Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence

Embedding System Dynamics in Agent Based Models for Complex Adaptive Systems

Maarika Teose
Applied Mathematics
Cornell University
mt524@cornell.edu

Kiyan Ahmadizadeh
Computer Science
Cornell University
kiyan@cs.cornell.edu

Eoin O'Mahony
Computer Science
Cornell University
eoin@cs.cornell.edu

Rebecca L. Smith
Pop Med & Diagnostic Sci
Cornell University
rls57@cornell.edu

Zhao Lu
Pop Med & Diagnostic Sci
Cornell University
zl73@cornell.edu

Stephen P. Ellner
Ecology & Evol Bio
Cornell University
spe2@cornell.edu

Carla Gomes
Computer Science
Cornell University
gomes@cs.cornell.edu

Yrjo Grohn
Pop Med & Diagnostic Sci
Cornell University
yrjo.grohn@cornell.edu

Abstract

Complex adaptive systems (CAS) are composed of interacting agents, exhibit nonlinear properties such as positive and negative feedback, and tend to produce emergent behavior that cannot be wholly explained by deconstructing the system into its constituent parts. Both system dynamics (equation-based) approaches and agent-based approaches have been used to model such systems, and each has its benefits and drawbacks. In this paper, we introduce a class of agent-based models with an embedded system dynamics model, and detail the semantics of a simulation framework for these models. This model definition, along with the simulation framework, combines agent-based and system dynamics approaches in a way that retains the strengths of both paradigms. We show the applicability of our model by instantiating it for two example complex adaptive systems in the field of Computational Sustainability, drawn from ecology and epidemiology. We then present a more detailed application in epidemiology, in which we compare a previously unstudied intervention strategy to established ones. Our experimental results, unattainable using previous methods, yield insight into the effectiveness of these intervention strategies.

1 Introduction

Complex adaptive systems are pervasive: from energy grids to supply chain networks, ecosystems, social diffusion and disease dynamics, we are surrounded by and personally engaged in complex adaptive systems of varying scales every day [Miller and Page, 2007]. *Complex adaptive systems (CAS)* are composed of interacting agents, exhibit nonlinear properties such as positive and negative feedback and tend to produce emergent system-level behavior. The field of *Computational Sustainability*, seeks to develop computational models and methods to study complex natural and engineered systems in the hopes of guiding them toward long-term, sustainable outcomes [Gomes, 2009].

Historically, the field of *system dynamics* has used ordinary differential equations to describe, analyze and qualitatively predict the macro-level behaviors of CAS [Sterman, 2000], but as computational costs have decreased *agent-based modeling* has gained popularity as an alternative experimental technique in CAS research [Miller and Page, 2007]. When studying CAS in which the quantities of interest are discrete (such as people or businesses), then the values of state variables in a system dynamics model of the CAS directly correspond to subpopulations of agents in an agent-based model of the same CAS.

The top-down approach of system dynamics allows for easy model construction and validation, while the bottom-up approach of agent-based modeling allows for sophisticated interactions between agents with heterogenous state space [Schieritz and Milling, 2003; Osgood, 2007]. On the other hand, both methods have significant drawbacks: system dynamics makes strong assumptions about the homogeneity of agents and complexity of agent interactions in order to achieve mathematical rigor, whereas agent based models are difficult to construct, parameterize, and validate [Rahmandad and Sterman, 2008]. A class of agent-based models with embedded system dynamics models can overcome these difficulties and provide new insights into complex adaptive systems.

Embedded (hybrid) models provide several advantages over pure system dynamics or agent-based models. A complete agent-based model need not be fitted, but individual-level granularity in the model is maintained and heterogeneity in agents can be exploited. This heterogeneity allows for the simulation of novel, complex intervention strategies at the level of agents that might otherwise be difficult or impossible to express succinctly in system dynamics terminology. Few studies, however, present hybrid models for complex phenomena [Martinez-Moyano *et al.*, 2007]. Although the software AnyLogic [Borshchev and Filippov, 2004] claims to have the capability of constructing such hybrid models, no work has resolved the issues behind the construction of embedded models by producing a general definition for such models or giving a formal semantics for their simulation.

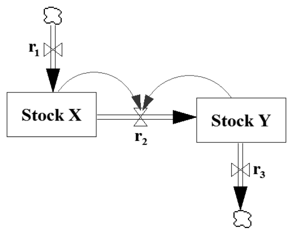


Figure 1: An example stock and flow diagram.

Herein we introduce a class of agent-based models with an embedded system dynamics model, and detail the semantics of a simulation framework for these models. The model definition and simulation framework integrate agent based and system dynamics approaches in a way that retains the strengths of both paradigms. Our embedded model definition streamlines model construction, eases validation, and provides new agent-level information that can be used to discover novel management and intervention strategies based on more realistic motivations and goals.

We demonstrate the benefits of our approach by first instantiating it for two example complex adaptive systems in the field of Computational Sustainability, drawn from ecology and epidemiology. We then present a more detailed application in the field of epidemiology, and give novel experimental results that compare the epidemiological and economic impacts of two disease control intervention strategies, one previously studied and one proposed by our research. Finally, we conclude by outlining future work that seeks to solve optimization problems defined on embedded models.

2 Background

System dynamics views a system as a set of *stocks* (variables) changing through *flows* (derivatives). When stocks are discrete quantities, they represent homogenous groups of well-mixed agents, whereas flows represent the movement of agents between these groups [Sterman, 2000].

$$\frac{dX}{dt} = (F_{gen} + F_{in}) - (F_{des} + F_{out})$$

System dynamics uses *systems of ordinary differential equations* (ODEs) to mathematically express models of stocks and flows [Brauer and Castillo-Chavez, 2001]. For a stock X, let F_{gen} stand for *generative flow* that increases the stock X with agents from outside the system, F_{des} for *destructive flow* that decreases X and removes agents from the system, F_{in} for the flow of agents from other stocks to X, and F_{out} for the flow of agents from X to other stocks. We note that the terms F_{gen}, F_{des}, F_{in}, and F_{out} may be nonlinear functions of other stocks and parameters that may change with time.

Figure 1 shows a simple *stock and flow diagram* of a system that contains two stocks X and Y. A generative flow introduces new agents to the system and stock X at a rate r_1. Agents flow out of stock X and into stock Y at a rate r_2. Finally, a destructive flow removes agents from stock Y and the system, occurring at a rate r_3. The diagram also indicates *feedback loops* within the system, in which r_2 depends on the sizes of stocks X and Y. The identification and modeling of such feedback loops is a critical part of system dynamics.

Although closed-form analytic solutions cannot be obtained for most systems of ODEs, system dynamics has many advantages: models can be constructed and validated relatively quickly using available data and simulation methods are computationally efficient. However, the expressive power of system dynamics is limited by its underlying assumptions of a homogenous and well-mixed population [Brauer and Castillo-Chavez, 2001], meaning that these models capture only the average behavior of a system.

One may improve the granularity of ODE models by increasing the number of variables (stocks) in order to represent a finer division of the state space. For instance, rather than assume all agents in stock X are identical in every respect, X may be divided into smaller stocks denoted by $X_1, \ldots, X_n$ based on another characteristic (e.g. age, see for instance [Medlock and Galvani, 2009]). Each of these stocks could be further divided into smaller stocks based on a secondary characteristic, and this process could conceivably lead to an explosion in the number of equations and parameters that would need to be fitted. It is difficult to know when to truncate this division of stocks in order to obtain the desired complexity and expressiveness of the model. Additionally, the amount of data needed to parameterize the model also increases with model complexity.

When modeling CAS, it is sometimes important to consider *interventions* (actions by outsiders), which can influence flows in the ODE model in predictable or unpredictable ways. Interventions may be included in an ODE model as a parameter that affects one or more flows in the model, or included as a new term in one or more equation. Such parameters may be difficult to compute for innovative strategies for which no data exist. Additionally, the homogenous nature of transition events provides no information on how to select agents for targeted interventions, nor the consequences of such interventions to the agent.

In contrast to system dynamics modeling, *agent-based modeling* views a system as a population of heterogenous agents with a state space that evolves through local interactions. From these interactions, complex emergent behavior often arises. The ability of heterogeneous agents to maintain a large state space also yields greater and more intuitive information that can be used by researchers and policymakers. Agent-based modeling, however, has several drawbacks. First, maintaining a larger state space decreases computational efficiency. Model construction is also more difficult, as it is hard to link observed behavior to local interactions and to capture all critical feedback loops. Finally, since agent-based models are composed of many interactions, model calibration, validation, and sensitivity analysis require large amounts of data and time [Osgood, 2007].

3 Definition of Embedded Model

Here we identify and define a class of agent-based models with embedded system dynamics models. We then provide an algorithmic description of our simulation framework for these models. The semantics of this framework solve the problem of embedding system dynamics within agent-based models while retaining the advantages of both paradigms and avoiding their pitfalls.

We use Gillespie's τ*-leap algorithm* for stochastically simulating the embedded system dynamics model. The τ-leap algorithm interprets the rates of flow from the ODE model as probabilities per unit time for *transition events* between system states (i.e. the movement of agents between stocks or into and out of the system). For Δt small enough, each transition event $E_i, i = 1, \ldots, n$ occurs at rate r_i (as given by the ODE system) and is Poisson distributed with parameter $r_i \Delta t$ [Keeling and Rohani, 2008].

Algorithm 1 τ-leap Method

```
while Time < MaxTime do
  for all event types i do
    ΔE_i ← Poisson(r_i Δt)
  end for
  Update size of each stock based on which transition events occur.
  Randomly select ΔE_i agents uniformly from the appropriate stock and transition according to event E_i.
  Time ← Time + Δt
end while
```

Our class of embedded models divides an agent's state variables into two sets: *local state variables* that hold general agent state and *ODE state variables*, one for each embedded model, that take values corresponding to the stocks of the ODE model. We use λ for the set of local state variables for an agent, θ forthe set of ODE state variables for an agent, and Λ and Θ for the updatable maps that store the values of state variables for agents. Let Name denote a datatype of identifiers (used to name variables, stocks, and agents). We reserve the names Gen and Des for the source of generative flow and the destination of destructive flow.

We define our class of embedded models as a tuple $M = (S, A, O, U, D, V)$. S is a set of sets of local state variables λ, one set of variables per agent ; A is a set of sets of ODE state variables θ, again one set of variables per agent; O is a set of tuples of the form $(\mathsf{Name}, \mathsf{Name}, \mathbb{R})$, specifying rates of flow in the ODE system (either between stocks, from Gen to a stock, or from a stock to Des); U is a set of *local state update functions* that use agent actions and interactions to update local state variables λ and suggest the generation or destruction of agents; D is a set of *demographic functions* that resolve suggestions on agent generation and destruction given by other model components; and V is a set of *intervention functions* that can model high-level actors by updating both local state variables and ODE state variables. A complete description of notation used is shown in [Table 1].

Algorithm 2 shows the simulation of an embedded model for one time step, given an initial set of agents and their state space. The algorithm enforces a set of data access rules; a function can read a variable if it receives it as an argument and can write by returning an updated copy. Functions in U can read both local state and ODE state variables in Λ and Θ, but can only update local state. We use $\mathbf{ODESimulation}$ to denote an algorithm for simulating a system of ODEs (such as the τ- leap method given in Algorithm 1). Such algorithms may only read and write to the ODE state map Θ. Demography functions in D can modify the set P of agents in the system. Finally, intervention functions, in accordance with their role in modeling high-level actors that influence the system, can read and update the entire state space, as well as the set P of agents in the system.

Algorithm 2 Execution of one time step for our simulation framework

```
Input: Embedded model M = (S, A, O, U, D, V), set of agent names P, local state map Λ and ODE state map Θ.
  P_gen ← P_des ← {}
  for all local state update functions u ∈ U do
    (P_gen, P_des, Λ) ← u(P, P_gen, P_des, Λ, Θ)
  end for
  (P_gen, P_des, Θ) ← ODESimulation(O, P_gen, P_des, Θ)
  for all demography functions d ∈ D do
    P ← d(P, P_gen, P_des, Λ, Θ)
  end for
  for all intervention functions i ∈ V do
    (P, Λ, Θ) ← i(P, Λ, Θ)
  end for
```

Algorithm 2 also shows how predictions on agent generation and destruction from different model components are resolved. The sets P_{gen} and P_{des} temporarily hold agents predicted by functions in U or the ODE simulation for generation or destruction, respectively. Demography functions D use these predictions to update the population of agents in P.

Finally, the ordering of Algorithm 2 specifies that intervention functions update Λ, Θ, and P last, overriding changes made by other model components. This facilitates the role of intervention functions in modeling high-level actors like policy-makers or researchers who can influence the system.

3.1 Examples of Embedded Models

We present two complex adaptive systems from Computational Sustainability and give the embedded models for these systems in order to demonstrate the flexibility of our model definition, although we do not conduct any further analysis of these systems. Our first example comes from ecology, where *logistic equations* are used to model the population dynamics of species [Brauer and Castillo-Chavez, 2001]. Here we are interested in a species that occupies a number of habitat patches. Assume that for each habitat i there is a known *carrying capacity* K_i that gives the maximum population size that habitat i can support. Let r_{gen} be the rate at which the species reproduces, and r_{ij} be the rate of migration between habitats i and j. The change in the stock of species members at habitat i is modeled as:

$$\frac{dN_i}{dt} = \left(\sum_{j \neq i} r_{ji} N_j + r_{gen} N_i \right) \left(1 - \frac{N_i}{K_i} \right) - \sum_{j \neq i} r_{ij} N_i$$

Species distribution models are often used to inform conservation efforts. One intervention commonly undertaken is *translocation*, in which species members are transported into

Symbol	Description
S	Set of sets of local state variables λ
A	Set of sets of ODE state variables θ
O	Set of tuples of the form $(\mathsf{Name}, \mathsf{Name}, \mathbb{R})$ specifying rates of flow in the ODE system (either between stocks from Gen to a stock, or from a stock to Des)
U	Set of *local state update functions* that use agent actions and interactions to update local state variables λ and suggest the generation or destruction of agents
D	Set of *demographic functions* that resolve suggestions on agent generation and destruction given by other model components
V	Set of *intervention functions* that model high-level actors by updating both local state variables and ODE state variables
λ	Set of local state variables for an agent
θ	Set of ODE state variables for an agent
Λ	Updatable map, mapping agents to local state variables
Θ	Updatable map, mapping agents to ODE state variables

Table 1: Variables in the embedded model definition.

a habitat with the hopes that they will survive, reproduce, and bolster the population there. Species members could be transported from another habitat in the system, or introduced from outside the system. System dynamics models translocation with new flows occurring at specified rates. Conservationists, however, may wish to model selection policies for translocation targets based on characteristics about the current habitat population. On the other hand, a completely agent-based model of species reproduction and migration would be difficult to construct and verify.

An embedded model (S, A, O, U, D, V) can solve these problems. Agents represent species members, and the set of local state variables S holds data such as age, weight, or spatial location within the habitat. A contains one ODE state variable taking values corresponding to the i habitat spaces from the ODE model, indicating which habitat a species member occupies. O specifies the logistic equations introduced above. The functions in U would model the biological behavior of species members to evolve local state variables. We assume that only the logistic equations provide suggestions on agent generation and destruction, and so D contains one function which accepts these suggestions. Finally, V contains functions to model translocation. Such functions would use the current state space of agents to obtain a view of the species population in each habitat. This view could then be used to target translocation decisions.

The next example comes from epidemiology, the study of the spread of an infectious disease through a population [Keeling and Rohani, 2008]. Epidemics are well-studied by both the agent-based and system dynamics communities [Brauer and Castillo-Chavez, 2001; Barrett *et al.*, 2005] and representative of other types of diffusion processes, such as innovation adoption, financial panics, and the spread of rumors [Rahmandad and Sterman, 2008].

The SIR model, one of the simplest used to model epidemics, models the changes in three stocks: organisms susceptible to, infected by, and recovered from the disease in question. Let β be the rate at which infected individuals transmit the disease to susceptible individuals and γ be the rate at which infected individuals recover from the disease. Then, the SIR model is defined by the equations:

$$\begin{aligned} \frac{dS}{dt} &= -\beta IS \\ \frac{dI}{dt} &= \beta SI - \gamma I \\ \frac{dR}{dt} &= \gamma I \end{aligned}$$

Consider a disease for which the SIR model is applicable and vaccination is available as an intervention strategy for disease control. This control strategy could be modeled using systems dynamics by introducing a rate of vaccination and a new flow at this rate from the susceptible to the recovered stock. In practice, however, policy-makers target vaccination programs toward specific groups based on characteristics that make the group vulnerable to the disease and require forecasts on how such targeting policies affect disease impacts. We show here how an embedded model for this system can solve these problems.

In this embedded model, S includes variables to track socioeconomic factors like age, occupation, income, and health. The set A contains one ODE state variable taking the values susceptible, infected, or recovered corresponding to an agent's state in the ODE model. O specifies the ODE model. The update functions in U can model an agent's socioeconomic evolution, as well as the generation or destruction of

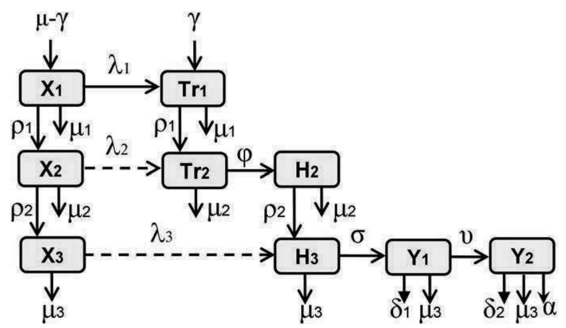


Figure 2: Stock and flow diagram for *Mycobacterium avium* subsp. *paratuberculosis* (MAP) in a typical dairy herd . X_1 and Tr_1 are susceptible and transiently shedding calves, resp.; X_2, Tr_2 and H_2 are susceptible, transiently shedding and latent heifers, resp.; X_3, H_3, Y_1 and Y_2 are suscepible, latent, low-shedding and high-shedding adult cows, resp. See text for detailed description.

agents through births and deaths (all of which can be influenced by the ODE state of agents). Since the embedded SIR model makes no suggestions regarding the generation or destruction of agents, the set D of demography functions can consist of a simple function that accepts the changes proposed by the functions in U. Functions in V can then model vaccination. Given the current state space of agents, machine-learning could be used to produce a function that predicts which agents seek vaccination. The effects of campaigns targeted toward at-risk groups can also be modeled. Policy-makers can test new vaccination strategies and forecast their impacts on disease control by changing the functions in V.

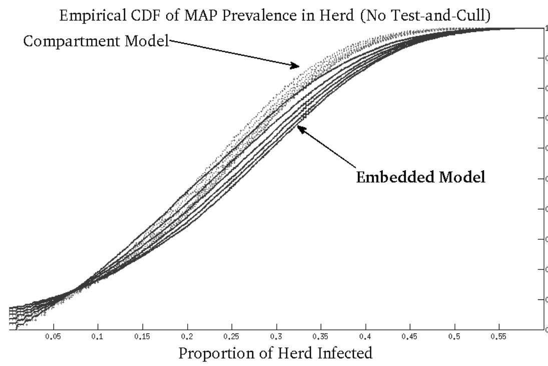
4 Case Study: Dairy Herds

We conduct a detailed study of our model class definition and simulation framework using an advanced application from epidemiology. We focus on *Mycobacterium avium* subsp. *paratuberculosis* (MAP) in dairy herds, a disease that is difficult to detect and control because of its slow progression. MAP adversely affects the milk production of an infected cow, and is estimated to cost $200 million to dairy herds in the United States. MAP may also be connected to Crohn's disease in humans [Ott, 1997]. The primary disease control strategy for MAP is a *test-and-cull* strategy implemented by the dairy farmer, in which the farmer periodically tests the herd for MAP and uses these results to decide which cows to remove, or *cull*, from the herd. The dynamics of MAP on a dairy farm can be considered a complex adaptive system. The farmer is able to make culling decisions based on the observed state of the system, and the system reacts accordingly, creating a feedback loop which (the farmer hopes) will drive the system to a disease-free equilibrium.

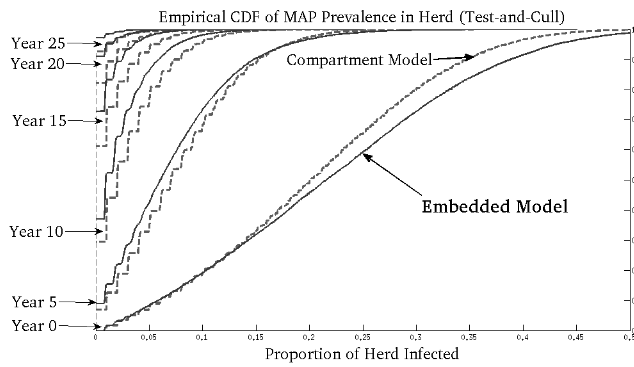
A fully parameterized and validated ODE model of MAP on a dairy farm is shown in Figure 3 [Lu *et al.*, 2010]. It is a complex system of 9 ODEs and 24 parameters, and divides the dairy herd into three age groups: calves (<1 year), heifers (1-2 years) and adult cows. The disease progresses through the following states: *Susceptible, Transiently Shedding, Latent, Low Shedding* and *High Shedding*. Calves are divided into susceptible (X_1) and transiently shedding (Tr_1) stocks; heifers are divided into resistant (X_2), transiently shedding (Tr_2) and latent (H_2) stocks; and cows are divided into resistant (X_3), latent (H_3), low shedding (Y_1) and high shedding (Y_2) stocks. Due to the long incubation period of the disease and the comparatively short lifespan of dairy cows, not all age groups are assumed to reach all disease states. For instance, if a calf has not become infected within its first year, it is assumed to be resistant to the disease for the remainder of its lifetime, since infection as a heifer or adult cow does not produce clinical signs of infection before the cow is removed from the herd. In Figure 3 this is indicated by the dashed flow lines λ_2 and λ_3; these parameters in the validated model are set to zero. The force of infection (λ_1) to susceptible calves is parameterized as $\lambda_1 = \beta_{Tr}(Tr_1 + Tr_2) + \beta_{Y_1}Y_1 + \beta_{Y_2}Y_2$, indicating that susceptible calves (X_1) are infected through one of three transmission routes: calf-calf transmission (β_{Tr}), or infection by either low (β_{Y_1}) or high (β_{Y_2}) shedding adult cows. The herd birth rate (μ) takes into account that some calves might be born with the disease (γ) as a result of in utero infection, and (μ_1, μ_2, μ_3) represent age-specific general mortality rates. Additional test-based culling of low shedding (δ_1) and high shedding (δ_2) adult animals is included, as well as an extra culling rate (α) for removal of high shedders (Y_2) due to clinical signs of MAP infection. For a more detailed description of this model, as well as a complete list of parameter values, please see [Lu *et al.*, 2010].

This model shows good fit to longitudinal data gathered as part of MAP disease control programs [Smith *et al.*, 2009]. However, the ODE model lacks the ability to identify or track individual cows within the system. Ultimately, milk yield is the measure of interest to the dairy farmer and the one used to make most culling decisions, and in order to compute this value for individual cows it is necessary to know their life histories (e.g. number of pregancies, MAP disease state, etc). If a modeler wished to analyze the effect that culling low-yield milk cows had on MAP prevalence within a farm, or the economic impact of test-and-cull strategies, the ODE model presented above would be inadequate and agent-based techniques would be more appropriate. On the other hand, the longitudinal data used to fit and validate the ODE model is not so detailed as to fully parameterize a complete agent-based model of MAP transmission on a dairy farm. Therefore it is in the modeler's best interest to utilize the aggregated information provided by the ODE model by embedding this model within an agent-based model of MAP transmission on a dairy farm.

We construct an embedded model M. In this model, the set S of local state variables λ track biological information about dairy cow agents, including reproductive history and levels of milk production. The set of ODE state variables A contains one variable θ that holds an agent's current MAP disease state. The local state update functions in U are responsible for maintaining the biological state held by the variables in S. For example, our model includes a local state update function that uses an animal's reproductive history and cur-



(a) Comparison of MAP prevalence at endemic equilibrium between ODE and embedded models with no intervention.



(b) Comparison of MAP prevalence at endemic equilibrium between ODE and embedded models with a culling strategy.

Figure 3: Results for experiments validating the Embedded model for MAP against the ODE based Coupled Model.

rent MAP disease state to predict the animal's current milk production level. U also includes functions that model the birth of new livestock. The demography functions in D ignore suggestions from O and accept all other newly generated agents into the system. Population size is bounded by culling animals with low production from the herd (a practice analogous to that carried out by dairy farmers in reality). The set of intervention functions V are aimed at controlling MAP. Intervention functions use an agent's local and ODE state to simulate MAP testing and culling decisions.

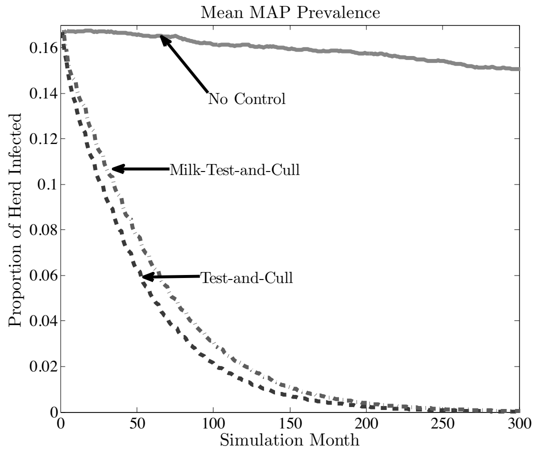
Our embedded model provides significant advantages over the existing ODE model. Our model includes a local state update function, extrapolated from field data and responsive to MAP infection, that predicts an animal's milk production [Smith *et al.*, 2009]. This allows our model to include functions that simulate the economically motivated actions of farmers and to provide metrics on how MAP infection and disease control impact the economic output of a farm. To our knowledge, ours is the first model to provide forecasts of herd-level economics under MAP infection.

We implemented our simulation framework in Java and used it to run three sets of 100,000 simulations on a herd of no more than 150 agents. Simulations differ in the intervention strategies they employ. The *no test-and-cull* strategy employs no control, while the *test-and-cull* strategy simulates the semi-annual testing of cows for MAP, removing any test positive individuals from the herd as replacements become available. Last, we propose the *milk-test-and-cull* strategy which simulates semi-annual testing of MAP, but delays the removal of test positive cows who have high milk yield.

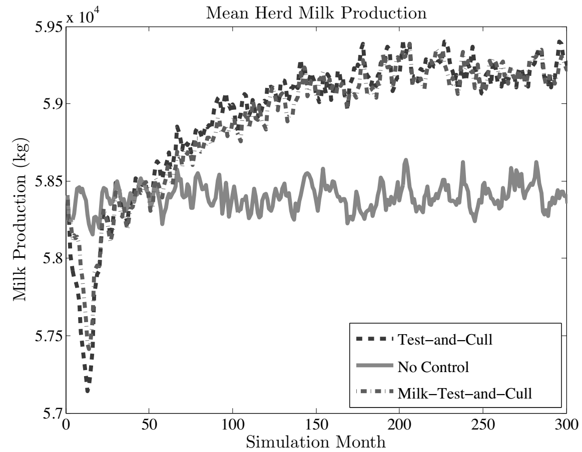
When a disease persists in a population over time rather than naturally fading out, it has reached an *endemic equilibrium*. In our simulated herd we introduced an infected animal and simulated a run-up period of 50 years until endemic equilibrium was reached. We then simulated a 25 year period employing disease control strategies to study their effects.

We first verify that the embedded model produces accurate disease transmission dynamics by comparison to the already validated ODE model. Each line in Figure 3(a) gives the CDF of prevalence across 100,000 simulation runs every five years after the run-up period. This figure visually confirms that the two models produce similar distributions of results for the prevalence of MAP in an uncontrolled setting over time. We confirm this analytically by noting that the KL divergence between the two distributions at the time horizon is 0.0742.

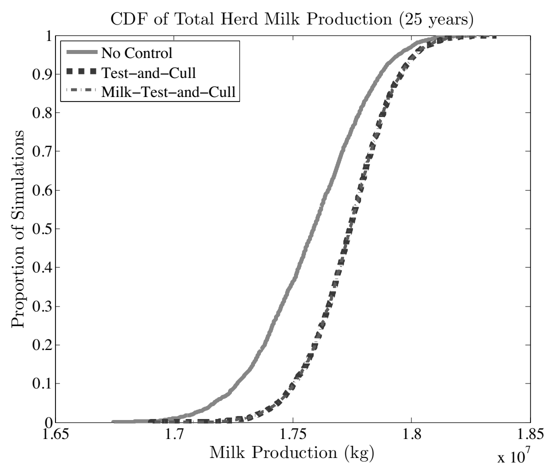
We now turn to using the embedded model to measure the impacts of the three interventions. Figure 4(a) gives the mean prevalence of MAP infection in the simulated herd over time. We first note that employing no control results in the



(a) Comparison of mean MAP prevalence over time between three control strategies.



(b) Comparison of mean herd milk production over time between three control strategies.



(c) CDFs of total herd milk production over 25 years for three control strategies.

Figure 4: Results on epidemiological and economic impacts of MAP obtained with an embedded model.

persistence of the endemic equilibrium, as expected. Next, we note that the test-and-cull and milk-test-and-cull strategies both result in the disappearance of MAP from the herd within 25 years, although fadeout under milk-test-and-cull is slightly slower. Since both strategies result in the eradication of MAP in similar time-frames, our results suggest that the more economically-minded strategy can be chosen by farmers for disease control in closed dairy herds (i.e. no movement of cows between farms).

Figure 4(b) reveals the economic impacts of MAP under the three control strategies by showing the mean total milk production of the simulated herd over time. It is clear that controlling MAP infection in the herd has an early cost: during the first four years of control, milk production under the two culling strategies is significantly decreased as infected cows that produce more milk than their younger replacements are removed. The milk-test-and-cull strategy suffers less from this effect. It is also clear that in the long term disease control is vital to economic output: milk production is significantly higher over time under the two strategies that control for MAP vs. the uncontrolled strategy.

Figure 4(c) compares total economic output under the three intervention strategies. Each line gives the CDF (across 100,000 simulations) of the total milk produced by the herd under the strategies for the 25 year control period. The distributions of results when controlling for MAP are clearly higher than with no control, demonstrating the economic benefits of interventions. Furthermore, Figure 4(c) indicates that the two culling strategies have similar performance in the long term. Combining this with the results from Figure 4(b), we conclude that the economically informed milk-test-and-cull strategy can be employed to mitigate the short-term costs of MAP control without sacrificing long term productivity and disease eradication.

5 Future Work

Future work will focus on solution techniques for optimization problems defined on our class of models, in order to provide policymakers with optimal intervention strategies for complex adaptive systems. In our MAP application, farmers must use the current state of their herd to make culling decisions that result in the best future outcome. We define a *policy function* $\psi(\Lambda, \Theta)$ that, given the current state of agents, chooses a set of agents for culling. An *optimal policy function* $\psi^*(\Lambda, \Theta)$ makes culling decisions resulting in the best economic productivity across all possible stochastic outcomes. Finding or learning an optimal policy function is computationally complex. The stochastic nature of simulations makes any optimization problem defined on the embedded model computationally difficult. In particular, we will focus on finding optimal *policies* in this stochastic setting.

6 Conclusion

System dynamics and agent-based techniques, each with their benefits and drawbacks, have been used to model many complex adaptive systems. In this work, we defined and presented a simulation framework for a class of embedded models that leverage the advantages of both system dynamics and agent-based modeling. Through examples and an advanced application in epidemiology, we demonstrated the applicability of our class of models, its ability to support novel intervention strategies, and its potential to generate innovative forecasts.

Acknowledgments

The authors gratefully acknowledge the support of the NSF Expeditions in Computing grant on Computational Sustainability (Award Number 0832782).

References

[Barrett *et al.*, 2005] Chris L. Barrett, Stephen G. Eubank, and James P. Smith. If smallpox strikes portland... *Scientific American*, pages 54–61, March 2005.

[Borshchev and Filippov, 2004] Andrei Borshchev and Alexei Filippov. From System Dynamics and Discrete Event to Practical Agent Based Modeling : Reasons , Techniques , Tools. *System Dynamics*, 2004.

[Brauer and Castillo-Chavez, 2001] Fred Brauer and Carlos Castillo-Chavez. *Mathematical Models in Population Biology and Epidemiology*. Springer, 2001.

[Gomes, 2009] Carla P. Gomes. Computational Sustainability: Computational methods for a sustainable environment, economy, and society. *The Bridge, National Academy of Engineering*, 39(4), 2009.

[Keeling and Rohani, 2008] Matthew James Keeling and Pejman Rohani. *Modeling infectious diseases in humans and animals*. Princeton University Press, 2008.

[Kocsis and Szepesvri, 2006] Levente Kocsis and Csaba Szepesvri. Bandit based monte-carlo planning. In *Machine Learning: ECML 2006*, volume 4212 of *Lecture Notes in Computer Science*, pages 282–293. Springer Berlin / Heidelberg, 2006.

[Lu *et al.*, 2010] Zhao Lu, Ynte H. Schukken, Rebecca L. Smith, and Yrjo T. Grohn. Stochastic simulations of a multi-group compartmental model for johne's disease on us dairy herds with test-based culling intervention. *Journal of Theoretical Biology*, 2010.

[Martinez-Moyano *et al.*, 2007] Ignacio Martinez-Moyano, David Sallach, and Mark Bragen. Design for a Multilayer Model of Financial Stability : Exploring the Integration of System Dynamics and Agent-based Models. In *System Dynamics*, pages 1–16, 2007.

[Medlock and Galvani, 2009] Jan Medlock and Alison P. Galvani. Optimizing influenza vaccine distribution. *Science*, 325:1705–1708, 2009.

[Miller and Page, 2007] John H. Miller and Scott E. Page. *Complex Adaptive Systems: An Introduction to Computational Models of Social Life*. Princeton University Press, 2007.

[Osgood, 2007] Nathaniel Osgood. Using Traditional and Agent Based Toolsets for System Dynamics : Present Tradeoffs and Future Evolution. *System Dynamics*, 2007.

[Ott, 1997] Stephen Ott. Johne's disease on u.s. dairy operations. Miscellaneous Publications 32763, United States Department of Agriculture, National Animal Health Monitoring System, 1997.

[Rahmandad and Sterman, 2008] H. Rahmandad and J. Sterman. Heterogeneity and Network Structure in the Dynamics of Diffusion: Comparing Agent-Based and Differential Equation Models. *Management Science*, 54(5):998–1014, May 2008.

[Schieritz and Milling, 2003] Nadine Schieritz and Peter Milling. Modeling the Forest or Modeling the Trees A Comparison of System Dynamics and Agent-Based Simulation. In *System Dynamics*, number Phelan 1999, 2003.

[Smith *et al.*, 2009] R.L. Smith, Y.T. Grohn, A.K. Pradhan, R.H. Whitlock, J.S. Van Kessel, J.M. Smith, D.R. Wolfgang, and Y.H. Schukken. A longitudinal study on the impact of johne's disease status on milk production in individual cows. *Journal of Dairy Science*, 92:2653 – 2661, 2009.

[Sterman, 2000] John D. Sterman. *Business Dynamics: Systems Thinking and Modeling for a Complex World*. McGraw-Hill, 2000.