Validation of the Virtual Design Team (VDT) Computational Modeling Environment

Raymond E. Levitt, Stanford University
Terman Room 294, Mail Code 4020, Stanford, California, 94305-4020
Phone: (650) 723-2677; Fax: (650) 725-6014; e-mail: ray.Levitt@stanford.edu

Ryan J. Orr, Stanford University
CIFE, Building 550, Stanford, California, 94305-4020
Phone: (650) 575-9365; email: rjorr@stanford.edu

Mark E. Nissen, Naval Postgraduate School
555 Dyer Road, Code GB/Ni, Monterey, CA 93943-5000
Phone: (831) 656-3570; Fax: (831) 656-3407; e-mail: MNissen@nps.navy.mil

Introduction

The Virtual Design Team (VDT) computational model has been under development at Stanford University for almost two decades. VDT is an example of a new breed of computational organization models that are becoming increasingly realistic in structure, parameters and predictions of real-world organization experience. As the VDT model has matured, researchers in the VDT program have continued to take strides towards calibration of the model parameters and validation of the model representation and reasoning. This article describes this new breed of emerging computational models, discusses the theoretical foundations of the VDT computational model, describes the VDT modeling environment, and reviews three important strands of the continued long-term effort towards validation of the VDT model.

Computational Organization Theory Research

Computational organization theory (COT) is an emerging multidisciplinary field that integrates aspects of artificial intelligence, organization studies and system dynamics/simulation (e.g., see Carley and Prietula 1994). Nearly all research in this developing field involves computational tools, which are employed to support computational experimentation and theorem proving through executable models developed to emulate the behaviors of physical organizations (e.g., see Burton et al. 2002, Carley and Lin 1997, Levitt et al. 1999).

As the field has matured, several distinct classes of models have evolved for particular purposes, including: descriptive models, quasi-realistic models, normative models, and man-machine interaction models for training (Cohen and Cyert 1965, Burton and Obel 1995). More recent models have been used for purposes such as developing theory, testing theory and competing hypotheses, fine-tuning laboratory experiments and field studies, reconstructing historical events, extrapolating and analyzing past trends, exploring basic principles, and reasoning about organizational and social phenomenon (Carley and Hill 2001: 87).

Our research through the Virtual Design Team (VDT) is a branch of COT, built upon the planned accumulation of collaborative research over almost two decades to develop rich theory-based models of organizational processes (Levitt 2004). Using an agent-based representation (Cohen 1992, Kunz et al. 1999), micro-level organizational behaviors have been researched and formalized to reflect well-accepted organization theory (Levitt et al. 1999). Extensive empirical validation projects (e.g., Christiansen 1993, Thomsen 1998) have demonstrated the representational fidelity, and shown how the qualitative and quantitative behaviors of VDT computational models correspond closely with a diversity of enterprise processes in practice.

The VDT research program continues today with the goal of developing new micro-organization theory and embedding it in software tools that can be used to design organizations in the same way that engineers design bridges, semiconductors or airplanes—through computational modeling, analysis and evaluation of multiple virtual prototypes. Clearly this represents a significant challenge. Micro-theory and analysis tools for designing bridges and airplanes rest on well-understood principles of physics (e.g., involving continuous numerical variables, describing materials whose properties are relatively easy to measure), and analysis of such physical systems yields easily differentiable equations and precise numerical computing.

Of course, people, organizations and business processes differ from bridges, airplanes and semiconductors, and it is irrational to expect the former to ever be as understandable, analyzable or predictable as the latter. This represents a fundamental limitation of the approach.

Within the constraints of this limitation, however, we can still take great strides beyond relying upon informal and ambiguous theoretical descriptions of organizational behavior. For instance, the domain of organization theory is imbued with a rich, time-tested collection of micro-theories that lend themselves to qualitative representation and analysis. Examples include Galbraith's (1977) information processing abstraction, March and Simon's (1958) bounded rationality assumption, and Thompson's (1967) task interdependence contingencies. Drawing on this theory, we employ symbolic (i.e., non-numeric) representation and reasoning

techniques from established research on artificial intelligence to develop computational models of theoretical phenomena. Once formalized through a computational model, the symbolic representation is "executable," meaning it can be used to emulate organizational dynamics.

Even though the representation is qualitative (e.g., lacking the precision offered by numerical models), through commitment to computational modeling, it becomes semi-formal (e.g., most people viewing the model can agree on what it describes), reliable (e.g., the same sets of organizational conditions and environmental factors generate the same sets of behaviors) and explicit (e.g., much ambiguity inherent in natural language is obviated). Particularly when used *in conjunction with* the descriptive natural language theory of our extant literature, this represents a substantial advance.

Additionally, although organizations are inherently less understandable, analyzable and predictable than physical systems, and the behavior of people is non-deterministic and difficult to model at the individual level, it is known well that individual differences tend to average out when aggregated cross-sectionally or longitudinally. Thus, when modeling aggregations of people, such as work groups, departments, or firms, one can augment the kind of symbolic model from above with certain aspects of numerical representation. For instance, the distribution of skill levels in an organization can be approximated—in aggregate—by a Bell Curve; the probability of a given task incurring exceptions and requiring rework can be specified—organization wide—by a distribution; and the irregular attention of a worker to any particular activity or event (e.g., new work task or communication) can be modeled—stochastically—to approximate collective behavior. As another instance, specific organizational behaviors can be simulated hundreds of times—such as through Monte Carlo techniques—to gain insight into which results are common and expected versus rare and exceptional.

Of course, applying numerical simulation techniques to organizations is hardly new (Law and Kelton 1991). But this approach enables us to *integrate* the kinds of dynamic, qualitative behaviors emulated by symbolic models with quantitative metrics generated through discrete-

event simulation. It is through such integration of qualitative and quantitative models—bolstered

by reliance on sound theory and devotion to empirical validation—that our approach diverges

most from extant research methods, and offers new insight into the organizational dynamics.

VDT Computational Modeling Environment

The VDT computational modeling environment consists of the elements described in Table 1 and has

been developed directly from Galbraith's information processing view of organizations. This view of

organizations has two key implications (Jin and Levitt 1996). The first is ontological: we model knowledge

work through interactions of *tasks* to be performed; *actors* communicating with one another, and

VDT Model Element	Element Description
Tasks	Abstract representations of any work that consumes time, is required for project completion and can generate exceptions.
Actors	A person or a group of persons who perform work and process information.
Exceptions	Simulated situations where an actor needs additional information, requires a decision from a supervisor, or discovers an error that needs correcting.
Milestones	Points in a project where major business objectives are accomplished, but such markers neither represent tasks nor entail effort.
Successor links	Define an order in which tasks and milestones occur in a model, but they do not constrain these events to occur in a strict sequence. Tasks can also occur in parallel. VDT offers three types of successor links: finish-start, start-start and finish-finish.
Rework links	Similar to successor links because they connect one task (called the *driver* task) with another (called the *dependent* task). However, rework links also indicate that the dependent task depends on the success of the driver task, and that the project's success is also in some way dependent on this. If the driver fails, some rework time is added to all dependent tasks linked to the driver task by rework links. The volume of rework is then associated with the project error probability settings.
Communication links	Define paths for the flow of communications between actors who must communicate with each other to ensure that the choices made for interdependent tasks are compatible. The amount of communication between the positions is determined by the information exchange probability.
Task assignments	Show which actors are responsible for completing direct and indirect work resulting from a task.
Supervision links	Show which actors supervise which subordinates. In VDT, the supervision structure (also called the *exception-handling hierarchy*) represents a hierarchy of positions, defining who a subordinate would go to for information or to report an exception.

Table 1 VDT Model Elements and Element Descriptions

performing tasks; and an *organization structure* that defines actors' roles, and constrains their behaviors. Figure 1 illustrates this view of tasks, actors and organization structure. As suggested by the figure, we model the organization structure as a network of reporting relations, which can capture micro-behaviors such as managerial attention, span of control, and empowerment. We represent the task structure as a separate network of activities, which can capture organizational attributes such as expected duration, complexity and required skills. Within the organization structure, we further model various *roles* (e.g., marketing analyst, design engineer, manager), which can capture organizational attributes such as skills possessed, levels of experience, and task familiarity. Within the task structure, we further model various sequencing constraints, interdependencies, and quality/rework loops, which can capture considerable variety in terms of how knowledge work is organized and performed.

As also suggested by the figure, each actor within the intertwined organization and task structures has a queue of information tasks to be performed (e.g., assigned work activities, messages from other actors, meetings to attend) and a queue of information outputs (e.g., completed work products, communications to other actors, requests for assistance). Each actor processes such tasks according to how well the actor's skill set matches those required for a given activity, the relative priority of the task, the actor's work backlog (i.e., queue length), and how many interruptions divert the actor's attention from the task at hand.
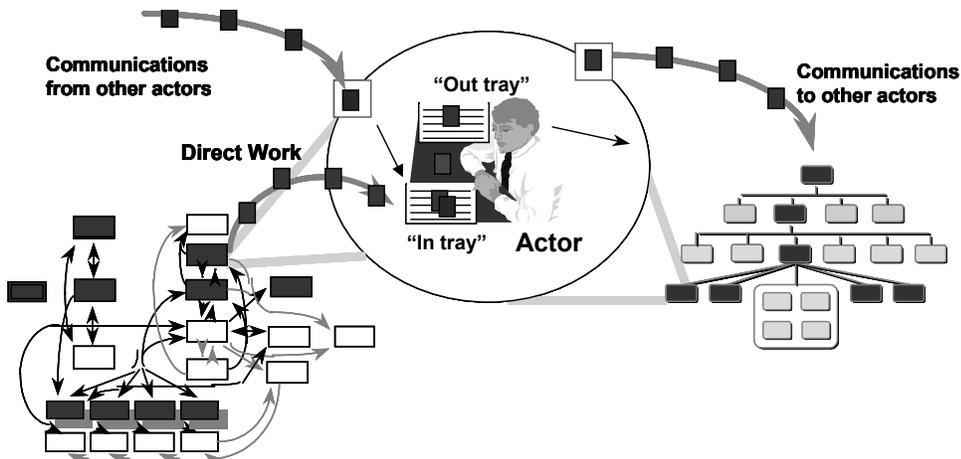

Figure 1   Information Processing View of Knowledge Work

5

The second implication is computational: *work volume* is modeled in terms of both *direct work* (e.g., planning, design, manufacturing) and *indirect work* (e.g., decision wait time, rework, coordination work)*.* Measuring indirect work enables the quantitative assessment of (virtual) process performance (e.g., through schedule growth, cost growth, quality).

Validation of VDT Computational Modeling Framework
Computational models of organizations can be viewed as providing a unique theoretical bridge between micro-organization theory and experience (formalized in theories of cognitive and social psychology) and macro organization theory and experience (formalized in theories of sociology, economics and political science). Computational emulation models of organizations like VDT embed well-accepted or "canonical" theories of micro-behavior in computational agents, instantiate a configuration of agents and tasks, and then explore the macro-outcome implications of this configuration. After they have been internally validated, organizational emulation models can be externally validated in two distinct ways—"intellective experiments" or "emulation experiments"—as shown in Figure 2.

- **Internal Validation:** Developers of computational models that attempt to emulate real-world behavior must first validate the micro-behavioral assumptions of their models, either by drawing on empirical micro-social science research findings, or by conducting their own ethnographies to describe and calibrate micro-behaviors. Once the behaviors have been captured and described, the implementation of the micro-behaviors in the model is internally validated (or "debugged") using very simple "toy problems" for which predictions can be simulated manually, or with simple spreadsheet calculations.

- **Two Kinds of External Validation**: Second, the predictions of the model for specific idealized or real world configurations of work processes and organizations must be externally validated against the predictions of macro theory, via theorem-proving or "**intellective experiments**"; or they must be validated against macro experience via "**emulation experiments**". A third, newer form of external validation involves cross

model docking in which pairs of models that address similar input and output variables are cross-validated by entering the same input data sets to both models and comparing their predicted outputs (Axtel et al. 1996). In the following section we discuss all three modes of external validation in more detail.

In the VDT research program, we carried out internal validation continuously as part of model development each time new micro-behaviors were defined and embedded in the model. For external validation, we performed emulation experiments through multiple Ph.D. theses and class term projects. Initially we used historical "back-casting" experiments to calibrate and validate the model's performance retrospectively in what might be viewed as curve-fitting exercises. Then we began to conduct "forecasting" experiments to validate the VDT model predictions prospectively. By 1996, managers at one of our test sites, Lockheed Martin, had developed sufficient confidence in VDT, following successful retrospective and prospective predictions for prior projects, that they began to use VDT predictions as a basis for intervening proactively in the design of their organizations. Validation of VDT through intellective experiments was initially viewed by our group as a more rigorous form of internal validation of the reasoning of VDT. Rigorous external validation of VDT through intellective experiments was carried out later, by us and others.
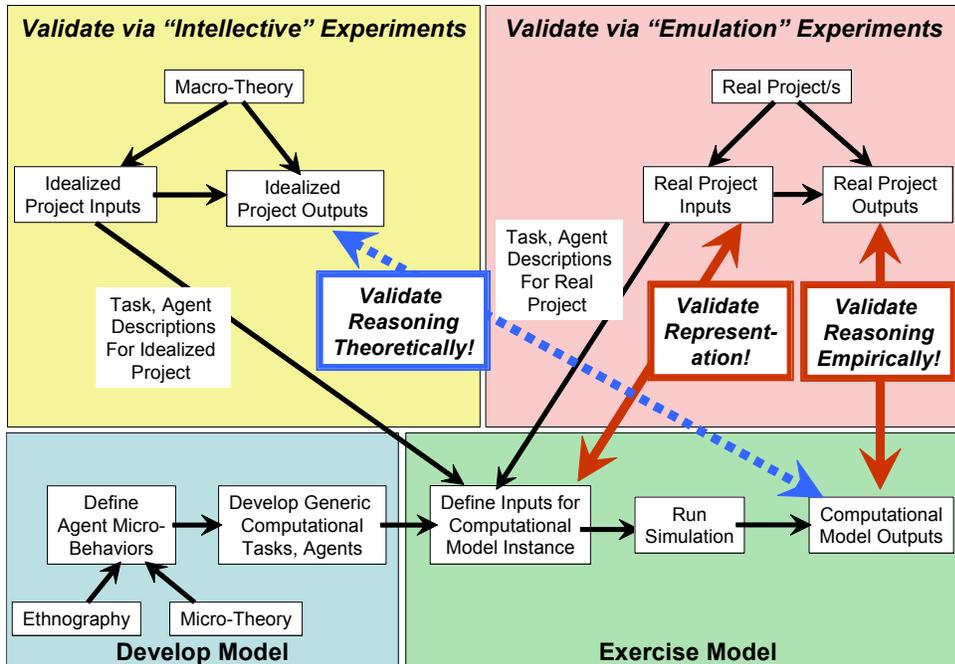
Figure 2   Two Kinds of External Validation for Computational Models of Organizations

Internal Validation of VDT

To validate VDT internally, we began by conducting two ethnographic experiments of several months each in which we observed project teams in real-world settings to gather data that would allow us to parameterize actor micro-behaviors. The first was a large engineering-construction firm engaged in design of an oil refinery with employees in two geographically distributed offices; the second was an engineering team engaged in the design of a satellite launch vehicle. In the first experiment, we relied on direct observation and participant interviews to gather data about information-processing, decision-making and communication behaviors such as: how an actor with multiple pending tasks prioritizes the pending tasks; to whom workers at each level of the organization would forward "exceptions" that they encountered in their tasks; the likelihood that actors would use particular communication technologies for posing exceptions, issuing instructions to resolve exceptions, etc.; and how long it would take an actor at the level of team participant, sub-team leader or project manager to process an "exception." In the second ethnography, we conducted interviews and observations, but were

also able to access time stamped electronic documents such as change orders, to see what paths they followed to the organization, and how long they resided at each step of the approval cycle. These electronic documents provided large amounts of data from which to develop average parameters for many of the internal variables in VDT. The students who carried out these ethnographies all had about 10 years of work experience in engineering project settings before starting their Ph.D. research, so that their intuition about many of these micro-behaviors turned out to be relatively accurate.

After validating and calibrating our initial assumptions about micro-behavior parameters for VDT through these two ethnographic experiments, we felt that the micro-behaviors embedded in VDT were relatively well validated. We then created simple "toy models" with one or two actors and a handful of tasks in which we could calculate the "correct" outcomes, to see whether we had implemented to behaviors correctly in the simulation model. Having done this, the next step was to carry out external validation experiments in which we compared the predictions of VDT models at the level of the project to real project outcomes in "emulation" mode, and to the predictions of organizational contingency theory in "theorem proving" or "intellective" simulation mode.

External Validation of VDT as an Emulator of Real Project Teams
To assess the extent to which VDT could emulate the performance of real-world project teams, we validated VDT in all of the ways shown in Figure 3 over a period of about six years (Thomsen et al. 1999). First, as described above, we performed internal validation of VDT on toy problems—problems with one or two actors and one or two tasks that were simple enough for the researcher to simulate outcomes manually—to be sure that we had correctly built-in the intended micro-behaviors in the agents. Next we developed a series of simple intellective experiments in which VDT model predictions were tested against macro predictions of

9

organizational contingency theory to test whether the overall model, which combines multiple

kinds of micro-behaviors, was producing accurate macro performance predictions. For example,
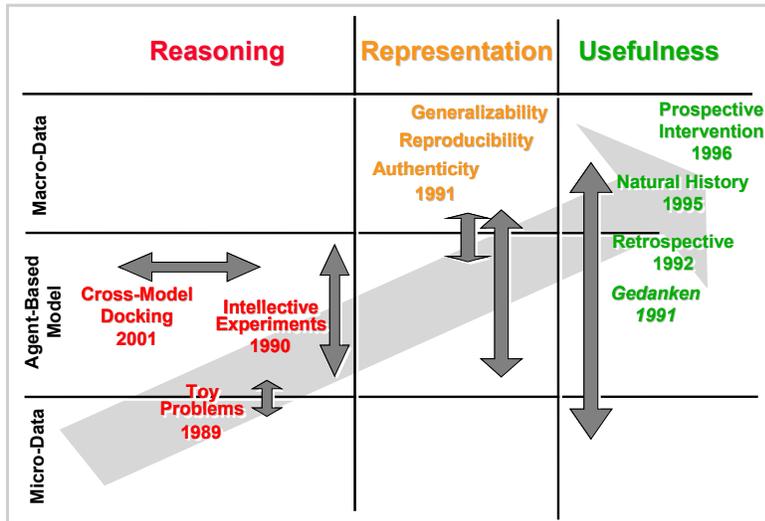

Figure 3   Validation Trajectory Used for VDT, and Proposed for Validating other Computational Emulation Models of Organizations *[Adapted from Thomsen et al. 1999]*

we modeled organizations engaged in highly uncertain tasks with varying centralization to check

that low centralization would produce better performance outcomes, in line with predictions of

contingency theory, noted in Burton and Obel's (2004) OrgCon model.

Next we tested the representational validity of the model by sending student teams out

to model real project teams in which we gathered data from managers and discussed the results

of model predictions with them. These experiments resulted in us renaming some of our model

variables to better match the natural idioms used by project managers. We tested the

reproducibility (inter-rater reliability) of the modeling framework by having multiple students

model the same organization and work process and comparing the models that they produced.

And we validated the generalizability of the modeling framework by using the framework to

develop models in multiple engineering, software and human resource project domains.

Finally, we validated the usefulness of VDT as an analysis tool by testing it on a series of

real-world projects in industries ranging across power stations, offshore oil, semiconductors,

aerospace, biotechnology, theme parks and consumer products. We started with a series of back-casting experiments in which we modeled the initial conditions of a completed project and compared the VDT predictions to the actual macro "as-built" project performance. Through a series of back-casting emulation experiments over five years, we calibrated the relative strengths of the various micro-behaviors in VDT to match the macro performance outcomes of the actual projects. Next, we began to use VDT to make real-time, prospective predictions of performance for projects that were in the very early stages of conception and development.

By 1995, this process was converging. It was now time to do some prospective prediction and validation of the model. In 1995-96, a group of VDT modelers from our team made a stunningly accurate prediction of exactly when and how the first Lockheed Launch Vehicle project organization would fail (Kunz *et al* 1999; Levitt *et al.* 1999). Following this successful prediction of a significant organizational failure, VDT had achieved a high enough level of external validity that a group of managers who had participated in the successful back-casting emulation experiments now had enough confidence in the model's predictions to start making proactive interventions in future projects, based on VDT predictions.

After almost eight years of development and testing, VDT had met the standard of being a "useful" analysis tool for designing project organizations!

External Validation of VDT through Intellective Experiments

Viewing VDT as a well-validated model of real project-oriented knowledge work, researchers began to use this computational modeling framework as a "virtual organizational test bench" to explore a variety of organizational questions starting in about 1997. The goals of such "virtual organizational experiments" conducted with VDT to date include: understanding the effects of geographical distribution of team members on project performance (Wong and Burton 2000); replicating empirical findings of the classical Bavelas and Leavitt (Leavitt 1951) communication experiments (Carroll & Burton 2000); exploring information processing and decision making
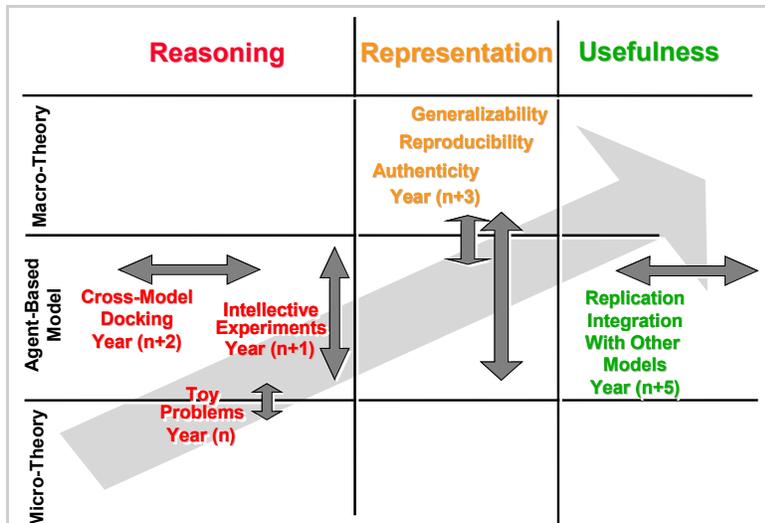
**Figure 4   Trajectory for Validating Computational Organizational Models
via "Theorem Proving "Intellective" Experiments** *[Adapted from Levitt 2004]*

behavior at the "edge of chaos" in organizations (Carroll & Burton 2000; Levitt *et al.* 2002); and

modeling the effect of cultural influences on project performance (Horii, Jin & Levitt 2005).

Based on the success of these intellective validation experiments, which we had not

anticipated when we began to develop VDT as an emulator of real world organizations, we

proposed an alternative trajectory for validating future computational emulation models in a

more traditional theorem-proving mode, as shown in Figure 4.

External Validation of VDT through Cross-Model Docking Experiments
In cross-model docking experiments, two models are used to represent the same set of data

that define an organization, work process and context; and the outcome predictions of the two

models are compared with one another. A variant of model docking involves embedding one

model within the other to create modeling and simulation at two levels of analysis. Starting in

about 2000, cross-model docking experiments were performed in which VDT was docked with:

OrgCon (Burton & Obel 2004); with the ORGAHEAD model of Kathleen Carley and her

collaborators (Louie et al. 2003); and with the BLANCHE knowledge network model of Noshir

Contractor and his colleagues (Pallazolo et al. 2002). In all cases, the predictions of VDT were

qualitatively similar to those of comparable models making similar predictions.

VDT's ability to predict the performance outcomes of alternative project organizations has thus been repeatedly validated externally against real-world project organizations, organizational contingency theory and other organizational simulation models.

Conclusion

The VDT model is an example of a computational organization model that has been validated through successive stages of internal validation, external validation and model cross-docking experiments. This validation effort has been underway for almost two decades and has involved more than 30 researchers in the Virtual Design Team program at Stanford University.

Once models like VDT have been validated to emulate accurately the qualitative behaviors of the field organizations, they can be used as a "virtual test bench," to examine a multitude of cases (e.g., many more and diverse than observable in practice) under controlled conditions (e.g., repeating the same events multiple times, manipulating only one or a few variables at a time through repeated trials, stopping the action for interpretation). As models like VDT become increasingly validated, they offer great promise for the development, enhancement and falsification of organization theory.

References

Axtel, R., R. Axelrod, J.M. Epstein, M.D. Cohen. 1996. Aligning Simulation Models: A case study and results. *Computation and Mathematical Organization Theory* **1**(2) 123-141.

Burton, R.M., J. Lauridsen, B. Obel. 2002. Return on Assets Loss from Situational and Contingency Misfits. *Management Science* **48**(11) 1461-1485.

Burton, R.M., B. Obel . 2004. *Strategic Organizational Diagnosis and Design: The Dynamics of Fit*, 3rd Edition New York, Kluwer.

Burton, R., B. Obel. 1995. The Validity of Computational Models in Organization Science: From Model Realism to Purpose of the Model. *Computational and Mathematical Organization Theory* **1**(1) 57-71.

Carley, K.M., Z. Lin. 1997. A Theoretical Study of Organizational Performance under

Information Distortion. *Management Science* **43**(7) 976-997.

Carley, K.M., M.J. Prietula . 1994. *Computational Organization Theory*, Hillsdale, NJ, Lawrence Erlbaum Associates.

Carley, K., V. Hill . 2001. Structural Change and Learning Within Organizations. In: Lomi, A. and Larsen, E.R., (Eds.) *Dynamics of Organizational Computational Modeling and Organization Theories*, Cambridge, MA, The MIT Press,

Carrol, T., R.M. Burton. 2000. Organizations and Complexity: Searching for the Edge of Chaos. *Computational & Mathematical Organization Theory* **6**(4) 319-337.

Christiansen, T.R. 1993. Modeling Efficiency and Effectiveness of Coordination in Engineering Design Teams. Stanford University.

Cohen, G.P. 1992. The Virtual Design Team: An Information Processing Model of Design Team Management. Stanford University.

Cohen, K.J., R.M. Cyert . 1965. Simulation of Organizational Behavior. In: March, J.G., (Ed.) *Handbook of Organizations*, Rand McNally,

Galbraith, J.R. 1977. *Organizational Design*, Addison-Wesley.

Horii, T., Y. Jin, R. Levitt. 2005. Modeling and Analyzing Cultural Influences on Project Team Performance. *Computational and Mathematical Organization Theory* **10** 305-321.

Jin, Y., R.E. Levitt. 1996. The Virtual Design Team: A Computational Model of Project Organizations. *Journal of Computational and Mathematical Organizational Theory* **2**(3) 171-195.

Kunz, J.C., R.E. Levitt, Jin Y. 1999. The Virtual Design Team: A computational simulation model of project organizations. *Communications of the Association for Computing Machinery* **41**(11) 84-92.

Law, A.M., D. Kelton . 1991. *Simulation Modeling and Analysis (2nd Ed.)* , New York, NY, McGraw-Hill.

Leavitt, H.J. 1951. Some effects of certain communication patterns on group performance. *Journal of Abnormal and Social Psychology* **46**(1) 38-50.

Levitt, R. E., Fyall, M., P. Bjornsson, W. Hewlett III , and Casebolt, T. 2002. When Information Flows in Organizations Becomes Turbulent: Toward an Organizational Reynolds Number. Conference on Agent-Based-Modeling in the Social Sciences. UCLA Lake Arrowhead Conference Center.

Levitt, R. 2004. Computational Modeling of Organizations Comes of Age. *Journal of Compuational and Mathematical Organization Theory* **10**(2) 127-145.

Levitt, R.E., J. Thomsen, T.R. Christiansen, J.C. Kunz, Y. Jin, C. Nass. 1999. Simulating Project Work Processes and Organizations: Toward a Micro-Contingency Theory of Organizational Design. *Management Science* **45**(11) 1479-1495.

Louie, M. A., Carley, L. M., Haghshenass, L., Kunz, J. C., and Levitt, R. E. 2003. Model Comparisons: Docking ORGAHEAD and SimVision. NAACSOS Conference. Pittsburgh, PA.

March, J.G. and Simon Herbert A. 1958. *Organizations*, New York: Wiley.

Palazollo, E. T., Ghate A., Dandi, R., Mahalingam, A., Contractor, N, and Levitt, R. (2002) Modeling 21st Century Project Teams: Docking Workflow and Knowledge Network Computational Models. Paper presented at the CMU conference. Pittsburgh, PA.

Thompson, J.D. 1967. *Organizations in Action*, New York, McGraw-Hill.

Thomsen, J. 1998. The Virtual Team Alliance (VTA): Modeling the Effects of Goal Incongruency in Semi-Routine, Fast-Paced Project Organizations. Stanford University.

Thomsen, J., R.E. Levitt, J.C. Kunz, C.I. Nass, D.B. Fridsma. 1999. A Trajectory for Validating Computational Emulation Models of Organizations. *Journal of Computational & Mathematical Organization Theory* **5**(4) 385-401.

Wong S.S., R.M. Burton. 2000. Virtual Teams: What are their Characteristics, and Impact on Team Performance? *Computational and Mathematical Org.Theory* **6**(4) 339-360.