



Web Appendix for “Network Meta-Metrics: Using Evolutionary Computation to Identify Effective Indicators of Epidemiological Vulnerability in a Livestock Production System Model”, by Wiltshire S. et al., published on 31 March 2019 in JASSS-Journal of Artificial Societies and Social Simulation, 22(2), 8: <http://jasss.soc.surrey.ac.uk/20/4/11.html> [doi: DOI: 10.18564/jasss.3991]

ODD+D Protocol: Regional U.S. Hog Production Network Biosecurity Model v.1.2

This material is based upon work that is supported by the National Institute of Food and Agriculture, U.S. Department of Agriculture, under award number 2015-69004-23273. Any opinions, findings, conclusions, or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the view of the U.S. Department of Agriculture. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

The following model description follows the ODD (Overview, Design concepts, Details) protocol for describing individual- and agent-based models (Grimm et al. 2006, 2010), with ODD+D amendments as proposed by Müller et al. (2013).

I) Overview

I.i Purpose

I.i.a What is the purpose of the study?

The Regional U.S. Hog Production Network Biosecurity Model (RUSHPNBM), version 1.2, is an agent-based susceptible / infective model developed to assess both supply chain network level and human-behavioral factors relevant to the spread of socioeconomically-important diseases through regional U.S. hog production chain networks. RUSHPNBM has been developed using AnyLogic v.8 software. The model is calibrated to represent hog production within the U.S. states of North Carolina, Iowa, and Illinois. These three states serve as case studies, since they are all major hog producers, while also having interesting supply chain network features that differentiate them from one another.

Model calibration was undertaken using available datasets coupled with an iterative expert advisory panel process. The model uses agricultural statistics and model calibration tools to generate realistic production chain networks of producers, feed mills, and slaughter plants within the spatial bounds of each study area state.

The epidemiological spread submodel is of the susceptible / infective (SI) type, with infections transmitted between agents probabilistically based on patterns of trade and contact honed through industry expert advisory panels and a review of the primary literature. Disease spread probabilities associated with the different types of inter-agent contact have been calibrated by reference to epidemiological data concerning disease spread dynamics

associated with previous real-world epidemic events in the hog industry, as well as input from livestock veterinary professionals within the expert panel sessions.

I.i.b For whom is the model designed?

The model was designed for use by university researchers, industry practitioners, veterinary specialists, and government agencies wishing to analyze the dynamics and consequences of disease spread in U.S. hog production systems under varying assumptions concerning disease characteristics, production chain network structures, and implementation of biosecurity measures and agent behaviors that may prevent or curb catastrophic outbreaks.

I.ii Entities, State Variables, and Scales

I.ii.a What kinds of entities are in the model?

Three classifications of hog production chain network agents, identified by industry experts as critical players in the transmission of disease, are represented in the model. These are (a) producers, (b) feed mills, and (c) slaughter plants. Producer agents are assigned one of six industry roles. Five of these encompass the USDA’s classification system for hog producers, these being (a) Farrow to Wean, (b) Wean to Feeder (a.k.a. Nursery), (c) Feeder to Finish (a.k.a. Finish Only), (d) Farrow to Feeder, and (e) Farrow to Finish. Upon the advice of industry experts, an additional producer classification, (f) Wean to Finish—which has recently become more popular in the industry—is also included in the model. Figure 1 below shows each agent type, its graphical representation, and an outline of the heuristics that govern inter-agent contact patterns.

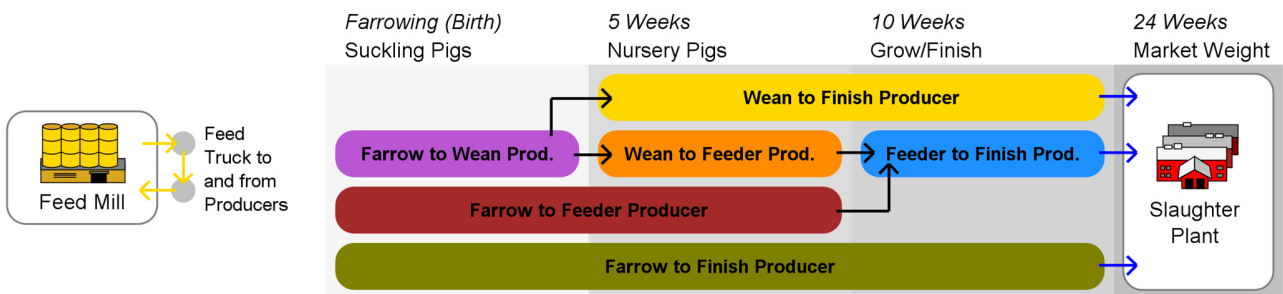


Figure 1: Structure of connections between agents, including livestock age transfer conditions where applicable

I.ii.b By what attributes (i.e. state variables and parameters) are these entities characterized?

Each agent class has a specific set of state variables and parameters relevant to its industry role. These are given in Tables 1–5, below.

Table 1: Parameters and variables common to all agents

Attribute	Description
<i>Static parameters (set at initialization)</i>	
My name	String representation encoding agent class and index (for tracking network connections)
<i>State variables (may change throughout simulation run)</i>	
Infectivity state	Either “clean” or “infected”.
Has been infected	Flag indicating whether the agent was ever infected during a run.
Total infection duration	Counter indicating the total number of days an agent was infected during a run.
Contact network out degree & in degree	List of the other agents with whom each agent had contact throughout a run, as well as the number of times contact between the two agents occurred. Out degree is incremented whenever an agent sends animals or feed; in-degree whenever animals or feed are received.
Infection network out degree	List of the other agents to whom an agent spread the disease, along with the number of times infection spread occurred between the two agents throughout a run.

Table 2: Parameters and variables for producer agents

Attribute	Description
<i>Static parameters (set at initialization)</i>	
Farm category	Encodes which of the 6 producer industry roles this agent falls into.
Total capacity	Total animal capacity.
Number of sows	Number of sows (calculated using parameters for the ratio of sows to pigs associated with each producer classification).
Non-sow capacity	Total capacity minus number of sows.
My transferee producers	List of potential producer trading partner agent objects.
My slaughter plant	Link to slaughter plant agent object (finishing producers only).
My feed mill	Link to feed mill agent object.
<i>State variables (may change throughout simulation run)</i>	
Current piglet inventory	Number of piglets currently available to wean and batch.
Last weaning day	Timecode of the last date piglets were batched and encoded in the pig batch tracker.
Current non-sow pig inventory	Number of pigs (that is, neither piglets or sows) currently on premises.
Pig batch tracker	Non-sow pigs are represented by a data structure encoding the size of each batch of stock currently making up the agent’s hoofstock inventory, along with a timestamp representing that pig batch’s “birthday”, which is used to calculate the age of the batch.
Pig shipments in & out	Lists of sizes (number of animals) of each shipment incoming from and outgoing to the producer (used for calibration).
Feed deliveries in	Number of feed deliveries incoming to the producer (used for calibration).

Table 3: Parameters and variables for feed mill agents

Attribute	Description
<i>State variables (may change throughout simulation run)</i>	
Truck infected	Flag indicating whether the feed mill's delivery truck is currently infected.
Feed deliveries out	Number of outgoing feed deliveries to producers (used for calibration).

Table 4: Parameters and variables for slaughter plant agents

Attribute	Description
<i>State variables (may change throughout simulation run)</i>	
Pig shipments in	List of sizes (number of animals) of each shipment incoming from producers (used for calibration).

I.ii.c What are the exogenous factors/drivers of the model?

Exogenous factors include the spatial and operational distributions of agents of each class within each study area (Table 5), as well as parameters held constant across model runs (Table 6). These exogenous parameters can be broken down into general disease parameters, disease spread probabilities specific to each modality of inter-agent contact, and parameters pertaining to each agent class.

Table 5: Study area network parameters

Parameter	Baseline Value			Data Source(s)
	<i>North Carolina</i>	<i>Iowa</i>	<i>Illinois</i>	
Num. producers	2217	6266	2045	Burdett et al. (2015)
Avg. producer capacity	4015	3265	2264	Burdett et al. (2015)
Proportion farrow to wean	0.050	0.026	0.038	Burdett et al. (2015)
Proportion farrow to feeder	0.005	0.010	0.009	Burdett et al. (2015)
Proportion farrow to finish	0.554	0.304	0.635	Burdett et al. (2015)
Proportion wean to feeder	0.102	0.064	0.023	Burdett et al. (2015)
Proportion wean to finish	0.003	0.077	0.055	Burdett et al. (2015)
Proportion feeder to finish	0.286	0.519	0.241	Burdett et al. (2015)
Num. slaughter plants	24	18	25	USDA NASS (2014)
Num. feed mills	40	114	37	Google search; EAP

Table 6: Parameters common to all study areas, remaining fixed throughout each model run

Parameter	Description / Units	Baseline Value	Data Source(s)
<i>Disease parameters</i>			
Percent to infect	Percent of producers initially infected	5%	-
Avg. producer infection length	Avg. duration of producer infection (days)	40	Goede & Morrison (2016); EAP
Avg. slaughter plant infection length	Avg. duration of slaughter plant infection (days)	7	EAP
Avg. feed mill infection length	Avg. duration of feed mill infection (days)	25	EAP
Suckling mortality rate	Proportion of suckling pigs dying if infected	0.9	Goede & Morrison (2016); EAP
Nursery mortality rate	Proportion of nursery pigs dying if infected	0.4	Goede & Morrison (2016); EAP
Grow/finish mortality rate	Proportion of grow/finish hogs dying if infected	0.1	Goede & Morrison (2016); EAP
<i>Producer disease spread probabilities</i>			
Prob. producer will become infected if returning pig truck is contaminated		0.3	EAP
Prob. producer will become infected if delivered feed is contaminated		0.8	EAP
Prob. feed truck will become contaminated if producer is infected		0.05	EAP
Prob. pig truck will become contaminated if producer is infected		0.2	EAP
<i>Feed mill disease spread probabilities</i>			
Prob. feed mill will become infected if returning feed truck is contaminated		0.1	EAP
Prob. feed truck will become contaminated if feed mill is infected		0.5	EAP
<i>Slaughter plant disease spread probabilities</i>			
Prob. slaughter plant receiving area will become infected if pig batch is infected		0.4	EAP
Prob. pig truck will become contaminated if receiving area is infected		0.2	EAP
<i>Producer farrow, wean, and batch parameters</i>			
Farrow to wean sow proportion	Relative to total capacity	.6	EAP
Farrow to feeder sow proportion	Relative to total capacity	.5	EAP
Farrow to finish sow proportion	Relative to total capacity	.2	EAP
Annual piglets per sow	Number of piglets	34	The Pig Site (2014)
Max. frequency of weaning	“farrow, wean and batch” function freq. (days)	7	EAP
Min. batch size	As proportion of total capacity	0.05	EAP
Capacity under which one batch	Small producers have only one pig batch	20	EAP
<i>Producer to producer transfer parameters</i>			
Min. capacity similarity ratio	Trading producers cannot be greatly different sizes	25	EAP
Max. producer connection distance	Max. distance between trading producers (km)	150	EAP
Max. potential transferees	Max. number of producer trading partners	15	FHPC
Max. shipment frequency	“evaluate pig shipments” function freq. (days)	5	FHPC
<i>Feed mill parameters</i>			
Producer to feed mill proximity λ	Producers connect to λ th closest feed mill (Poisson)	1.5	EAP
Avg. daily trips	Avg. num. daily feed deliveries per mill	10	FHPC
Num. producers visited λ	Expected to visit λ per delivery (Poisson)	1	EAP
<i>Slaughter plant parameters</i>			
Producer to slaughter plant proximity λ	Producers connect to λ th closest slaughter plant (Poisson)	2	EAP

Note: “EAP” indicates that the value was derived through expert advisory panel sessions. “FHPC” refers to the family-owned hog production chain system dataset.

I.ii.d If applicable, how is space included in the model?

The model is spatially situated in a continuous, two-dimensional GIS environment. Distances between agents are calculated “as the crow flies” and measured in kilometers. In some cases, distance is a factor in determining inter-agent contact patterns.

I.ii.e What are the temporal and spatial resolutions and extents of the model?

The model’s time scale is based on real-world days, with the initial model date set to January 1st, 2012. 2012 was chosen because FLAPS initialization data are drawn from the 2012 USDA Census of Agriculture (Burdett et al. 2015). The model’s stop date can be set as desired depending on the experimental phenomena the user is interested in studying. Model time is continuous (i.e., events may occur part-way through a day). The model’s spatial extents correspond to the extents of the selected U.S. state study area. Space is also continuous in the model, utilizing a two-dimensional GIS framework.

I.iii Process Overview and Scheduling

I.iii.a What entity does what, and in what order?

Four classes of functions define the operation of the model, presented in order of the point(s) in the simulation that they occur (see the *Implementation Details* section). First are the initialization functions, which define how the agents will be physically situated in the space, set each agent’s individual operational parameters, and identify lists of potential trading partners based on the classification and industry role of the agent, as well as spatial proximity to other agents. Second are the cyclically-executing functions, which make up the agents’ decision rules, determining how and when contact between agents will occur (through the transfer of livestock and the distribution of feed), and thereby opening potentials for infection to spread. These functions also determine and implement the consequences of an infection upon the agent. Third is the initial infection function, which is called after the initial transient period in each run. Finally, fourth are the set of functions facilitating the output of model data for further analysis, including post-experiment scripts to parse model outputs and analyze results across multiple runs. All event scheduling in the model follows a Last-In-First-Out (LIFO) protocol.

II) Design Concepts

II.i Theoretical and Empirical Background

II.i.a Which general concepts, theories or hypotheses are underlying the model’s design at the system level or at the level(s) of the submodel(s) (apart from the decision model)? What is the link to complexity and the purpose of the model?

Because real-world epidemics are fundamentally phenomena which propagate through networks (social, business, transportation, etc.), the formulation of a suitably-realistic network structure within which agents operate is a fundamental basic principle of the model. A corollary to this basic principle concerns the model’s balance between context specificity and analytic transparency. The model’s network generation algorithm strives to maintain sufficient context specificity to capture the critical complexities underpinning observed epidemiological spread phenomena, while bracketing superfluous elements of real-world production chain networks which have not been implicated in previous epidemiological events. For example, the model contains only feed mill, producer, and slaughter plant agent classes, because these were identified by industry experts in our Delphi panels as the critical players underpinning the spread of fecal-oral livestock diseases. Whereas in real-world hog production chain networks there may be a multitude of other actor typologies (e.g. equipment suppliers,

construction contractors, insurance agents, auction houses, and many more), these were intentionally excluded from the model's design to simplify analysis.

Another guiding principle is the geospatial situation of the model within real U.S. states. The states of North Carolina, Iowa, and Illinois were chosen because they produce a large number of the nation's hogs, as well as being amongst the most hog-dense states. In many epidemiological studies, agent density has been shown to impinge directly upon spread characteristics. With high enough density, complex phenomena such as percolation thresholds may emerge (Wiltshire, 2018).

II.i.b On what assumptions is/are the agents' decision model(s) based?

The primary set of assumptions driving agent behavior relate to trade patterns associated with the industry role each agent plays, with agents in the model operating in accordance with general industry norms. For example, it is assumed that, as soon as their livestock batches reach the transfer age appropriate for their industry role, producer agents will search the agent space for trading partners until an appropriate partner is found, at which point the pig batch will be immediately transferred.

Another assumption concerns the spatial locations of non-producer agents. Since fine-grained spatial data were not available—and the FLAPS tool only covers livestock production units (Burdett et al. 2015)—locations of non-producer agents are initialized by distributing them at random positions within each county, in proportion to the number of producers in the county.

Several assumptions also come into play concerning the distance, similarity, and number of other industry actors with which each agent may interact. These assumptions were parameterized using the maximum distance, minimum capacity similarity, and maximum connection number global parameters, as well as the characteristic connection distance “lambda” values that underlie how feed mill and slaughter plant service areas are generated. For example, in the case of feed mills, $\lambda = 1.5$, indicating that most producers purchase feed from the nearest or second-nearest mill, with fewer purchasing from the third-nearest, fewer still from the fourth-nearest, etc. Alternatively, for slaughter plants, $\lambda = 2$, indicating that the most likely outcome is for a producer to ship hogs to the second-closest plant. We note that a limitation of the model is that, since all agents exist within the bounds of a single U.S. state, inter-state trade is not accounted for.

A further assumption is our representation of livestock in batches (or metapopulations) of animals of the same age. This was primarily done to reduce computational overhead (vs. storing each animal's parameters individually). In addition to having the same theoretical birthday, it is assumed that if a batch is infected, all of its members are infected.

Finally, it is assumed that if a producer agent becomes infected, all of its livestock batches become infected. While a simplifying assumption to be sure, this is reasonably realistic, owing to the high observed virulence of the PED virus, which tends to sweep quickly through entire herds.

II.i.c Why is/are certain decision model(s) chosen?

Agents' decision heuristics primarily relate to their day-to-day operations, and rely on parameters including industry role, size, and spatial location. Based on industry standards along with individual parameters, agents make decisions pertaining to when and with whom inter-agent contact will occur. These contact patterns go on to impact the susceptible / infective state variable of each agent, according to probabilities associated with each modality of inter-agent contact

II.i.d If the model/submodel (e.g. the decision model) is based on empirical data, where do the data come from?

We use the Farm Location and Agricultural Production Simulator (FLAPS) tool—which draws upon USDA Census of Agriculture data along with aerial imaging to impute realistic distributions of livestock farms within a specified U.S. region—to set producer agent locations and key operational parameters including industry roles and capacities (Burdett et al. 2015). While the FLAPS tool serves as our primary means to set production unit locations and operational parameters, our team also gained access to internal records from a large family-owned hog production chain system—identified as “FHPC” in Table 6—which was used to impute realistic contact rate and shipment size parameters. Several other sources of empirical data were also used to parameterize the model, also indicated in Table 6.

II.i.e At which level of aggregation were the data available?

The FLAPS system uses several core datasets to impute producer agent locations and operational characteristics. The system primarily relies upon the 2012 USDA Census of Agriculture, which is aggregated at the county level. However, FLAPS also leverages aerial imaging land use data to position agents more precisely within each county. Slaughter plant data from USDA NASS (2014) were aggregated at the level of the U.S. state.

II.ii Individual Decision-Making

II.ii.a What are the subjects and objects of the decision-making? On which level of aggregation is decision-making modelled? Are multiple levels of decision making included?

Decision-making is modeled at the level of the agent, be it a producer, feed mill, or slaughter plant. When transferring livestock, for example, a producer agent will wait until a pig batch matches the appropriate age corresponding to its industry role. This subject will then search among its potential trading partners (objects, in this case), which were pre-selected at model initialization to be of the appropriate industry role, until a suitable agent is found that has sufficient capacity to accept the shipment.

II.ii.b What is the basic rationality behind agent decision-making in the model? Do agents pursue an explicit objective or have other success criteria?

Agents in the model act according to accepted industry operational standards. They do not pursue a specific “objective,” per se, other than to efficiently take in new livestock and ship them out at the appropriate life cycle stage. Future versions of the model will incorporate adaptive agent decision-making, for example allowing for decisions which will reduce an agent’s vulnerability if a disease is present in the network.

II.ii.c How do agents make their decisions?

A series of cyclically-executing functions (outlined in the *Implementation Details* section) govern how and when agents make decisions. These are based on the industry role of each agent.

II.ii.d Do the agents adapt their behavior to changing endogenous and exogenous state variables? And if yes, how?

Producer agents desiring to transfer livestock to the next production phase adapt their behavior based on the operational variables associated with their potential trading partners. This primarily comes down to finding a partner within a certain distance, of the appropriate industry role, of sufficiently-similar size, and with sufficient excess capacity to accept the shipment. The number of pigs housed at each production unit is constantly updated as the model runs, so the agents have to perform this search with each new outgoing shipment. However, agents in the current model do not adapt their decisions based on the prevalence of disease in the system, or other global factors.

II.ii.e Do social norms or cultural values play a role in the decision-making process?

Agents follow basic industry standards when making their decisions, which could be considered a kind of social norm. Values do not play into their choices, however.

II.ii.f Do spatial aspects play a role in the decision process?

Producer agents looking to transfer animals to another producer are constrained by a maximum distance parameter. The service areas of slaughter plant and feed mill agents are governed by Poisson distributions, with producers most likely to connect to the λ th-closest of each.

II.ii.g Do temporal aspects play a role in the decision process?

Producer agents will only transfer livestock to their producer trading partners if the partner has sufficient excess capacity. Since the inventory of each agent is constantly in flux, the time when the transfer function is executed will determine the trading partner that is chosen. Farrowing producers will also wait to batch weaner pigs until the quantity of piglets is greater than or equal to the minimum batch size, as a proportion of their capacity.

II.ii.h To which extent and how is uncertainty included in the agents’ decision rules?

Agent behavior does not account for uncertainty in the current model. However, we are pursuing behavioral research which will be used to parameterize agents in future model versions to react to uncertainty as regards disease prevalence in the networks (Merrill et al. 2008).

II.iii Learning

II.iii.a Is individual learning included in the decision process? How do individuals change their decision rules over time as consequence of their experience?

The agents' decision rules remain non-adaptive in the current model. Decision heuristics are based on industry roles, and are designed to realistically replicate throughput in the production chain system as a whole. Thus, an agent will transfer animals to an appropriate trading partner as soon as possible, farrowing will proceed regularly wherever a producer has sufficient excess capacity, and feed deliveries take place at a set frequency. The agents' behavior does not change as a result of model conditions, for example the presence of a disease within the network, however each agent will necessarily adapt to market conditions resulting from the available spare capacity of its trading partners.

II.iii.b Is collective learning implemented in the model?

No.

II.iv Individual Sensing

II.iv.a What endogenous and exogenous state variables are individuals assumed to sense and consider in their decisions? Is the sensing process erroneous?

Producer agents "sense" the operational variables associated with potential trading partners when making decisions concerning livestock transfers. An agent's perception of these factors is not erroneous, as it is based on a direct query of the potential recipients' operational variables at the time the transfer is desired.

II.iv.b What state variables of which other individuals can an individual perceive? Is the sensing process erroneous?

The primary state variables agents perceive when making livestock transfer decisions are the potential trading partner's industry role, capacity, inventory, and proximity.

II.iv.c What is the spatial scale of sensing?

At model initialization, producer agents generate a list of potential producer trading partners only within a given distance (150 km). It could be said that their "sensing" of other producers does not extend beyond this distance. Note that (based on the Poisson distributions discussed above), producers may very well interact with slaughter plants and feed mills beyond the 150-km limit imposed upon producer to producer transfers.

II.iv.d Are the mechanisms by which agents obtain information modelled explicitly, or are individuals simply assumed to know these variables?

Agents are simply assumed to know the relevant operational variables necessary to make the choice of which trading partner to choose.

II.iv.e Are the costs for cognition and the costs for gathering information explicitly included in the model?

No.

II.v Individual Prediction

II.v.a Which data do the agents use to predict future conditions?

Agents do not engage in predictive behavior concerning the likely outcomes of their actions. They simply operate according to industry standards.

II.v.b What internal models are agents assumed to use to estimate future conditions or consequences of their decisions?

None, although future versions of the model may incorporate the Theory of Planned Behavior with regard to agent decisions to increase biosecurity measures or limit livestock movements in response to a perceived disease threat in the system.

II.v.c Might agents be erroneous in the prediction process, and how is it implemented?

Since prediction is not currently a decision-making factor, no.

II.vi Interaction

II.vi.a Are interactions among agents and entities assumed as direct or indirect?

Interactions among agents take the form of transfers of livestock and feed. These transfers are assumed to be mediated by transportation equipment, and so can be conceived as indirect. For example, if a truck delivers livestock to an infected premises, a probability parameter governs whether the truck will become contaminated. The truck is then assumed to return to its origin, at which point (if it is contaminated), another probability parameter determines whether the truck contamination will result in infection of the original agent. Similarly, a feed truck originating from a mill may become contaminated upon reaching an infected producer premises, and then may pass the infection along to another producer on the same route, or to the mill itself upon its return.

II.vi.b On what do the interactions depend?

Interactions depend upon the operational variables associated with each agent. Producer agents will only initiate a transfer when a pig batch reaches the designated transfer age. Spatial proximity and underlying network structure—encoded at model initialization—also mediate interactions. For example, feed mills periodically generate delivery routes which encompass a subset of producer agents within their service areas.

II.vi.c If the interactions involve communication, how are such communications represented?

The only communication that is represented during livestock transfers is the size and age of pig batches, along with the spare capacity of the potential trading partner. This may be conceived as a phone call or email to potential trading partners inquiring whether they are able to accept the batch; or alternatively as predetermined business arrangements, common in private production system networks.

II.vi.d If a coordination network exists, how does it affect the agent behavior? Is the structure of the network imposed or emergent?

Agents coordinate only in the sense that, during livestock transfers, the receiving agent must concurrently have the spare capacity to accept the shipment. Thus, to the extent that coordination occurs in the model, it is emergent, not imposed.

II.vii Collectives

II.vii.a Do the individuals form or belong to aggregations that affect and are affected by the individuals? Are these aggregations imposed by the modeler or do they emerge during the simulation?

Livestock in the model may be considered as collectives (or metapopulations), as they are encoded in groups of animals with the same theoretical age. This is imposed by the model structure.

In addition, while not defined explicitly as such, groups of agents in the model exhibit emergent collective characteristics due to their differential spatial distribution across the model's GIS space. For example, in densely-packed areas, groups of agents tend to interact heavily within connected clusters, potentially leading to localized disease outbreaks. This type of emergent collective behavior is not directly imposed by the modeler, although the fixed spatial location and network structure that is imposed at model initialization impacts how and where such phenomena arise.

II.vii.b How are collectives represented?

Metapopulations of livestock are tracked as passive objects within a concurrent hash map data structure associated with each agent object. This data structure encodes the size and age of each pig batch.

Emergent “collectives” of heavily-connected agent clusters are captured by tracking agent contact patterns over the course of each run, and outputting these data as weighted edge lists.

II.viii Heterogeneity

II.viii.a Are the agents heterogeneous? If yes, which state variables and/or processes differ between the agents?

As described in the *Entities, state variables, and scales* section above, agents fall into three main classes: (a) producers, (b) slaughter plants, and (c) feed mills. Producer agents are assigned one of six industry roles, based on the USDA classification system for hog producers, along with expert advisement. A producer agent's industry role determines the initial age of its livestock, its livestock age transfer condition, as well as the appropriate trading partners which make up its set of potential trading partners. These relationships are visualized in Figure 1.

II.viii.b Are the agents heterogeneous in their decision-making? If yes, which decision models or decision objects differ between the agents?

Agents' decision-making will differ depending upon their industry roles. For example, a farrow-to-wean producer will only send pigs to wean-to-feeder or wean-to-finish producers; slaughter plant agents will only receive hogs from finishing producers within their service areas; and feed mills will only distribute to producers within their service areas.

II.ix Stochasticity

II.ix.a What processes (including initialization) are modelled by assuming they are random or partly random?

RUSHPNBM uses both fixed-seed stochasticity (for initialization) as well as random-seed stochasticity (to mediate infection spread). All initialization procedures that are not drawn directly from the FLAPS data—for example the list of potential transferees for each producer—utilize a fixed seed in all draws from stochastic functions. This is important because we are interested in analyzing the dynamics associated with disease risk, and changing the basic supply chain network structure across runs would confound results. We also use fixed-seed draws from custom distributions—i.e., those based on the number of observations in a series of categories—to establish the spatial location of slaughter plants and feed mills to correspond with producer density by county.

Poisson distributions are utilized in some cases in model initialization, for example to determine which slaughter plant or feed mill to which each producer will connect. Poisson distributions are used in the model because they require only a λ parameter corresponding to the expected value of the distribution; because they are discrete, returning a whole-number; and because they cannot return a value less than zero. For initialization procedures, the fixed seed is utilized in these Poisson draws, maintaining the same initial “latent” network structure for each study area across runs.

The remainder of the stochasticity in the model uses a random seed, yielding a dataset representing a distribution of contact and infection patterns across model runs. Random-seeded stochasticity is first used to populate each producer's initial pig batches. Both the size and age of the pig groups associated with each producer are drawn from a uniform distribution bounded according to the producer's industry role. Since each producer will always start with a slightly different animal inventory, this ensures that the trade patterns that unfold throughout each model run are not repeated exactly. Random-seeded Poisson distributions are used to determine the number of producers to visit for each feed distribution trip.

Random-seed stochasticity is also used for all disease-spread calculations. Uniform probability distributions returning “true” if a randomly-drawn value between zero and one is less than p are used to determine if the infection will spread. Thus, whereas the disease spread probabilities stay constant across runs, the result of any given random draw using these probabilities may differ.

Finally, a random-seeded triangular distribution centered at the average infection duration for each agent type, and limited to the range between 50% and 150% of this value, is used to determine the length of time an agent will

remain in the infected state. Triangular distributions were used here because they are a good stand-in for the normal distribution, while offering an intuitive means to establish upper and lower limits.

II.x Observation

II.x.a What data are collected from the ABM for testing, understanding and analyzing it and how and when are they collected?

The model tracks in real-time the current livestock inventory of all producers in the model, the number of currently infected animals, the number of currently infected agents, and the cumulative number of infected agents, which can be output as time-series data to examine infection-spread dynamics. Tabular data including each agent's class, operational parameters, and other information is also output at the conclusion of each run.

In addition, a contact network adjacency matrix with link weights encoding the number of times each agent interacted throughout the model run is exported as tabular data after each run, and later parsed using a series of Python functions. An infection-spreading network is similarly tracked, output, and parsed. Key statistics on trade and infectivity patterns across a series of model runs—both at the individual agent as well as the whole-network level—may then be analyzed.

Finally, for calibration purposes, we include an option to track the flow of feed and livestock between different types of agents, for example the distribution of hog shipment sizes and delivery frequencies, which can then be compared with available real-world data.

II.x.b What key results, outputs or characteristics of the model are emerging from the individuals? (Emergence)

Emergent phenomena in the model occur as a result of the contact patterns mediated by agents' decision heuristics. For example, this could take the form of differential disease spread dynamics resulting from the structures of contact networks that emerge throughout each run.

III) Details

III.i Implementation Details

III.i.a How has the model been implemented?

The model was implemented using AnyLogic v.8 software, which relies upon the Java programming language for all scripts and functions. The sections below use pseudocode to describe in detail the algorithmic structures underlying each model function.

Notes on pseudocode used in this document:

- The characters “//” will be used to designate a descriptive comment (i.e., the line of text following the “//” is not part of the actual function logic).
- Parameters referenced in all functions refer to those associated with the agent object from which a function has been called. In some cases, to disambiguate, the terms “self” or “my” may be used to refer to the function-calling agent object or its associated parameters.
- “ADD OR INCREMENT [sender] in [receiver]'s [network edge list]” is defined here as:

IF [sender] is not in [receiver]'s [network edge list] ADD [sender] to [receiver]'s [network edge list] with contact counter set to 1

ELSE INCREMENT contact counter associated with [sender] in [receiver]'s [network edge list] by 1

- “RANDOM DRAW using [probability]” is defined here as the Boolean value resulting from:

(DRAW random number from uniform distribution between 0 and 1) < [probability]

III.i.b Is the model accessible, and if so where?

While the source code for the model is not accessible due to limitations of the AnyLogic software, the pseudocode below explicates the code at a high level of detail. In principle, these pseudocode functions may be used to implement the model using any desired programming language.

III.ii Initialization

III.ii.a What is the initial state of the model world, i.e. at time $t = 0$ of a simulation run?

The model is initialized by progressing through a series of functions. Several agent parameters are set as each agent object is generated by the model. Next, further agent parameters are set by reference to the model database. Finally, upon completion of the preceding, producer agents initialize their networks of potential trading partners to be referenced throughout the model run.

- Initialization functions called from main object (in order of function calls):

Initialize map view function:

SET GIS map boundaries and zoom on U.I. dashboard to correspond to study area

Initialize agents function:

Numbers, locations, and typological distributions of producer agents within the model are generated heuristically using the Farm Location and Agricultural Production Simulator (FLAPS) system developed through a collaboration between Colorado State University and the United States Department of Agriculture (Burdett et al. 2015). FLAPS parses USDA National Agricultural Statistics Service (USDA NASS) databases along with land use data to impute spatially-explicit datasets depicting the distribution of livestock production units throughout the desired study area. Thus, while not representing actual farm locations, the producer agents in the model are distributed geographically and with characteristics including production volume and classification category in such a way as to be consistent with real-world distributions.

An implicit assumption we have made is that the distribution of slaughter plants and feed mills mirrors the distribution of producers by county. Once a non-producer agent's county has been assigned by drawing from this distribution, the agent is placed at a random set of coordinates within the selected county. Thus, counties with higher producer density will tend to have higher numbers of non-producer industry actors as well.

FOR EACH producer agent

// read in data for each agent from database and set appropriately

READ latitude and longitude from database table corresponding to study area

SET agent's spatial location

READ total capacity from database table corresponding to study area

```

SET agent's total capacity
READ producer industry role from database table corresponding to study area
// in consultation with industry experts, producers classified as "other" in the USDA NASS data are assumed to be
    "wean to feeder"
IF (database query result = "other")
    SET agent's industry role to "wean to finish"
ELSE
    SET agent's industry role to correspond with database query result
SET agent's icon color to match producer type
IF (agent is a farrowing type)
    SET agent's sow inventory to MAX of 1 and (total capacity * sow ratio parameter appropriate for agent's type)
ELSE
    SET agent's sow inventory to zero
SET agent's non-sow capacity to (total capacity – sow inventory)

// iteratively generate initial pig batches
SET minimum batch size to (agent's total capacity * minimum batch size as proportion of capacity parameter)
WHILE (minimum batch size < remaining capacity)
    IF (agent's total capacity <= capacity under which a producer is assumed to have only one batch parameter)
        SET batch size to agent's non-sow capacity
    ELSE
        SET batch size to a random integer between minimum batch size and non-sow capacity
        IF ((batch size + current pig inventory) > non-sow capacity)
            SET batch size to minimum batch size
        SET batch birthday to a random integer between the maximum and minimum age of a pig for the agent's
            industry role
        ADD batch size and birthday to pig tracker
        INCREMENT agent's current inventory by batch size

// distribute other agents by county to correspond with producer density
FOR EACH county in study area
    READ number of producers in county from database
    ADD county name to county distribution array as many times as there are producers in that county
FOR EACH slaughter plant agent
    DRAW RANDOM county name from county distribution array
    SET agent's spatial location to a random point inside the county drawn
REPEAT above FOR loop for feed mill agents

// now that locations and characteristics are set, run each producer's network initialization function in turn
FOR EACH producer agent
    CALL agent's "initialize network" function

```

- Producer agent initialize network function:

Once the agents' locations and industry roles have been initialized, a network initialization function generates a set of potential trading partners for each agent. All producer agents are assigned one feed mill, and finishing producers are also assigned one slaughter plant, both connections being to the n th-closest of that agent type, with n being drawn from a Poisson distribution using the appropriate λ parameter. A pool of potential transferee producers is also generated for each non-finishing producer according to their industry role. These relationships are shown in Figure 1. The potential transferee producers in this potential transferee pool are filtered according to (a) the maximum producer-to-producer connection distance parameter, (b) the minimum capacity similarity ratio parameter, as well as (c) the maximum number of transferee producers parameter.

IF industry role is NOT a finishing type

Potential farms list = FILTER other producer agents s.t. (industry role of other producer is the next step in the production chain) AND (distance to the other producer \leq max producer-producer connection distance global parameter) AND NOT ((total capacity / other producer's total capacity \geq minimum capacity similarity ratio parameter) OR (other producer's total capacity / total capacity \geq minimum capacity similarity ratio parameter)))

WHILE (there are still potential farms AND (my transferee producers \leq max number of transferee producers global parameter))

DRAW RANDOM from potential farms list and ADD to my transferee producers

IF my transferee producers is empty

ADD nearest farm of appropriate industry role

IF industry role is a finishing type

SET my slaughter plant to Nth closest slaughter plant where N is drawn from a Poisson distribution where lambda = the global proximity lambda parameter for slaughter plant connections

SET my feed mill to Nth closest feed mill where N is drawn from a Poisson distribution where lambda = the global proximity lambda parameter for feed mill connections

III.ii.b Is the initialization always the same, or is it allowed to vary among simulations?

The initialization of the spatial location, operational characteristics, and potential trading partners for each agent remains consistent across runs within each of our three study area states. Thus, there are in essence three distinct initial states with regard to the above parameters, defined by the study areas. However, the initial livestock population housed at each producer premises differs between runs, as do the real-time trading choices and infection spread patterns experienced by each agent throughout the run. For more detail, see the *Stochasticity* section.

III.ii.c Are the initial values chosen arbitrarily or based on data?

Initialization parameters rely upon several datasets, including the University of Colorado / USDA FLAPS system (Burdett et al. 2015), USDA NASS data (USDA NASS 2014), Google Maps queries, and livestock industry internal records. For more details, see the initialization function descriptions and pseudocode above, section *III.iv.c*, as well as Table 6.

III.iii Input Data

III.iii.a Does the model use input from external sources such as data files or other models to represent processes that change over time?

The model relies upon an external database to store many of the initialization parameters. Once set, these values remain static throughout each model run.

III.iv Submodels

III.iv.a What, in detail, are the submodels that represent the processes listed in ‘Process Overview and Scheduling’?

- Producer agent cyclically-executing functions:

Farrow, wean, and batch piglets function:

If a farm which farrows piglets (Farrow to Wean, Farrow to Feeder, or Farrow to Finish types) is left with excess capacity after a livestock transfer, a farrowing function fills that capacity with a new batch of piglets, whose birthday is set to the current model day. Once again, to eliminate unrealistically-small pig groups, a minimum farrowing size as a proportion of the farrowing farm’s total capacity is required for the farrowing function to proceed. Thus, a farm which is already almost at maximum capacity will not farrow a new batch of piglets until another batch has been shipped to an appropriate trading partner.

****Recurrence time is the frequency of weaning global parameter****

IF industry role is a farrowing type

 // calculate number of farrowed piglets ready to wean and batch

 Current piglet inventory = MIN of remaining pig capacity and (days since last weaning day * number of sows *
 (global parameter for piglets weaned annually / 365))

IF infectivity state is “infected”

 DECREMENT current piglet inventory according to suckling mortality rate global parameter

```

// wean and batch piglets
Number to wean and batch = MIN of current piglet inventory and remaining pig capacity
IF (number to wean and batch >= my minimum batch size)
    ADD number to wean and batch and birthday (current day – 35) to pig batch tracker
    INCREMENT non-sow pig inventory by batch size
    DECREMENT piglet inventory by batch size
    SET last weaning day to current day

```

Evaluate pig shipments function:

Non-finishing producers transfer hoofstock to a transferee farm as soon as the hoofstock reach the age corresponding to the transfer condition associated with the industry role of the producer. If it is determined that the transfer age requirement of a pig batch has been met, the transferee producers in the transferring producer’s pool of possible producer trading partners are sequentially evaluated to determine whether they are able to receive the shipment. To eliminate the transfer of unrealistically-small groups of livestock, transfers will only proceed if the pig batch size exceeds the minimum transfer quantity, as a proportion of the transferee’s total capacity. If the excess capacity of a potential transferee producer is less than the size of the pig batch, the pig batch will be split such that the transferee producer’s capacity will be filled, and the remaining animals will stay with the transferring producer. The transferring producer will then continue to assess producers until all remaining pigs in the pig batch have been transferred to appropriate trading partners.

If the transferring producer is infected but the transferee is not, the transferred hoofstock will automatically spread the infection to the transferee producer. If the transferee producer is infected but the transferring producer is not, the “delivery trailer” returning from the infected transferee producer may infect the transferring producer according to a probability set at model initialization.

The birthday parameter associated with the batch of transferred stock is maintained as it is passed to the transferee(s), such that the pig batch will once again be appropriately transferred to the next production phase at the correct transfer age. In the rare case that a pig batch grows too old before a suitable transferee producer can be located, the pig group is culled, making room for a new batch of pigs.

Finishing producers (Feeder to Finish and Farrow to Finish types) ship hoofstock their slaughter plant as soon as the hoofstock reach the designated slaughtering age. If the transferring producer is infected, the receiving area of the slaughter plant may become contaminated according to a probability set at model initialization. If the receiving area of the slaughter plant is already contaminated, the “delivery trailer” returning to the transferring producer may carry the infection back to that producer according to another probability set at model initialization.

****Recurrence time is the maximum frequency of pig shipments global parameter****

```

// to eliminate continually changing inventory levels during execution
SUSPEND farrow, wean, and batch piglets function countdown

```

```

// determine where pigs will be shipped

```

```

IF industry role is NOT a finishing type

```

```

    FOR EACH pig batch meeting age transfer requirement

```

```

        FOR EACH transferee in my transferee producers

```

```

            IF (batch size <= transferee’s spare non-sow capacity) AND (batch size >= transferee’s minimum batch size)

```

```

                IF (transferee’s infectivity state is “infected”) AND (infectivity state is “clean”)

```

```

                    DECREMENT batch size according to mortality rate global parameter associated with pigs’ age

```

```

                    REMOVE pig batch from pig batch tracker

```

```

DECREMENT non-sow pig inventory by batch size
ADD pig batch and birthday to transferee's pig batch tracker
INCREMENT transferee's non-sow pig inventory by batch size

// update contact network trackers
ADD OR INCREMENT transferee in contact network out-degree list
ADD OR INCREMENT self in transferee's contact network in-degree list

// update pig shipment trackers
ADD batch size to pig shipments out list
ADD batch size to transferee's pig shipments in list

// infection brought to transferee via infected pigs
IF infectivity state is "infected"
    SET transferee's infectivity state to "infected"
    ADD OR INCREMENT transferee in infection-spreading network out degree list

// infection brought home via trailer from transferee farm
IF (transferee's infectivity state is "infected") AND (RANDOM DRAW using Prob. pig truck will
    become contaminated if producer is infected) AND (RANDOM DRAW using Prob. producer
    will become infected if returning pig truck is contaminated)
    SET infectivity state to "infected"
    ADD OR INCREMENT self in transferee's infection-spreading network out degree list

// cull pigs that are too old and were never able to be transferred
FOR EACH pig batch over 168 days old
    REMOVE batch from pig batch tracker
    DECREMENT non-sow pig inventory by batch size

ELSE IF industry role is a finishing type
    REMOVE pig batch from pig batch tracker
    DECREMENT non-sow pig inventory by batch size

// update contact network trackers
ADD OR INCREMENT slaughter plant in contact network out-degree list
ADD OR INCREMENT self in slaughter plant's contact network in-degree list

// update pig shipment trackers
ADD batch size to pig shipments out list
ADD batch size to slaughter plant's pig shipments in list

// infection brought to slaughter plant via infected pigs
IF (infectivity state is "infected") AND (RANDOM DRAW using Prob. slaughter plant receiving area will become
    infected if pig batch is infected)
    SET slaughter plant's infectivity state to "infected"
    ADD OR INCREMENT slaughter plant in infection-spreading network out degree list

```

```

// infection brought home via trailer from slaughter plant
IF (slaughter plant's infectivity state is "infected") AND (RANDOM DRAW using Prob. pig truck will become
    contaminated if receiving area is infected) AND (RANDOM DRAW using Prob. producer will become
    infected if returning pig truck is contaminated)
    SET infectivity state to "infected"
    ADD OR INCREMENT self in slaughter plant's infection-spreading network out degree list

```

```

IF (non-sow pig inventory is zero)
    SET infectivity state to "clean"
RESUME farrow, wean, and batch piglets function countdown

```

- Feed mill agent cyclically-executing functions

Feed mills periodically generate delivery routes encompassing a subset of producers within their latent feed-mill-to-producer link set. Each route encompasses a subset of the producers in the feed mill's service area, with the number of stops in each trip resulting from a draw from a Poisson distribution. While there is no actual "feed truck" object in the model, the logic of the following function is based on the way such a truck would move between agents and possibly spread disease.

Beginning from the mill, this conceptual feed truck will visit the previously-drawn number of randomly-selected producers within the feed mill's service area before finally returning to the feed mill. If the feed mill is infected, the truck may be contaminated initially. Should the truck encounter an infected producer on its route, it may become contaminated at that point. Once a truck is contaminated, the infection may be spread to subsequent producers on the route. If a contaminated truck returns to the feed mill, the mill itself may become infected.

Distribute feed function:

```

**Recurrence time is the frequency of feed deliveries global parameter**
// determine whether truck is initially infected
IF (infectivity state is "infected") AND (RANDOM DRAW using Prob. feed truck will become contaminated if feed mill
    is infected)
    SET truck infected to "true"
ELSE
    SET truck infected to "false"

// determine number of farms on delivery route
SET number to visit to MIN of (number of farms in service area) and (DRAW from Poisson distribution with lambda
    equal to global parameter encoding average number of producers visited per route)

// generate delivery route
FOR number to visit
    ADD random producer in service area (that is not already in delivery route list) to delivery route list

// parse infectivity consequences of delivery route
FOR EACH producer in delivery route list
    // update contact network trackers

```

```
ADD OR INCREMENT producer in contact network out-degree list
ADD OR INCREMENT self in producer's contact network in-degree list
```

```
//infected truck infects farm it's delivering to
IF (truck infected is true) AND (RANDOM DRAW using Prob. producer will become infected if feed truck is
    contaminated)
    SET producer's infectivity state to "infected"
    ADD OR INCREMENT producer in infection-spreading network out degree list
```

```
//truck becomes infected from delivery to infected farm
IF (producer's infectivity state is "infected") AND (RANDOM DRAW using Prob. feed truck will become
    contaminated if producer is infected)
    SET truck infected to "true"
    SET "truck-infecting producer" to current producer
```

```
// infected truck infects feed mill
IF (truck infected is "true") AND (RANDOM DRAW using Prob. feed mill will become infected if returning feed truck
    is contaminated)
    SET infectivity state to "infected"
    ADD OR INCREMENT self in truck-infecting producer's infection-spreading network out degree list
```

- Initial infection function

The system is initialized with all agents free of infection. After one model year has passed, an infection is introduced to a random subset of producer agents. The proportion of agents which are infected by the initial infection function may be set at model initialization.

The reason for the one-year lag is to skip the transient period and allow the model to stabilize before analyzing the effect of an introduced disease. This lag is necessary because, as in a real production chain, a certain amount of slack, or a difference between the theoretical production capacity and actual production, is characteristic in the modeled production chains. In the model, this economic slack is due to the producers sometimes temporarily operating at less than maximum hoofstock capacity until an appropriate shipment of livestock becomes available. In general, after about 9 months, the level of slack in the model has stabilized.

Initial infection function:

```
**Function is called only once, after one model year**
FOR number to infect global parameter
    SET randomly chosen producer agent's infectivity state to "infected"
```

- Infection control functions

Susceptible/infective state charts:

Each agent has an embedded state chart which encodes its infectivity status. Should an agent become infected, a function is called which calculates the number of its stock which are to die of the disease. The proportion of livestock which succumb to the disease is based on the age of the pig groups, with uniform mortality rates set at model initialization for suckling pigs, nursery pigs, and grow/finish hogs. After die-off is calculated for pig

groups of each life stage within an infected producer’s inventory, the producer’s inventory data are updated accordingly. An agent will remain infected for a duration whose mean length in days is controlled by parameters specific to each agent type by drawing from a triangular distribution limited to between 50% and 150% of the mean value. Upon transition back to “clean,” the total infection duration is updated for later analysis. In some cases, such as when a producer has an inventory of zero after transferring pigs, the transition back to a “clean” state may also be triggered manually.

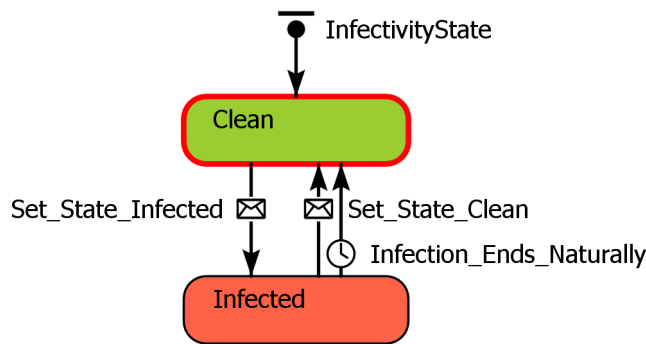


Figure 2: Infectivity State Chart

Livestock mortality calculation:

Calculate mortality function:

```

DECREMENT piglet inventory by global parameter encoding piglet mortality proportion
FOR EACH pig batch in pig batch tracker
    DECREMENT pig batch size by global parameter encoding mortality proportion appropriate for age of pigs
    
```

III.iv.b What are the model parameters, their dimensions and reference values?

See Table 6.

III.iv.c How were the submodels designed or chosen, and how were they parameterized and then tested?

Due to the inherent variability of epidemic events within complex networked systems, we are interested less in empirically-validating the model to be used as a forecasting tool, and more in developing sufficient structural and face validity to allow for a deeper understanding of the dynamics of the modeled systems. Even given identical starting conditions, deviations in contact patterns over the course of a real-world disease incursion render precise forecasts unfeasible. For example, while reducing outcome volatility in RUSHPNBM could easily be accomplished by eliminating stochasticities associated with disease transmission, calibrating the model such that outcomes correspond precisely to a single observed epidemic event misses the point. Our aim is rather to uncover and better understand the fundamental network features that lead to epidemiological vulnerability in livestock production systems more generally.

Calibration and validation procedures that leverage concrete historical data are often regarded as the best way to bring a model in line with empirical evidence. Unfortunately, there is a marked lack of publicly-available data in the agricultural sector beyond aggregated county- or state-level statistics. To the extent that datasets containing

explicit locations, operational parameters, livestock and feed movements, and disease histories exist; these data tend to be held by private enterprises, which view them as sensitive internal records. In light of this, following Windrum et al. (2007), we employ several alternative calibration procedures that have been widely-used in previous modeling endeavors in which fine-grained data are scarce.

The spatial locations and basic operational parameters of RUSHPNBM agents associated with each study area are calibrated using the "indirect" approach, whereby stylized facts about the distribution of agents in the system are gleaned from statistical datasets. Statistical datasets used in this process include the FLAPS output data, USDA data, and livestock and feed movement records we obtained from a large U.S. family-owned hog production chain system (discussed below).

To calibration additional model elements that define how and when inter-agent contact occurs, as well as epidemiological submodel parameters, we leveraged an iterative companion modeling approach (Barreteau et al. 2003). 2015 to 2017, we convened several Delphi panels consisting of livestock industry and veterinary experts at national research team meetings and livestock veterinary conferences. In these meetings, we used both qualitative focus groups as well as questionnaires to elicit and hone parameter values. Using this participatory methodology, the modeled system was brought in line with the collective understandings of stakeholders who are intimately familiar with the operational details of U.S. livestock production systems. As model development progressed, these same experts also provided input to ensure the face validity of the distribution of epidemic patterns, scales, and durations produced by the model.

The model's parameters and functions controlling pig movement and feed deliveries were further validated with the help of data provided by a large U.S. family-owned hog production chain system (as a result of our confidentiality agreement, the company's name is not disclosed here). The database contains two-year records of each pig movement and each feed delivery involving producers in the system, although spatial data on premises locations were not provided. The family-owned hog production chain system consists of a network of 161 producer partners that raise pigs from birth to market. This production chain system has two characteristics that allowed for the parameterization and validation of RUSHPNBM. The first characteristic is that the farm sizes vary from small (300 pigs) to large (8800 pigs, Figure 3). The second characteristic is that pigs are grown at specialized sites including farrowing, wean-to-finish, nursery, and finishing premises; and are moved across the production network according to their growing stage. The pig movement records were used to derive realistic estimates of hog transfer frequencies and number of hogs per transfer relative to farm size, including both producer to producer transfers, and producer to slaughter plant transfers (Table 7). The feed delivery records were used to estimate delivery frequencies (Figure 4).

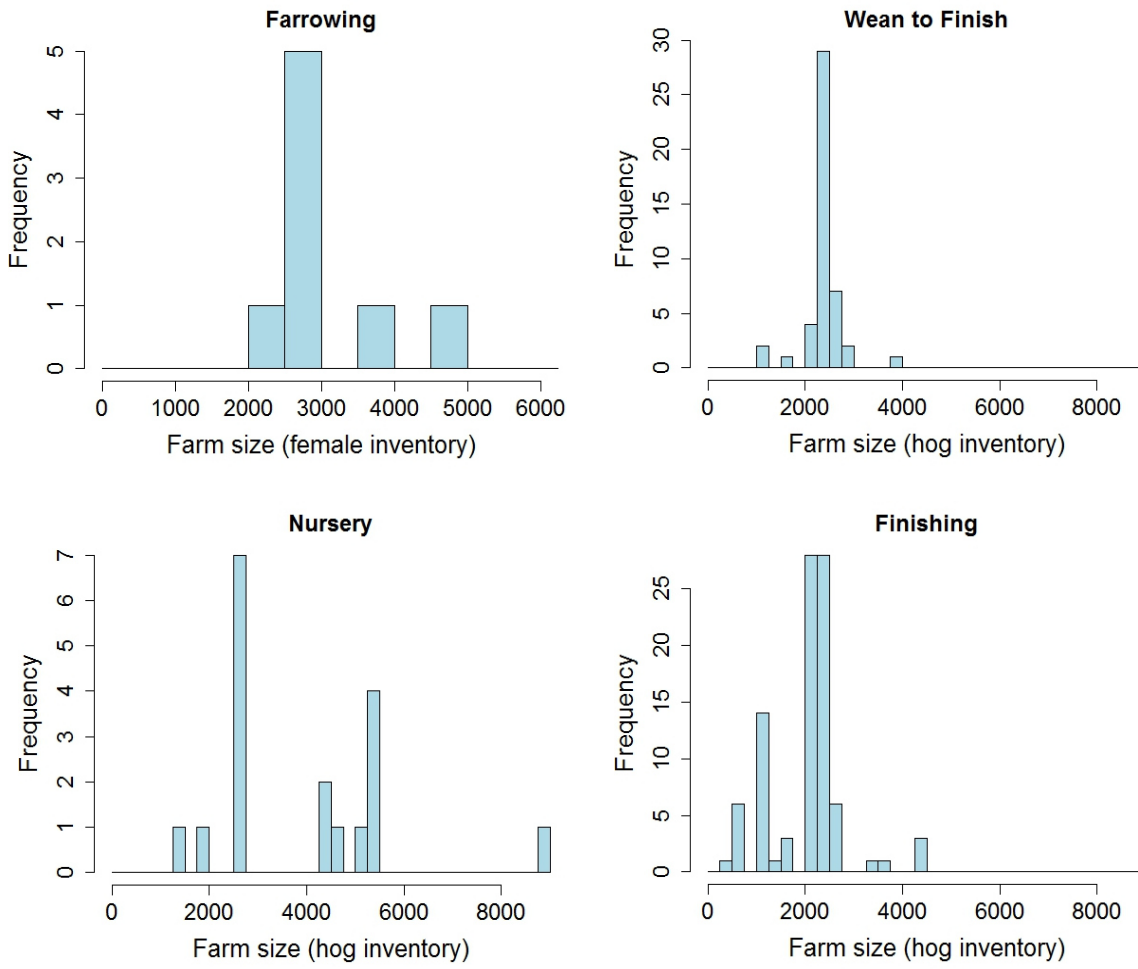


Figure 3: Farm size distribution by farm type (farrowing, wean-to-finish, nursery, finishing)

Table 7: Average annual number of pig shipments within the study period by operational classification

FROM	TO	N shipments FROM	N shipments TO	Avg. shipment size
Wean to Finish	Wean to Finish	1.9	1.6	259
Wean to Finish	Finishing	2.5	1.7	608
Finishing	Finishing	0.5	0.5	176
Nursery	Nursery	1.2	0.8	161
Nursery	Wean to Finish	1.5	1.5	652
Nursery	Finishing	45.7	8.6	512
Supplier	Nursery	156.6	41.2	582
Supplier	Wean to Finish	60.9	10.8	406
Nursery	Customer	0.8	4	3
Finishing	Packer / Customer	27.9	276.6	153
Wean to Finish	Packer / Customer	22.4	179.5	153
Nursery	Packer / Customer	0.5	1	2

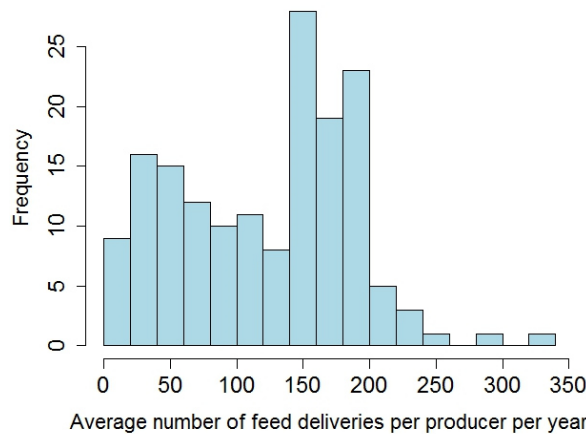


Figure 4: Average number of feed deliveries per producer per year

Finally, we conducted a sensitivity analysis focusing on four key parameters, each representing a specific aspect of the model's architecture. Each parameter is varied in steps between 50% and 150% of the baseline values given in Table 6, with ten replications per step. Table 8 shows the elasticity of the response variable---average vulnerability---across this range. Figure 5 visualizes the sensitivity analysis data along with linear correlations.

Table 8: Elasticity of response variable (Avg. Vulnerability) resulting from variation between -50% and 50% of baseline values for four key parameters

Parameter	% Change in Avg. Vulnerability			
	North Carolina	Iowa	Illinois	All Study Areas
Prob. producer to pig truck infection	49.93	109.9	491.1	94.56
Avg. producer infection length (days)	2382	3164	6077	3010
Max producer connection distance (km)	2.182	-20.32	108.6	7.976
Prob. feed truck to feed mill infection	36.21	114.8	119.4	50.82

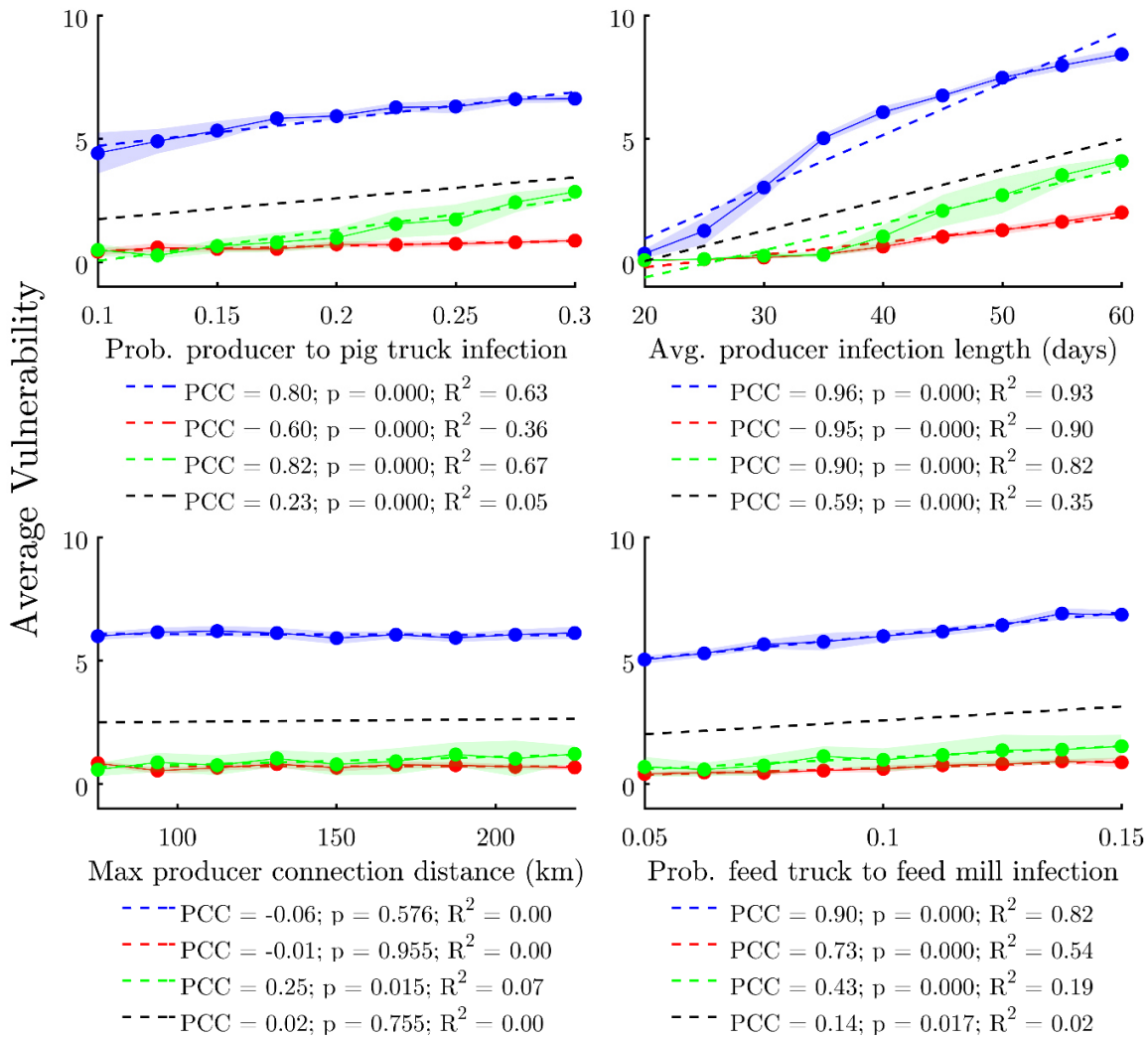


Figure 5: Sensitivity analysis plots for four key parameters. Scatter points show average values at each step, colored regions show 95% CIs, and dashed lines show linear trends. Blue represents North Carolina, red Iowa, green Illinois, and black the combined dataset. Pearson correlation coefficients, p-values, and R^2 values of linear regressions appear in legends below each figure.

Results of the sensitivity analysis show that the model is moderately-sensitive to changes in the probability of producer to pig truck infections and the probability of feed truck to feed mill infections; with the effect on average vulnerability being positive in all study areas, and generally highest in Illinois. For maximum producer connection distance, the magnitude and direction of the effect varies considerably between study areas, with Iowa demonstrating a negative relationship with the response variable. The R^2 values suggest that, overall, the model is not particularly sensitive to maximum producer connection distance. By contrast, increasing the average producer infection length causes significant increases in average vulnerability across all study areas. In light of previous SIR / SI model studies, the observation that average infection duration heavily impacts average vulnerability is not a surprise. The shape of the elasticity curves suggest that percolation dynamics may exist, with the nonlinearity---or percolation threshold---being lowest for North Carolina and higher for the other two study areas, corroborating findings from Wiltshire (2018).

References

- Barreteau, O., Antona, M., D'Aquino, P., Aubert, S., Boissau, S., Bousquet, F., . . . Mathevet, R. (2003). Our companion modelling approach. *Journal of Artificial Societies and Social Simulation*, 6(1).
- Burdett, C. L., Kraus, B. R., Garza, S. J., Miller, R. S., & Bjork, K. E. (2015). Simulating the distribution of individual livestock farms and their populations in the United States: An example using domestic swine (*Sus scrofa domestica*) farms. *PLoS one*, 10(11), e0140338.
- Goede, D., & Morrison, R. B. (2016). Production impact & time to stability in sow herds infected with porcine epidemic diarrhea virus (PEDV). *Preventive Veterinary Medicine*, 123, 202 - 207.
- Grimm, V., Berger, U., Bastiansen, F., Eliassen, S., Ginot, V., Giske, J., . . . others. (2006). A standard protocol for describing individual-based and agent-based models. *Ecological Modelling*, 198(1-2), 115-126.
- Grimm, V., Berger, U., DeAngelis, D. L., Polhill, J. G., Giske, J., & Railsback, S. F. (2010). The ODD protocol: A review and first update. *Ecological Modelling*, 221(23), 2760-2768.
- Muller, B., Bohn, F., Dressler, G., Groeneveld, J., Klassert, C., Martin, R., . . . Schwarz, N. (2013). Describing human decisions in agent-based models: ODD+ D, an extension of the ODD protocol. *Environmental Modelling & Software*, 48, 37-48.
- The Pig Site. (2014). Pig Farm Targets: Stockmanship Standards. Retrieved from <http://www.thepigsite.com/stockstds/3/pig-farm-targets/>
- USDA NASS. (2014). Livestock Slaughter, 2013 Summary: Federally Inspected Plants and Head Slaughtered by Species, States and United States, 2013. <http://usda.mannlib.cornell.edu/usda/nass/LiveSlauSu/2010s/2014/LiveSlauSu-04-21-2014.pdf>
- Wiltshire, S. W. (2018). Using an agent-based model to evaluate the effect of producer specialization on the epidemiological resilience of livestock production networks. *PLoS one*, 13(3), e0194013.
- Windrum, P., Fagiolo, G., & Moneta, A. (2007). Empirical validation of agent-based models: Alternatives and prospects. *Journal of Artificial Societies and Social Simulation*, 10(2), 8.