PROCEEDINGS OF SCIENCE

# From Internal Validation to Sensitivity Test: How Grid Computing Facilitates the Construction of an Agent-Based Simulation in Social Sciences

**Frank C.S. Liu**[1]
*Institute of Political Science, National Sun Yet-San University*
*No. 70 Lian-Hai Rd., Kaohsiung 804, Taiwan, R.O.C.*
*E-mail:csliu@mail.nsysu.edu.tw*

**Simon C. Lin**
*Academia Sinica Grid Computing Center*
*128, Sec.2, Academia Road, Nankang, Taipei 11529, Taiwan, R.O.C.*
*E-mail: simon.lin@cern.ch*

**Jing-Ya You**
*Academia Sinica Grid Computing Center*
*128, Sec.2, Academia Road, Nankang, Taipei 11529, Taiwan, R.O.C.*
*E-mail: jingya.you@twgrid.org*

**Yu-Ting Chen**
*Academia Sinica Grid Computing Center*
*128, Sec.2, Academia Road, Nankang, Taipei 11529, Taiwan, R.O.C.*
*E-mail: yuting.chen@twgrid.org*

**Jing-Lung Sun**
*Department of Computer Science, National Tsing Hua University*
*30013 No. 101, Section 2, Kuang-Fu Road, Hsinchu, Taiwan 30013, R.O.C.*
*E-mail: s9962571@m99.nthu.edu.tw*

[1] Speaker

PROCEEDINGS OF SCIENCE

Over the past decades, we see a trend that social scientists adopt the experiment approach to study our social and political world. Particularly, agent-based modelling (ABM) is employed as a tool for “thought experiment” because theorists usually (1) fall short of empirical data to contrast with experiment results and (2) are more interested in solving theoretical puzzles than empirical puzzles. Consequently, current application of ABM in social sciences (except the field of business management) has not reached the stage of sound validation and verification (V&V). Researchers are usually not sure now stable their model will perform. To take a further step out of this situation, we suggest that researchers focus on internal validation and conduct sensitivity tests. We argue that this step at least ensures that simulation process and results pass such tests are more trustworthy than those that fail the tests. Moreover, we demonstrate the utility of using grid computing for sensitivity tests. We show how we identify a model’s problem by analysing results of 8,470 runs of simulation derived from grid computing.

Keywords: Agent-based modeling; empirical validation; empirical validation.

*The International Symposium on Grids and Clouds and the Open Grid Forum*
*Academia Sinica, Taipei, Taiwan*
*March 19 - 25, 2011*

## 1. Introduction

Agent-based modeling (ABM) as a method of studying “mechanism,” “patterns,” and “complexity” has been applied to disciplines that emphasize objective analysis of the external physical world, such as ecology, physics, biology, economics, management, anthropology, engineering, sociology, and psychology over the past two decades. It has also attracted attention from political scientists and resulted in creative applications in this discipline in the late 1990s (for examples, see [1]; [2]; [3]; [4]; [5]). Although it can take pages to describe the innovative idea of agent-based modeling and how it is connected to current development in philosophy of science, recent practice of ABM is relatively straightforward: use an object-oriented computing language, such as C++, Objective-C, or JAVA, to create self-organizing objects and environment that facilitate observation of patterns emerging from the interaction between such agents.

Compared to their natural science colleagues, social scientists, particularly economists and political scientists who cannot conduct large-scaled field experiments to study causal relationships among conceptual variables, are more concerned about how ABM is used to aid intuition or facilitate “thought experiments” more than to use it for prediction [1]. As it is difficult to model empirical world comprehensively, it is important to justify a model through verification and validation (V&V) processes by asking: “How to make sure that my simulation results are communicable to those who may cast doubts on my report?”[1]

In this paper we will first introduce the two terms, verification and validation then focus our attention on validation issues, setting aside the equivalently important issues on verification. Then we will present an example of how to enhance a model’s validity. Particularly, we will present a procedure of identifying potential problems of an agent-based model (beyond debugging) using the grid computing resources of which social scientists have been unaware. We believe that making an ABM model valid is the first and the most important step to facilitate communication within a discipline and across disciplines. Below we will first summarize the issues regarding validation and ABM, and then introduce SRAS, a model of the formation of political preference that we used for this demonstration. Next, we will describe how we connect this model to grid computing and in the fourth part present a result of validation, followed by a brief discussion about how this procedure benefit future work on model validation.[2]

## 2. Validation of Agent-Based Modeling

### 2.1 General concepts about V&V

Model verification and validation (V&V) are challenging tasks to most ABM modelers. Verification is about the soundness of a model, which refers to whether or not a software program

[1] It is equally important to consider applying sophisticated designed agent-based models to examine the linkage between simulation results and empirical data. The present research on validating conceptual models is better seen as one of streams of this emerging field of research.

[2] Due to the page limit, we put materials supplemental to this paper online at http://www2.nsysu.edu.tw/politics/liu/main/FrankCSLiu.htm, including the SRAS program (the updated version free of the coding problem identified in this paper) and the four appendices: the user manual of SRAS (Appendix 1), comparison of toolkits (Appendix 2), simulation results of the 121 experiments (Appendix 3), the procedure of conducting grid computing with ASGC (Appendix 4), and the results of the second round of simulation (Appendix 5).

will function well. In other words, verification is about program debugging. While debugging is important to programmers, it is even more critical that a model is designed well. Validation, the focus of this paper, is an action that researchers make sense of the model design and can produce outcomes that readers of the results would like to trust [6].

In general, the term “validity” means a good correspondence between an artificial model and the world that we project. A validated model allows one to learn about aspects of the real world trough performing “what if” computational experiments with the model.

There are a number of approaches to achieve model validation [6]. And they can be re-categorized into two aspects: internal validation and external validation. Internal validation refers to the concept that a model needs to make sense to its readers and users, including face validity (that modeling based on plausible assumptions and that results “look right”), theory validity (the theory included in model design is valid and the model makes a valid use of the theory), and requirements validity (clearly defined requirements and research question). External validity refers to the concept that the model needs to mirror, or at least consistent with, the functionality of the real word.[3] It includes the following meanings: agent validity (agents in the model are like those in real-world in terms of behavior, relationships, and interaction processes), process validity (rules and steps of the program correspond to real-world process), model output validity (results correspond to the empirical observation), and data validity (data used in the “calibration” process or the parameter values inserted into the model have been validated). It is difficult to see a model that meets all of these requirements.

Moreover, there is tension between achieving internal validity and accomplishing external validity. It is usually difficult to build a model that is both internally and externally valid. A good internally valid model may sacrifice its external validity [7, 8]. “Where internal validity often requires abstraction and simplification to make the research more tractable, these concessions are made at the cost of decreasing external validity” [8]. Such conflict and artificiality exist in four areas or goals of experimental research. From low to high demand of external validity, these four areas of research are (1) testing theory, (2) theory stress tests, (3) searching for empirical regularities, and (4) advising policy-makers. The goal of testing theory requests the least external validity of an ABM model, while the goal of advising policy makers demands the most external validity, because policy makers need more empirically justified simulation results to propose or negate a policy idea. If simulation results cannot be prove (to be) empirically valid, the scenario about the policy consequence can become unconvincing [8].

As a model that makes sense to policy makers should be based on valid information derived from validated models, we see internal validation an important step prior to external validation. Following this flow of thinking, we like to focus our efforts on internal validation. We see a newly-formed consensus among scholars suggest that we evaluate carefully the theoretical elements of an agent-based model, and inspect how and to what extent the variation of the values of targeted parameters affect the results. Precisely speaking, when the goal of research is not making an empirical prediction or when data for external validation are unavailable, researchers using ABM are expected to (1) carefully choose parameters derived from theory (construct validity) and, (2) if the theory does not explicate some assumptions or parameters, formulate assumptions based on empirical findings (analytical adequacy), and (3) conduct sensitivity tests (or sensitivity analysis) of target parameters.

[3] It is important to note that it is difficult to model real-world phenomena.

### 2.2 Sensitivity Test for Internal Validation

There are some options to achieve internal validity [6]: one can (1) propose that the project using ABM is a special case of analytical modeling (this is the option with the least effort); (2) consult subject matter experts (SMEs) or third-party institute or professionals for approval of model design; or (3) conduct parameter sweeping. We see that the third option, parameter sweeping, is the most fundamental work to validate a model, although it is equally important to consider the first two methods. After all, readers and users of an agent-base model need a clear idea if the system would yield reasonable results when they input certain parameter values. Although the whole purpose of ABM is to observe patterns that emerge from unattended simulation, we need to keep in mind that most models are still like a fish tank, a close system that its capacity needs to be understood first.[4]

Parameter sweeping refers to "exploring the kinds of results and behaviors the model is capable of producing to identify the most interesting cases that warrant further exploration." Sensitivity test is a smaller scale of parameter sweeping; the former means basic inputs are varied in a single direction, while latter means basic inputs are varied in systematic ways [6]. A model that passes parameter sweeping or sensitivity test will release users and readers' doubts about the stability of simulation process. Below we will describe how sensitivity test is applied to examining the stability of two parameters of a model.

## 3. The Procedure of Conducting Sensitivity Test using Grid Computing Services

### 3.1 Grid Computing for Sensitivity Test

Grid computing means combining computer resources from multiple administrative domains to reach a common goal. Unlike conventional high performance computing systems such as cluster computing, grid computing is a distributed system and more loosely coupled (on the Internet), heterogeneous, and geographically dispersed [9, 10]. With assistance of general-purpose grid software libraries, known as middleware, a grid facilitates turn to be large-scale resources for computing, storage, databases and other devices.

To access the grid a researchers needs to rely on middleware to communicate with the resources. The middleware we employ is gLite, sophisticated and trusted software that has been widely used in world-scale physics studies, such as CERN-LHC. Middleware like gLite has two "gate keepers" to locate resources and schedule works. First, one needs to submit job (in our case, a batch file that specifies the combination of parameter values) to a "resource broker" (RB), a computer agent that search for available computers and deploy the piles of works to these computers. The RB of gLite is called Workload Management System (WMS). Next, given the resources, one needs to assign a scheduler to each assigned job that communicates with "local resource

[4] To accomplish external validity, one can adopt the following methods: (1) compare model results for selected cases with the real system, with another model's results, or with what would be expected by subject matter experts; (2) use model calibration (use specific cases to estimate and set values of model parameters); and/or (3) compare multiple models (compare results across multiple models). More discussion on external validation can be found in [9, 10].

management system" built in these available computers so that the computing resources as a whole will be used efficiently.[5]

### 3.2Experiment Design

Among the promising toolkits for ABM, such as Swarm, Netlogo, Mason, and Repast Java, Repast Python, and Repast Simphony, we chose Repast Simphony for this demonstration of sensitivity (Appendix 1 gives a report of our evaluation of these toolkits). We think that Repast Simphony is preferable to other toolkits for its extended functions, simpler installation process, and sufficient concepts to develop sophisticated simulation projects.

[Figure 1 is about here]

Based on our evaluation of these toolkits, we recreate the S-RAS model (the Swarm version of the Receive-Accept-Sample model) [11] on Repast Symphony and rename it as SRAS (S-RAS without "-", standing for the Repast Simphony version of the RAS model), where the "RAS" refers to a theory-based model describing how an individual citizen forms its political preferences [12]. As Figure 1 (adopted from [5]) illustrates, the key rules of SRAS (same as those used to build S-RAS) model are briefed below:

1. There are 1,600 citizen agents randomly located on a 40*40 lattice. Only two options available to all citizens: 0 and 1. Half of agents are initiated with *Opinion 0.7* (*Preference* 1) and the other half agents 0.3 (*Preference* 0). This initialized environment can be seen as a very polarized society.
2. A citizen agent's *Opinion* is an average of the past 10 *Preferences* received from its self-selected neighbors (most likely to have the same voter preference) **or** from self-selected news media (based on its ideology, 1 or 0).
3. Each citizen agent's *Preference* is recorded at a time step is based on its *Opinion* (0.0 ~ 1.0) at that time step. At any given time step, if an agent's *Opinion* is equal or larger than 0.5, its *Preference* will be recorded as 1 and all of its contacts who interact with this agent will receive 1; if the agent's *Opinion* goes lower than 0.5, its *Preference* will be updated to and perceived by its neighbor agents as 0.

Our sensitivity test goes as follows. First, in the original model of SRAS, there are two types of citizen agents: C1 agents represent ordinary citizens and C2 agents represent the politically aware who are more likely to discuss politics, access the media, and are more capable of retaining information collected from past events. The default proportion of C2 agent is 0.02 and in our simulation the parameter "*propExperts*" is set to 0. This means that all agents in the coming tests are ordinary citizens.

Second, we chose citizen agents' propensity to talk politics (TP) and propensity to access the news media (MP) as they key parameters for sensitivity test. By design, each agent in SRAS has its propensity (or "personality") to discuss politics and to access news media, ranging from 0.0 to 1.0. For example, an agent of TP 0.3 and MP 0.8 means that at each time step it is more likely to access self-selected media than to discuss politics, if it is not "caught" and occupied by another agent for political discussion. We vary the parameter values of TP from 0.0 to 1.0 with increments of 0.1, while fixing the value of all other parameters. And then vary MP from 0.0 to 1.0. This results in 11 * 11 =121 sets of parameter combination (to be detailed below). We chose to monitor the proportion

[5] The executable of a pilot job is the "worker" program. The load of computing is determined by the number of workers or pilot jobs. The more workers we give, the more grid jobs are running simultaneously; this increases the load of the system.

of agents and run each set of parameter combination for 70 times, each of which is given a unique seed and runs for 10,000 time steps. This sequence will lead to a total of 11 * 11 * 70 = 8,470 "tulples" or works.

Third, we write up the above design into a batch file that specifies the values of each parameter. Details about parameter settings for this paper are listed in Table 1.

[Table 1 is about here]

Fourth, while sending the job to grid computing, we adopt the "pull mode," which is also known as Master-Worker mode, a more advanced and efficient alternative to the conventional "push mode". Pull mode is executed this way: a researcher locates some computers on grid and assigns software to these computers (Workers). Then the researcher uploads all of the works to WMS (Master). The workers on the grip machines will come to their master to check out works and, based on their load and capacity, execute these works, and return the results to the master. Because the workers are the ones determine how much works to check out, the whole system load can be balanced instead of overloaded. The execution of this pull mode in gLite requires WMS to be the master, which is rely on a self-developed Master server program, we develop such program to assist this task (see details in Appendix 4).[6]

## 4.Simulation Results

SRAS allows researchers to observe customized phenomena at the aggregated level, such as the proportion of agents perceiving diversity and the proportion of agents changing their voter preferences. While we can monitor multiple phenomena, we choose one phenomenon that makes most sense for this study of sensitivity test: the proportion of agents favoring preference "1" or "Yes". Based on findings of previous simulation projects [5, 11], we expected that a polarized society (half of agents favoring "1" and the other half favoring "0") under two polarized news environment will remain polarized. This gives us a basic guideline to evaluate how stable the model is if we modify the combination of parameters, in this case, TP and MP. We expect that, the model will pass the sensitivity test if "no surprise" (abnormally high or low proportion, and/or irregular high standard errors of the means) found in these 121 experiments. Note that before the tasks were sent to grid computing, we had done all debugging for SRAS and made sure that the program runs. Even though, like all modelers none of us was able to foresee and guarantees that all situations fall into this expectation. (Again, this is why we think sensitivity test is a necessary task.)

[Figure 2 is about here]

Figure 2 presents the distribution of the 121 means of the proportion of agents favoring "1" (the Y-axis). The X-axis shows only TP values, while MP values, from 0.0 to 1.0 are put between TP marks. (The details are put in Appendix 3.) We see that the average goes apparently "abnormal" and puzzling (0.59) in the case where TP is 0.9 and MP is 0.1.

[Figure 3 is about here]

Let's take a look of the variance of the means across these 121 experiments. Again, due to the difficulty to show each of the 121 parameter combinations, we identify only TP in the X-axis, while we need readers to imagine that MP values, from 0.0 to 1.0, are inserted sequentially between two TP marks. As shown in Figure 3, we find that in cases where TP is higher than 0.4 and MP values

[6] The conventional "push" mode is the one by which researchers assign works to available computers. This method takes more efforts to deploy the jobs and more recourse to complete them, because one needs to manage to understand the performance and loading of the resources. (For more information, see http://www.eu-egee.org/fileadmin/documents/UseCases/Pilotjobs.html.)

are low, standard errors of the means becomes larger. Specifically, the combinations of (TP, MP) are less stable in the following cases: (.5, .0-.3), (.6, .0-.3), (.7, .0-.3), (.8, .0-.3), (.9, .0-.4), (1.0, .0-.4). This pattern implies that in the experiments where agents discuss politics very frequently and access the news media only occasionally, any single result of a run of simulation cannot represent all of runs of simulation with other seeds. That is, in these cases, the stable pattern should be presented by the average of the means, as we have done for this paper.

The puzzle shown in Figure 2 led us to inspect the specific cases. We found that, for these cases and among the 70 replications, the proportion of agents favoring "1" can be 1.0 in some cases, about 0.5 in some cases, and the other 0. This explains why the standard errors are going higher in these cases. We suspect that the distribution of the means 1, 0, and 0.5 which has more 1s than 0s, is the cause of the peak of the situation shown in Figure 2.

We then went back to examine our model SRAS. We found the major cause of this problem that we could not find when transforming SRAS from its Objective-C version to the Repast Symphony version and when we debugging the transformed version: the preferences of the two news sources that by design are supposed to be fixed 1 and 0, respectively, in S-RAS (Objective-C) fluctuate in SRAS (Repast Symphony). In other words, in the original (and correct) version of SRAS the news media's preferences are consistent whenever a citizen agent accesses them. The unattended error we made is that we let the news media change its preferences whenever a citizen agent accesses them. We suspect that this error will lead to a scenario where both news sources favoring 1 (or 0) at the same time and therefore can homogenize the whole society's preferences. (The correct version of SRAS that is available online, see footnote 1).

[Figure 4 is about here]

Given the updated SRAS we conducted sensitive tests (121 experiments and each lasts 1,0000 ticks or time steps) over again. This time we ran each case for 100 times, resulting in a total of 11 * 11 * 100 = 12,100 "tulples" or works. The average of the means shown in Figure 4 shows a pattern that is less surprising and consistent with our earlier expectation. The average of the targeted case (TP= 0.9 and MP=0.1) is 0.52, not the highest one among the simulation results.

[Figure 5 is about here]

The pattern of the distribution of the standard errors shown in Figure 5 is consistent with that shown in Figure 3. Given the second round of sensitivity test, we come to a more confident summary, which is consistent with the one stated earlier: in cases where agents discuss politics very frequently and access the news media only occasionally, the simulation results may vary a bit more than the other cases. Specifically, the mean of proportions of agents favoring "1" will be a little higher in the following cases, in terms of (TP, MP): (.6, .1), (.7, .0-.2), (.8, .0-.1), (.9, .0-.2), (1.0, .2-.4) (see Appendix 5 for the details).

## 5.Conclusion

This paper is devoted to introduce to social scientists how grid computing resources facilitates social simulation, particularly agent-based modeling. We have to acknowledge, first, that agent-based modeling is not the only approach that social scientists should adopt when they like to use computers for their research. (In effect, the majority of researchers who use computers to facilitate their studies focus on statistical analysis, hypotheses testing, content analysis, and discourse analysis). Neither do we suggest that validating a model using sensitivity test should be prior to other types of application of agent-based modeling. The example presented by this paper is one that that the construction of an agent-based model is benefited from the advances of grid computing.

Even though, we believe that there exist other innovative ways to make the best use of this magnificent resource.

Second, although our original goal of this project is to make it easy to a researcher using agent-based simulation to access grid computing resources, it turns out that there remain a few technical details that are not easy to explain to social scientists, particularly the part of programming the scripts and of submitting parameter settings to the middleware. Even though we manage to explicate the procedures of replicating what we have done for this paper, we much note that there remain some parts that require assistance from experts from institutes like ASGC. It is still our goal, however, that in a near future this procedure will be standardized, more simplified, and made easier to potential users of grid computing in general and modelers using ABM in particular.

Third, by identifying the coding issue in SRAS, we have demonstrated how important it is to conduct sensitivity test to validate an agent-based model. Sometimes we hardly identify a problem of a program through a debugging procedure, and when the model grows complicate it becomes more difficult to ensure how consistent the coding complies with the original design. When we focus on debugging and experiment design, we usually fall short of attention and method to check if there is no other logical problems embedded in the codes. Therefore, we highly recommend that ABM modelers consider validate their own models though sensitivity tests (by reporting the results of sensitivity test of the targeted parameter values of their model in a footnote) before generating interpretations and generalization of their simulation results.

Finally, there is a puzzle left by this paper. We find a pattern that the simulation results of models where agents talk politics more than accessing the news media are less stable. We welcome more research on discovering the mechanism behind this pattern, associating such a pattern to a theory, and/or give reasonable explanation of such patterns. We believe that such efforts will not demise the value of applying ABM to social studies; instead, we see more cross-disciplinary insights may inspire scholars to join this format of thought experiment.

Figure 1: The Flow Chart of How Citizen Agents of SRAS Update Their Preference

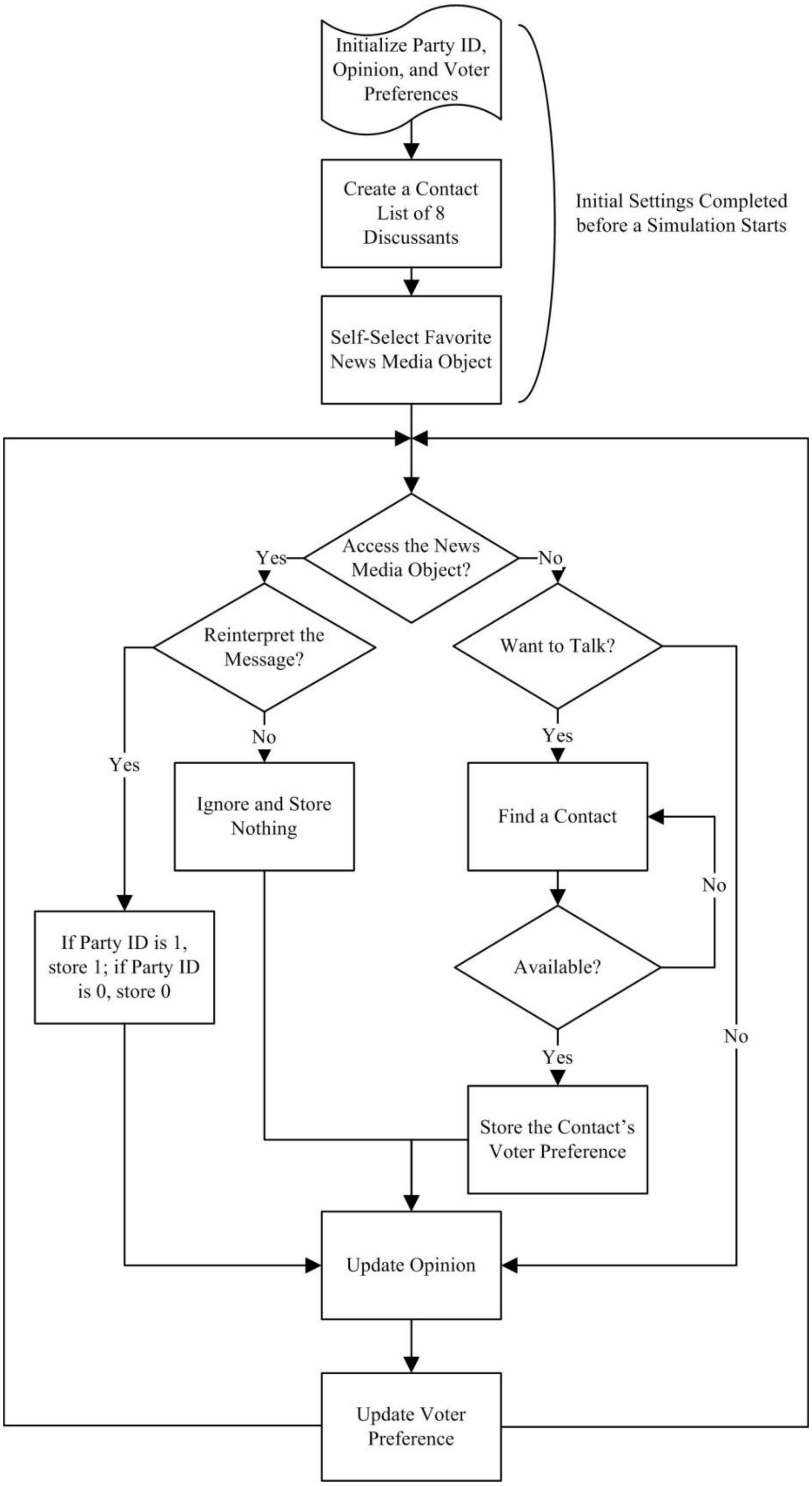

Initialize Party ID, Opinion, and Voter Preferences
Create a Contact List of 8 Discussants
Self-Select Favorite News Media Object
Initial Settings Completed before a Simulation Starts
Access the News Media Object?
Yes
No
Reinterpret the Message?
Want to Talk?
Yes
No
Yes
Ignore and Store Nothing
Find a Contact
No
If Party ID is 1, store 1; if Party ID is 0, store 0
Available?
No
Yes
Store the Contact's Voter Preference
Update Opinion
Update Voter Preference

Table 1: The Specification of Parameter Values of SRAS for Grid Computing

| Parameters | Description | Set Value |
|---|---|---|
| duration | The number of time steps. | 10,000 |
| worldXSize | The column size of the world. | 40 |
| worldYSize | The row size of the world | 40 |
| GUIshots | Take a shot at the end | 0 |
| numMedia | How many media in simulation | 2 |
| propExperts | The proportion of citizen2 in the model | 0 |
| propYES | The proportion of favoring "YES" for citizen1 | 0.5 |
| c1CHECKMEDIA | Citizen1 can access media or not. | 1 |
| c1Conform | The critical value of Opinion to change Preference | 0.5 |
| memLength | Citizen1's memory length or the capacity of storing past political information that influences vote preferences | 10 |
| c1TPMax | The maximum proportion of finding somebody to discuss politics of citizen1. | Max=Min<br>[0.0, 1.0] |
| c1TPMin | The minimum proportion of finding somebody to discuss politics of citizen1. | |
| c1ExpertiseMax | The maximum level of expertise of citizen1. | 5 |
| c1ExpertiseMin | The minimum level of expertise of citizen1. | 1 |
| c1MPMax | The maximum proportion of accessing media of citizen1. | Max=Min<br>[0.0, 1.0] |
| c1MPMin | The minimum proportion of accessing media | |
| c1SPMax | The maximum proportion of selective perceiving mass media messages of citizen1. | 0.5 |
| c1SPMin | The minimum proportion of selective perceiving mass media messages of citizen1. | 0.5 |

Figure 2: Simulation Results: The Mean Proportion of Agents Favoring "1" across the 121 Experiments

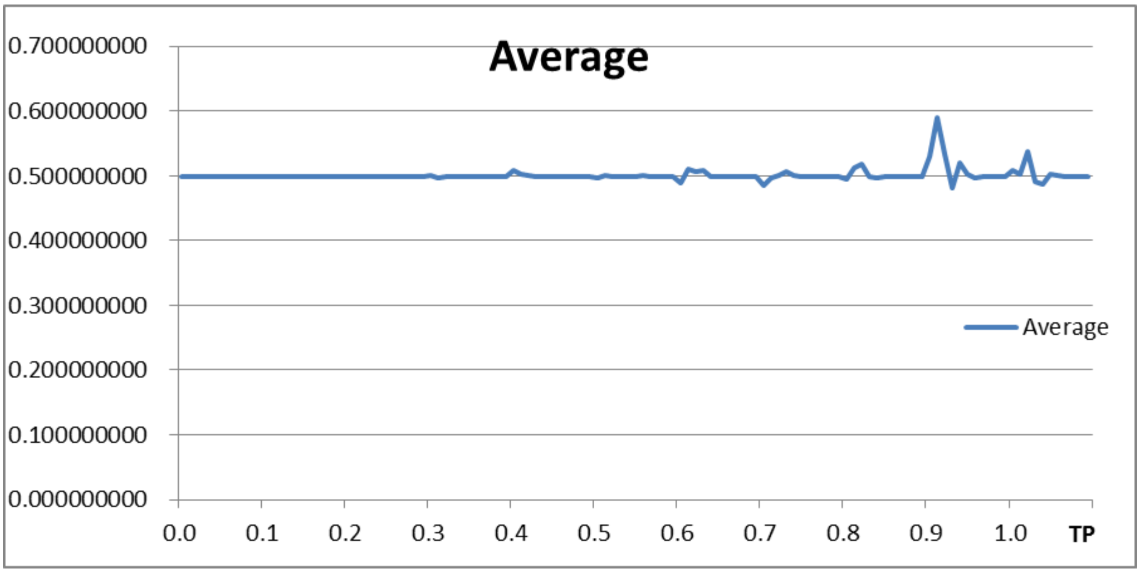


Figure 3: Simulation Results: The Standard Error of the 121 Experiments

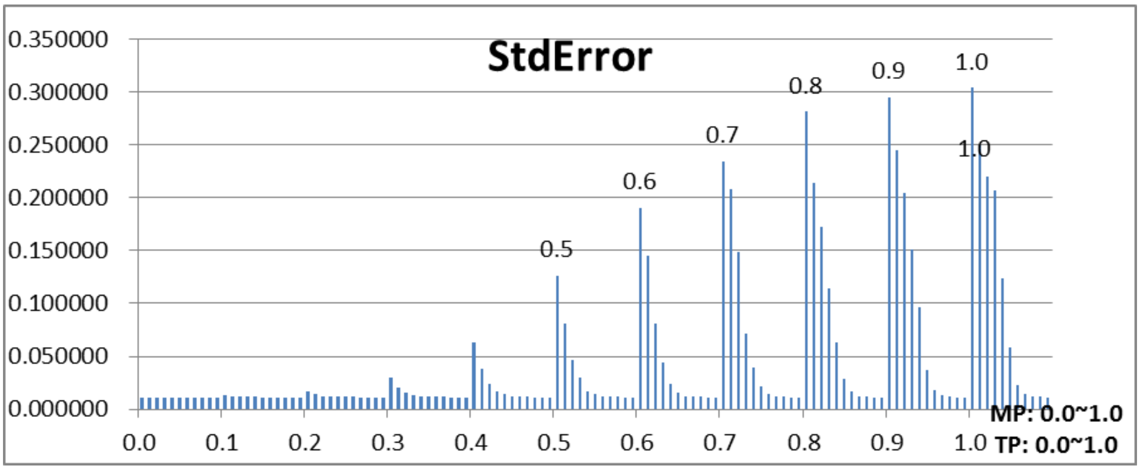


Figure 4: Simulation Results: The Mean Proportion of Agents Favoring "1" across the 121 Experiments based on the updated SRAS

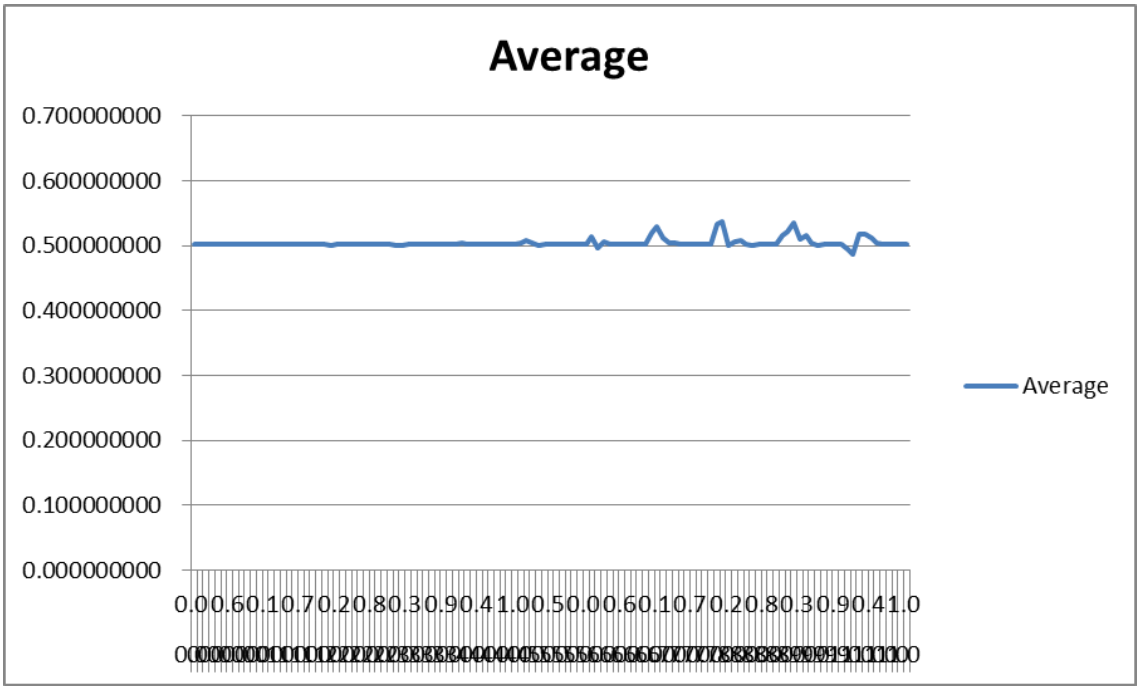
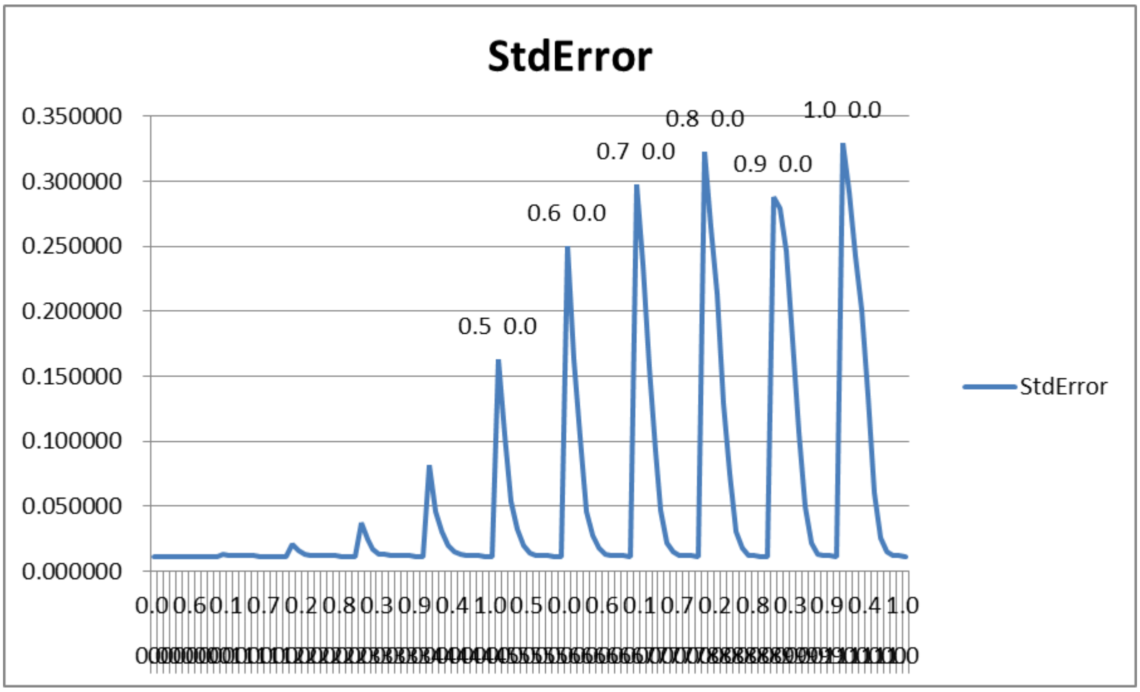


Figure 5: Simulation Results: The Standard Error of the 121 Experiments based on the updated SRAS



## References

1. Axelrod, R.M., *The Complexity of Cooperation: Agent-Based Models of Competition and Collaboration*. Princeton studies in complexity.1997, Princeton, N.J.: Princeton University Press.
2. Cederman, L.-E., *Computational models of social forms: Advancing generative process theory.* American Journal of Sociology, 2005. **110**(4): p. 864-893.
3. Huckfeldt, R.R., P.E. Johnson, and J.D. Sprague, *Political Disagreement: The Survival of Diverse Opinions within Communication Networks*2004: New York: Cambridge University Press.
4. Laver, M., *Policy and the dynamics of political competition.* American Political Science Review, 2005. **99**(2): p. 263-281.
5. Liu, F.C.S., *Polarized news media and the polarization of the electorate.* International Journal of Artificial Life Research, 2010. **1**(1): p. 35-50.
6. North, M.J. and C.M. Macal, *Managing Business Complexity: Discovering Strategic Solutions with Agent-Based Modeling and Simulation*2007: Oxford University Press.
7. Laurent, G., *Improving the external validity of marketing models: A plea for more qualitative input.* International Journal of Research in Marketing, 2000. **17**(2-3): p. 177-182.
8. Schram, A., *Artificiality: The tension between internal and external validity in economic experiments.* Journal of Economic Methodology, 2005. **12**(2): p. 225-237.
9. Wikipedia. *Grid computing* [cited 2011 Feburary 20]; Available from: http://en.wikipedia.org/wiki/Grid_computing.
10. TWGrid. *What is Grid?* [cited 2011 January 3]; Available from: http://www.twgrid.org/en/index.php?option=com_content&task=view&id=5&Itemid=90.
11. Liu, F.C.S., *Modeling political individuals using the agent-based approach: A preliminary case study on political experts and their limited influence within communication networks.* Journal of Computers, 2009. **19**(4): p. 8-19.
12. Zaller, J., *The Nature and Origins of Mass Opinion*1992: New York, NY: Cambridge University Press.