. An alternative to downloading is purchasing a cd from them, at $13.50. Installation is fairly easy – you download a setup program (`http://prdownloads.sourceforge.net/miktex/setup.exe`), which takes care of downloading the MiKTeX package you choose. There are three packages, small, medium, and large – unless you are a very experienced user the small package should do fine. Once you have downloaded the package set, you have to run setup again to install MiKTeX. Further instructions can be found at the MiKTeX website.

WinEdt[9] is a good text editor that makes using LaTeX a bit easier. Of course, you can type your input files in any text editor as long as you save your files in ascii format and with the .tex extension. But I strongly recommend using WinEdt or Winshell because they provide a front end that takes care of the LaTeX commands that you will otherwise have type into the old command

[6]If you are unfamiliar with the command prompt, the command prompt is a throw back to the old DOS operating system days, and can be accessed from the accessories menu of Windows 2000.

[7]The latest versions of Ghostscript and ghostview for windows can be downloaded from `ftp://mirror.cs.wisc.edu/pub/mirrors/ghost/AFPL/gs703/gs703w32.exe` and `http://www.cs.wisc.edu/~ghost/gsview/get40.htm` respectively. Installation instructions and links can also be found in the main website at `http://www.cs.wisc.edu/~ghost/`

[8]`http://www.miktex.org/`

[9]Note: there is a similarly named but completely unrelated program called WinEdit. WinEdt can be downloaded from `http://www.winedt.com`.

prompt. WinEdt has a template of buttons that will run LaTeX commands for you, as well as icons for various symbols and commands that you can insert into your input file. Although Winshell[10] is free, and has some of WinEdt's functions, it is more limited. Another alternative for price conscious users is the Emacs editor familiar to Unix users.[11] WinEdt registration for student users is $30.

Where to go from here

So far I have only reviewed a small fraction of the features available in LaTeX. There are many other useful and important functions that users should learn. Commands to section articles, the tabular environment that makes tables, and many packages that allow, for instance, the use of hyperlinks and graphics are but a few of these. Rather than continuing to dribble on about LaTeX in such a limited and random manner, it would be more beneficial to point interested readers to the widely available documents and resources that detail these feature more completely.

The most important resource for all things TeX is the Comprehensive TeX Archive Network, at `http://www.ctan.org/`. It contains links to various programs as well as links to many instructional documents. A good place to start is the "The not so short introduction to LaTeX2e" by by Tobias Oetiker, Hubert Partl, Irene Hyna and Elisabeth Schlegl, available at ctan under the search link. The TeX Catalogue Online maintained by Graham Williams is another important resource for a huge number of LaTeX packages and tools. This latter resource is where most style files are described and where they can be accessed. Also, once you download LaTeX you can find helpful documents in your own hard drive. There are many .dvi and pdf files within the doc subdirectory in the LaTeX directory, texmf, that can be very useful. Also, MikTeX comes with a mediocre local guide, which is good at times when you want something easily searchable.

There are many books on LaTeX – the most popular are Lamport's (1994) and Goossens, Mittelbach, and Samarin (1994). Lamport's is a good beginner's guide, while Goossens, et al. provide a discussion of the more advanced features to LaTeX. While Lamport (1994) is a good source for the fundamentals, it isn't a must read. Most of what is covered in Lamport can is found and learned elsewhere. I would recommend Goossens, et al. as the more useful guide to advanced features, and you will most likely need a reference guide to properly use them. But if you are really having difficulty with LaTeX then, Lamport would be very helpful.

[10]Winshell can be downloaded from the TeX Archive Network, at `http://www.ctan.org`.

[11]If you are interested in enabling your Emacs editor to issue LaTeX commands, you might find information at `www.ctan.org`, the TeX Archive Network.

References

Michel Goossens, Frank Mittelbach and Alexander Samarin. 1994. *The LaTeX Companion.* Addison-Wesley, Reading, Massachusetts.

Leslie Lamport. 1994. *LaTeX: A Document Preparation System.* Addison-Wesley, Reading, Massachusetts, second edition.

Review of Wooldridge's *Introductory Econometrics*

George A. Krause
University of South Carolina
george.krause@sc.edu

Finding the appropriate econometrics text for training political science graduate students depends upon whether the intent is to train individuals as competent applied empirical researchers, or those with an advanced background that is necessary for having the skills to become a political methodologist. The new text *Introductory Econometrics: A Modern Approach (IE)* by Jeffrey Wooldridge is the type of text that falls squarely into the former category. Like Damodar Gujarati's popular text *Basic Econometrics (BE)*, it is marketed to an undergraduate and non-technical graduate student audience that does not require familiarity with matrix algebra nor differential and integral calculus. The mathematical sophistication of *Introductory Econometrics* lies somewhere between *Basic Econometrics* and Jan Kmenta's *Elements of Econometrics* (1998), with it being much closer to the former text given the latter's use of calculus and matrix algebra[1]. As a result, any comparisons that are made in this review are limited to Gujarati's *Basic Econometrics.*

Wooldridge has done an excellent job of writing an accessible introductory regression/econometric analysis text that retains the easy-to-follow applications that are the real strength of *BE*, yet provides new applied wrinkles. In addition, *IE* provides a more thorough analytical treatment of the concepts without placing heavy technical requirements on the reader. For instance, *IE* provides a more theoretically satisfying treatment of the general linear (OLS) model by showing its analytical connection to the methods of moments. Another unique feature of

[1]More technical treatments of introductory econometrics can be found in texts such as Greene (2000), Davidson and MacKinnon (1993), but to name a few. Gallant (1997) is a rigorous mathematical treatment emphasizing a pure (asymptotic) theoretical approach to the topic.

IE is the author's focus on disturbance terms and conditional expectations as a theoretical vehicle for understanding the general linear model and corresponding departures from it. This treatment is more theoretically satisfying than found in other non-technical texts such as *BE*, which fail to provide one with an adequate conceptual understanding of regression analysis since its emphasis is almost exclusively applications-oriented. At the same time, political science graduate students should not find Wooldridge's analytical treatment as being esoteric since the author carefully intertwines the theoretical with data illustrations presented in a "non-technical" scalar algebra form.

IE exhibits more breadth and depth than *BE* in several ways. *IE* spends an entire chapter on the consistency and efficiency of statistical estimators, plus one finds more detailed information throughout the text on these issues as well as the unbiasedness property than what is found in *BE*.[2] *IE* also expends considerably more effort in handling the analytic issues underlying time series analysis, as well as dealing with the testing of general linear restrictions in the general linear model. *IE* also spends brief, but effective time discussing issues of functional form and the interpretation of nonlinear relationships (e.g., finding the maximum or minimum value for the dependent variable for a regression specification containing a quadratic functions). Further, *IE* also has a more intuitively straightforward way of interpreting marginal effects from Logistic and Probit regression models compared to *BE*. The author provides a clear discussion analyzing the distinction between censored versus truncated samples that gets short shrift in *BE*.

There are several important features that *IE* possesses that are lacking altogether in *BE*. Wooldridge dedicates two chapters to panel data designs and estimation, although the treatment is rudimentary in nature (Fixed and Random Effects, and Simultaneous Equation Models). Moreover, *IE* not only makes note of White, Newey-West, and quasi-maximum likelihood estimation (QMLE) robust standard errors, but also shows the reader how they are analytically derived! Still yet, *IE* goes above the treatment of *BE* in the area of limited dependent variable models by providing a thoughtful discussion of Poisson regression and how to handle overdispersion within a QMLE context as well as incidental truncation resulting in the use of the Heckit two-stage sample selection procedure. Finally, *IE* contains a detailed 15 page glossary near the end of the text that provides helpful definitions for concepts ranging from "Adjusted R^2" to the "Zero Conditional Mean Assumption" that students will find extremely valuable.

[2] *BE* devotes only two pages to discussing this issue with respect to BLUE. Furthermore, *IE* integrates these concepts with integrated illustrations to show how these issues manifest in actual empirical research.

Jeffrey Wooldridge's *Introductory Econometrics* does an admirable job of conveying a sound working knowledge of regression analysis by providing analytical depth and subject breadth that is typically not found in a non-technical econometrics textbook. While *Introductory Econometrics* is not a suitable text for those wishing to provide their students with a more rigorous treatment of regression analysis requiring use of matrix algebra and/or asymptotic probability theory, it provides greater analytical rigor and broader topical coverage than Gujarati's *Basic Econometrics* and texts of a similar nature.

References

Davidson, Russell, and James G. MacKinnon. 1993. *Estimation and Inference in Econometrics.* New York: Oxford University Press.

Gallant, Ronald A. 1997. *An Introduction to Econometric Theory.* Princeton, NJ: Princeton University Press.

Greene, William H. 2000. *Econometric Analysis.* Fourth edition. New York: Macmillan.

Gujarati, Damodar N. 1995. *Basic Econometrics.* Third edition. New York: McGraw-Hill.

Kmenta, Jan A. 1998. *Elements of Econometrics.* Third edition. Ann Arbor, MI: University of Michigan Press.

The Society for Political Methodology

ad indicia spectate

Section Activities

Political Analysis Moves to Oxford University Press

Nathaniel Beck
University of California, San Diego
beck@ucsd.edu

Political Analysis has now completed its first two quarterly volumes (8 and 9). These were published by Westview Press. Starting with Volume 10, we will be published by Oxford University Press. Our contract with Oxford runs from Volume 10 through 19. The first Oxford issue, 10(1) is scheduled to mail on Feb. 1, 2002 (with all articles for that issue currently sitting with the copy editor). Articles for 10(2) are due at the press on Dec. 10, 2001.

We currently publish about 20 articles per year. The article submission rate is irregular, but is approximately 50 per year. This gives us a relatively high acceptance rate, but there is also good self-selection by authors.

Oxford is undertaking a strong promotional program, both in the US and outside. Royalties to the section are based on library sales. Oxford will soon be asking section members to talk with their university librarians to make sure that we have good library coverage at least in institutions of members. You will be receiving a mailing from Oxford about this very soon. Libraries do not acquire new journals without strong faculty input.

Oxford is also helping us in other important ways. Of most importance, I expect that we shall very soon be covered in the Social Science Citation Index and then Current Contents. Oxford also has its own mailing list for widely distributing our Table of Contents for each issue.

Our on-line presence will change a bit. The electronic version of *PA* will be published by Highwire Press, the leading on-line publisher. Oxford will maintain the actual web site. Members will receive a password to that site; libraries that subscribe will allow their university communities access to the site in the usual manner. Members and libraries will, of course, continue to receive hard copies. As is current practice, the on-line version will contain ancillary materials, data sets and full replication articles. Volumes 8 and 9 will remain, through the courtesy of our webmaster Jeff Gill, on the polmeth web site (`web.polmeth.ufl.edu/pa/pa_main.html`).

In terms of submissions, I am particularly interested in publishing issues that join related articles. These are not special issues in that there is no guest editor and all articles are reviewed individually; in addition, there is no need for the grouped articles to occupy a full issue. But I do think that publishing related articles in new arenas is a good use of my space. Thus, 9(3) was made up of articles related to legislative scaling, and in 10(1) I have three articles on analyzing multi-party elections and two on dynamic panels. Future plans include an issue on experimental methods and another on spatial methods. I am also working on some articles that relate to the joining of qualitative and quantitative methodology. I am, of course, open to suggestions (or joint submissions) in other new or currently exciting areas of research.

And, of course, the quality of the journal depends on high quality submissions of any kind. *PA* publishes all articles that are related to methodology, broadly defined. If you are unsure if your article is suitable, I am happy to look at a submission and discuss with the author whether it fits. I am also happy to take articles that are too short for standard journals (and can work with authors of papers that are too long).

So when asked what you can do for your journal, the answer is: submit, review and work on your librarians. Thanks.

Notes from the Section President

Jonathan Nagler
New York University
jonathan.nagler@nyu.edu

The Political Methodology Section continues to move along in good shape. The most notable development of the past year was the continued success of *Political Analysis*, and its move to Oxford University Press. We expect this will enhance the visibility of the journal. If you are reading *TPM*, then chances are you are a section member and already receive *Political Analysis.* But in order to keep your colleagues from borrowing your copy of *PA*, and to make *PA* more accesible to your graduate students, you should have your university's library subscribe to *PA*! If your library does not get the journal, you will hear from us.

The Political Methodology Summer Meeting at Emory University was successful. Many thanks to David Davis, Chris Zorn, and everyone at Emory who took care of the needs of over 100 visiting political scientists. The 2002 meeting will be at University of Washington in Seattle. A request for proposals to participate in the 2002 meeting will be posted in January. Anyone interested in hosting the 2003 meeting at their university should contact me for information regarding what the host institution is expected to provide, and information on the range of costs that previous host institutions have incurred.

But *Political Analysis*, the summer meeting, and of course *TPM* are not the only things the section does. Jeff Gill continues to serve as section webmaster. The latest addition to the website is an archive of methodology syllabi, covering both quantitative and qualitative methodology.

A mystery of the website is the low number of papers submitted. All other indications are that the productivity of people in the section has been increasing. So yet another reminder that there are many benefits to posting: 1) you can get useful feedback; 2) people can find out what an innovative scholar you are; and 3) when committees in charge of awarding lucrative cash prizes are reviewing worthy papers they tend to look at the website to find the year's best papers.

The section has usually provided new services through individual initiative. We continue to welcome new proposals. Political Science still lags behind other disciplines in not having keywords to index things by. Fixing that would be useful.

2001 APSA Political Methodology Panels

Bradford S. Jones
University of Arizona
bsjones@email.arizona.edu

The 2001 American Political Science Association Meetings were held in San Francisco, CA. The Political Methodology group had seven panels that dealt with a variety of issues ranging from vote irregularities in the 2000 presidential election to methods and models for legislative behavior. About 28 papers were given dealing with an extensive array of issues in political methodology. Apart from the seven panels, our section was well represented in the poster session, with about 20 political methodology posters.

As coordinator for our section's panels at the 2001 annual meeting, I was happy to see that nearly all of the panels were well attended with some of the panels approaching "standing room only" status. As past, present, and future APSA panel organizers have stressed, our section's allocation of panels for subsequent APSA meetings is strongly related to attendance at the previous year's meetings. Of course the best way to ensure attendance is to have a slate of outstanding papers and posters. For the 2001 meetings, we satisfied this condition and so attendance was quite good. Thanks to all who presented, discussed, and attended.

Reflections on the Eighteenth Annual Summer Meeting

David Davis
Emory University
poldd@emory.edu

Christopher Zorn
Emory University
czorn@emory.edu

When we were asked to write this column, the initial impulse was for something along the lines of the following:

"The Eighteenth Annual Political Methodology Summer Conference was held at Emory University on July 19-21, 2001. 103 participants attended this year's meeting, including 56 faculty and 47 graduate students from more than 40 different institutions around the U.S. and the world, etc. etc."

We even had plans to spice it up with a little humor, e.g.:

"Professor Issen Teer (Miskatonic University) delivered the keynote address, which was as brief as it was unmemorable."

Instead, we thought we'd share a few thoughts about the meeting, from the perspective of the organizers. Some are simply observations; others will look more like advice (or perhaps suggestions) for future meeting planners. All are our own, and do not reflect the views of the Emory Department of Political Science, Emory University, the Atlanta Chamber of Commerce, the state of Georgia, or the editor of this newsletter.

1. *The meeting isn't small anymore.* Looking at the population growth rates alone, an ecologist might mistake *homo methodologus* for a particularly virulent strain of bacteria. From an organizational perspective, arranging lodging, meals, and so forth for 120 is far different from doing so for a group of 12. Despite the challenges it posed, we felt the meeting came off well, and the size was not unmanageable. At the same time, were it to grow any further, the Society may want to consider changing its format somewhat (e.g., to relieve organizers of responsibility for securing lodging and meals).

2. *We're not as mean as everyone says.* A remarkable air of civility prevailed throughout the paper presentations: there seemed to be fewer interruptions, calmer voices, and a near-absence of people walking around with their guts for garters.

3. *If you're running the meeting, forget about attending the panels – or sleeping.* This is a corollary to (1); the logistics for a meeting this size require the full-time attention of at least one, and realistically two, people. For the 2001 meeting, Davis handled most of the local arrangements (catering, venues, etc.) while Zorn addressed "external" matters (participants, program, etc.), though there were significant exceptions and a good deal of overlap as well. This arrangement worked well for us.

4. *Graduate students are a lot smarter than they used to be.* The consensus of the organizers, following the poster session, is that neither of us would like to have to compete against the current ABD crowd for academic jobs with a methodology component. We suspect we're not the only ones relieved about this.

5. *Money* ***can*** *buy happiness.*The generous support of Emory, the section, and others made our lives substantially easier before, during, and after the meeting. This point is aimed primarily at all those readers who are considering hosting the meeting yourself sometime in the future: Overbudget everything. Moreover, remember that adequate institutional support is key, since it often comes with the fewest strings attached.

6. *The Bayesians are taking over.*

7. *You can't do it alone.* This is the place where we thank the people who made the meeting possible: Charles Franklin, Jonathan Nagler, Jeff Gill, and the Political Methodology section of the American Political Science Association; Tom Walker, then-Chair of the Department of Political Science; Steven Sanderson, former Dean of

Emory College; Robert A. Paul, past Dean of the Graduate School of Arts and Sciences of Emory University, and the National Science Foundation. We also greatly appreciate the hard work of our intrepid graduate students – Melissa Cannon, Amy Williamson, Zaryab Iqbal, Susan Allen and Jesse Hamner – and of the Political Science Department staff – Denise Brubaker, Elizabeth Fricker, Anthony Matthews and Esther Nerenbaum – all of whom were instrumental to making the meeting a success.

Announcement for The 19th Annual Summer Meeting

The 19th Annual Summer Meetings of the Political Methodology Section of the American Political Science Association will be held from July 18th to July 20th, 2002, at the University of Washington, Seattle, USA. The local sponsors of the meeting include the Center for Statistics and Social Science (`www.csss.washington.edu`), the Department of Political Science (`http://www.polisci.washington.edu`), and the College of Arts and Sciences. Local organizers are Kevin Quinn (`quinn@stat.washington.edu`) and Michael Ward (`mdw@u.washington.edu`). Details about the conference, as well as registration materials will be posted on the (`www.csss.washington.edu`) web site in the near future.

Section Awards

Three awards were were given at the section business meeting at APSA, August 31 2001. The winners were:

Kevin Quinn, University of Washington &
Andrew Martin, Washington University

The 2001 *Gosnell Prize* for "Bayesian Learning about Ideal Points of U.S. Supreme Court Justices, 1953-1999."[1] The *Gosnell Prize* is awarded for the best work in political methodology presented at any political science conference during the preceding year.

Joshua Clinton, Stanford University

The *Society for Political Methodology Poster Award* for "Representation and the 106th Congress: Legislators' Voting Behavior and their Geographic and Party Constituencies."[2] The *Society for Political Methodology Poster Award* is given for the best poster in the area of political methodology at any political science conference.

Keith Poole, University of Houston

The *2001 Miller Prize* for "The Geometry of Multidimensional Quadratic Utility in Models of Parliamentary Roll Call Voting."[3] The *2001 Miller Prize* was awarded for the best article published in *Political Analysis*, Volume 9. The Miller Prize is named for the late Warren Miller, who early on understood the importance of good methodology for the study of politics, and who was instrumental in providing support for the formative meetings of what was to become the Society for Political Methodology. The award carries an honorarium of $500.

[1] `www.stat.washington.edu/quinn/papers/papers.html`
[2] `http://www.stanford.edu/~jclinton/`
[3] `http://web.polmeth.ufl.edu/pa/vol9no3.html`

The Political Methodologist
Department of Political Science
Pennsylvania State University
University Park, PA 16802

The Political Methodologist is the newsletter of the Political Methodology Section of the American Political Science Association. I gratefully acknowledge the support of the Department of Political Science of the Pennsylvania State University in helping to defray the editorial and production costs of the newsletter.

Subscriptions to *TPM* are free to members of the APSA's Methodology Section. Please contact APSA (202 483-2512, `https://www.apsanet.org/about/membership-form-1.cfm`) to join the section. Dues are $25.00 per year and include a free subscription to *Political Analysis*, the quarterly journal of the section.

Submissions to *TPM* are always welcome. Articles should be sent to the editor by e-mail (sdeboef@la.psu.edu) if possible. Alternatively, submissions can be made on diskette as plain ascii files sent to Suzanna De Boef, Department of Political Science, 108 Burrowes Building, Pennsylvania State University, University Park, PA 16802. LaTeX format files are especially encouraged. See the *TPM* web-site [http://web.polmeth.ufl.edu/tpm.html] for the latest information and for downloadable versions of previous issues of TPM.

TPM was produced using LaTeX on a PC running MikTex and WinEdt.