ABMSync: An Approach to Agent Based Modeling
Stephen Pelhan, Randall Fradin
Faculty Advisor: Barry Silverman

Abstract:
- Create a developer-oriented agent based modeling (ABM) platform implemented in Python, with a well-documented API and file I/O specification that can be used to create scientifically useful and interesting simulations
- Create an example reimplementation of the Identity Repertoire Model used by Political Science-Identity (PS-I)

Background on Agent Based Modeling (ABM):
- ABM is a computational model that runs a simulation in order to examine interactions between groups or actors
- ABM introduces a new concept of individual agents, that each have rules for how they will evolve and interact
- The results from these individual interactions can often produce much larger, emergent behavior

Reasons for creating a new program for ABM:
- Some current programs are too complex for the intended audience of political scientist
  - Programs such as SWARM and RePast are capable of creating complex simulations but have an incredible learning curve
  - These programs also require extensive coding knowledge
- Other current programs are too limited in their scope
  - PS-I is capable of handling only the Identity Repertoire Model and still has huge flaws
- Our program seeks a balance between these two approaches

Creating a program with both flexibility and capability:
- In order to address both the problems of needing to be able to handle a large range of potential applications while still maintaining performance levels we structured our program as follows:
  - The three levels are our program allow for both flexibility and power
  - The top level is the backbone needed to run any ABM simulation
  - Under that level are specific types of ABMs
  - This second level is where developers will be able to access our API and create new types of models for their needs. While this level is more complicated it will only need to be created once, and then individual models will be able to run repeatedly on that type of model. As part of our project we reimplemented a sample model called the Identity Repertoire Model that is currently used by PS-I
  - The last level is the individual test. This is the level that the political scientist will work at. It will be their responsibility to create individual models and evolution rules that best allow them to predict the evolution of the area

Advantages over previous programs
- AMBSync allows for the same flexibility as SWARM and RePast because it allows developers to create their own type of model and then distribute that to the political scientists that will create the model
- The modelers only need to create a model file through a well defined XML format
- Because the program is written in Python it is easily readable and changeable. This allows other developers to add features on top of what has already been implemented

Conclusions:
- AMBSync represents a vast improvement over the current technology for agent based modeling. In addition to maintaining the same computation power that previous programs had it increases the flexibility while at the same time increasing the ease of use by allowing for abstraction and modularization
- AMBSync also allows for other features to seamlessly be integrated on top of the program, such as a model viewer, model editor, and statistics viewer

Above is an example of one type of ABM known as the Identity Repertoire Model as illustrated by the program PS-I

The identity Repertoire Model is characterized by a 2-Dimensional field of agents with each agent being “activated” on one identity and evolve based on their neighbors

This model is a representation of Bangladesh, India where each agent corresponds to a section of the population

Above is the same model evolved for 55 time steps. The important changes to note are the increased clumping of like colors, most notably the blues, purple, and light green

Senior Project Poster Day 2009, CIS Dept. University of Pennsylvania