Similarity Measures and Validation in Automated Modeling

Dev A. Nag[1,2], Glen E.P. Ropella[1,2], C. Anthony Hunt[2]

[1]*Tempus Dictum, Inc.* [2]*University of California, San Francisco*

ABSTRACT

Developments in the fields of artificial intelligence (AI) and software engineering are making it feasible to automate much of the modeling process. Automated modeling promises to eventually reduce the cost and effort expended in model selection, optimization, and validation.

However, due to the diminished experimenter oversight in certain phases of this process, effective model validation plays an especially key role in ensuring successful results. Performing this validation in a consistent and reliable manner requires a structured formalism for the entire modeling process. We approach automated modeling from two sides. From one side, we explore a generic class of similarity measures used in model validation; from the other, we produce generic frameworks for the overall modeling and validation processes.

We first consider related work on model validation techniques with motivation. In the context of output-driven validation, we describe a generic vector time series similarity measure. We then provide a formal category theoretic framework for the modeling process and the specific role of similarity measures. Finally, we discuss future directions of research aimed at tying these two approaches together, providing a clear theoretical foundation for validation in the context of automated modeling.

INTRODUCTION

Automated modeling can be defined in many ways, but always involves shifting some of the responsibility for one or more phases of the model generation, verification, simulation, validation, and optimization processes to a computational entity, typically a software program (Rickel and Porter, 1994) (Farquhar, 1993) (Rickel and Porter, 1997) (Iwasaki and Levy, 1994) (Amsterdam, 1992). The goal of automated modeling is to reduce the cost, effort, and modeling skill required to create a model of a given utility in a problem domain, and to allow modelers to focus on building domain expertise rather than on particular modeling techniques.

Of these phases, model validation alone asks whether the model created serves its intended purpose; in the words of Schlesinger, model validation is ``substantiation that a computerized model within its domain of applicability possesses a satisfactory range of accuracy consistent with the intended application of the model." (Schlesinger, 1979) Model validation plays a key role in any modeling exercise, whether manual or automated; however, the nature of automated modeling requires that validation take on the additional role of implicit conceptual verification. In a manual modeling process, the experimenter implicitly verifies their own conceptual model when they generate it. In automated modeling, the experimenter must have a means of determining whether the modeling algorithm has sufficiently identified the system structures and processes required to satisfy the modeler's ultimate needs. Model validation provides this implicit verification.

Our ongoing research on biological component modeling (Ropella and Hunt, 2003) has included a variety of automated and manually generated models for validation with the reference data from *in situ* biological components (Ropella et al., 2005) (Ropella et al., 2003). We began by describing a family of generic similarity measures on real-valued scalar time series, showed that many measures in broad use were members of this family, and proved that this class of similarity measures could be decomposed into a pair of mathematical operators (Nag et al., 2004) that allowed for the selection of particular features in the time series data.

Here, we extend our generic similarity measure to vector time series in Euclidean space, encouraging a methodology for automated modeling across a broader scope of application. We then describe a category-theoretical schema of the modeling process, and then of the model validation process with respect to similarity measures. This top-down approach, in conjunction with our bottom-up work on real-valued scalar and vector time series measures, should eventually lead to a more complete and ontologically robust understanding of the automated modeling process.

MODEL VALIDATION AND SIMILARITY MEASURES

Model validation can occur on many levels using a variety of techniques, including animation, model comparison, degenerate tests, event validity, internal validity, multistage validation, and parameter variability-sensitivity analysis (Sargent, 1998) (Page et al., 1997) (Alexopoulos and Seila, 2000). Other authors have discussed the varying usefulness and proper application of these methods. However, the most common technique used in practice is historical data validation---that is, a comparison between output data from the model and output data from the referent system of interest (eg, historical system data). These data often take the form of time series, allowing for the application of automated series comparisons---namely, mathematical similarity measures (Ropella and Hunt, 2003). This regularization of the similarity measurement process is a key step in automated modeling (Ropella et al., 2005).

In a previous paper (Nag et al., 2004), a generic similarity measure for real-valued time series was described and applied to outflow concentration data taken from isolated perfused rat liver (IPRL) as well as from two prevailing models of the liver---the reference extended convection-dispersion model (Hung et al., 2001) (Hung et al., 2002) and the articulated FURM model (Ropella et al., 2003). The purpose of this generic similarity measure was twofold---to underscore the structural homologies of different similarity measures (such as the difference of means, F-index, and Piecewise Constant Approximation measures) in common use, and to provide a mechanism for generating new similarity measures with desired behavior in the context of automated modeling applications.

Clearly, there are many types of experimental data beyond real-valued scalar series. One useful generalization is to extend our measure to vector time series. These may arise in cases where the modeler is interested in simultaneous attributes of a reference system, or when a single attribute (such as position or velocity) may have multiple dimensions or degrees of freedom (Vlachos et al., 2002a) (Vlachos et al., 2002b) (Vlachos et al., 2003). We will discuss similarity measures in this context.

SIMILARITY MEASURES FOR VECTOR TIME SERIES

Vector time series (or 'spacetime' series) can be treated in two different manners. First, we can flatten the vector time series by mapping spatial dimensions into time dimensions (eg, a time series of *n* vectors in R^m would become a flattened series of *nm* elements (Karimi and Hamilton, 2000)), and use a parametrization of our earlier measure (Nag et al., 2004). However, if we maintain the separation of the space/time indices, we have more flexibility in defining measures that behave in certain experimenter-desired manners, such as emphasizing differences in one space dimension but minimizing differences in another space dimension, or throwing away time index information and treating the spatial values as unordered (but comparable) sets.

Reviewing our original time series measure, we begin with two vectors x_1 and x_2 in R^T, where each dimension in *T* represents a time index. We apply the linear characterization matrix *P*, which is a *TxN* matrix, thus creating two vectors y_1 and y_2 in R^N.

$$y = \mathrm{p}(x) = Px$$

We proposed a general distance metric (defined strictly as a metric) on these two characteristic vectors in R^N. This metric has an arbitrary number of terms J as well as arbitrary fractional exponents $0 < e_j \leq 1$ and nonnegative scalar coefficients α_j, thus implementing a 'spectral' Minkowski metric:

$$D(y, y') = \sum_{j=1}^{J} \left[\alpha_j \left(\sum_{n=1}^{N} |y_n - y'_n|^j \right)^{\frac{e_j}{j}} \right]$$

Or, in shorthand: *M(x, x') = D(p(x), p(x'))*.

Earlier (Nag et al., 2004), we showed through a series of theorems on distance metric space that *D(y,y')* fulfilled the four criteria for a true distance metric:

I. D(y, y') ≥ 0 (nonnegativity)
II. D(y, y') = 0 if and only if $y = y'$ (strict equality)
III. D(y, y') = D(y', y) (symmetry)
IV. D(y, y') ≤ D(y, y") + D(y', y") (sublinearity)

We propose a generalized distance measure for vector time series:

$$E(y, y') = \sum_{j=1}^{J} \left[\alpha_j \left(\sum_{t=1}^{T} \beta_t \sum_{i=1}^{N} \gamma_i |y_{i,t} - y'_{i,t}|^j \right)^{\frac{e_j}{j}} \right], \quad \alpha_j, \beta_t, \gamma_i > 0$$

In this similarity measure on two vector time series (in R^N and over a time range of length T), we have free parameters for the spectral terms (α_j), for the time index (β_t), and for the space index (γ_i). Given that these free parameters are all nonnegative, *E(y, y')* satisfies all four criteria of a distance metric.

In *E(y, y')*, both the β_t and γ_i coefficients are nonnegative, and the double sum (over *i* and *t*) ends up creating what is in effect a single sum with coefficients $\beta_t\ \gamma_i$ on a difference series $|y_{\{i,t\}}\text{-}y'_{\{i,t\}}|$ of length *TN*. Thus, *E(y, y')* is sublinear and a true distance metric.

As in Nag et al. (2004), we can now construct a similarity measure by composing this distance metric with a linear projective mapping on the original data. This mapping, defined by a linear characteristic matrix *Q* of dimension *N x N'*, acts on original vector elements (of dimension *N*) of the time series to create a series of dimension *N'* vectors.

$$y = q(x) = Qx$$

These resulting vectors (of dimension *N'*) become the arguments to *E(y, y')*, or in compressed form:

$$K(x, x') = E(q(x), q(x'))$$

where *K(x,x')* is the vector time series metric.

OUTPUT DATA VALIDATION

Our families of time series similarity measures focus on a quantitative comparison of output data. As described above, this is not the only form of model validation. However, validation based on output data is an especially well-accepted and intuitively satisfying manner of judging a model (Wright and Jr., 1997). This preference is not an accident, but a byproduct of our reliance on induction (Ropella et al., 2005).

In inductive modeling, data is the basis for generating models, which then in turn generate data. To borrow an analogy from linear algebra, models and data are duals of each other. Data itself can take many forms---time series, structural graphs, pictorial bitmaps, and so forth. However, there is at least one common division that applies to data collected in any context; the separation between data indices and data values.

Data indices refer to the background structure of a set of observables, while data values refer to the measured 'foreground' output of those observables. In an intuitive sense, data indices answer the question, 'When/Where are you looking?', while data values answer the question, 'What did you find there?'.

For example, in a time series, the data index might be a single, real-valued time parameter, while the data values might be a complex value (at each time index). In a two-dimensional bitmap, the data indices might be two positive integers (representing coordinates) while the data values might be three real values (representing the intensity of red, blue, and green at that point in the bitmap image).

Typically, we think of data as a given mathematical scalar/vector/series and measures as a function on that mathematical entity. From our high-level perspective, we can see that these are merely degenerate forms of the high-level case. Any data set can be represented as a function from index to output space, and measures are operators on those functions.

CATEGORY THEORY OF MODEL VALIDATION

With the benefit of this data decomposition, we would like to turn our attention to an abstract characterization of similarity measures on systems (and system models). One high-level overview of the modeling process was proposed by Robert Rosen (Rosen, 1991) (Rosen, 1999). In his framework, diagrammed in Figure 1, a referent system is 'encoded' into a corresponding model, behavior is 'inferred' within the model environment, and the predicted behavior is then 'decoded' or mapped into an analogous prediction for the referent system. Rosen used the language of category theory (also known as homological algebra (Jacobsen, 1985) (Jacobsen, 1989)) to identify this three part sequence of modeling with the internal 'causality' of the referent system; in other words, the modeling diagram 'commutes' for a valid model.

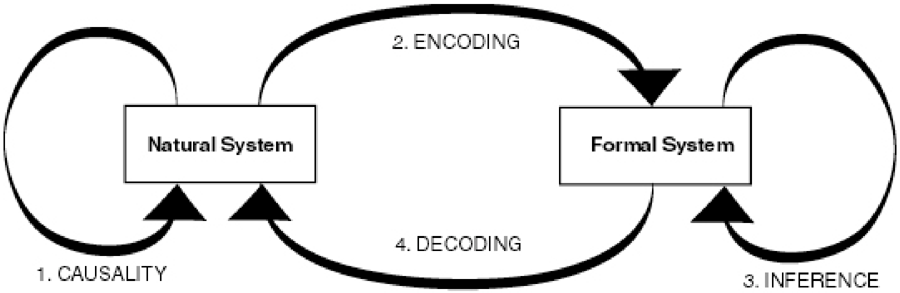

Figure 1. Rosen's Modeling Diagram

Category theory can be described as a general mathematical theory about structures, encompassing not only structures within given fields of mathematics such as algebra, analysis, and topology, but also abstract relationships between these fields in a clear and well-defined framework. As such, category theory is an ideal language for describing the act of modeling, a field with enormous scope across heterogeneous applications. Category theory shows that non-trivial statements about one field of mathematics often have precise analogies in other fields (Herrlich and Strecker, 1973); applying category theory to modeling can likewise show that structural insights in one specific type of modeling can be extended to other domains.

The most common form of these non-trivial, generalized statements are commutation diagrams. Commutation is essentially the equivalence of two distinct sequences of operators. In Rosen's modeling diagram, the causality within the natural, referent system is equivalent to the three operators of encoding, inference (whether theoretical or experimental), and decoding within the formal model. This commutation holds true for any type of modeling. Rosen showed, crucially, that the act of modeling was not merely a collection of unrelated techniques, but a unified process.

In this view, the act of modeling is really the act of relating two systems in a subjective way. That relation is at the level of observables; specifically, observables which are selected by the modeler as worthy of study or interest. Thus, any model of a referent system is primarily 'constrained' (in the sense of limiting degrees of freedom) by its observables.

Models are constrained by how their observables can be identified with the observables of a referent system. If there is no potential identification between observables, the model cannot be related to the system. Thus, in theory, a modeler could simply build a stand-alone model without any knowledge of the referent system and randomly assign model observables to system observables. In practice, of course, an experimenter is more likely to construct a model from the bottom-up; that is, modeling sub-components of the system and then identifying larger features of the model with the relevant features in the system.

In this methodology, the identification process is implied by the structure of the system and the interests of the modeler. The modeler will approach a natural system with some kind of driving question---how will the system evolve (prediction), how can the system be made to evolve in a certain way with an external impulse function (control), or what factors make the system behave the way it does (understanding)? Based on answers to these questions, the modeler will select certain system observables, and build a conceptual model of how the internal components of the system interact to generate these observables, following the process represented in Figure 2.

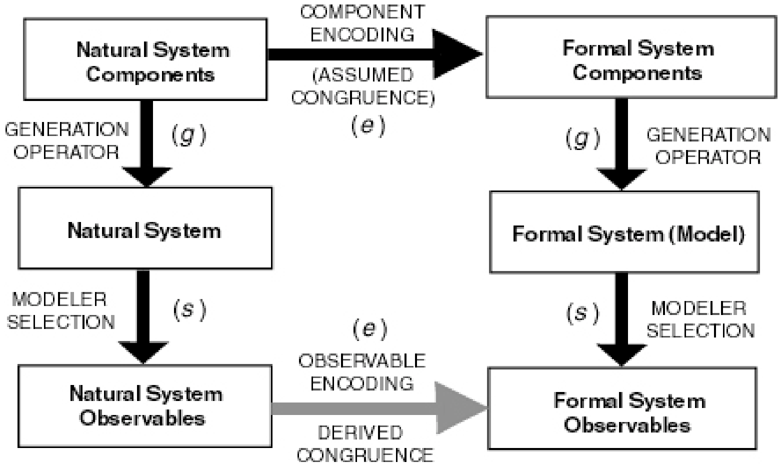

Figure 2. The model creation process

This generation operator *g* acts on the system components to produce the system, which is then acted on by the observable selection operator *s* to produce the system observables. Conversely, to create model observables, the modeler will then create model components (an act that can be identified with the encoding operator *e*) which are assumed to behave congruently with the system components, but without the benefit of a deeper substructure. In other words, the system and model components will act externally in an identifiable manner, but the internal generators of component behavior are not assumed to be congruent at all. Then, the generation structure operator is applied to the model components to build the final model, which will have observables that are naturally identifiable with system observables (due to the assumptions made by the modeler in building the model components).

In this sense, the modeler assumes congruence (between the system and the model) at the component level, and is rewarded with derived congruence at the model/system level. Expressed in category theoretic language, we have a commutation between the generation/selection operators and the encoding operators in the model creation process:

$$e \circ (s \circ g) = (s \circ g) \circ e$$

Ultimately, the modeler makes a series of informational trade-offs in order to build the model---by projecting a natural system down to observables, making a hypothesis about how internal components of a system interact to generate those observables, building model components that behaviorally mimic the system components without explicit structural justification, and then applying the system component generation operator to the model components to build a model with multiple levels of congruence to the referent system. With all of these assumptions at work, the validation mechanism for the model becomes critical; the final model must be compared to the system at one or more levels.

Similarity measures are essentially a quantification of a comparison of two entities. In our context, a similarity measure quantifies how similar a model is to a referent system. There are many ways in which to judge this similarity; for example, we might compare their description lengths, apparent complexity, or other internal attributes. However, in our inductive, data-driven environment, we wish to validate and compare models at the observational output level.

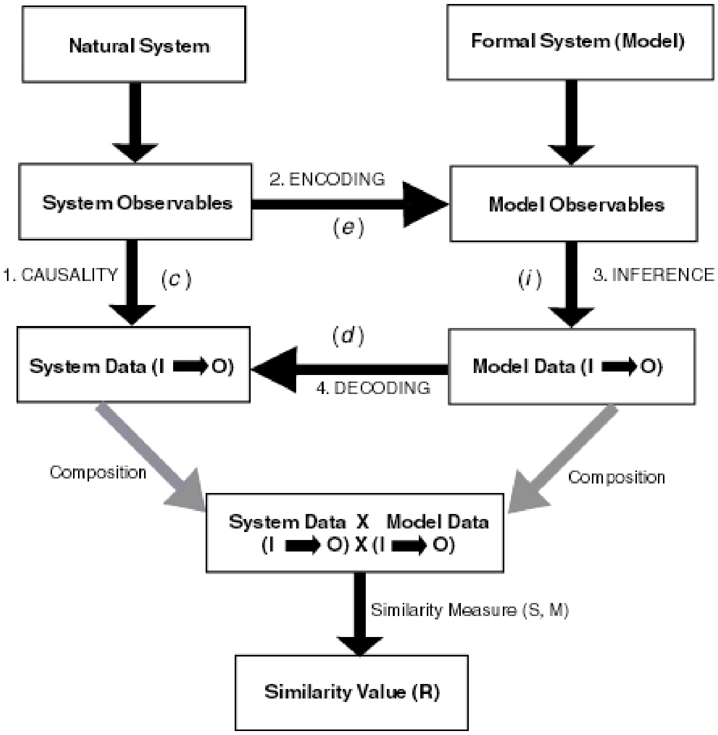

Figure 3. Schematic diagram for model similarity measurement

Observables are modeler-defined structural attributes of a system (or model); however, we need to measure them in order to create *observations*, or data. We then apply the similarity measure to the unordered pair of data (system and model). This measure will typically output a positive, real-valued number.

Our diagram in Figure 3 includes the Rosen commutation, and provides specific instantiations of each entailment. The causal processes within a natural system are encapsulated, from the perspective of an external observer, in the measurement operator from the system's observables to the system's observations (data). Encoding from the natural system to the formal system takes place at the level of 'percepts', or observables. Inference within the formal system is encapsulated within the model observables to model observations pathway. Finally, decoding is the function that takes a set of model observations back to system observations. In particular, similarity measures provide a functionalized comparison of this final decoding step, simplifying the differences to arrive at an easily communicated value.

We frame the two operators of 'causality' and 'inference' in a slightly different way from Rosen's original diagram. The diagram in Figure 1 makes these two operators seem like automorphisms---that is, operators which map an object to itself, in a sparse category class with only two objects. We delineate the components of the system by separating the system from its observables (and further from its observations). In other words, there are no automorphisms in our diagram, only morphisms. Our diagram commutes as well, but across a greater range of basic objects. In category theory, we preserve the commutation between the causality operator *c* and the encoding (*e*), formal inference (*i*), and decoding (*d*) operators:

$$c = d \circ i \circ e$$

This framework captures meaningful structure in the modeling process without explicitly designating any particular operator, whether compositional or validative, manual or automated. It represents a starting point for further decomposition and specification in particular domains of application, and a common formal language/foundation for automated modeling.

CONCLUSIONS

We approach model validation in the context of automated modeling from two sides--from the bottom-up, by describing computable similarity measures on time series data, and from the top-down, conceptual frameworks for the overall modeling and validation processes. Our primary contributions are the description of a generic similarity measure on vector time series data, extending an earlier generic similarity measure; and our theoretical framework, expressed formally in the language of category theory, for model generation (on a particular constructivist class of models) as well as output-driven model validation.

Ultimately, it is our hope that these two sides will be bridged. Similarity measures will continue to be extended and genericized, encompassing broader classes of specific measures (whether in practical use or merely possible). Formal frameworks for modeling will continue to become more richly endowed with specific modeling entities and concrete operations, elucidating the internal structure of previously indivisible processes. Future work may bring these approaches successively closer, providing a clear ontology, theoretical foundation, and practical methods for the nascent field of automated modeling.

REFERENCES

Christos Alexopoulos and Andrew Seila. Output analysis for simulations. In *Winter Simulation Conference*, pages 101–108, 2000. URL citeseer.ist.psu.edu/alexopoulos00output.html.

J. Amsterdam. Automated qualitative modeling of dynamic physical systems, 1992. URL citeseer.ist.psu.edu/amsterdam93automated.html.

Adam Farquhar. Automated modeling of physical systems in the presence of incomplete knowledge. Technical Report AI93-207, 1, 1993. URL citeseer.ist.psu.edu/farquhar93automated.html.

Horst Herrlich and George E. Strecker. *Category Theory*. Allyn and Bacon, Inc., Boston, MA, 1973.

D.Y. Hung, P. Chang, M. Weiss, and M.S. Roberts. Structure-hepatic disposition relationships for cationic drugs in isolated perfused rat livers: transmembrane exchange and cytoplasmic binding process. *Journal of Pharmacology and Experimental Therapeutics*, 297(2):780–789, May 2001.

D.Y. Hung, P. Chang, and K. Cheung. Cationic drug pharmacokinetics in diseased livers determined by fibrosis index, hepatic protein content, microsomal activity, and nature of drug. *Journal of Pharmacology and Experimental Therapeutics*, 301:1079–1087, 2002.

Yumi Iwasaki and Alon Y. Levy. Automated model selection for simulation. *In National Conference on Artificial Intelligence*, pages 1183–1190, 1994. URL citeseer.ist.psu.edu/iwasaki94automated.html.

Nathan Jacobsen*.* *Basic Algebra II*. W.H. Freeman and Co, New York, NY, 1989.

Nathan Jacobsen. *Basic Algebra I*. W.H. Freeman and Co, New York, NY, 1985.

Kamran Karimi and Howard J. Hamilton. Finding temporal relations: Causal bayesian networks vs. c4.5. In *International Syposium on Methodologies for Intelligent Systems*, pages 266–273, 2000. URL citeseer.ist.psu.edu/karimi00finding.html.

D.A. Nag, G.E.P. Ropella, and C.A. Hunt. Decomposition of similarity measures for time series analysis. Technical report, University of California, San Francisco, San Francisco, CA, 2004.

Ernest H. Page, Bradford S. Canova, and John A. Tufarolo. A case study of verification, validation, and accreditation for advanced distributed simulation. *Modeling and Computer Simulation*, 7(3):393–424, 1997. URL citeseer.ist.psu.edu/page97case.html.

Jeff Rickel and Bruce W. Porter. Automated modeling for answering prediction questions: Selecting the time scale and system boundary. In *National Conference on Artificial Intelligence*, pages 1191–1198, 1994. URL citeseer.ist.psu.edu/rickel94automated.html.

Jeff Rickel and Bruce W. Porter. Automated modeling of complex systems to answer prediction questions. *Artificial Intelligence*, 93:201–260, 1997.

G.E.P. Ropella and C.A. Hunt. Prerequisites for effective experimentation in computational biology. In *Proceedings of the 2003 IEEE Engineering in Medicine and Biology Society Conference*, Cancun, Mexico, September 2003. IEEE.

G.E.P. Ropella, D.A. Nag, and C.A. Hunt. Similarity measures for automated comparison of in silico and in vitro experimental results. In *Proceedings of the 2003 IEEE Engineering in Medicine and Biology Society Conference*, Cancun, Mexico, September 2003. IEEE.

G.E.P. Ropella, C.A. Hunt, and D.A. Nag. Using heuristic models to bridge the gap between analytic and experimental models in biology. In *Proceedings of the 2005 Spring Simulation Multiconference*, San Diego, CA, April 2005. SMSI.

Robert Rosen. *Anticipatory Systems: Philosophical, Mathematical, and Methodological Foundations*. Pergamon Press, New York, NY, 1985.

Robert Rosen. *Life Itself*. Columbia University Press, New York, NY, 1991.

Robert Rosen. *Essays on Life Itself*. Columbia University Press, New York, NY, 1999.

Robert G. Sargent. Verification and validation of simulation models. In *Winter Simulation Conference*, pages 121–130, 1998. URL citeseer.ist.psu.edu/sargent98verification.html.

Schlesinger et al. Terminology for model credibility. *Simulation*, 32(3):103–104, 1979.

M. Vlachos, G. Kollios, and D. Gunopulos. Discovering similar multidimensional trajectories, 2002a. URL citeseer.ist.psu.edu/vlachos02discovering.html.

Michail Vlachos, Dimitrios Gunopulos, and George Kollios. Robust similarity measures for mobile object trajectories, 2002b. URL citeseer.ist.psu.edu/vlachos02robust.html.

Michail Vlachos, Marios Hadjieleftheriou, Dimitrios Gunopulos, and Eamonn Keogh. Indexing multi-dimensional time-series with support for multiple distance measures. In *ACM KDD*, 2003. URL citeseer.ist.psu.edu/vlachos03indexing.html.

Samuel A. Wright and Kenneth W. Bauer Jr. Covalidation of dissimilarly structured models. In *Winter Simulation Conference*, pages 311–318, 1997. URL citeseer.ist.psu.edu/490575.html.