Developing a vision for executing scientifically useful virtual biomedical experiments

Brenden K. Petersen
The UC Berkeley-UCSF Graduate Program in Bioengineering
University of California, Berkeley, CA, USA
brenden.petersen@berkeley.edu

C. Anthony Hunt
Department of Bioengineering and Therapeutic Sciences
University of California, San Francisco, CA, USA
a.hunt@ucsf.edu

ABSTRACT

A virtual biomedical experiment (VBE) is a simulation of a wet-lab or clinical biomedical experiment. The goal of VBEs is to provide a scientifically useful method of in silico experimentation that can challenge concrete hypotheses and provide explanatory, mechanistic insight into the referent system. We envision virtual experimentation not as a supplement to traditional wet-lab experimentation, but rather as an essential part of the scientific method itself. The goal of this work is to lay preliminary groundwork for realizing this vision, through outlining requirements and describing agent-based models demonstrative of this vision. VBEs focus on reasoning by analogy; thus, a VBE includes model components analogous to particular relevant aspects of the referent experiment—from hypothesis formation to data analysis, and key concepts in between. We explore five exemplary categories of scientifically useful VBEs: hypothesis, experiment context, living counterparts, experiment agents, and measurements. We discuss how to develop model components of each category in the context of agent-based modeling. We highlight the importance of two overarching requirements: concreteness and modularity. Finally, we demonstrate this vision by describing an in silico liver model that partially fulfills the VBE vision and requirements, including components corresponding to each of the five VBE categories.

Author Keywords

virtual experiment; agent-based model; modeling and simulation

ACM Classification Keywords

I.6.m [Simulation and Modeling]: Miscellaneous

Abbreviations

virtual biomedical experiment (VBE), modeling and simulation (M&S), agent-based model (ABM), agent-directed simulation (ADS), object-oriented programming (OOP), in silico liver (ISL), in silico hepatocyte culture (ISHC)

SpringSimSim-ADS 2016 April 3-6 Pasadena, CA, USA

1. INTRODUCTION

1.1 The virtual biomedical experiment vision.

Envision a biomedical R&D landscape in which researchers plan detailed wet-lab experiments and execute them in a virtual laboratory—all before putting on their lab coat. They choose virtual reagents and lab equipment needed for their use case. They select among a range of in vitro, in vivo, and/or human specimens. They design, customize, and execute virtual protocols and issue virtual treatments. They observe their virtual system and take virtual measurements using virtual instrumentation. They share their results with wet-lab researchers and other stakeholders who can follow, interpret, and comment (unassisted) on the details and results of the virtual experiment. They use the results of virtual experiments to design new or refocused wet-lab experiments, which they then conduct in a physical laboratory.

Envision virtual experimentation not as a supplement to traditional wet-lab experimentation, but rather as an essential part of the scientific method itself. The goal of this work is to lay preliminary groundwork for realizing this vision, through outlining requirements and providing models demonstrative of this vision. The discussion and models presented herein move toward that vision.

1.2 What is a virtual biomedical experiment?

A *virtual biomedical experiment* (VBE) is a simulation of a wet-lab or clinical biomedical experiment. In some sense, simulations using most existing biological models can be considered VBEs because they mimic aspects of the underlying biological processes that occur during wet-lab experiments, possibly in addition to any treatments issued and measurements taken. However, the VBE concept used here encompasses a broader, more "open" vision than current modeling practices, in which the modeler aspires to mimic particular relevant aspects of the referent experiment—from hypothesis formation to data analysis, and key concepts in between—not just features of the underlying biological processes. It requires that models are designed for particular use cases, but are also flexible and reusable for future, yet to be specified use cases.

1.3 Existing support for virtual experimentation

Experimentation via simulation is not a new concept. Several recent works focus on managing simulation experiments throughout their entire life cycle [1-3]. With an eye toward reproducibility, these studies combine model-driven engineering concepts with intelligent software agents to aid in the design, batch execution, and iterative refinement of simulation experiments. The methods allow one to transform an experiment description using domain-specific vocabulary to one using an experiment ontology, which is then translated into an executable script. Simulations fulfilling the VBE vision can make use of such tools to manage virtual experiments.

To date, virtual experiments have been primarily used as pedagogical tools. For example, consider the first two results of a Google Image search of "virtual experiment" (without the quotations), shown in Figure 1. The first image depicts the classic "celery experiment" used to demonstrate capillary action. The image comes from an interactive classroom teaching tool [4]. While this may seem a trivial example, it actually highlights several important aspects of virtual experiments. We see, of course, the celery experiment itself, but also instrumentation used for observations (magnifying glass) as well as recorded measurements on the right. The second image represents a virtual assay used to teach laboratory skills remotely [5]. Users select virtual reagents and perform virtual tasks (e.g. mixing reagents, centrifuging liquids, and separating supernatants). The model uses rules and embedded "tutoring agents" that track user progress and inform the user of mistakes. Note it is not the visualization that renders these virtual experiments; rather, it is the fact that many aspects of the experiment—besides the biological processes themselves—are modeled explicitly.

Pedagogy is useful in itself; however, the question remains whether virtual experimentation can be used to advance biomedical science. We argue that virtual biomedical experiments can be scientifically useful when simulations are used to challenge hypotheses and encompass multiple aspects of referent wet-lab or clinical experiments.

1.4 Prerequisite: challenging mechanisms with analogs.

The context is using modeling and simulation (M&S) to challenge and improve explanatory, mechanistic hypotheses of biological phenomena. Models developed in support of this goal are software devices, suitable for scientific experimentation. We refer to such models—and the object-oriented software components that comprise them—as *analogs*. We prefer the term "analog" over "model" to highlight the fact that analogs are analogous to their referent system in both structure and function. Analogs are expected to be perpetual works-in-progress that undergo cycles of falsification, refinement, and validation. Analog mechanisms are validated by comparing analog measurements to existing, commensurate wet-lab validation data. When an analog measurement exhibits a prespecified acceptable level of similarity to validation data, the analog has achieved its *validation target* for that use case. If the analog cannot achieve validation under any biologically reasonable parameter settings, it is falsified. A falsified analog is then iteratively, parsimoniously modified until it can achieve its validation targets. Once validated, additional validation targets are added. As analogs achieve increasingly large sets of validation targets, they become increasingly analogous to their referent system.

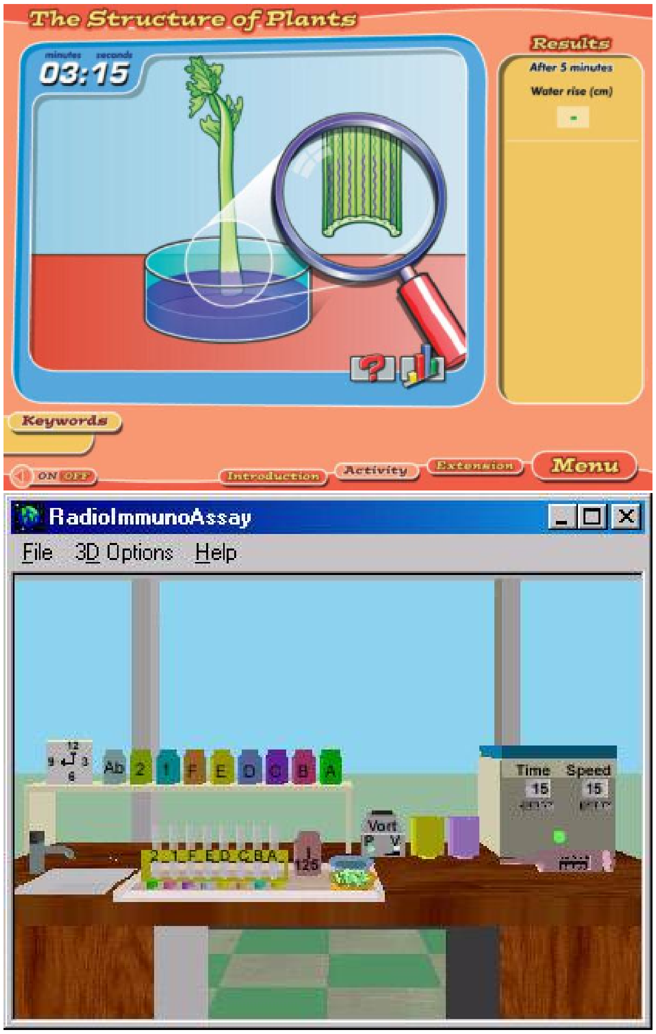

Figure 1. **Virtual experiments as pedagogical tools.** The first two search results found in a Google Image search for "virtual experiment." Top: A virtual celery experiment used to demonstrate aspects of the scientific method and capillary action. Bottom: A virtual assay used to teach laboratory skills remotely.

1.5 Agent-based models are ideal for VBE

Simulations fulfilling the VBE vision must include analog counterparts to several aspects of a referent experiment, including hypotheses, the environment, protocols, instrumentation, and measurements. Further, they require the flexibility to mimic a variety of wet-lab experiments; thus, it must be straightforward to change model context. Lastly, analogs must be falsifiable if they are to be used to challenge mechanistic hypotheses. Such additional requirements render traditional equation-based modeling approaches difficult, if not problematic. Instead, modular approaches (e.g. object-orientation; component-based) are more appropriate.

Agent-based models (ABM) and agent-directed simulations (ADS) are particularly well suited to achieving the VBE vision. In ABM, agents are software objects situated within a virtual environment that can sense, be a part of, and interact with their environment as well as with other agents; agents are capable of scheduling their own events in pursuit of their own agenda [6]. ABM components and mechanisms are discrete and can be made semi-autonomous and modular. Agents may map to biological components, as well as other parts of an experiment (e.g. the experimental environment or an observer). Using ABM, the modeler can easily instantiate two or more competing mechanisms and test them in parallel; thus, they are falsifiable and suitable for experimentation. Agents can be flexible to changes in context: when modular, agents can switch among different model environments. These features render ABM ideal candidates for VBEs. Before outlining more specific VBE requirements, we first limit subsequent discussion to analogs utilizing agent-based methods.

2. CATEGORIES OF THE VBE ANALOGY

An experiment on an analog is a VBE, precisely analogous to a wet-lab or clinical experiment. But this analogy has many parts. Below we describe five major categories of the VBE analogy. Each category requires development of one or more analogs which, when composed, result in a VBE. Developing analogs of each category is necessary to achieve the VBE vision. Each category carries a set of associated technical modeling requirements. Two overarching requirements are concreteness and modularity, which are expounded in “Meeting VBE requirements.” To avoid ambiguity between analog components and their referent biological counterpart, we use SMALL CAPS when referring to the former, e.g. HEPATOCYTE.

2.1 Category 1: Hypothesis

Before discussing this category of analogy, consider the following observation. A tacit hypothesis underlies most wet-lab experiments: there exists some degree of phenotypic overlap between model (i.e. in vitro or animal) and referent (i.e. humans). In other words, researchers hypothesize that results found in the model system may correspond to analogous findings (if/when they are available) in the referent system. The concept is illustrated in Figure 2A. This observation explains why scientists bother conducting in vitro and animal experiments at all. If this hypothesis were not supported (indeed, follow-up clinical experiments often reveal that it is not), in vitro and animal experiments would be scientifically useless for human applications. Of course, phenotypic overlap is partial: there are always aspects of the referent system that are different (or not present) in the model system. Indeed, many research efforts focus on quantifying the areas of phenotypic overlap, as well as improving model systems to increase the amount of overlap, e.g. [7].

The importance of the above observation is that *the in silico hypothesis is the same*: there exists some degree of phenotypic overlap between model and referent. In this case, the model is an analog, and the referent is a wet-lab or clinical system. Much like an in vitro system exhibits phenotypic overlap with the referent human system, agent-based analogs share a portion of phenotype space with their referent. An important difference is that synthetic analog phenotypes are not necessarily intended to mimic humans; rather, an analog’s referent is often an in vitro or in vivo system (purple circles in Figure 2B). While different analogs may have different referent systems, they may share components and mechanisms, which allows switching between different referents (see “Experiment Context” below). Further, we can observe the transformation that occurs when switching among analogs of different referent systems (light purple areas in Figure 2B); such transformations may provide insight into the differences between referent systems.

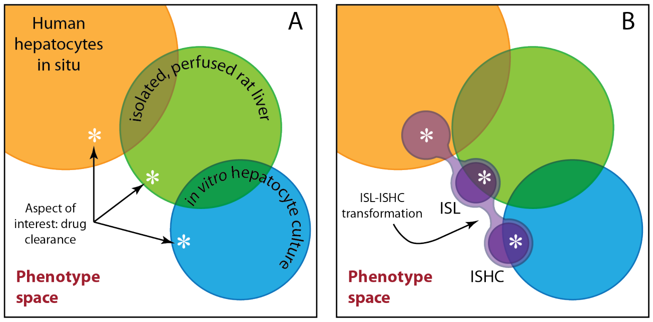
Human hepatocytes in situ
A
isolated, perfused rat liver
in vitro hepatocyte culture
*
*
*
Aspect of interest: drug clearance
Phenotype space
B
*
*
*
ISL-ISHC transformation
ISL
ISHC
Phenotype space

Figure 2. **Phenotypic overlap.** A) Shaded areas represent hypothetical phenotype space occupied by two different wet-lab platforms and their referent human system in the context of drug clearance. Asterisks represent specific, measurable system attributes. There is clear overlap of some measured attributes of an isolated, perfused rat liver (green circle) and corresponding human hepatocytes in situ (orange circle). The same can be said of in vitro hepatocyte culture cells (blue circle) and an isolated, perfused rat liver. In non-overlapping regions, the mapping between related attributes is complex. If phenotypic overlap exists between two platforms, then findings in one system may correspond to analogous findings in the other system. B) The ISL and ISHC (dark purple circles) are in silico analogs with their own measurable phenotypes. Overlapping regions represent targeted attributes that have achieved quantitative measures of similarity. The light purple connecting the two analogs illustrates that the transformation between the ISL and ISHC need not be one-to-one. Exploring ISL-SHC transformations may be instructive of the transformation that occurs between when in vivo cells are isolated into in vitro cultures.

While the hypothesis of phenotypic overlap is usually unspoken, we strive to keep the hypothesis explicit. Thus, the first category of analogy includes the hypothesis of phenotypic overlap. Analog mechanisms must be able to generate phenomena that can be measured and compared to referent data. This hypothesis is supported when measured phenomena fall within ranges that are acceptably similarity to referent measurements; the hypothesis is otherwise falsified. Similarly, there is the hypothesis of mechanistic overlap. Analog mechanisms should be similar to referent mechanisms during execution. Validation results in plausible mechanisms, and falsification shrinks plausible mechanism space.

The ability to mimic hypothesis testing in silico entails several requirements, including the ability to produce a mechanism upon execution, the ability to measure the mechanism during execution, and the ability to develop and refine several competing mechanisms in parallel. These requirements are all characteristics of analogs aimed at improving explanatory, mechanistic insight.

2.2 Category 2: Living Counterparts

Of course, a VBE requires analogs of living counterparts: “the biology.” This category is typically the main (or only) focus of publications related to biological M&S and often requires the most attention when developing a model. It includes analogs of biological components and biological mechanisms. Components are discrete software objects that maintain state information. Components include both agents and non-agents. Agents are "active" (i.e. they make decisions and execute actions), e.g. objects that map to cells. Non-agents are "passive" (i.e. they are acted upon by agents), e.g. objects that map to drug molecules. Mechanisms need sets of operating principles (i.e. rules, equations, and/or governing logic) that instruct agents how to interact with other components.

One requirement of analog components that map to living counterparts is that they are *biomimetic*. A *biomimetic analog* is an analog aspiring to emulate aspects (including phenomenal, componential, causal, and organizational) of a referent biological system. Analogs are made increasingly biomimetic through iterative refinement and the achievement of an increasingly large set of validation targets [8].

2.3 Category 3: Experiment Context

A VBE does not simulate biology or biological processes in isolation. Rather, multiple aspects of the experiment context are modeled explicitly. For example, an experiment on a rat liver analog is not simply a model of (say) drug metabolism in a generic sense; rather, it is a model of (say) acetaminophen metabolism in a single-pass, isolated, perfused rat liver. Explicitly modeling the experiment context stands in stark contrast to the many existing biological models that simulate biological processes either in isolation or in hypothetical or idealized contexts that cannot be reproduced in the wet-lab. For example, many existing ABMs simulate a block of cells [9] or a portion of a cell membrane [10] in an otherwise unspecified environment. VBEs must make experiment contexts explicit. The experiment context includes the structure of the wet-lab system along with any external and/or environmental influences. For example, an in vitro cell culture analog may include analog components mapping to a tissue culture flask, the surrounding culture media, and the cell incubator and/or cryogenic freezer environments. There are many variations of in vitro systems, each of which constitutes a unique experiment context. The particular components to include (or not) in analogs of the experiment context depends on the use case.

Analog components can be switched among different analog experiment contexts between simulations. For example, an in vitro hepatocyte culture analog can use the same HEPATOCYTE components as an isolated, perfused liver analog or the liver component of a whole mouse analog. Switching contexts may be accomplished using model parameters that control which parts of the experiment context to include, or by developing multiple models with shared components. An analog unable to switch experiment contexts can only mimic wet-lab experiments in a single context, thereby limiting its usefulness. This ability to switch contexts is important to widespread adoption of the VBE vision. When an analog can exist in various contexts, it is more like its referent, which can also exist in various contexts.

Components can even switch during a single simulation. Consider a wet-lab protocol in which rats are injected with drug to trigger a whole-body response; after 24 hr, they are sacrificed, their livers are excised and then homogenized, and microsomes are separated via centrifugation, as in [11]. A VBE of this experiment would require analogs in both whole rat and microsome contexts (and possibly excised and homogenized liver contexts). Analog components (e.g. microsomes containing drug-metabolizing enzymes) must be able to switch during a single simulation in order to simulate this VBE.

Switching contexts requires that analog components be modular and semi-autonomous. Modularity allows components to be reused without significant model refactoring (see "Modularity"). When components are semi-autonomous, they can sense and interact with their environment, but do not completely rely on that environment to operate. Modular, semi-autonomous analogs are well supported by the object-oriented programming paradigm.

A common concern with reusing analog components (e.g. analog cells) in different experiment contexts (e.g. in vitro vs. in vivo) is the observation that real cells behave differently in different contexts. How, then, can the same analog cell be reliably used in different analog environments? Consider a hypothetical HEPATOCYTE in an in vitro versus in vivo analog context. There are several ways in which HEPATOCYTE can behave differently in the two environments. Firstly, there may be differences in parameterizations. For example, a parameter that maps to oxygen available to the cell may be a constant value for the in vitro analog but a gradient within the in vivo analog. HEPATOCYTE mechanisms that depend on an oxygen parameter value will then operate differently. Secondly, there may be differences in the in vitro and in vivo contexts that indirectly affect HEPATOCYTE behavior. For example, due to the nature of the in vivo architecture, centrilobular HEPATOCYTES will experience greater concentrations of mobile objects, which will lead to downstream zonation effects not present in the in vitro analogs. Lastly, there may be analog mechanisms that are exclusively turned "off" or "on" in particular contexts. For example, HEPATOCYTES in the in vitro analog context may include a mechanism that favors forming flattened monolayers. Such a mechanism may be manually turned "on/off" by changing the value of a simulation parameter. However, when analogs are semi-autonomous, a more biomimetic solution would be for analogs to self-regulate which mechanisms they turn "on/off" based on the environment they sense. Continuing the previous example, HEPATOCYTES can sample their local environment: if they detect nearby acellular surfaces, they may form a flat monolayer; otherwise, they may assume a cuboidal structure, which can change exposure to the local environment. Thus, there are several methods whereby analogs in one context may adapt their behavior in another context.

2.4 Category 4: Experiment Agents

An experiment agent is a software object analogous to a scientist conducting the experiment. Experiment agent activities include all actions that may be relevant to the experiment, from setting up the experiment to analyzing the data. Experiment agents are not only intuitive, but also increase transparency to other research and development stakeholders, simply because VBE protocols map directly to familiar wet-lab protocols. Most experiment agent practices can be labeled under the following categories:

- Setting up the experiment. This wet-lab category includes assembling platforms and setting the conditions (e.g. temperature); it maps to the in silico process of setting model parameters, initializing the model, and instantiating the analog.
- Running batch simulations. This in silico process is analogous to running wet-lab replicate experiments or testing multiple participants in a clinical experiment. In silico, experiment agents can run batch simulations, the results of which are later combined and/or averaged. Batch simulations can be run in series or in parallel.
- Executing protocols. Experiment agents execute actions analogous to wet-lab protocols. For in vitro analogs, actions may be analogous to changing culture media, passaging cells, moving cells from a cryogenic freezer to a cell incubator, and placing cells on a shaker. For in vivo analogs, this may include sacrificing animals and isolating cells. These activities are distinct from biomimetic processes (e.g. simulated metabolism), which are handled within the "living counterparts" category of analogy.
- Issuing treatments. This category is a special case of executing protocols that includes issuing treatments (e.g. drugs), which may include analogs of devices used to administer drugs, like a pipette or intravenous injection.
- Taking measurements at predefined time points. Experiment agents measure analog phenomena just as a wet-lab scientist measures the objects of their experiments. Experiment agents of this kind are referred to as observer agents. Measurements may or may not interfere with the rest of the simulation (e.g. removing an aliquot of culture media interferes by changing the volume of remaining media).
- Analyzing data. Experiment agents can perform calculations and produce plots. At the software level, this may require a set of algorithms or scripts that perform calculations on and/or plot analog measurements. Analyses may be executed during the simulation or "offline" (after the simulation has finished). Notably, experiment agents can follow a protocol to compare simulation data to validation data and provide a measurement of similarity.
- Tuning parameters to achieve validation targets. When a VBE does not produce results acceptably similar to validation data, experiment agents can adjust parameter values before executing the next series of simulations. The process is similar to Bayesian updating, in which simulation results are used to inform the choice of subsequent parameter values. This experiment agent practice is loosely analogous to a scientist adjusting experimental conditions when refining experimental protocols: the scientist uses a combination of rigorous "parameter sweeps" (e.g. trying a 10^6-fold range of doses) and intuitive, heuristic decisions (e.g. compensating for a change in variable X by changing variables Y and Z) to achieve an optimal protocol.

The above activities facilitate automating ADS experiments; analogously, a VBE facilitates the shift toward employing automation tools. Quasi-automated experiment agents can execute a VBE, then (if falsified) adjust parameter values until validation is achieved or (if validated) add new validation targets and repeat the cycle. Thus, experiment agents will play key roles in automating portions of falsification-refinement-validation cycles. Options for parallel computing will also increase computational speed when running batch simulations.

Observer agents should be only loosely coupled with the rest of the simulation. That is, their existence should not affect analog system operation (except when intentional, for example when the act of taking measurements interferes with the analog). Achieving loose coupling can be facilitated using object-oriented programming and modularity.

2.5 Category 5: Measurements

Analogs executing VBEs are not simply input/output machines. Rather, they are concrete software devices that can be measured and observed. As stated above, observer agents take measurements on the analog under prespecified conditions. These measurements stem from analog state information. The term "measurement" is preferred over "output" to emphasize the fact that something (i.e. an observer agent) is performing an action on the analog (i.e. taking a measurement). Measurements can even interfere with the rest of the simulation when the act of taking the referent wet-lab measurement is thought to affect the wet-lab experiment itself.

Mechanisms that generate phenomena during wet-lab and clinical experiments have different features. Darden describes five categories of features: 1) phenomenon, 2) components (e.g. entities and activities, modules), 3) spatial arrangement of components (e.g. localization, structure, orientation, connectivity, compartmentalization), 4) temporal aspects of components (e.g. order, rate, duration frequency), and 5) contextual locations (e.g. location within a hierarchy, location within a series) [12]. Each feature can be measured differently. For example, one may measure 1) how a system perceives the systems it influences, 2) the number and types of subcomponents involved in a mechanism, 3) location-dependent effects of a mechanism, 4) time-course changes in phenotype, and 5) relationship networks amongst subcomponents.

Typical journal reports of wet-lab experiments include a variety of measurement types spanning the above mechanism features to reach their conclusions; thus, reports based on VBEs are expected to do the same. However, conventional modeling methods (e.g. equation-based models) typically output one type of measurement. For example, many biological models output concentrations of species as a function of time [13]. However, what if a future use case requires a new type of measurement—for example, the number and location of dead cells, or the current cell cycle stage? Adding additional measurement types may require significant model re-engineering, or perhaps even an entirely new model. In the latter case, the different models may have little in common. For example, consider a pharmacokinetic model of the hepatic outflow profile of propranolol [14]. The model estimates propranolol kinetic parameters by fitting data to a two-phase physiologically based organ pharmacokinetic model [15]. When fit to in vivo data, the model produces an outflow profile that closely matches the targeted data. Say the modeler was also interested in modeling drug-induced hepatotoxicity. The pharmacokinetic model would have to be completely re-engineered to allow for measurements of toxicity. If the modeler were interested in location-dependent effects of toxicity, the problem would be compounded. The modeler could refer to (say) a Boolean network model of hepatotoxicity as in [16]; however, the two models would have little in common and their integration would be problematic.

An alternative approach is to make models inherently multi-attribute, which is an important VBE requirement. Multi-attribute analogs are designed to facilitate adding new measurement types at will based on use case requirements. By so doing, analog mechanisms also share Darden's five features of mechanisms, each of which can be measured differently. Including analogs of different measurements may require analogs of instrumentation used to obtain those measurements. Realizing multi-attribute models requires avoiding limiting oneself to one particular modeling formalism. For example, consider a model developed strictly using differential equations. While continuous measurements come more naturally to this modeling formalism, adding a discrete measurement like the number of dead cells would be problematic. In contrast, it is infeasible to add continuous measurements to a discrete modeling formalism like a Boolean network model. Analogs using agent-based methods are suitable for a wide range of measurement types.

2.6 Are all categories necessary?

Many potential analog components developed in part to satisfy the VBE vision might seem superfluous or unnecessary. For example, does one really need an analog of a pipette (to mimic the addition of volume) to capture the effects of a non-instantaneous bolus injection into culture media? The answer to this question (and others like it) is that it depends on use case and the granularity of available wet-lab validation data. If validation data is coarse-grain—say, the amount of drug remaining in a cell culture at each hour following injection for 24 hr—then no, an analog pipette is likely unnecessary to achieve all validation targets. However, if the validation data are sufficiently fine-grain—say, measurements of the amount of drug remaining each minute following dosing—then pipette effects (volume changes) may indeed be consequential. Regardless of the granularity, fine-grain analog components (i.e. analog pipette) should not be included in the simulation until falsification and iterative refinement "forces" the modeler to include it. This is true of all analog components—components that fulfill the VBE vision are no exception. However, as validation targets are achieved and the set of wet-lab validation data increases, increasingly fine-grained analog components and mechanisms will be required. The VBE vision supports that path toward finer granularity.

3. MEETING VBE REQUIREMENTS

The above categories entail many requirements, some of which are only briefly mentioned as their details fall outside the scope of this paper. We begin by discussing object-oriented programming, which (although not a requirement) we have empirically found to be both useful and natural for constructing analogs conforming to the VBE vision. We then describe two overarching requirements that are necessary for all or most VBE categories: concreteness and modularity.

3.1 Object-oriented programming

Our implementations of almost all parts of the VBE analogy rely on employing object-oriented programming (OOP). The VBE vision does not require any one particular programming paradigm; OOP is no exception. However, we have found OOP to be a useful, natural way to develop ABMs and analogs of the VBE categories. Specific to the VBE vision, we use OOP extensively for developing both Category 2: Living Counterparts and Category 3: Experiment Context. Experiment context typically comprises some spatial representation, e.g. grids (discrete or continuous), "well-mixed" compartments, or a directed graph. Analogs of living counterparts are often agents that reside within and can move about the spatial representations. Further, agent types often maintain a class hierarchy, which is facilitated using class inheritance. Additional aspects of OOP (including encapsulation, polymorphism, message passing, and composition) can be leveraged for developing VBEs but are outside the scope of this paper. Instead, we turn our focus to two additional requirements that OOP facilitates: concreteness and modularity.

3.2 Concreteness

For an analog to satisfy Category 1: Hypothesis, simulation mechanisms must be analogous to hypothesized mechanisms tested during referent experiments. Thus, the analog must be able to undergo hypothesis testing, which means that its mechanisms must be falsifiable. Falsifiability requires analog components and mechanisms to be sufficiently concrete when executed. Concreteness is the property of being real, actual, or standing for that which so exists. When an analog is implemented and instantiated in software, it is concrete. Concreteness is important because it is a prerequisite for causation and falsifiability, both of which allow concrete analogs to be scientifically useful. When analogs are concrete, their execution results in a causal cascade of events that provides a hypothesized, mechanistic explanation for the generated phenomena. This stands in contrast with mathematical descriptions of conceptual mechanisms (e.g. ordinary differential equation models), which can (when implemented and executed in software) reproduce desired phenomena, but do not necessarily provide a causal cascade or "story" of how those phenomena were generated in a particular simulation. An analog's causal cascade is analogous to that of the referent experiment. It can be measured and compared to the referent system to assess the strength of the analogy. When measurements are acceptably similar to the referent, the analog simulation then stands as a challengeable (falsifiable) hypothesis about causal events that may have occurred during referent experiments.

3.3 Modularity

Modularity is a multifaceted (sometimes overloaded) term in M&S. Here, we focus our modularity discussion on the ability to reuse and repurpose model components with minimal code refactoring, including the ability to exchange ("plug and play") modules during or between simulations [17]. Modularity is particularly important yet uniquely challenging in biological models because real biological systems are inherently heterogeneous. Cells do not have standardized "parts" in the same way that automobiles do; even individual cells within the same cell population can vary in phenotype [18]. At the same time, biology is modular in its own sense, in that many phenomena are functionally separable [19]. The nature of biology and the challenge presented call for modular biological models designed to satisfy a variety of use cases.

Modularity is crucial for achieving the VBE vision because each simulation recapitulates a particular (use case specific) experiment. Each experiment challenges a particular hypothesis, may be composed of a particular set of living counterparts, has a particular experiment context, requires particular experiment agents to follow a particular set of protocols, and includes a particular set of measurement types. Thus, all categories require modular components that can be recomposed ("plugged" and "unplugged") to simulate a variety of use case experiments. Further, components must be easily repurposed to simulate future, yet to be specified use case experiments.

For example, consider an analog mimicking a wet-lab experiment on a hepatocyte monolayer cell culture. The experiment context is a simple two-dimensional grid, with grid points containing cell analogs. During the simulation, an experiment agent adds analog drug to the analog culture, and then observes the culture by measuring the amount of drug remaining over time. Newer literature may involve more sophisticated wet-lab systems, in which, for example, cells are placed in a hepatocyte bioreactor that forces the flow of oxygen across cells. Further, cells in the bioreactor are organized into a three-dimensional scaffold. Mimicking this experiment using VBEs would entail a repurposed experiment context (say, a three-dimensional grid) and a new protocol for experiment agent (forced oxygen flow). These processes are greatly facilitated when components and mechanisms are modular.

Often, wet-lab or clinical experiments cannot directly measure attributes of interest, so they resort to using surrogate measures. Simulations must be able to do the same to ensure simulation results are commensurate with wet-lab validation data. For example, a hypothetical dead CELL may release DEATH MARKER that maps to a wet-lab biomarker for cell death. An observer agent can then measure the DEATH MARKERS as a surrogate measure for cell death, which can then be compared to validation data. Of course, simulations enjoy the benefit of the having more complete system information (including that which is experimentally inaccessible in the wet-lab) in addition to surrogate measures.

4. DEMONSTRATION MODEL: IN SILICO LIVER

The in silico liver (ISL) is an analog used to study mechanisms related to drug metabolism and toxicity. It is an object-oriented, discrete event ABM implemented in Java, utilizing the MASON multi-agent simulation toolkit [20]. A typical use case might be to inject analog drug into an analog liver, and then measure the amount of drug exiting the liver over time as it is slowly metabolized. Full details of the ISL structure, mechanisms, and use cases are detailed elsewhere [21-23]. Here, we focus specifically on how the ISL includes analogs of the five VBE categories, and how employing the VBE approach allows the ISL to mimic a variety of referent experiments.

4.1 Category 1: Hypothesis

Experiments on the ISL can mimic wet-lab drug metabolism and toxicity experiments. We hypothesize phenotypic overlap between an executed ISL and the referent system, which ranges from in vitro to whole animal (see "Category 3: Experiment Context" below). Phenotypic overlap is supported when measured ISL phenomena are acceptably similar to referent validation data. We have supported this hypothesis across a range of use cases, including experiments involving drug clearance, cellular toxicity, enzyme induction/elimination, inflammation, and immune-mediated enzyme regulation [21-23]. The area of phenotypic overlap increases as we add additional validation targets (see Figure 2B). Because the ISL satisfies many use cases spanning several experiment contexts, its phenotype space spans several distinct areas (purple areas in Figure 2B).

We also hypothesize mechanistic overlap between ISL mechanisms and real mechanisms occurring during referent wet-lab experiments. An experiment on an ISL is a concrete, mechanistic hypothesis. Concrete mechanisms produce causal cascades that can be compared to the referent. For example, an inflammatory mechanism can test the following hypothesis: inflammation occurs via cytokine production once a threshold level of inflammatory agent is exceeded; inflammation stops after reaching a threshold level of cytokine. We can instantiate this hypothesis using the following parsimonious, coarse-grain operating principles within ISL immune cell agents:

```
If # INFLAMMATORY AGENT $> inflammatory\_threshold$
    If # CYTOKINE $< cytokine\_threshold$
        Then produce CYTOKINE
```

This simple mechanism may achieve initial validation targets, but will eventually fail to achieve additional validation targets. When failure occurs, the simpler mechanistic hypothesis is falsified. The system is then refined to overcome that failure. The refined mechanism may either be an alternative (but equally fine-grain) mechanism, or may be a finer-grain version of the falsified mechanism. For example, the amount of CYTOKINE to produce may depend on the current level of CYTOKINE, instead of using a simple cutoff threshold. As the falsification-refinement-validation cycles continue, mechanistic overlap improves.

4.2 Category 2: Living Counterparts

The ISL consists of many concrete components that map to biological counterparts. A CELL object is the basic ISL agent. There are different types of CELLS, the most relevant of which is HEPATOCYTE. Important subcellular components are SOLUTE and ENZYME objects. SOLUTES are mobile objects that percolate through CELLS and other spaces. An important SOLUTE type is DRUG, which can act upon CELLS and be "metabolized" by ENZYMES. Additional SOLUTE types, which play roles in the examples below, include METABOLITE, DAMAGE PRODUCT, MARKER, INFLAMMATORY AGENT, and CYTOKINE, the functions of which depend on use case. ENZYMES exist within CELLS and can interact with (i.e. bind and metabolize) SOLUTES.

CELLS contain mechanisms that map to processes within cells. Mechanisms are sets of operating principles (typically stochastic) that govern interactions among objects. For example, a mechanism that maps to immune-mediate enzyme regulation can add or remove ENZYMES based on the presence of INFLAMMATORY AGENTS and CYTOKINES [23]. Details of operating principles are not relevant here; rather, the important aspect is that the analog mechanisms are concrete and designed to be biomimetic. Through iterative rounds of falsification-refinement-validation, mechanisms become increasingly biomimetic.

4.3 Category 3: Experiment Context

The ISL can be used to mimic several wet-lab experiments spanning in vitro to whole animal platforms. In doing so, it switches among several experiment contexts, each of which is used for various use case experiments; we describe three. The simplest, Experiment Context 1, maps to an in vitro hepatocyte culture [17]. The model structure consists of two stacked, rectangular grids (bottom of Figure 3). HEPATOCYTES are contained within the bottom grid, CELL SPACE. The top grid, MEDIA SPACE, maps to acellular culture media and cannot contain CELLS. The more complex Experiment Context 2 maps to experiments on an isolated, perfused liver (top of Figure 3, excluding MOUSE BODY). In this case, the structure spans multiple scales. It consists of a directed graph—or sinusoid network—of interconnected nodes and edges, which maps to a portion of a liver lobule. Each node is a SINUSOID SEGMENT, which consists of concentric, cylindrical grids containing hundreds of CELLS and an innermost CORE that maps to blood. Graph edges map to the direction of blood flow. The final Experiment Context 3 maps to a whole mouse (top of Figure 3). The previous structure is augmented with a MOUSE BODY analog that collects DRUG objects at the "bottom" of the sinusoid network and then slowly recirculates them back to the "top." When executed, ISL simulations using different experiment contexts result in new areas in phenotype space (purple areas in Figure 2B).

All three use cases utilize the modular HEPATOCYTE agent, as depicted in Figure 3. Modularity facilitates switching components among experiment contexts in different simulations, typically by simply adjusting a parameter value. On a more logistical level, use cases for Experiment Context 1 sufficiently differed from those of Experiment Contexts 2 and 3 that we developed the former analog as a separate program: the in silico hepatocyte culture (ISHC) [17]. Because HEPATOCYTE and other ISL components were modularized, it was straightforward to reuse them in the ISHC with minimal refactoring. Since then, the ISL and ISHC HEPATOCYTE have slightly diverged; namely, ISL HEPATOCYTE mechanisms have been repurposed to fit the additional finer mechanistic granularities needed to achieve new validation targets for Experiment Context 3 [23]; corresponding changes in ISHC HEPATOCYTE mechanisms were not needed. Thus, repurposed versions of modular components can be developed and tested in parallel among different experiment contexts. If desired, it is straightforward to merge changes via software repository tools.

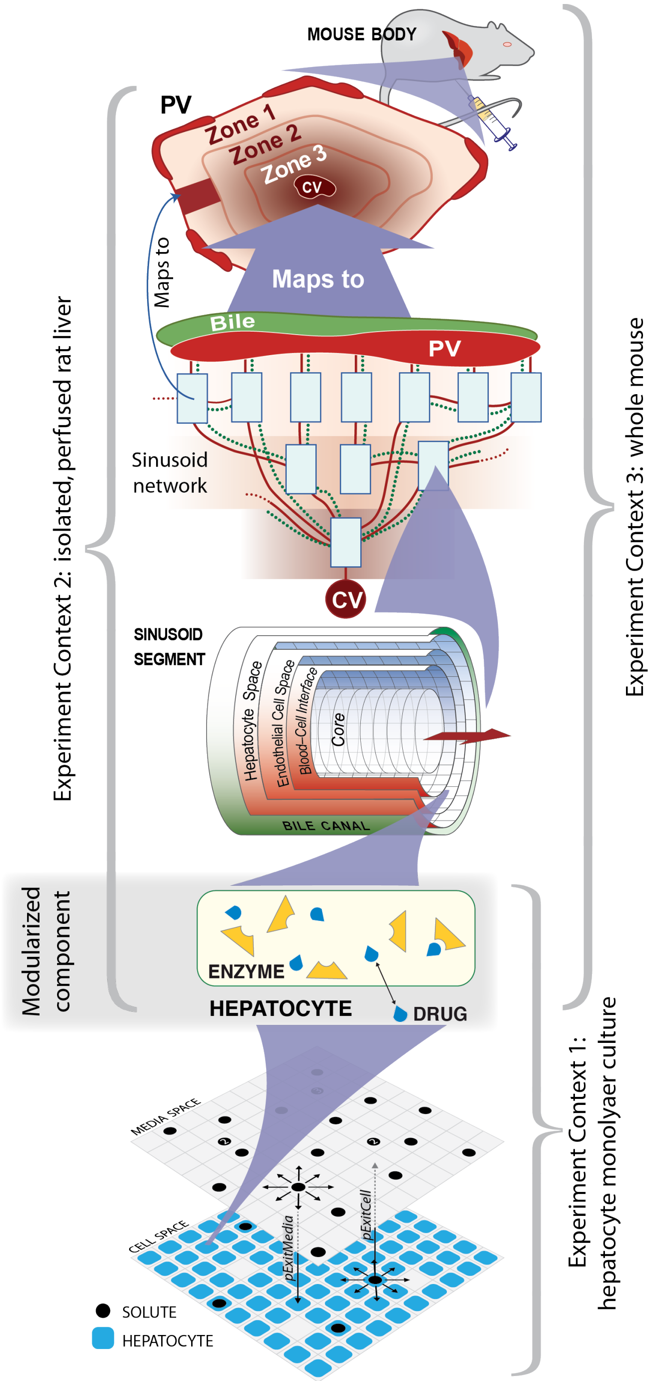

Figure 3. ISL structures and modularity for each experiment context. Large braces show which components are used for each experiment context. Experiment Context 1 (bottom) includes two grids, the bottom of which contains HEPATOCYTES. Experiment Context 2 (top) includes SINUSOID SEGMENTS (containing many HEPATOCYTES) organized into a sinusoid network. Experiment Context 3 is similar, but additionally includes MOUSE BODY. All three experiment contexts repurpose the modularized HEPATOCYTE agent.

4.4 Category 4: Experiment Agent

Experiment agents instantiate and execute ISL experiments. Many experiment agent activities are use case specific. Typically, ISL use cases involve the addition (injection) of DRUG at one or more points during the simulation, followed by time-course measurement of some set of attributes. The location in which DRUG is injected depends on the experiment context: namely, DRUG is injected into MEDIA SPACE, the top of sinusoid network, or MOUSE BODY for Experiment Context 1, 2, or 3, respectively. Further, experiment agents inject DRUG using different use case specific protocols that map to different routes of administration (i.e. oral, intravenous). When a new use case requires, say, transdermal DRUG delivery (i.e. because the referent wet-lab experiment administered drug transdermally), experiment agent activities will expand accordingly. Observer agents differ similarly in where (from which components) and how they take analog measurements.

Automation tools in wet-lab experimentation—from a simple parallel pipette to a sophisticated automated tissue culture system—are indispensable. Analogous tools for VBE are expected to greatly enhance productivity. Automation tools facilitate experiment agent activities. Currently, we utilize scripts to run batch simulations (including parameter sweeps), software objects to conduct within-simulation activities (e.g. inject DRUG, measure toxicity), and additional scripts to analyze simulation data (including falsifying/validating simulation measures against validation data). A near-term goal is to develop "smart" experiment agents that automate parts of the refinement process within the falsification-refinement-validation cycle. So doing may entail providing an experiment agent a set of candidate mechanisms, which the experiment agent then tests in parallel.

4.5 Category 5: Measurements

The ISL is not merely an I/O system; it does not simply output, for example, the concentrations of SOLUTES as a function of time. Rather, the ISL is measured by observer agents, analogous to how a wet-lab system is measured by a scientist, and that enables direct comparison of comparable features. For example, an ISL observer agent might measure toxicity by randomly sampling a population of CELLS in three separate regions of the sinusoid network. Observer agents perform a variety of analog measurements, similar to the many types of measurements made for a typical wet-lab publication. Use case dictates which measurements to take and how to take them. New use cases require configuring new analog measurements; modular, object-oriented components greatly facilitate this task.

Observer agents in the ISL and ISHC can perform a variety of measurements, according to use case. For example, the ISL can measure the time and location of CELL death events, CELL repair events as a function of lobule zone, and the amounts of species as a function of time; the ISHC can measure a dose-response curve between INFLAMMATORY STIMULUS and CYTOKINE response, and scatter plots between ENZYME levels and DRUG clearance [23]. We can add new observer agents as new use cases and availability of new data types dictate new measurement types.

5. CONCLUSION

The VBE approach is intended to enable biomedical M&S efforts to become more scientifically useful. It is easy to succumb to the urge to model "for the sake of modeling;" that is, to produce models that mimic some aspects of the referent but without clearly identifying model use case or context. Such models can be characterized as augmenting the scientific process. By focusing M&S efforts on use case and experimentation, VBEs are envisioned to become an integral part of the scientific process. This concept is illustrated in Figure 4. The scientific processes involved in in silico science parallel that of wet-lab science. As new insights give rise to new questions and hypotheses, one can alternate between cycles of wet-lab and virtual experimentation. The ISL and analogs like it have demonstrated use in support of the VBE vision. We anticipate the use of such analogs to further meaningful scientific progress, in part by adopting principles of the VBE vision.

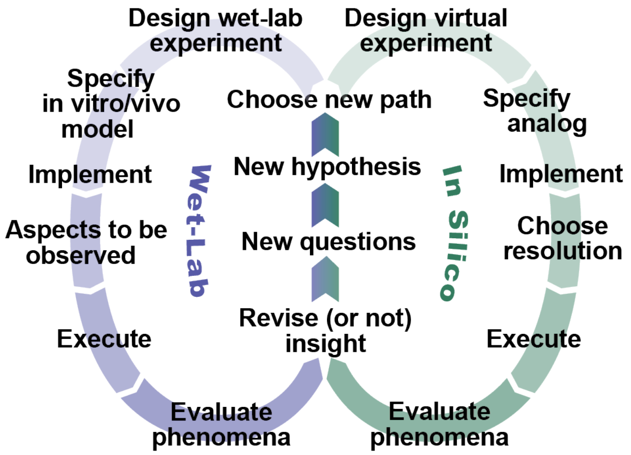

Figure 4. **Integrating wet-lab and in silico scientific protocols.** The two protocols are analogous at every step. Part of the VBE vision includes virtual experimentation as an integral part of the scientific method. When cycling through a particular scientific protocol (wet-lab or in silico), new insights lead to new questions and hypotheses can lead one to switch between wet-lab and in silico paths.

ACKNOWLEDGMENTS

This work was supported in part by the CDH Research Foundation, the Alternatives Research and Development Foundation, and the EPA (G10C20235).

6. REFERENCES

1. Yilmaz L (2015) Toward agent-supported and agent-monitored model-driven simulation engineering. In *Concepts and Methodologies for Modeling and Simulation* (pp. 3-18). Springer International Publishing.
2. Teran-Somohano A, Smith AE, Ledet J, Yilmaz L, and Oguztuzun H (2015) A model-driven engineering approach to simulation experiment design and execution. In *Proceedings of the 2015 IEEE/ACM Winter Simulation Conference* (pp. 2632-2643).
3. Teran-Somohano A, Dayibas O, Yilmaz L, and Smith AE (2014) Toward a model-driven engineering framework for reproducible simulation experiment lifecycle management. In *Proceedings of the 2014 IEEE/ACM Winter Simulation Conference* (pp. 2726-2737).
4. SMART Exchange (2012) Virtual experiments 2. Retrieved from http://exchange.smarttech.com

5. Yu J, Brown D, and Billett E (2007) Design of virtual tutoring agents for a virtual biology experiment. *European Journal of Open and Distance Learning* 11.
6. Ören TI and Yilmaz L (2009) Agent-directed Simulation. In *Agent-Directed Simulation and Systems Engineering*. Wiley-VCH.
7. Allen JW, Khetani SR, and Bhatia SN (2005) In vitro zonation and toxicity in a hepatocyte bioreactor. *Toxicological Sciences* 84(1): 110-119.
8. Hunt CA, Ropella GEP, Lam TN, Tang J, Kim SH, Engelberg JA, and Sheikh-Bahaei, S (2009) At the biological modeling and simulation frontier. *Pharmaceutical Research* 26(11): 2369-2400.
9. Zhang L, Athale CA, and Deisboeck TS (2007) Development of a three-dimensional multiscale agent-based tumor model: simulating gene-protein interaction profiles, cell phenotypes and multicellular patterns in brain cancer. *Journal of Theoretical Biology* 244(1): 96-107.
10. Jamali Y, Jamali T, and Mofrad MR (2013) An agent based model of integrin clustering: Exploring the role of ligand clustering, integrin homo-oligomerization, integrinligand affinity, membrane crowdedness and ligand mobility. *Journal of Computational Physics* 244: 264-278.
11. Rockich K and Blouin R (1999) Effect of the acute-phase response on the pharmacokinetics of chlorzoxazone and cytochrome P-450 2E1 in vitro activity in rats. *Drug Metabolism and Disposition* 27(9): 1074-1077.
12. Darden L (2006) *Reasoning in biological discoveries: Essays on mechanisms, interfield relations, and anomaly resolution*. Cambridge University Press.
13. Mallavarapu A, Thomson M, Ullian B, and Gunawardena J (2009) Programming with models: modularity and abstraction provide powerful capabilities for systems biology. *Journal of The Royal Society Interface* 6(32): 257-270.
14. Hung DY, Siebert GA, Chang P, Whitehouse MW, Fletcher L, Crawford,DH, and Roberts MS (2006) Hepatic pharmacokinetics of propranolol in rats with adjuvant-induced systemic inflammation. *American Journal of Physiology-Gastrointestinal and Liver Physiology* 290(2): G343-G351.
15. Siebert GA, Hung DY, Chang P, and Roberts MS (2004) Ion-trapping, microsomal binding, and unbound drug distribution in the hepatic retention of basic drugs. *Journal of Pharmacology and Experimental Therapeutics* 308(1): 228-235.
16. Jack J, Wambaugh JF, and Shah I (2011) Simulating quantitative cellular responses using asynchronous threshold Boolean network ensembles. *BMC Systems Biology* 5(1): 109.
17. Petersen BK, Ropella GEP, and Hunt CA (2014) Toward modular biological models: defining analog modules based on referent physiological mechanisms. *BMC Systems Biology* 8(1): 95.
18. McCullagh E, Farlow J, Fuller C, Girard J, Lipinski-Kruszka J, Lu D, Noriega T, Rollins G, Spitzer R, Todhunter M, and El-Samad H (2009) Not all quiet on the noise front. *Nature Chemical Biology* 5(10): 699-704.
19. Hartwell LH, Hopfield JJ, Leibler S, and Murray AW (1999) From molecular to modular cell biology. *Nature* 402: C47-C52.
20. Luke S, Cioffi-Revilla C, Panait L, and Sullivan K (2004) Mason: A new multi-agent simulation toolkit. In *Proceedings of the 2004 SwarmFest Workshop* (Vol. 8, p. 44).
21. Park S, Kim SH, Ropella GEP, Roberts MS, and Hunt, CA (2010) Tracing multiscale mechanisms of drug disposition in normal and diseased livers. *Journal of Pharmacology and Experimental Therapeutics* 334(1): 124-136.
22. Ropella GEP, Kennedy RC, and Hunt, CA (2012) Falsifying an enzyme induction mechanism within a validated, multiscale liver model. *International Journal of Agent Technologies and Systems (IJATS)* 4(3): 1-14.
23. Petersen BK, Ropella GEP, and Hunt, CA (2015) Virtual experiments enable exploring and challenging explanatory mechanisms of immune-mediated P450 down-regulation. *PLOS ONE* (submitted).