Spatially Aware Agents: An effective and efficient use of GIS data within an Agent-based Model

Ryan C. Kennedy1, Kelly E. Lane2, Agustín Fuentes3, Hope Hollocher2, Greg Madey1
1Department of Computer Science and Engineering
2Department of Biological Sciences
3Department of Anthropology
University of Notre Dame

Keywords: performance, scientific simulation, raster data, vector data, epidemiological modeling

Abstract
Agent-based modeling (ABM) is quickly becoming a choice tool for researchers, as it is very adept at modeling natural phenomena. Real-world data has the ability to further increase the usefulness of agent-based models. This can be achieved through the inclusion of geographical information system (GIS) data. Such integration between ABM and GIS may sound trivial, but is difficult to do effectively, particularly as the complexity of GIS data and the amount of agents increase. Here, we present methods and recommendations on including GIS data in a complex simulation model. We demonstrate our techniques to an advanced epidemiological model.

1. INTRODUCTION
Real-world simulations have the potential to be very valuable to researchers. ABMs are intuitive because their implementations rely on more natural behavioral rules than on complex equations that approximate average behaviors in the system. This leads to a more direct translation from natural phenomena to a simulation. It is logical to integrate real-world data into the simulation environment; however, as Gilbert [1] pointed out, utilizing this GIS data for dynamic agents is a difficult challenge that has not been adequately solved. GIS data has successfully been integrated into ABMs for several years now; however, the ability to run complex simulations with thousands of spatially aware agents is computationally challenging. In this paper, we present several methods to integrate GIS data into a simulation environment. We describe an epidemiological model that utilizes GIS data and offer insight on how to efficiently integrate GIS data into a model, depending on the model’s complexity and needs.

The organization of this paper is as follows. In section 2, we discuss the integration of ABMs and GIS. Section 3 details our simulation model. We finish with a discussion in section 4.

2. GEOGRAPHIC INFORMATION SYSTEM DATA AND AGENT-BASED SIMULATIONS
GIS data has a variety of applications and spans many fields. Simply, a GIS is a system in which real-world environmental data is represented. Examples can include roads, rivers, coastlines, governmental boundaries, rainfall, temperature, population, elevation, disease prevalence, among many others. GIS data is typically stored in either the raster or vector formats. Raster data is characterized as a collection of pixels, or cells. These cells typically make up a grid-like structure, with each cell having its own attributes and properties. Vector data is coordinate-based whereby environmental data is represented as points, lines, and polygons. These features then have associated characteristics. While raster data lends itself directly to the grid-like frameworks of ABMs, it is subject to spatial resolution issues and requires more storage space, making it computationally expensive. Vector data is more realistic and suffers less from data loss than raster data, while requiring less storage space. However, querying vector data can also be computationally expensive. For example, querying a set of complex polygons representing forests in an environment would potentially require multiple queries of each polygon for each agent, unless some sort of indexing of the polygons was performed. Even so, querying a complex polygon is expensive. Figure 1 visually compares raster and vector data. A means to combine the benefits of raster and vector data to create spatially aware agents would be an important step in the advancement of ABMs with GIS.

While previous studies have described ABM coupled with GIS, most existing models do not have agents that intelligently move based on their current environment. Castle et al. [2] mention numerous toolkits and applications yet fail to go beyond the incorporation of GIS data into a model and into the realm of its effective use. Crooks [3] more deeply describes the realm of space within ABM and offers example applications but does not specifically address the underlying issue of how agents can most efficiently access GIS data. Gimblett [4], Keeling et al. [5], and Brown et al. [6] further describe aspects of the integration of ABMs and GIS data, but also do not go into detail regarding approaches to efficiently create spatially aware agents.
Moreover, standard means of accomplishing this are computationally expensive and therefore not feasible for complex, large-scale simulation models. In many cases, only particular parts of a GIS are necessary; utilizing a feature-rich GIS toolkit at simulation runtime is not typically advisable. We next describe the problem and offer increasingly better solutions.

2.1. GIS Data in ABMs

In traditional ABMs, agents typically move about a grid-like structure. Spatially aware agents must move about the same structure, but in a manner such that each move is influenced by the surrounding environment in addition to other agents. A simple example would be allowing agents to move preferentially into one landscape over another. When an ABM environment is built upon GIS data, queries can be expensive, particularly with complex data or movement. As a general rule, the more complex the GIS data, the more difficult it is to efficiently utilize in an ABM. Additionally, the more GIS data that is available, such as multiple landscape features, the more time consuming it will be for agents in an ABM to query. Put simply, at each timestep, an agent needs to query its unknown surroundings and make a decision regarding movement. The more GIS data there is, the longer this will take. A common solution is to approximate GIS data to the level of granularity required for a given model. As such, the amount of GIS data is decreased, while maintaining a sufficient amount of environmental data. We next describe several ways to access GIS data from a simulation, offering advantages and disadvantages for each.

2.1.1. Raster Queries

Raster-based spatial queries made through a spatial package can be costly, as the mechanisms by which agents access this data are typically not optimized. Additionally, the storing and loading of large raster data files is inefficient at simulation runtime, particularly when not all of the data is necessary. Raster files are also not ideal for complex GIS data or in simulations where a fine-scaled granularity is required, particularly if there are a substantial number of agents. An advantage of utilizing raster files is that they do easily map to traditional ABM grid spaces.

2.1.2. Spatial Queries

Spatial queries on vector-based GIS data are the most accurate way an agent can interact with GIS data. Here, an agent queries the loaded GIS data to determine its surroundings. While very accurate, the cost of performing a spatial query increases as the complexity of the data increases. For example, it may be a simple query to determine if an agent is within a simple rectangle while it would be very complex to do the same query on a large, many-sided polygon. Repeatedly performing such queries is expensive, and the problem is exaggerated as the number of agents and the amount of spatial data increases.

2.1.3. Simplified Spatial Queries

Simplified spatial queries are identical to spatial queries, with the exception that the vector data has been approximated in a manner such that the number of vertices in a line or polygon is decreased, while maintaining an appropriate level of data integrity. The Douglas-Peucker algorithm [7] is commonly used to perform such simplifications. This technique offers a speedup over traditional spatial queries, but at a cost of less accurate environmental data. However, repeatedly performing similar or identical spatial queries is redundant and can be remedied. Figure 2 shows a near 100% data simplification.

2.1.4. Precalculated Query Matrix

Recognizing the drawbacks of earlier techniques, we developed a technique we call the precalculated query matrix. This technique relies on the advantages of raster data while utilizing the accuracy of vector data. Here, vector files are used in conjunction with spatial queries to build..
arrays of spatial data. Specifically, we iterate through the vector data at a chosen granularity and perform queries at each point. This is time consuming, but only needs to be performed once, prior to simulation runtime. The results of building this precalculated query matrix can then be accessed by agents in constant time. The only disadvantage to this method is that the finer the granularity, the longer the initial calculations will take, which results in more data that needs to be loaded. The advantages include agents that can more quickly query their environment and also that this method scales well, both in terms of the amount of GIS data available and in the number of agents. Researchers also can choose a granularity that is adequate for their needs. Currently, we use multiple precalculated query matrices in our model, with some environmental data being combined.

2.2. Spatially Aware Agents

In the previous subsections, we listed ways by which agents can query their environment. Once agents are able to adequately and efficiently survey their surroundings, they must be able to make use of that data to become spatially aware. In our model, we use our precalculated query matrix as a space on which our agents move. This matrix, which has been built at a specified granularity, acts as the foundation on which the agents interact and move about their environment. To display this movement on the native vector GIS data, we use hash tables to “map” the native GIS latitude and longitude points to our matrix, and vice versa. This mapping avoids repetitive calculations, while allowing the agents to find their true coordinates with ease. This also assists in enabling agents to move with complex rules, which we next describe.

2.2.1. Movement

Adding intelligent movement to agents in a GIS-based environment is challenging. With raster data, agents must perform tedious queries through the GIS system to determine the surrounding landscape. Spatial queries are also not efficient, as the queries are redundant and take considerable time. Utilizing a precalculated query matrix enables us to create many agents with complex and realistic movement in rapid time.

In traditional ABM cellular automata spaces, agent behavior is based on a von Neumann or Moore neighborhood. Specifically, von Neumann neighborhoods describe the four cells adjacent to the current cell in a traditional square grid. A Moore neighborhood extends this to the surrounding eight adjacent cells, including those diagonally adjacent. Performing spatial queries on such spaces would be tedious and inefficient, particularly if the neighborhood was extended beyond a Moore neighborhood. In our model, spatial movement is based on a Moore neighborhood, but it will be extended to allow for an arbitrary amount of lookahead. For agents to move intelligently, they must know the landscape they are currently in, as well as the landscape that surrounds them. In our model, we have eight environmental layers that influence movement, described further below. Agents are more likely to remain in “suitable habitat” as opposed to move into or cross “unsuitable habitat” as described by their biological properties and GIS dataset. To represent these transitions, we use a matrix of probabilistic movement values. This table consists of values representing the “probability” that an agent would move from a given landscape to a given landscape. This calculation is performed for each of the eight surrounding cells. A directional bias is also added to agents so that they are more likely to continue in the same general direction. Once the values for the surrounding cells have been calculated, we normalize the values and then use probabilities to determine the next location for the agent, if it moves at all. This is all performed very quickly, as the lookups for the surrounding cells can be performed in constant time, and allows for realistic movement of our agents. Figure 3 shows a simplified version of our movement on an example grid. We
next describe the simulation model we have developed in more detail.

In the artificial intelligence agent realm, intelligent agents are classified as simple reflex, model-based reflex, goal-based reflex, utility-based, or as learning [8]. Our agents could be classified as utility-based, but with a stochastic-based utility function and with stochastic-based decisions. This classification is fits because our agents make movement decisions based upon utility – they are happier in certain landscapes than others, and this happiness is determined by their previous location and current landscape.

Figure 3. Agent Movement. We show a macaque in a forest environment. The macaque will query surrounding cells and base its movement on them and its current environment.

3. A SIMULATION MODEL FOR THE MODELING OF PATHOGEN TRANSMISSION

We have created a model, named LiNK [9], to aid in the understanding of pathogen transmission patterns. This model was designed to simulate the spread of infection amongst long-tailed macaques (Macaca fascicularis) on the Indonesian island of Bali. We have coupled detailed GIS data with a deep knowledge of the macaque population to create a rich simulation.

3.1. Background

Several zoonotic diseases have recently emerged on the Asian landscape, and primates have been implicated as both hosts and reservoirs in these disease emergences in humans. Increasing anthropogenic landscape changes have increased the incidence of human to non-human primate interaction, potentially leading to bi-directional pathogen transmission events [9,10,11]. In our model, we evaluate how landscape changes might influence pathogen transmission patterns, based on the behavior and dispersal patterns of long-tailed macaques across the island of Bali. We specifically aim to address the following research questions:

1. What are potential rates and routes of pathogen transmission in macaques across the island?
2. How do pathogen life history parameters impact this transmission?
3. Do the answers change with the inclusion of humans as a component of the landscape?

Landscape plays a very important role in these questions, necessitating the use of GIS data in our simulation.

On the island of Bali, a unique system of temples has existed in the landscape for centuries; these temples and their associated forests act as refugia for the large populations of long-tailed macaques that live there [13]. Each temple population consists of between 30 and 250 individuals. Existing behavioral and preliminary genetic evidence has documented the matrilocial society of the macaques, resulting in strong female philopatry [9,13,14]. Females remain at their birth groups throughout their lives and dominance is inherited maternally. Typically, subdominant and subadult males disperse from their natal population around age seven, traveling to nearby temple populations. Currently, actual dispersal distances and rates are unknown. Figure 4 shows a selection of the 44 temple sites with macaques on Bali [15].

The ability of long-tailed macaques to coexist with humans has enabled a number of macaque populations to thrive in areas where other primate species have become extinct [13]. In Bali, human land-use patterns have resulted in a mosaic of riparian forest, small forest patches, agricultural lands, and urban areas across much of the island. The broad distribution of the macaque populations on Bali suggests that these macaques are utilizing the human modified landscape as it currently exists. Due to the protection and resource availability available at the temples, macaques are able to exist in moderately high densities alongside high human densities. This co-existence with humans, particularly surrounding the temples, has created an ideal study setting for evaluating how primate behavior and anthropogenic landscape changes influence pathogen transmission [11].

3.2. Conceptual Model

The conceptual model was developed by K.E. Lane, with support from A. Fuentes and H. Hollocher. This group has closely studied the macaques and an array of pathogens for a number of years. The basic model consists of a display of Bali with temple sites and macaques. We also display the contents of a given temple and provide the user with multiple model and pathogen parameter options. More details on the model characteristics follow.

Agents: Our agents are macaques, each with their own properties, such as location, sex, age, natal temple, and
Figure 4. Selected temple locations on the island of Bali, Indonesia. Total macaque-associated temples number 44 and are spread throughout the island.

Infection status. Macaques move in accordance to their surrounding environment, and males have the ability to enter and leave temples. Our model can support thousands of agents.

Behavior: Macaques have the ability to move through their environment, interact with other macaques, reproduce, and die. Movement is dictated by their surrounding environment and the associated probability values; macaques query their neighborhood and move appropriately. Macaques within a temple move in a more randomly. All macaques also have the ability to transmit pathogens when within a specified distance of one another. Reproduction is handled by allowing female macaques to produce offspring, with inherited traits, after they reach a specified age. As macaques age, they have a higher probability of dying.

Interface: Researchers interact with the model through a simple control panel that allows them to tweak simulation parameters. Once the parameters are set, the user can begin running the simulation. The simulation is displayed via OpenMap [16], shown in Figure 5. Users can also see within temples.

Pathogens: Our model has the ability to simulate a wide array of pathogens through the incorporation of several important parameters. Infectivity is how infectious the pathogen being modeled is, while virulence is the proximity a macaque must be to another macaque to have the ability to transmit a pathogen. Latency is how long a macaque takes to become symptomatic after becoming infected. Acquired immunity refers to the amount of time a macaque is immune to contracting a pathogen after having been previously infected. Clearance time is the amount of time a macaque takes to be cleared of a pathogen. Finally, natural resistance is the proportion of macaques that are immune to a given pathogen. Selected pathogen-related variables and their relationships are shown visually in Figure 6.

Figure 5. LiNK Display. Macaques are shown as circles and temple sites as squares. All layers are enabled.

Space: The macaques move about on 2D grids that represent temples sites and the island. The sizes of the temple site grids reflect the actual size of the temples. The island grids are extrapolated from GIS data, at a customizable granularity. For our purposes, a grid cell has sides of roughly 100m, leading to over one million possible locations. Each grid is called a layer; we have a total of eight layers: cities, forests, lakes, rice fields, rivers, roads, temples, and the actual island (called coast). These eight layers are melded together and use the same coordinate system. The coast and temple layers are required, while the rest can be turned on or off.

3.3. Implementation

There are several tools and technologies that made this study possible. Everything is coded in Java with the Repast simulation toolkit [17]. We utilize Repast and OpenMap to display the model and use GeoTools [18] and JTS Topology Suite [19] to interact with the spatial information. The choice of the tools used in this study was primarily driven by the necessity to process and visualize geographic information.

3.4. Performance

The model has utilized the aforementioned techniques to interact with GIS data. We started with hefty raster files and refined our method until we achieved the balance of specificity and speed we desired. Table I shows the initial load time for the model for each technique, and Table II (and Figure 7) show the number of timesteps simulated per second for each query mechanism. These tables (and figure) show data with either the coast and lakes or the coast, lakes, and forests layers enabled, all with the same number of initial agents. Spatial queries were predictably slowest, as the raw vector files contain an enormous amount of data,
making calculations expensive. Utilizing raster data offers a significant improvement but with the drawback of the long initial startup time. Our simplified spatial query greatly improves upon the traditional spatial query, but performance drops significantly as more layers are added. Table III and its corresponding Figure 8 shows the scalability, in terms of number of agents, for the raster and precalculated query matrix method. The precalculated query matrix method scales very well as the amount of GIS data increases and adequately as the number of agents increases. The precalculated query matrix offers the best, scalable results. All performance tests were run on a single core as a single thread on a Core 2 Duo 2.0 GHz laptop, highlighting further potential in scalability.

### Table I: Performance Comparison

<table>
<thead>
<tr>
<th></th>
<th>Start Up Time (s)</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Coast, Lakes</td>
<td>Coast, Lakes, Forests</td>
<td></td>
</tr>
<tr>
<td>Raster Query</td>
<td>35</td>
<td>42</td>
<td></td>
</tr>
<tr>
<td>Spatial Query</td>
<td>3.5</td>
<td>3.5</td>
<td></td>
</tr>
<tr>
<td>Simplified Spatial Query</td>
<td>1.8</td>
<td>2.5</td>
<td></td>
</tr>
<tr>
<td>Precalculated Query Matrix</td>
<td>1.6</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>

### Table II: Performance Comparison

<table>
<thead>
<tr>
<th></th>
<th>Timesteps/s</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Coast, Lakes</td>
<td>Coast, Lakes, Forests</td>
<td></td>
</tr>
<tr>
<td>Spatial Query</td>
<td>1.6</td>
<td>0.15</td>
<td></td>
</tr>
<tr>
<td>Raster Query</td>
<td>18.5 (11x faster)</td>
<td>19 (126x)</td>
<td></td>
</tr>
<tr>
<td>Simplified Spatial Query</td>
<td>39.5 (25x)</td>
<td>15.8 (105x)</td>
<td></td>
</tr>
<tr>
<td>Precalculated Query Matrix</td>
<td>126.2 (79x)</td>
<td>124.1 (827x)</td>
<td></td>
</tr>
</tbody>
</table>

### Table III: Performance Comparison

<table>
<thead>
<tr>
<th>Number of Initial GIS Agents</th>
<th>Timesteps/s</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Raster Query (3 Layers)</td>
<td>51.3</td>
<td>29</td>
<td>19.9</td>
</tr>
<tr>
<td>Raster Query (7 Layers)</td>
<td>33.6</td>
<td>27.6</td>
<td>11</td>
</tr>
<tr>
<td>Precalculated Query Matrix (3 Layers)</td>
<td>140.7</td>
<td>131.4</td>
<td>83.8</td>
</tr>
<tr>
<td>Precalculated Query Matrix (7 Layers)</td>
<td>137.5</td>
<td>129.5</td>
<td>82.9</td>
</tr>
</tbody>
</table>

### Table IV: Advantages and Disadvantages

<table>
<thead>
<tr>
<th></th>
<th>Raster Query</th>
<th>Spatial Query</th>
<th>Simplified Spatial Query</th>
<th>Precalculated Query Matrix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy of GIS Data</td>
<td>Good</td>
<td>Excellent</td>
<td>Very Good</td>
<td>Very Good</td>
</tr>
<tr>
<td>Amount of GIS Data</td>
<td>Good</td>
<td>Poor</td>
<td>Fair</td>
<td>Excellent</td>
</tr>
<tr>
<td>Complexity of GIS Data</td>
<td>Fair</td>
<td>Excellent</td>
<td>Very Good</td>
<td>Very Good</td>
</tr>
<tr>
<td>Load Time</td>
<td>Poor</td>
<td>Very Good</td>
<td>Very Good</td>
<td>Excellent</td>
</tr>
<tr>
<td>Memory Requirement</td>
<td>Fair</td>
<td>Good</td>
<td>Very Good</td>
<td>Excellent</td>
</tr>
<tr>
<td>Number of Agents</td>
<td>Very Good</td>
<td>Poor</td>
<td>Fair</td>
<td>Very Good</td>
</tr>
<tr>
<td>Timesteps/s</td>
<td>Very Good</td>
<td>Poor</td>
<td>Fair</td>
<td>Excellent</td>
</tr>
</tbody>
</table>

### Figure 6. Visual Schematic of Pathogen Parameters.

The diagram above shows the relationship over time between pathogen parameters in our simulation.

### 4. DISCUSSION

When designing an ABM that utilizes GIS data to simulate spatially aware agents, there are a number of factors that should be considered. Scalability in terms of the number of agents is probably the most important factor to consider. Other important issues include the complexity of the GIS data and the amount of GIS data the simulation will rely upon. An adept modeler will utilize the GIS data at a granularity appropriate for the simulation at hand. In terms of speed, raster data scales reasonably with increasing GIS complexity, but not as well with the number of agents. Spatial queries scale poorly with an increase in the amount of GIS data and complexity, as well as with an increase in the number of agents. Regarding accuracy, utilizing vector data via spatial queries offers the highest accuracy, but at the highest performance cost. Raster data and our precalculated query matrix offer varying levels of accuracy, while offering faster speed. Table IV summarizes general ratings for each approach. Possible ratings are Poor, Fair, Good, Very Good, and Excellent. Accuracy of GIS data refers to the faithfulness to the original GIS data, while amount of GIS data refers to the ability of each technique to handle multiple layers of GIS data. The remaining metrics are self-explanatory. Our precalculated query matrix scales best in terms of number of agents and particularly in amount of GIS data present, as shown in Figure 8.

### 3.5. Results

LiNK has demonstrated the importance of landscape heterogeneity in the scope of epidemiological modeling [8]. The model has been improved in terms of speed and scalability through an abstraction of typical GIS data representation. We have shown the ability to have many agents interact with complex spatial data in a timeframe adequate for a simulation while addressing the research concerns at a high-level.
Figure 7. Speed comparison of varying query methods.

Figure 8. Scalability with respect to number of agents and amount of GIS data.
In this paper, we have integrated ABM and GIS, utilizing GIS shapefiles as the basis of our model environment. We have gone a step further by introduction a precalculated query matrix, which our agents use to intelligently and efficiently move about based on an abstraction of the GIS data. As such, this has led to a more informative and realistic model that yields important scientific data.

5. ACKNOWLEDGMENT

Funding for this work was provided in part by the National Science Foundation (BSC-0629787) and by the University of Notre Dame’s Institute for Scholarship in the Liberal Arts.

6. REFERENCES