Do ideologically intransigent parties affect the policy positions of other parties?

And … rigorously characterizing output from computational models of party competition

Michael Laver and Ernest Sergenti
Department of Politics
New York University

ABSTRACT
Building on a particular agent based model (ABM) of party competition, we aim to do two things in this paper. First we aim to make a clearer distinction between using ABMs as discovery tools, on one hand, or as platforms for systematically designed suites of simulation experiments that offer computational solutions when the underlying model is analytically intractable, on the other. Second, we aim to work towards a set of methodological standards for the design of simulation experiments designed to further the computational analysis of a model that is intractable using conventional formal analysis. Throughout the paper we maintain a substantive focus on party competition and address a particular problem in this area – the extent to which the presence of an intransigent and unresponsive ideological “sticker” party, located at one extreme of the policy spectrum, results in outcomes whereby more responsive parties also adopt more extreme policy positions. We consider both unimodal and bi-model distributions of voter ideal points in investigating this problem. We find the effect on other parties of an intransigent party is greatest with moderately polarized bimodal voter densities, especially when one mode of voter ideal points is smaller than the other.

Paper for presentation at the Annual Meetings of the American Political Science Association
Chicago 2007
INTRODUCTION

This paper is motivated by two self-evident truths about multi-party competition in a multi-dimensional policy space. First, most informed observers of real politics do not for one second think of party competition as a system that is either at, or en route to, static equilibrium. They think of party competition as a dynamic process that continually evolves. Indeed most people would probably be quite alarmed in substantive terms by the notion of a party system in equilibrium – a system flat-lining in steady state and perturbed only by random shocks. Second, the complex system generated by the dynamic process of party competition is analytically intractable. It is intractable for analysts and, with far deeper implications, intractable for real decision-making agents.

Intractability has two important implications. First, if general results are sought, intractability for the analyst implies a methodological shift from formal analysis to computation – noting that computation does not necessarily involve using electronic computers, though almost invariably does. Second, as an empirical matter, intractability for real agents involves a substantive shift in the most plausible behavioral assumption about agents’ decision-making inside the complex system – from deep strategic look-forward to adaptive learning. It is vital to maintain a clear distinction between these two different implications of analyzing a complex dynamic system such as that generated by multidimensional, multi-party competition (MDMPC). There are computational models of party competition that are not based on an assumption of adaptive learning (Jackson 2003; Smirnov and Fowler 2007). And there are adaptive agent models of social behavior that do not need to be implemented computationally (Schelling 1978). When it is possible to resolve a substantively important problem using formal analysis, few would argue computation is the better way to go. However, since so much of real politics involves complex and/or dynamic and/or non-linear interactions, this leaves a huge intellectual terrain that can be investigated by computational methods but not tractable formal analysis.

Agent-based models (ABMs) are almost invariably, though not inherently, computational models of complex dynamic systems that make the behavioral assumption of adaptive learning by real agents, as opposed to assuming deep strategic look-forward. Compared to classical formal models, therefore, they simultaneously make the methodological move to computation and the substantive move to the assumption of adaptive learning by real agents. Both moves, which are essentially epiphenomenal, are made with the intellectual aim of generating theoretically tractable (use computational methods) and empirically realistic (assume adaptive learning) accounts of a complex process such as MDMPC.

Notwithstanding the many benefits of computational models of complex political processes, there are also costs. Well-established professional canons set out what constitutes good formal analysis. These are taught in the formal modeling courses of good graduate programs and policed with great rigor by referees for good journals. Computational modeling is much newer, becoming a feasible technology for the intellectual mainstream only with very recent advances in computing power and, more importantly, the software to exploit this new resource. There are far fewer established canons setting out what constitutes good computational modeling in the social sciences. Thus our main methodological motivation in this paper is to do two things. First we aim to make a clearer distinction between the use of computational models as discovery tools, as opposed to their use in deriving computational solutions to analytically intractable problems. Second, we aim to work towards a set of methodological standards for the design of the suites of simulations that underpin any rigorous computational analysis.

---

1 By “formal” or “classical” analysis in this paper we mean the analysis of models that assume agents follow the rational choice axioms and are investigated using analytical techniques including (but not limited to) non-cooperative game theory.
This paper also has a substantive focus. We elaborate our methodological arguments by developing and presenting new findings about MDMP in a complex dynamic setting, building on an existing ABM of this (Fowler and Laver 2006; Laver 2005; Laver and Schilperoord 2007). These findings concern the extent to which the presence of an intransigent and unresponsive ideological “sticker” party, located at one extreme of the policy spectrum, results in outcomes whereby more responsive parties also adopt more extreme policy positions. We consider both unimodal and bi-model distributions of voter ideal points in investigating this problem. We find the effect on other parties of an intransigent party is greatest with moderately polarized bimodal voter densities, especially when one mode of voter ideal points is smaller than the other. Before doing any of this, however, we discuss in more detail some of the issues arising from using computational models rather than formal analysis.

INTELLECTUAL RATIONALE FOR COMPUTATIONAL MODELS

Computer simulations as discovery tools
Parsimonious models tend to be easier to analyze in a rigorous way than complicated models; implications are typically easier to interpret. The price paid is that a parsimonious model is sparser, more abstract, less realistic, than a more complicated model of the same thing. This price is usually seen as repaid with handsome profits, given the analytical rigor and clearer intuitions that a good parsimonious model can provide. In the realm of formal analysis, the canon of parsimony is to a large extent self-policing. Theorists have few incentives to complicate a simple model that has been solved and seems to be doing a good job at explaining something of substantive interest. Analyzing a more complicated model is typically more difficult, even completely intractable, while systematically inferring general substantive implications can be more problematic. In stark contrast to this, it is shockingly easy to take a baseline computational model and graft on layer after layer of complication, each layer added in an understandable quest for enhanced realism. Parsimony is not self-policing in computational models and, if the behavior under investigation does indeed seem more realistic as a result of complicating the model, the trade-off between parsimonious but less realistic models, as opposed to complicated but more realistic ones, is drawn in sharper relief.

Thinking of spatial models of party competition, we see a spectrum. At one end are sparse and abstract models that can be solved rigorously only for simple scenarios we find nowhere in the real world. No sane analyst claims the model applies to any real setting. The claimed virtues of such a model are typically analytical rigor and stimulus to useful intuition about real politics. The latter virtue makes it a model of politics rather than a (possibly quite beautiful) construction of abstract statements. In this spirit, variables and parameters in such a model are typically given names drawn from the real political world. Strong conventions have evolved about what constitutes analytical rigor. Useful intuition, however, is inherently subjective, even aesthetic, and subject to the constant ebb and flow of intellectual fashion. This leaves open the possibility that substantive intuitions arising from impeccably rigorous formal models can be claimed much more in terms of informal aesthetics and gut feeling than of intellectual rigor.

At the other end of the spectrum, we see a complicated simulation\(^2\), involving a large and heterogeneous set of variables and parameters we expect a priori have an impact on party competition. Such a model might, in extremis, look like a more or less realistic “SimPolitics” game. The power of modern computers allows us to build and run very complicated simulations,

---

\(^2\) Note that, throughout this discussion, we make a clear distinction between a complicated model and a model of a complex system – there may be complicated models of non-complex systems, and simple models of complex systems.
but is there any well-defined intellectual point in doing so? Setting aside the need to control complicated processes\(^3\) in real time – while nonetheless noting that the ability to do this is an impressive intellectual achievement – the virtues of such a model might be: it is computationally rigorous\(^4\); it is realistic, offering substantive intuitions; it offers the possibility of unexpected discoveries and intuitions, which can emerge short of having a fully solved model. The latter is one of the signal virtues of a fully programmed and working computational model. A good example can be found in one of Axelrod’s famous computer tournaments for strategies in the repeat-play Prisoners’ Dilemma game, where he used the genetic algorithm to evolve completely new strategies from those in the strategy set at the start of the simulations (Axelrod 1997). Strategies emerged from this evolutionary process that resembled Tit-for-Tat, the most successful strategy in previous tournaments. But completely new types of successful strategy also evolved, which beat Tit-for-Tat and which no one had submitted to earlier tournaments. These new strategies were discoveries arising from Axelrod’s simulations although, once discovered, they proved amenable to formal analysis.

**Computer simulations as analysis**

A fundamentally different intellectual rationale for using computational models is as a substitute for formal analysis when the latter is intractable – a typical problem, as we noted above, when the model deals with a complex dynamic system. In this situation, the epistemological roles of analytical and computational models are in essence identical; at issue is how the model is interrogated. In the context of party competition, Smirnov and Fowler (2007) present a computational analysis of an analytical model of party competition credited to Wittman, which has been published for many years and has attracted considerable intellectual attention, but has proved intractable, using formal analysis, in many realistic settings (Wittman 1977). Rather than using analytical techniques to find the partial derivatives of the key variables of interest, Smirnov and Fowler systematically vary these in a suite of simulations, investigating their effect on some output variable of interest. This is a standard method in many branches of the natural sciences, and is increasingly used to gain analytical traction with difficult models in the social sciences. A well designed suite of simulations can reveal as much about the effects of some output variable of interest as can a model solved using formal analysis. This allows us to derive the computational equivalent of analytical comparative statics. The computational approach may be considered uglier and less pure than formal analysis, and it is hard to see why computation would be used when formal analysis can do the job. But computation can offer solutions when formal analysis cannot. In such settings, the choice is between “ugly” simulation and “beautiful” nothing, where rigorous simulation can generate results as solid as those derived using formal analysis, had formal analysis been possible.

Doing this properly may involve very intensive computation and resource constraints quickly bind, even given the power of modern computers. Say we are interested in substantive output from a model with four free parameters, and decide we can safely sweep the feasible range of each parameter by investigating each of 100 evenly spaced values within this range. We thereby define a grid on our four-dimensional parameter space, for which model output must be investigated, with \(100^4 = 100,000,000\) different points. We may need multiple runs, or very long runs, to estimate model output at each point on the grid. We run smack into a computational wall, given limitations in processor power, compounded by banal constraints on our ability to store, manage, analyze and interpret huge volumes of computer output.

There are three basic ways around this problem. The first is to build parsimonious computational models with as few free parameters as possible; adding parameters compounds the parameter-sweeping problem at an exponential rate. Computational models, in this context, face

\(^3\) For example automatic pilots for passenger airliners, or control systems for nuclear reactors.

\(^4\) This point is emphasized by both Epstein (2006) and Miller and Page (2007).
precisely the same intellectual constraints as analytical models – for which the over-riding
professional canon is that parsimonious models are better. A second way round the computational
wall is to sweep parameters on a coarser grid – perhaps investigating 20 evenly spaced values
rather than 100. In our hypothetical example, even retaining four free parameters, we thereby cut
down the grid to \(20^4 = 160,000\) points, a lot less than 100,000,000. Our grid collapses to 8,000
points if we now strip the model down to three free parameters. If we had the resources to
investigate 1000 different parameter settings of the model every day, we are down to a mere eight
days’ computation to solve our problem, as opposed to the 100,000 days needed to sweep a four-
parameter fine grid. Of course non-linear interactions might mean the model does something
unusual and/or interesting at an “off-grid” point in the parameter space. This worry can be
partially addressed by additional suites of Monte-Carlo simulations that use random in-range but
off-grid settings for free parameters, though it always remains a lurking possibility.

The third way round this wall involves efficient computing. During our work for this
paper, for example, we refined our computer code to yield, not untypical, 10-fold speed increases
that effectively multiplied our computational resources by a factor of ten, greatly enhancing our
parameter-sweeping capability within any fixed time horizon. A more challenging solution
involves ensuring we budget sufficient, but not excessive, computational resources to allow valid
inferences to be drawn; this is essentially a problem of statistical inference. We argue below that
we must characterize significant model outputs in a rigorous way, in the sense that longer, or
more, simulations would not result in a significantly different estimate. We must budget enough
computing to ensure valid statistical inference; but we do not want to do “too much” computing,
since we prefer to budget finite computing resources for sweeping more parameters, or sweeping
a finer grid in a given parameter space.

**Computational analysis and/or agent based modeling of party competition?**

Summing up our argument thus far in the context of party competition, MDMPC is an inherently
dynamic process involving the interaction of a substantial number of actors. A rigorous model of
this process is analytically intractable unless confined to the simple (essentially pathological)
settings with very few parties, very few voters with preferences accurately characterized using
one dimension of policy, and almost no real dynamics. This suggests building more realistic
models that are computationally tractable. John Jackson, as well as Smirnov and Fowler, analyze
models such as these (Jackson 2003; Smirnov and Fowler 2007); in each case we find
computational interrogations of formal analytic models, as opposed to ABMs in the conventional
sense. Such computation is a technical substitute for classical analysis. The desire to model large
numbers of autonomous decision-making agents, especially when these are in a complex dynamic
setting, also imposes an extraordinarily heavy computational load on real decision-makers,
suggesting a shift of behavioral assumption from deep strategic-look-forward to adaptive
learning, and thus to computational ABMs.

Recent examples of ABMs in this tradition include the seminal Kollman, Miller and Page
model, as well as work by De Marchi, and by Laver and co-authors (De Marchi 1999; De Marchi
Kollman, Miller, and Page 1998; Laver 2005; Laver and Schilperoord 2007). This work has not,
for the most part, been situated in the methodological landscape we map out above – sometimes
using ABMs as discovery tools, less often as simulations that substitute for formal analysis. Most
ABMs of MDMPC have been used as discovery tools and we do not consider this further here –
the beauty of such discoveries is in the eye of the beholder. In what follows, we focus on the
rigorous investigation of computational ABMs as a technical substitute for formal analysis.
A BASELINE COMPUTATIONAL ABM OF MDMPC

Building on earlier work by various authors (De Marchi 1999; De Marchi 2003; Kollman, Miller, and Page 2003; Kollman, Miller, and Page 1992; Kollman, Miller, and Page 1998; Laver 2005), Laver (2005) developed an ABM of party competition in a two-dimensional policy space. Briefly, this assumed non-strategic “proximity” voters with ideal points randomly drawn from a bivariate normal density function, each voter supporting the closest party. Party leaders set policy positions adaptively. They do not know the ideal point of any voter, but respond to published information about voter support for each party, given parties’ published policy positions, at any given time. Given the adaptation of party policy positions, voters re-evaluate which party to support. Given this, parties re-adapt. This process runs forever. The four policy-adaptation rules for party leaders investigated by Laver drew from traditional empirical literatures on intra-party decision-making:

• STICKER: never change position (an “ideological” party leader);
• AGGREGATOR: set party policy on each dimension at the mean preference of all current party supporters (a “democratic” party leader responding perfectly to supporter preferences);
• HUNTER: if the last policy move increased support, make the same move; else, reverse heading and make a unit move in a heading chosen randomly from the arc ±90° from the direction now being faced (an autocratic party leader who is a Pavlovian vote-forager);
• PREDATOR: identify largest party; if this is you, stand still; else, make a unit move towards largest party (an autocratic party leader seeking votes by attacking larger parties).

A striking finding of this work was that party leaders using the Pavlovian Hunter rule, despite its simplicity, are systematically more successful at finding popular policy positions than party leaders using any other rule investigated. Agents using Hunter tend to seek votes near the center of the distribution of voter ideal points (recall they so not know this), but also systematically to avoid the dead center of this distribution.

STREAMLINING THE BASELINE MODEL

The substantive problem we investigate here concerns whether an unresponsive party located away from the center of the policy space causes other parties, using more responsive decision rules, to move towards its position. This is certainly an informal intuition about why ideologically off-center parties might stay put rather than adapt their policy positions in the fight for votes, an intuition we investigate by systematic interrogation of Laver’s original ABM. To do this, we first strip the baseline model of some of its complications in order to reduce the number of free parameters and thereby facilitate rigorous parameter sweeping.

Laver’s original model, and its extension to endogenous political parties, treated every voter as an independent decision-making agent (Laver 2005; Laver and Schilperoord 2007). Reported simulations involved 1000 voters and varying numbers of political parties – thus over 1000 interacting agents. Each voter in each simulation was given an ideal policy position drawn randomly from an underlying density function (bivariate normal with \( \rho = 0 \) – ideal points on each dimension are assumed to be uncorrelated). Different model runs thus generate different results, in part because of stochastic elements in some decision rules, but also because each run is based on a different random draw of voter ideals. This is substantively appropriate if we are interested in voter behavior, or in the impact on party behavior of stochastic local clusters of voters. In other circumstances, describing the voting population in terms of a discrete random draw from an underlying density function may not be helpful, for example if we want to
characterize party behavior, abstracting away from the discrete nature of any particular finite voting population. The substantive problem investigated in this paper involves characterizing party behavior. Given this, the random draw of voter ideal points generates an additional implicit parameter in the model (the seed used by the pseudo-random number generator), that is of no substantive interest to us. We thus strip this parameter out of the baseline model by replacing a finite population of voters, each with a discrete ideal point drawn from an underlying density function, with an infinite population of voters described by the underlying density function itself. The same configuration of party positions will now, over the very long run, always generate precisely the same vector of party support levels. This characterizes the voting population, not as autonomous agents but as a density function, in a manner directly analogous to the “electoral landscapes” used by Kollman, Miller and Page, and by de Marchi (De Marchi 1999; Kollman, Miller, and Page 1998). We now expect a series of independent runs of the same model with the same parameter settings to converge on the same results, allowing us to investigate in a rigorous way how these results respond to the manipulation of parameters of substantive interest.

RESEARCH DESIGN

What we want to find out in substantive terms is whether an ideologically intransigent (Sticker) party at one extreme of the policy space can result in a situation in which other parties move towards it as a result of the process of party competition. We thus design a suite of simulations that use our ABM to investigate this problem in a systematic way, in two different empirical settings. The first is a setting in which the density of voter ideal points has a unimodal distribution across the policy space – the setting used in the original Laver ABM, as well as in the classical Downsian model of party competition. We adopt the Laver (2005) assumption of bivariate normal voter density and without loss of generality fix our coordinate system for ideal points and party positions by assuming this has a mean at the origin of the space and standard deviations on each dimension of 1. The second is a setting with bimodal spatial distributions of voter densities – a setting that has not previously been investigated in the context of this type of ABM. We focus here on the interaction between Stickers and Hunters, leaving the interaction between Stickers and other party decision rules for future work.

The research design involves first benchmarking the analysis using simulation runs with four Hunters and no Sticker. This allows us to measure typical sizes and locations of the four parties using the Hunter rule, absent a Sticker. We then add a Sticker party and run a series of simulations, during which the Sticker’s ideal point is first set at the origin, and is then moved progressively away from the origin on the x-axis. Each simulation increments the x-coordinate of the Sticker ideal point by 0.2 units, leaving the y-coordinate at zero, until an “extreme” Sticker location of (3.4, 0) is reached. At this point, the Sticker x-coordinate is the same as that of a (very extreme) voter located 3.4 standard deviations away from the centroid of the voter distribution. This lays down a one-dimensional 18-point grid on the parameter space – reflecting Sticker positions on the x-axis. One output variable of substantive interest is the mean x-coordinate of the four Hunter parties, conditional on the position of the Sticker. Another output variable of interest is the x-coordinate of the “rightmost” Hunter. The latter is interesting in terms of the substantive argument that a more realistic objective for an intransigent party at the ideological extreme might well be to pull the closest party’s position towards it, as opposed to the position of all parties in the system. An ideological party on the right might be much more concerned with the positions of other right-wing parties than with the positions of parties on the left.

This research design has the virtue that our model generates outputs on a number of variables for which we have sound analytical expectations. For example, since voter ideal points,

---

5 That is, by an amount equal to 0.2 standard deviations of the voter density function.
Sticker ideal points, and Hunter decision rules are all completely unbiased in relation to the origin of the $y$-axis, we expect mean party locations on the $y$-axis to be zero over the long run for all parameter settings of the model. This is also true for both the $x$- and $y$-coordinates of all parties when the Sticker ideal point is fixed at $(0, 0)$. These clear analytical expectations prove very useful in diagnostic tests on the output from particular simulations, increasing our confidence in simulation-based estimates of the key quantities for which formal analysis is intractable – mean and rightmost Hunter locations in systems with an eccentric Sticker.

Having completed our sweep of Sticker positions in a setting with a unimodal spatial distribution of voter densities, we turn to a bimodal setting. We build bimodal voter density functions by assuming that an aggregate population of voters comprises two (or quite possibly more) distinct subpopulations, each with normally distributed voter densities, but with potentially different sizes, means and standard deviations. Thus, while all unimodal distributions have essentially the same “shape”, the universe of possible bimodal distributions can have many different shapes. The shift from unimodal to bimodal distributions, even tightly constraining such distributions to be aggregations of two bivariate normal distributions, involves introducing at least three new parameters – the relative locations of the means of the subpopulations, their relative variances, and their relative sizes. In what follows, we investigate two possibilities of interest. First, we consider bimodal voter density distributions in which the subpopulations are of equal size. Keeping the position of the eccentric Sticker fixed at $(2.0, 0)$, we investigate the effect on Hunter parties in settings where the subpopulation modes are moved progressively further apart, generating ever more “polarized” electorates. Second, we consider bimodal voter density distributions in which the subpopulations are of unequal size – specifically in which the rightmost subpopulation is half the size of the leftmost. Previewing our results, we find that the impact of an ideologically intransigent party is most striking in this type of setting.

METHODOLOGICAL ISSUES

Overview
Before we implement the research design set out in the previous section and interpret the results, we deal with a range of methodological issues that face anyone who sets out to characterize outputs from a computational dynamic model in a rigorous way. These concern, *inter alia*, the following:

- The need to establish when outputs generated by a model run have “burnt-in” after an arbitrary start
- The need to take account, during statistical analysis, of the time series structure of model outputs
- The need to establish how long a simulation should run after burn-in, and in particular to establish whether characterizations of model outputs are biased by the point at which the simulation run is terminated
- The need to estimate the effect on simulation outputs of using different model parameters

Figure 1 provides an overview of these issues. The model generates an output variable of substantive interest – a dependent variable, $y$. (For our problem, this is the mean $x$-coordinate of

---

6 Our substantive interest involves estimating quantities-of-interest that do not converge on a fixed or steady state but rather converge on a limiting distribution or ergodic state. Hence the procedure that we lay out below is designed for this (more general) case. For cases in which a quantity-of-interest converges on a steady state, the end of the burn-in period is obvious and controlling for serial correlation is not necessary.
the four Hunter parties or the x-coordinate of the rightmost Hunter party.) Figure 1 shows output, recorded over time, from one simulation run of a hypothetical dynamic model. The run has an arbitrary start. This might be a pure random start. It might be a particular start that is set up by the analyst for some reason. The point is that the start is “arbitrary” in terms of the long run dynamics of the system. We do not want our characterization of these long run dynamics to be affected in an arbitrary way by any particular start of any particular simulation run. In common professional parlance, we want to “burn-in” the simulation before we analyze its outputs. The arbitrary start and the “burn-in” era of the simulation run can be seen on the left hand side of Figure 1. Model outputs during burn-in move in atypical ways, not characteristic of model outputs over the long run (and recall it is these long-run outputs that we seek to estimate if we wish to use simulation in place of the analytical equivalent of comparative statics.) The right hand side of Figure 1 shows model output after burn-in – during what we can think of as the “burnt in” era of the simulation. Here, the burnt-in output of the simulation is in fact a sine wave oscillating around a long-run mean of zero, shown as the horizontal line; because of the analytical intractability of the model, however, the analyst does not know this and is seeking to estimate it from model output.

The first task of the analyst is to estimate when the burn-in era is over, a task simplified by the fact that statistical features of the burn-in problem are relatively well understood. Figure 1 makes this problem look simpler than it is in reality. Figure 2 plots the x-coordinate of one of the Hunter parties for a simulation run of our model with the Sticker position set at the origin. Each Hunter party starts the simulation run at an arbitrary spatial location and burn-in is blindingly obvious, simply by looking at Figure 2, during the first 200 model ticks. After this, it is difficult to tell with the naked eye at which point system dynamics can be said to be running independently of the random start.

The second task identified in Figure 1 is to characterize model output during the burnt-in era of the simulation run, on the right hand side of the figure. Two important issues arise here. The first is that model output is a time series not a set of independent observations; the second arises from the periodic structure of the output time series. Figure 3 summarizes, using a median spline, the mean Hunter x-coordinate, over 10,000 simulation ticks during the burnt-in era of the run reported in Figure 2. We note that the model output has long “periods”, lasting several hundred or even a thousand ticks, during which it is systematically above, or below, its long-run mean. Such periods arise because, while our model generates a Markov process, this is at the very least a second-order Markov process. Party leaders using the Hunter rule select a position at tick \( t \), conditional on the difference between their performance at both ticks \( t-1 \) and \( t-2 \). Such higher-order Markov processes have the potential to lead to a periodic structure in model outputs.

When there is a periodic structure in model outputs, a third potential problem arises if we want to estimate quantities of interest by summarizing outputs over the entire-burnt-in era. Every finite run has a stopping rule, which may stop the run at a point when model output has been away from its long run mean for some time, potentially biasing our estimates. Figure 4 illustrates this in a schematic way, focusing on the hypothetical burnt-in sine wave output shown in the right hand part of Figure 1.

If the run was stopped at “arbitrary stop 1”, estimates of model output would be biased above the long-run mean of the time series of interest. This bias would remain present, though reduced, at
“arbitrary stops 2 and 3”. Even if the model ran for a long time, the bias would still be present, albeit to a much lesser degree, at “arbitrary stop 4”. Since every real simulation is finite, the potential for this type of bias arises whenever output has a periodic structure. The problem this poses is to know when the burnt-in era is “long enough” to reduce such potential bias to acceptable levels.

A fourth problem, related to the previous problem, is outlined schematically in Figure 5, which shows output from two independent runs of the hypothetical sine wave model during their burnt-in eras. Each run uses different settings, a or b, of some key model parameter, X, the effect of which we want to estimate. The starts of each run are arbitrary so there is no reason to expect their periodic structures to be synchronized. Setting \( X = b \) generates long run output (colored green) with a lower mean than setting \( X = a \) (colored red), but model output is often higher when \( X = b \) than when \( X = a \). Our fundamental reason to run simulations is systematic parameter sweeping, which boils down to estimating whether the long run mean when \( X = b \) is lower than the long-run mean when \( X = a \). Figure 5 also provides a further illustration of why we need to be confident that the burnt in era of the simulation has been running “long enough”. If we stopped this simulation “too soon”, left of the dashed vertical line for example, we would incorrectly conclude that \( Y \) is systematically higher when \( X = b \) than when \( X = a \). As we see when we run the simulation for much longer, it is actually systematically lower.

We treat each of these problems in turn, illustrating our method with the first of our quantities of interest, the mean \( x \)-coordinate of the Hunters.

**Estimating model burn-in**

First, to eliminate the effects of the random start, we must determine the number of ticks to discard as burn-in. Note that the mean \( x \)-coordinate of the Hunter parties is dynamic. It does not asymptotically converge to a fixed state but to a dynamic equilibrium, or ergodic state; it “roams around” a long-run mean within a quantifiable range. We must therefore quantify a converged probability distribution, with a mean and standard deviation. This task is similar to the estimation of a posterior probability distribution, which analysts employing Bayesian techniques approximate using Markov chain Monte Carlo (MCMC) iterative simulation. We use the same procedure developed by these analysts to measure the convergence of the probability distribution of the mean \( x \)-coordinate.

Specifically, we use the potential scale reduction factor or R-hat statistic, proposed by Gelman and Rubin and generalized by Brooks and Gelman (Brooks and Gelman 1998; Gelman and Rubin 1992). The potential scale reduction factor estimates the factor by which the scale of the current distribution could be reduced if the simulation were continued indefinitely. In the limit, R-hat tends to 1. For values of R-hat close to 1, the scale of the distribution cannot be reduced much further; the statistic of interest will oscillate within the range defined by its mean and standard deviation. At this point the statistic of interested has “converged” to its ergodic state. The R-hat statistic is calculated by running several chains of the same type of simulation and, at each number-of-iterations of interest, taking the “second halves” of each chain and comparing between-chain variance with total within-chain variance from all chains – a technique similar to an analysis of variance test. The R-hat statistic approaches 1 as the between-chain variance becomes less important and is eventually completely dominated by the within-chain variance.

In Table 1, we present the mean, standard deviation, 95% confidence intervals, and R-hat statistics for the mean \( x \)-coordinate position of the Hunter parties with the Sticker party position set at \((0,0)\). These values are calculated using the second-half observations from five separate parallel model runs at various number-of-ticks. In addition, for comparison, we include the same information from one chain after 250,000 iterations. A priori, we expect the long-run mean of the
mean x-coordinate position to approach zero, as any effect the Sticker may have should be symmetrical around the origin. And this is in fact what we find with the mean after 250,000 ticks. After 50 ticks, however, the calculated mean using the second 25 ticks is still somewhat away from zero, at -.053. What’s more, the standard deviation at .224 is more than 20% larger than its long-run value and thus the confidence intervals are also much larger. The R-hat statistic at 2.85 indicates that the scale of the distribution can be reduced by increasing the number of ticks used. By 500 ticks, the mean, standard deviation, and confidence intervals have moved much closer to their long run values, but at 1.23, the R-hat statistic is still relatively high. Only after 5,000 and 10,000 ticks, with an R-hat of 1.01, does the distribution look similar to its long-run form. To be conservative in estimating the length of the burn-in era for our particular model, we take 10,000 ticks as our base-line. Hence, when calculating the mean x-coordinate position of the Hunters, we discard the first 5,000 ticks as burn-in, using simulation results from tick 5,001 onwards only.

[Table 1 about here]

**Time series structure of model output**

Second, in order to calculate the correct standard errors of our quantity of interest, we must control for serial correlation. In our model, the position of each Hunter is a function of the position it occupied during the previous two periods. Therefore, any time series of the x-coordinates of each Hunter party should follow an autoregressive process of at least the second order. By extension, the mean x-coordinate of the Hunter parties, and the x-coordinate position of the rightmost Hunter, should also follow an autoregressive process of at least the second order. If we failed to account for this serial correlation and instead treated each observation as if it were independent of the others, our standard error estimates would be incorrect. In fact, we find in this setting that not controlling for serial correlation leads to a reduction of the estimated standard error by a factor of more than 10. Hence, a crucial task in determining the correct standard errors for our estimates is to control for the time-series properties of the quantity of interest.

The first task is to determine the autoregressive, moving average (ARMA) model that best fits our data. Following the Box-Jenkins procedure (Box and Jenkins 1976), we first analyze a correlogram of the first 10 lags for the mean Hunter x-coordinate, using the last 245,000 ticks of a 250,000-tick simulation. As one can see from Table 2, the autocorrelation function (ACF) exhibits a steady decay, while the first 3 lags of the partial autocorrelation function (PACF) are significantly different from zero and most of the other lags are not. This would lead us to suspect that our series is best represented by either an AR(3) or an ARMA(2,1) process.

[Tables 2 and 3 about here]

To be sure, we conduct a systematic analysis of different potential ARMA models. In Table 3, we compare the ARMA(2,1) and AR(3) models. First, note, that the AR and MA coefficients are all significant with both models. However, the log-likelihood is higher and both the Akaike Information Criterion (AIC) and the Schwartz Bayesian Criterion (BIC) are lower with the AR(3) model. What’s more, the Portmanteau Q-statistics of the lag of the residuals from the AR(3) exhibit less autocorrelation than those from the ARMA(2,1) model, indicating that more of the movement of the dependent variable has been captured by the AR(3) model than by the

---

7 In this subsection, we present results using 245,000 observations. Similar time-series results were obtained using different numbers of observations. In the next subsection, we justify our choice of stopping our simulations after 250,000 ticks.
ARMA(2,1). Therefore, we use an AR(3) model to control for serial correlation with the mean \(x\)-coordinate of the Hunter parties.\(^8\)

Once we’ve identified an ARMA model, the second task is to check for stationarity. If a time-series is not stationary, its mean – our quantity of interest – is meaningless. Given that an AR(3) model fits our series best, we employ an augmented Dickey-Fuller test with an AR(3) process. Results assuming random walk, drift, and trend stationary models are presented in Table 4. The t-statistic on the lagged term at roughly -64 for all three models is enormous. What’s more, the coefficient on the trend term for the trend model is practically indistinguishable from zero. Hence we reject the hypothesis that our series is non-stationary.\(^9\)

![Table 4 about here](image_url)

**Length of the burnt-in simulation era**

Finally, having established the number of ticks required for burn-in and having analyzed the time-series components of our series, our last step is to determine how long to run the burnt-in period in order to establish precise estimates of our quantities of interest. (This task takes on extra significance given the periodic structure of the series.) Precision, of course, comes at a price. If we could run our simulation for an infinite number of ticks, and there were minimal costs to doing so – storage space costs, opportunity costs of not running another simulation, etc. – we would obtain a standard error of zero around our measure. If we ran our simulation for only 100 or 1000 ticks after burn-in our standard errors would be huge. One must, therefore, find a middle ground between these two extremes. Desired levels of precision vary by context and quantities of interest, and there is always a certain degree of arbitrariness or judgment or art with any decision rule. We propose the following three desiderata which we think are the most general and the most applicable to a variety of cases.

1. Quantities of interest, for which we have sound analytical expectations, approach the analytically expected values in the long-run.

2. Quantities of interest, which a priori we do not expect to be zero or close to zero, should be significantly different from zero. That is, there should be some noticeable effect; and

3. For a given level of fineness or coarseness of the interval between the parameter-settings of the parameter that is being analyzed (swept), to the extent possible, the estimated effect given one parameter setting should be statistically different from the estimated effect of parameter settings close to it on the grid.

To illustrate this rule, we analyze the mean of the \(x\)-coordinate for the Hunter parties for a set of simulations, moving the Sticker location away from the origin on the \(x\)-axis, in steps of 0.2 standard deviation units, to the extreme position of a party 3.4 standard deviations away from the voter centroid. (We treat substantive interpretations of these results in the next section. Here, we concentrate on methodological concerns.)

We start with Desideratum 1. Recall that the mean \(y\)-coordinate for the Hunter parties should be zero for all Sticker settings and the mean \(x\)-coordinate should be zero when the Sticker

---

\(^8\) We also investigated an AR(2), ARMA(2,2), ARMA(3,1), and AR(4). The AR(3) model performed the best.

\(^9\) We also ran a suite of augmented Dickey-Fuller tests with 50 lags to control for the possibility that our series contained an MA process. Results were equally as strong, with t-statistics at roughly -49 for all three specifications.
is set at (0,0). Figure 6 plots the cumulative mean $x$-coordinate of the Hunter parties when the Sticker is set at (0,0), estimated by an AR(3) model, at 10,000-tick intervals, as the simulation runs from 10,000 ticks to 250,000 ticks. Each point plots the long-run mean, as this would have been estimated had the simulation been stopped at that point in the run. After 10,000 ticks, the mean is over .005 standard deviation units away from its analytically known value of zero and the standard error of the estimate is roughly .020 units, implying a 95% confidence band that ranges from -.034 to .045. Thereafter, the mean is less than .005 away from zero, and the standard error continues to decrease. By 200,000 ticks the mean $x$-coordinate Hunter is within .0002 units of zero, and the standard error has decreased to .003 standard deviation units, or less than .005% of the total range of the $x$-coordinate. We find the same pattern when we analyze the mean $y$-coordinates. Hence, Desideratum 1 is satisfied. Note, however, that even though the closeness to zero and the precision of the estimate increases as we increase the number of ticks used in the simulation, the expected value of zero always falls within the 95% confidence interval even for the shortest of runs. Hence we turn to Desiderata 2 and 3 for further guidance.

Turning to Desideratum 2, Table 5 shows estimates of a quantity of interest that a priori we expect to differ from zero – the mean $x$-coordinate of four Hunters in a system with an eccentric Sticker. Running the simulations long enough (and what constitutes “long enough” is precisely what we are trying to find out), we see that these quantities are indeed statistically different from zero at the 95% confidence level. However, after simulations have been running for 50,000 ticks, only 11 of the 17 quantities of interest differ from zero to an extent that is statistically significant. After 100,000 ticks, 13 quantities were significantly different from zero; while after 150,000 ticks 14 quantities were significant. By contrast, after 200,000 ticks, 16 of the 17 quantities differed significantly from zero – and the quantity that is not statistically significant concerns the effect of the most eccentric Sticker, which a priori we expect to be close to zero.

Figures 7 and 8 illustrate our consideration of Desideratum 3. Figure 7 plots estimated effects after 50,000 ticks; Figure 8 plots these after 250,000 ticks. After the simulations have been running for only 50,000 ticks, we cannot statistically distinguish effects generated when the Sticker is located at $x = 1.2, 1.4, 1.6,$ or $1.8$. Nor can we distinguish effects when the Sticker is at $x = 2.0, 2.2, 2.4,$ or $2.6$. By contrast, after 250,000 ticks, although one can still not statistically distinguish the effect when the Sticker is at $x = 1.4$ as opposed to 1.2 or 1.6, i.e. the effect of a parameter setting that is one step away, the effect is statistically significant when the Sticker is at 1.4 as opposed to 1.8, a two-step difference. Note that taking pains to satisfy Desiderata 1 through 3 also should help alleviate the potential problem, noted above and illustrated in Figure 5, of incorrectly concluding that the long-run mean effect at one parameter setting is greater than the effect at another setting, when in fact the inverse is true. We see this in relation to our estimates for Stickers at $x = 1.2$ and 1.4. From Figure 7 we see that after only 50,000 ticks, the estimate for a Sticker at $x = 1.2$ is greater than the estimate when the Sticker is at 1.4. This would be a flawed inference. As shown in Figure 8 after 250,000 simulation ticks, the effect of a Sticker at $x = 1.4$ is greater than the effect when it is at 1.2.

Of course we could continue running simulations for ever-longer periods and will always get more precise estimates if we do so, but we feel we have achieved sufficient precision at this stage.
Thus, in what follows, we use 250,000-tick simulation runs when calculating the mean \( x \)-coordinate of the four Hunter parties (and also of the “rightmost” Hunter).

**Summary of method**

We summarize the procedure we advocate for analyzing outputs from a long-run dynamic model as follows:

1. **Burn-in.** Use the R-hat statistic to determine the number of model ticks that should be discarded as burn-in from an arbitrary start.

2. **Time series estimation.** If there are *a priori* reasons to suppose an output series is serially correlated, follow the Box-Jenkins procedure to identify the ARMA model that best fits the series and check the series is stationary using Dickey-Fuller or augmented Dickey-Fuller tests.

3. **Burnt-in era.** Determine the number of burnt-in simulation ticks needed to insure that:
   - (a) quantities of interest for which there are analytical expectations approach their expected values
   - (b) effects expected *a priori* to differ from zero are significantly different from zero; and
   - (c) for the chosen parameter grid, to the extent possible, the estimated effect of one parameter setting is statistically different from the estimated effect of parameter settings close to it on the grid.

**ESTIMATING THE EFFECTS OF MODEL PARAMETERS**

**Mean \( x \)-Coordinate of the Hunter Parties**

Using the above method, we return to our substantive analysis of the effect on the mean \( x \)-coordinate of the four Hunter parties of locating a Sticker at various \( x \)-coordinates. Returning to Figure 8, we present the first substantive results in this paper, plotting a fundamentally non-linear effect. First note that we know analytically, from the perfect symmetry of the model in a setting with a Sticker at the origin and four parties using the Hunter rule, that the mean \( x \)-coordinate of all parties will be zero over the very long run. Our simulations generate the same result. We are therefore interested in what happens to the mean \( x \)-coordinate of the four Hunter parties as the Sticker’s \( x \)-coordinate is progressively moved to the right. With Sticker positions up to, but not including, one standard deviation away from the voter centroid, the long-run mean \( x \)-coordinate of the Hunters moves significantly to the right. The effect is greatest when the Sticker is at 0.60; the mean position of the other parties is 0.13 standard deviation units away from the voter centroid, a not insubstantial effect. As the eccentric Sticker is located progressively further away from the center and towards the right, however, the mean position of the other parties moves systematically to the left. This effect is greatest at an eccentric Sticker location of 1.40, when the mean \( x \)-coordinate of the other parties is 0.07 standard deviation units left of center. As the Sticker is located at ever-more-eccentric locations, its effect on the positions of the other parties approaches zero. A far-out eccentric Sticker has effectively no effect on the positions of other parties, which interact with each other towards the center of the policy space, as if the Sticker was not there.

The substantive story behind the non-linear relationship in Figure 8 might seem on the face of things to be counter-intuitive – an ideologically intransigent party that in some sense “attracts” other parties when it is relative close to the center, “repels” them as it gets further away, and finally become irrelevant to other parties when it takes the most extreme policy positions. Fortunately, one of the great benefits of having a systematic suite of long-running simulations is
that we have a lot of information against which to assess this naïve interpretation and, as we shall see, find it misleading.

Figure 9 shows where, on the x-axis, Hunter parties tend to be found over the 245,000 burnt-in ticks of a four-Hunter “benchmark” run with no Sticker party at all. The Hunter parties essentially search for votes within the range ±1 standard deviation units from the voter centroid. More tellingly, Figure 10 shows the location of the rightmost party in the four-Hunter benchmark run. This shows that the rightmost of the four Hunters was typically 0.4 – 0.7 standard deviations to the right of the voter centroid; in fact the mean x-coordinate of the rightmost Hunter in the benchmark run was 0.56 standard deviations right of center. Note that it was at precisely this right-of-center location that the Sticker had the greatest effect. Investigating, in greater detail, what happens when the Sticker location is 0.6 standard deviations right of center, Figure 11 plots the x-coordinates of the four Hunters in this setting. Comparing this figure with the benchmark four-Hunter run, we see a very striking pattern. The intransigent Sticker party is not “attracting” other parties towards its position. On the contrary, the Sticker is “repelling” the other parties from its position.

[Figures 9, 10 and 11 about here]

Watching the simulations in motion, the reason why Stickers tend to repel Hunters makes analytical sense. If a Hunter party approaches an eccentric Sticker, it may gain votes from the Sticker but tends to be punished overall, as other Hunters are rewarded for moving into its former territory, which tends to have higher voter densities. The original Hunter then reverses direction and moves away from the Sticker. We have generated kernel density plots equivalent to that in Figure 11 for every Sticker location from (0, 0) to (3.4, 0) and it is clear that a single very systematic effect is being observed. This is the tendency of intransigent Stickers to repel nearby Hunters, for the reason we have just sketched, and explains the pattern we observe in Figure 8. When the intransigent Sticker is right-of-center but relatively close to the center (in the 0.0 – 0.8 standard deviation range), there will tend to be some Hunter parties to the right of it. These will tend to be repelled, and shift to the right. When the Sticker occupies a more right wing position, even the rightmost Hunter will tend to be to the left of it, as we see from Figure 10. Hunter parties will again be repelled by the eccentric Sticker, but this time the result will be a shift to the left. This effect declines in significance as the Sticker becomes ever more extreme, since its repulsive effect on the other parties, which are now located far away from it, becomes ever less significant.

Substantive results from bimodal spatial distributions of voter densities
As we shall now see, however, the effect of an eccentric Sticker is quite different when the spatial distribution of voter densities is bimodal, conforming more to popular informal intuitions. We investigate two different types of bimodal voter density distribution in what follows. The first is aggregated from two equal-sized unimodal subpopulation distributions with different degrees of polarization between their centroids; the second is aggregated from two unequal sized population distributions.

“Balanced” bimodal spatial distributions of voter ideal points
As noted in the introduction to this paper, we model multimodal spatial distributions of the ideal points of a population of voters as aggregations of subpopulations of voters, each characterized by some (unknown) feature that gives them distinct unimodal spatial distributions of (subpopulation) ideal points. The simplest case, analyzed in this section, involves a population that is an aggregation of two-equal sized “unimodal” subpopulations, with subpopulations differing in terms of the ideological locations of the centroid of their distributions of ideal points. In order to norm what would otherwise be arbitrary units of spatial location, we adopt the following strategy. Our analyses of unimodal voter populations assumed a bivariate normal distribution of mean 0 and standard deviation 1. Our coordinate system for party policy positions
was thus calibrated in terms of standard deviations of the underlying distribution of voter ideal points. A Sticker located at \((2.0, 0)\) was thus located at the ideal point of a voter two standard deviations from the population mean on the \(x\)-axis, and precisely at the population mean (and mode) on the \(y\)-axis. The situation is not quite so straightforward in relation to bimodal population distributions, especially when subpopulations are of unequal sizes. In what follows we thus keep things simple and intuitive by modeling aggregations of voter subpopulations such that, taking account of the means and standard deviations of subpopulation locations, an eccentric Sticker at \((2.0, 0)\) – the focus of our subsequent attention – will also be two standard deviation units away from the closest subpopulation mode. Thus a Sticker at \((2.0, 0)\) will be precisely as eccentric from the local subpopulation mode as from the mode of a unimodal population with a centroid at the origin. This is achieved by modeling the following bimodal distributions, all based on aggregating equal-sized bivariate normal distributions: means at \(x = \pm 0.7\), standard deviations 0.65; means at \(x = \pm 0.8\), standard deviations 0.6; means at \(x = \pm 0.9\), standard deviations 0.55; means at \(x = \pm 1.0\), standard deviations 0.5; and means at \(x = \pm 1.1\), standard deviations 0.45. In each case it can easily be seen that a Sticker at \((2.0, 0)\), in relation to the local subpopulations, is two local standard deviation units to the right of the local mean. If the local modes are set closer together, the aggregate population is unimodal; when the modes are set at \(\pm 1.1\), with standard deviations 0.45, the aggregate voter population is more polarized, “touching” only at the extremities of the subpopulations, more than two standard deviations from their local means.

What we are interested in is the effect of having an intransigent Sticker to the right of the rightmost voter subpopulation. We expect the effect to be quite different from that which we observed in the unimodal voter populations. To illustrate why, Figure 12 shows the population density distribution on the \(x\)-axis generated by aggregating subpopulations with means at \(x = \pm 1.0\) and standard deviations 0.5.

Consider the impact on a set of four Hunter parties of introducing a Sticker at \((2.0, 0)\). This effect will obviously be complex, but one scenario in which the impact will be clear is during periods when three Hunters are seeking votes in the left-hand voter subpopulation, leaving one seeking votes in the right-hand subpopulation. Absent an eccentric right-wing Sticker, the Hunter contesting the right hand subpopulation faces no punishment for moving to the left. It is moving down a steep slope in the voter density function, but there is no cost in doing this, while votes are gained to the right. The situation is quite different with an eccentric Sticker on the far right. Now the lone right-wing Hunter, when moving left and descending the slope down from the right hand voter density mode, quickly reaches a point at which votes lost to the Sticker outweigh votes gained from other parties by moving left. In this sense, we expect an eccentric Sticker to “anchor” at least one vote-seeking party in its local sub-population mode. This particular phenomenon does not arise in a unimodal voter distribution, since, other things equal, moving away from an eccentric Sticker typically involves climbing a slope in the voter density function, gaining more votes than are lost.

First, using our standard 250,000-tick runs, we benchmark the locations of four Hunter parties in these bimodal voter populations, with no Sticker party. Then, we add a Sticker at \((2.0, 0)\) to each of these settings and measure the locations of the four Hunter parties during a new suite of 250,000-tick runs. Burn-in for the bimodal populations with means at 7 and 8 required 5,000 ticks; burn-in with means at 9 and 10 required 10,000 ticks; and burn-in with the mean at 11 required 20,000 ticks. We estimate both of our quantities of interest – \(x\)-coordinate of the
rightmost Hunter and mean x-coordinate of the four Hunters – with an AR(3) model. Figure 13 sets out to summarize a lot of information as efficiently as possible. For each of the five, increasingly polarized, bimodal voter populations that we investigated, it shows the mean estimated deviation of two quantities of interest from the benchmark four-Hunter runs, together with 95% confidence intervals around these estimated deviations. The first is the mean Hunter location on the x-axis – the lower of the two lines in Figure 13. We conclude that, in each of the bipolar populations investigated, the mean location of all four Hunters is unaffected, relative to benchmark, by the introduction of a recalcitrant right wing Sticker. The mean position of the rightmost Hunter party is given by the upper line in Figure 13. This shows that, when there is an intransigent right-wing Sticker at (2.0, 0), the rightmost Hunter typically occupies a position significantly to the right of the no-Sticker benchmark. The effect is greatest with moderately polarized voter populations, having subpopulation means at $x = \pm 0.9$, standard deviations 0.55. In such environments, the mean x-coordinate of the rightmost Hunter in the four-Hunter benchmark run was 0.80, somewhat to the left of the right hand subpopulation mode. When there was a Sticker at (2.0, 0), the mean position of the rightmost Hunter was 1.08, significantly to the right of this subpopulation mode.

[Figure 13 about here]

This is an example of the effect we conjectured. Contrary to the situation with a unimodal voter population, where the presence of an eccentric Sticker makes little difference to the positions of the other parties, the presence of an eccentric right-wing Sticker (positioned precisely the same distance from the local subpopulation mode) has a significant effect in “anchoring” the rightmost party near the local subpopulation mode.

This effect is dramatically enhanced when voter subpopulations are of different sizes. We repeated the suite of simulations reported in Figure 13, setting right-hand voter subpopulations at half the size of the left-hand subpopulations. Figure 14 shows voter densities on the x-axis for an example of such a “lopsided” spatial distribution of voter densities. Of course, the lopsided case is far more general than setting voter subpopulations at precisely equal sizes. Figure 15 reports the results of repeating the suite of benchmarking four-Hunter simulations for each type of “lopsided” bimodal voter density distribution, followed by adding a Sticker at (2.0, 0) to each setting. It thus shows the same information as Figure 13. The effect of the eccentric Sticker is now very striking, and greatest when subpopulation means are at $x = \pm 1.0$ and standard deviations 0.5. In this setting, the mean position for the rightmost Hunter in the benchmark runs is 0.16 – well to the left of the rightmost population mode and not far from the local minimum at the center of the aggregate voter distribution. Adding an eccentric Sticker at (2.0, 0), the mean position of the rightmost Hunter is 1.26, well to the right of the rightmost mode – a huge effect.

[Figures 14 and 15 about here]

---

10 With some bimodal settings, other ARMA models, such as the AR(4) model, fit the data better than the AR(3) model. In these cases the substantive difference from the AR(3) model was negligible. For consistency we use the AR(3) model throughout. The absolute value of the t-statistics from the augmented Dickey-Fuller tests were greater than 20 for all bimodal setting, so we reject the hypothesis of non-stationarity with all settings. All quantities of interest for which we have expectations were close to their expected values after 250,000 ticks. And as can be seen in Figure 13, the estimated effect on the mean rightmost Hunter party was statistically different from zero for all settings and statistically different from other parameter setting close to it on the grid. This is not the case for the mean x-coordinate measure.

11 Burn-in required 5,000 ticks for the means at 7, 8, and 9; 10,000 ticks for the means at 10; and 30,000 for the means at 11. All other tests were in line with the balanced bimodal analysis above.
Since this is the largest and most interesting substantive effect we report in this paper, we crave the reader’s indulgence with two more figures. Figure 16 plots $k$-densities of the $x$-coordinate of the rightmost Sticker in the benchmark run for the lopsided bimodal voter density distribution shown in Figure 14. Figure 17 reports the same information for the same setting, with the addition of an eccentric Sticker at (2.0, 0). The effect is really very striking. In the benchmark run with no eccentric Sticker, the rightmost Hunter is sometimes close to the right hand subpopulation mode, but much more frequently at some point between the two subpopulation modes. In the run with a recalcitrant right-wing Sticker, the rightmost Hunter is most frequently to the right of the right hand subpopulation mode, and almost never in the zone between the two modes. The recalcitrant Sticker clearly “anchors” the rightmost Hunter at, or to the right of, the mode of the rightmost subpopulation and punishes it for moving to the left of this.

[Figures 16 and 17 about here]

CONCLUSIONS

We set out to do several things in this paper. First, we set out to develop a coherent rationale for using computational ABMs to study party competition. While noting the role of ABMs as discovery tools, we stressed their role as substitutes for classical analysis in complex and evolving settings. Given this rationale, we then set out to fortify the methodological standards for systematic computational interrogations of ABMs in settings where classical analysis is intractable. Our main conclusion on this matter is that such methodological standards are already available within the profession’s statistical literature, but that rigorously applying these standards may well imply much more heavy-duty computation than hitherto used by ABM practitioners.

We also set out to investigate what we take to be an interesting substantive problem in party competition, about which intuitions might be strong but firm analytical results are hard to come by. This concerns the competitive effect, on the positions of other parties, of a recalcitrant and unresponsive party located at some ideological extreme. (We can, for example, locate this in the context of French politics and ask whether the location on the far right of the FN under Le Pen results in a situation in which other parties, or other relatively right-wing parties, tend to move to the right.) Our conclusions on the matter, we think, are interesting. When the spatial distribution of voter densities is unimodal, the effect of the recalcitrant Sticker is modest. It somewhat repels other parties and this can “push” other parties to the right when the Sticker has a relatively moderate location. An extreme Sticker, however, seems likely to be more or less ignored by the other parties. The effect that conforms to popular intuitions kicks in when the spatial distribution of voter ideal points is bi-modal – especially when subpopulation modes are of unequal size. Now, the recalcitrant Sticker does tend to affect the location of vote-seeking parties, in particular those closer to it. By “punishing” parties that move away from it far enough to find themselves climbing down a local voter density gradient, the eccentric Sticker serves to “anchor” at least one party close to its local subpopulation mode.

Would it have been better to have come to these conclusions analytically? Of course it would. Could this have been achieved? We cannot see how it could, given the complex dynamic system we are investigating. We would never claim that computational results are better when equivalent analytical results are available. But we do claim that they are better than no result, when analytical results are otherwise out of our intellectual reach. And we further claim that the strength of our computational results can be greatly enhanced by applying rigorous methodological standards to the computational investigations involved.
REFERENCES


Figure 1: “Burn-in” and “burnt-in” eras in output from a computational dynamic model

Figure 2: Burn-in in a “real” simulated dataset
Figure 3: Median spline of mean Hunter x-coordinate for burnt-in system

Figure 4: Impact of arbitrary stopping rules on estimating quantities for the burnt-in era

Figure 5: Estimating the effects of key model parameters
Figure 6: Long-run mean x-coordinate (with 95% confidence intervals) of the four Hunter parties at various stages of the burnt-in period – Sticker at (0,0)

Figure 7: AR(3) estimates of mean Hunter x-coordinate during each of a suite of 18 independent 45,000-tick burnt-in runs, sweeping Sticker x-coordinate from 0 to 3.4, in increments of 0.2. (Sticker scale positions multiplied by 10 in this figure)
Figure 8: AR(3) estimates of mean Hunter x-coordinate during each of a suite of 18 independent 245,000-tick burnt-in runs, sweeping Sticker x-coordinate from 0 to 3.4, in increments of 0.2. (Sticker scale positions multiplied by 10 in this figure)

Figure 9: Kernel densities of party x-coordinates in four-Hunter benchmarking run. (Tick >5000 in 250,000-tick run. (Scale positions multiplied by 10)
Figure 10: Kernel densities of rightmost party’s x-coordinates in four-Hunter benchmarking run. (Tick >5000 in 250,000-tick run. (Scale positions multiplied by 10)

Figure 11: Kernel densities of party x-coordinates in four-Hunter, one-Sticker run, with Sticker at (0.6, 0) Tick >5000 of 250,000-tick run. (Scale positions multiplied by 10)
Figure 12: Spatial distribution the ideal points on the x-axis of a voter population aggregated from two subpopulations with means at $x = \pm 1.0$ and standard deviations 0.5.

Figure 13: Deviations from four-Hunter benchmarks for: rightmost Hunter (top) and mean Hunter x-coordinate (bottom) with 95% confidence intervals with Sticker at (20, 0), by subpopulation mode eccentricity – Balanced Bi-modal Voter Distribution
Figure 14: Spatial distribution the ideal points on the x-axis of a voter population aggregated from two subpopulations with means at $x = \pm 1.0$ and standard deviations 0.5. Right hand subpopulation half the size of the left hand subpopulation.

Figure 15: Deviations from four-Hunter benchmarks for: rightmost Hunter (top) and mean Hunter x-coordinate (bottom) with 95% confidence intervals with Sticker at (20, 0), by subpopulation mode eccentricity – Unbalanced Bi-modal Voter Distribution.
Figure 16: K-densities of rightmost Hunter x-coordinate in four-Hunter benchmark run, two subpopulations with means at $x = \pm 1.0$ and standard deviations 0.5, right subpopulation half the size of left subpopulation
(For tick > 5000 of 250,000-tick runs)

Figure 17: K-densities of rightmost Hunter x-coordinate in four-Hunter one-Sticker run, Sticker at (2.0, 0), two subpopulations with means at $x = \pm 1.0$ and standard deviations 0.5, right subpopulation half the size of left subpopulation
(For tick > 5000 of 250,000-tick runs)
Table 1: Confidence intervals and estimated potential scale reduction factors for mean Hunter x-coordinate; five independent simulation runs; Sticker at (0, 0)

<table>
<thead>
<tr>
<th>Ticks</th>
<th>Mean</th>
<th>SD</th>
<th>2.5%</th>
<th>97.5%</th>
<th>R-hat</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>-.053</td>
<td>.224</td>
<td>-.480</td>
<td>.310</td>
<td>2.85</td>
</tr>
<tr>
<td>100</td>
<td>-.044</td>
<td>.206</td>
<td>-.403</td>
<td>.350</td>
<td>2.87</td>
</tr>
<tr>
<td>250</td>
<td>-.090</td>
<td>.162</td>
<td>-.394</td>
<td>.213</td>
<td>1.24</td>
</tr>
<tr>
<td>500</td>
<td>.021</td>
<td>.187</td>
<td>-.334</td>
<td>.382</td>
<td>1.23</td>
</tr>
<tr>
<td>1,000</td>
<td>.011</td>
<td>.189</td>
<td>-.368</td>
<td>.368</td>
<td>1.04</td>
</tr>
<tr>
<td>2,500</td>
<td>.025</td>
<td>.187</td>
<td>-.338</td>
<td>.383</td>
<td>1.04</td>
</tr>
<tr>
<td>5,000</td>
<td>-.007</td>
<td>.181</td>
<td>-.368</td>
<td>.326</td>
<td>1.01</td>
</tr>
<tr>
<td>10,000</td>
<td>-.013</td>
<td>.186</td>
<td>-.365</td>
<td>.354</td>
<td>1.01</td>
</tr>
<tr>
<td>250,000</td>
<td>.0001</td>
<td>.184</td>
<td>-.356</td>
<td>.357</td>
<td></td>
</tr>
</tbody>
</table>

Table 1 Notes: Inferences from the second halves of 5 parallel simulations at various number-of-ticks. For comparison, information from one chain after 250,000 ticks is also included.

Table 2: Correlogram: lag ≤ 10, for 245,000 burnt-in ticks, Sticker at origin

<table>
<thead>
<tr>
<th>Lags</th>
<th>AC</th>
<th>PAC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.9804</td>
<td>0.9804</td>
</tr>
<tr>
<td>2</td>
<td>0.9535</td>
<td>-0.1957</td>
</tr>
<tr>
<td>3</td>
<td>0.9234</td>
<td>-0.0662</td>
</tr>
<tr>
<td>4</td>
<td>0.8932</td>
<td>0.0038</td>
</tr>
<tr>
<td>5</td>
<td>0.8634</td>
<td>-0.0003</td>
</tr>
<tr>
<td>6</td>
<td>0.8344</td>
<td>-0.0021</td>
</tr>
<tr>
<td>7</td>
<td>0.8061</td>
<td>-0.0066</td>
</tr>
<tr>
<td>8</td>
<td>0.7784</td>
<td>-0.0041</td>
</tr>
<tr>
<td>9</td>
<td>0.7513</td>
<td>-0.0052</td>
</tr>
<tr>
<td>10</td>
<td>0.7249</td>
<td>-0.0051</td>
</tr>
</tbody>
</table>

Table 2 Notes: AC: Autocorrelation; PAC: Partial Autocorrelation. The standard error of each PAC estimate is: 1/sqrt(245,000) = .002.
### Table 3: ARMA model selection

<table>
<thead>
<tr>
<th>Dependent variable:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean Other’s X Position</td>
</tr>
<tr>
<td>(1) ARMA (2,1)</td>
</tr>
<tr>
<td>(2) AR(3)</td>
</tr>
<tr>
<td>Constant</td>
</tr>
<tr>
<td>(.)</td>
</tr>
<tr>
<td>AR Coefficient 1</td>
</tr>
<tr>
<td>(.)</td>
</tr>
<tr>
<td>AR Coefficient 2</td>
</tr>
<tr>
<td>(.)</td>
</tr>
<tr>
<td>AR Coefficient 3</td>
</tr>
<tr>
<td>(.)</td>
</tr>
<tr>
<td>MA Coefficient 1</td>
</tr>
<tr>
<td>(.)</td>
</tr>
<tr>
<td>Degrees of Freedom</td>
</tr>
<tr>
<td>Log Likelihood</td>
</tr>
<tr>
<td>AIC</td>
</tr>
<tr>
<td>BIC</td>
</tr>
<tr>
<td>Q(1)</td>
</tr>
<tr>
<td>(.)</td>
</tr>
<tr>
<td>Q(2)</td>
</tr>
<tr>
<td>(.)</td>
</tr>
<tr>
<td>Q(8)</td>
</tr>
<tr>
<td>(.)</td>
</tr>
<tr>
<td>Q(16)</td>
</tr>
<tr>
<td>(.)</td>
</tr>
<tr>
<td>Q(24)</td>
</tr>
<tr>
<td>(.)</td>
</tr>
</tbody>
</table>

**Table 3 Notes:** For the AR and MA coefficients, standard errors are in parentheses. Statistically different from zero at 90% (*), 95% (**), 99% (***); confidence. For the Portmanteau Q tests, probability of zero autocorrelation for lags 1 through x in brackets.
Table 4: Augmented Dickey-Fuller tests

<table>
<thead>
<tr>
<th>Dependent variable: Change in Mean Others’ X Position (Sticker at origin)</th>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Walk Drift Trend</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mean Hunters’ x, t-1</td>
<td>-.025</td>
<td>-.025</td>
<td>-.025</td>
</tr>
<tr>
<td>(92.74)</td>
<td>(92.74)</td>
<td>(92.74)</td>
<td>(92.74)</td>
</tr>
<tr>
<td>Change in Mean Hunters’ x, t-1</td>
<td>.184</td>
<td>.184</td>
<td>.184</td>
</tr>
<tr>
<td>(33.17)</td>
<td>(33.17)</td>
<td>(33.17)</td>
<td>(33.17)</td>
</tr>
<tr>
<td>Change in Mean Hunters’ x, t-2</td>
<td>.066</td>
<td>.066</td>
<td>.066</td>
</tr>
<tr>
<td>(33.17)</td>
<td>(33.17)</td>
<td>(33.17)</td>
<td>(33.17)</td>
</tr>
<tr>
<td>Constant</td>
<td>4.26x10^-6</td>
<td>-1.06x10^-4</td>
<td></td>
</tr>
<tr>
<td>Trend</td>
<td>8.80x10^-10</td>
<td>(0.89)</td>
<td></td>
</tr>
</tbody>
</table>

Table 4 Notes: T-stats in parenthesis.

Table 5: Length of burnt-in era; estimates of mean four-Hunter x-coordinate at different points in simulation runs

<table>
<thead>
<tr>
<th>Sticker Position</th>
<th>Mean Hunter x-coordinate</th>
<th>Statistically Significant at the 95% Level After:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>50,000 ticks</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0004</td>
<td>-</td>
</tr>
<tr>
<td>0.2</td>
<td>0.0523</td>
<td>x</td>
</tr>
<tr>
<td>0.4</td>
<td>0.0983</td>
<td>x</td>
</tr>
<tr>
<td>0.6</td>
<td>0.1314</td>
<td>x</td>
</tr>
<tr>
<td>0.8</td>
<td>0.0921</td>
<td>x</td>
</tr>
<tr>
<td>1.0</td>
<td>-0.0132</td>
<td>-</td>
</tr>
<tr>
<td>1.2</td>
<td>-0.0618</td>
<td>x</td>
</tr>
<tr>
<td>1.4</td>
<td>-0.0729</td>
<td>x</td>
</tr>
<tr>
<td>1.6</td>
<td>-0.0640</td>
<td>x</td>
</tr>
<tr>
<td>1.8</td>
<td>-0.0552</td>
<td>x</td>
</tr>
<tr>
<td>2.0</td>
<td>-0.0430</td>
<td>x</td>
</tr>
<tr>
<td>2.2</td>
<td>-0.0361</td>
<td>x</td>
</tr>
<tr>
<td>2.4</td>
<td>-0.0297</td>
<td>x</td>
</tr>
<tr>
<td>2.6</td>
<td>-0.0236</td>
<td>x</td>
</tr>
<tr>
<td>2.8</td>
<td>-0.0142</td>
<td>-</td>
</tr>
<tr>
<td>3.0</td>
<td>-0.0170</td>
<td>x</td>
</tr>
<tr>
<td>3.2</td>
<td>-0.0083</td>
<td>-</td>
</tr>
<tr>
<td>3.4</td>
<td>-0.0075</td>
<td>-</td>
</tr>
</tbody>
</table>

Total Significant: 11 13 14 16 16
Percent Significant: 64.7 76.5 82.4 94.1 94.1