

## THINKING STRATEGICALLY ABOUT THINKING STRATEGICALLY: THE COMPUTATIONAL STRUCTURE AND DYNAMICS OF MANAGERIAL PROBLEM SELECTION AND FORMULATION

MIHNEA MOLDOVEANU\*

Marcel Desautels Centre for Integrative Thinking, Rotman School of Management, University of Toronto, Toronto, Ontario, Canada

*A new model of managerial problem formulation is introduced and developed to answer the question: 'What kinds of problems do strategic managers engage in solving and why?' The article proposes that a key decision metric for choosing among alternative problem statements is the computational complexity of the solution algorithm of alternative statements. Managerial problem statements are grouped into two classes on the basis of their computational complexity: P-type problems (canonically easy ones) and NP-type problems (hard ones). The new model of managerial cognitive choice posits that managers prefer to engage with and solve P-type problems over solving NP-type problems. The model explains common patterns of managerial reasoning and decision making, including many documented 'biases' and simplifying heuristics, and points the way to new effects and novel empirical investigations of problem solving-oriented thinking in strategic management and types of generic strategies, driven by predictions about the kinds of market- and industry-level changes that managers will or will not respond to. Copyright © 2009 John Wiley & Sons, Ltd.*

### INTRODUCTION

What kinds of problems do strategic managers engage with and try to solve, and why those and not others? How much should and how much *do* managers think about a strategic decision problem? Under what conditions do they *switch* among different problem statements to make sense of a given predicament? I introduce a model of strategic problem formulation based on distinctions drawn from computational complexity theory (reviews can be

found in Garey and Johnson, 1979; Cormen, Leiserson, and Rivest, 1990), and contribute a canonical representation and taxonomy of managerial problems based on their computational complexity to create a framework for addressing such questions. I develop a model of cognitive choices that managers implicitly make among alternative problem complexity classes. The model uses a well-known, established quantization of problem complexity measures to build a lexicographic model of managerial problem formulation. *P*-hard, or polynomial-time hard problems, are those with solution algorithms that require a number of operations that is at most a polynomial function of the number of independent variables and constraints. By contrast, *NP*-hard problems, or nondeterministic polynomial-time-hard problems, have solution algorithms that require a number of operations that is a greater than any polynomial (e.g., exponential) function of the problem's variables and constraints.

---

Keywords: managerial problem solving; strategy formulation; complexity

\* Correspondence to: Mihnea Moldoveanu, Marcel Desautels Centre for Integrative Thinking, Rotman School of Management, University of Toronto, 105 St. George Street, RSM 473, Toronto, Ontario, Canada, M5S 3E6.

E-mail: micamo@rotman.utoronto.ca

The model of problem selection put forth in this article posits that managers, *ceteris paribus*, prefer to conceptualize their predicaments in terms of *P*-hard problems over conceptualizing them in terms of *NP*-hard problems. The model allows researchers to make sense of a wide-ranging literature on cognitive simplifications, and to make predictions about the kind of problems managers attempt to solve, the environmental variables they pay attention to, and the types of strategies they consequently adopt.

### **Background: shortcomings of current models of problem selection and formulation**

Following the example of Simon and his co-workers, (Simon, 1947; March and Simon, 1958; Cyert and March, 1963), researchers have looked at the phenomena of strategy formulation and selection using the vocabulary of information processing operations. In this framework, the organization is represented as processing information from its environment so as to produce adaptive patterns of behavior, whose instantiation, reliability, and success are constrained by the bounded rationality of organizational actors (Simon, 1996; March, 1994). The resulting picture of the organization is that of a large device for registering new information and carrying out computations that 'solve problems' (March and Simon, 1958). Early work guided by this model simulated the processes by which organizations solve problems, make decisions, and address 'issues' as haphazard and unstructured (Cohen, March, and Olsen, 1972) and formulated preliminary process measures, such as the difficulty of making an organizational decision, but stopped short of formalizing organizational information processing to the point where specific decisions and problems can be mapped into different levels of difficulty or complexity. Accordingly, I focus here specifically on the question of problem formulation and selection: If a problem statement (i.e., a mismatch between an identified 'current' state of the world and a 'desired' state of the world, coupled with a desire to get to the desired state of the world subject to some constraints [Newell and Simon, 1972; Simon, 1996; Newell, 1990]) is what guides computation, then how are problem statements themselves chosen or generated? Are there general patterns of problem selection that would give greater

explanatory punch to the thrust of information-processing explanations of managerial problem-solving behavior? I introduce a way of thinking about such patterns and argue that it is possible to make predictions about the problems that managers engage with and choose to solve, the strategies they will come up with, and the variables they pay attention to. In this way, I connect the problem selection and formulation process to models of the expected span, structure, and dynamics of managerial attention (Ocasio, 1997), which has also been studied in the information processing tradition.

Apart from the formal information processing approach to representing organizations, the processes of problem diagnosis (Mintzberg, Raisinghani, and Theoret, 1976), definition (Kilmann and Mitroff, 1979), or discovery (Pounds, 1969) have been studied empirically and analytically for some time (Taylor, 1975; Lyles and Mitroff, 1980; Nutt, 1984; Cowan, 1986) with an eye to identifying either correlations between environmental conditions and structural elements of a problem statement, or typical problem-formulation behaviors that together would amount to a model of problem formulation or selection. Because 'problems' are difficult to define (Agre, 1982) as a class of entities, it has been difficult to speak of problems in general, without referring to either a specific set of problems (Nutt, 1984), or to a specific set of processes that qualify as 'problem-formulation behavior' (Kilmann and Mitroff, 1979). The complexity of a problem has been difficult to capture by empirically minded researchers, as attested to by the difficulties of Volkema's (1988) attempts to characterize the complexity of problem statements using the number of independent logical clauses in the problem statement. These difficulties stem from the fact that we can construct 'simple' many-clause problem statements and complicated few-clause problem statements, which renders the use of clause-based complexity measures ambiguous. The present work makes describing the problem formulation process precise by introducing a complexity measure along which strategic problems can be described and separated into different classes, making it possible to discuss problem formulation in terms of complexity class selection.

Complexity considerations have not been irrelevant to researchers' study of managerial cognition. Managers have been argued to be cognitive simplifiers (Schwenk, 1984)—as are lay persons more

generally (Rokeach, 1960; Tversky and Kahneman, 1986; Hogarth, 1975, 1980). Cognitive simplification processes, such as anchoring, escalating cognitive commitment to a previously chosen course of action, prior hypothesis bias, limited and anchor-biased search among lists of alternatives, have been identified as sources of 'skew' in managerial information processing, presumably relative to a non-skewed, 'ideal' mind. However, the question of finding an underlying mechanism for all or most of the identified simplification processes is open. I argue that computational complexity-guided problem choice is a viable candidate. The question I take up is: Can we reach more deeply than a description of surface heuristics and come up with a model of managerial cognition that is explanatory of the cognitive behavior of managers and amounts to more than a collection of stylized facts? To answer this, I deploy a model of cognitive operations, which explains the simplifying moves observed by researchers by introducing a space in which managerial problems can be described, and shows how various simplification maneuvers can be understood as complexity-driven choices among and within regions of this space.

The model of managerial cognitive choice among alternative problem statements developed here is relevant to the classification of different managerial strategies; it helps state conditions under which we can expect certain strategies to exist in the first place, such as those whose pursuit requires managers to have antecedently thought a complex problem through to an optimal solution. Researchers who developed new taxonomies of firm-level strategies (Chrisman, Hofer, and Boulton, 1988; Hambrick, 1983; Miller and Friesen, 1978, for instance) have focused on the presence (rather than the *absence*) of strategies and strategy types, as Inkpen and Choudhury (1995) point out. This 'presence bias' in strategy research—together with the multiplicity of possibly equally well corroborated strategy types—raises questions for the researcher interested in more realistic insight into managerial thinking and reasoning: How does one choose among the 'presence' and 'absence' hypotheses when searching for and classifying strategies? How does one choose among various possible taxonomies of strategies, given all of them are well corroborated and have led to fruitful programs of empirical research?

A methodological answer that comes from the computational study of 'mind' (Newell, 1990;

Simon, 1996) is a form of psychological realism, which posits that we want to seek strategy typologies that are maximally plausible or explicable from a psychological perspective (but, additionally, are based on algorithmically tractable representations of mental processes, that are themselves amenable to machine implementation (Turing, 1997). Rather than treating managers *as if* they were intendedly and strategically (albeit boundedly) rational, psychological realism encourages researchers to investigate in greater detail the cognitive and epistemological states of their subjects. The approach proposed here is a contribution to this psychologically realistic research program in the study of managerial cognition. It is based on the introduction of a space in which managerial problem solving and problem formulation processes can be represented in a psychologically intuitive fashion.

A direct intellectual precursor of the approach of this study is the behaviorist and cognitive reappraisal of game theory and game-theoretic results brought about by experimental economists. Classical game-theoretic analyses of strategic management (Kreps, 1990; Gintis, 2000) assume that agents are rational, know that they are rational, and reasonably assume that others are rational as well, in spite of the empirically unrealistic 'logical omniscience' that such assumptions require agents to possess, which entails that all agents know or believe all of the logical consequences of their current beliefs. By contrast, behavioral game-theoretic analyses test assumptions about individual rationality piecemeal and craft parametric models of individual cognition and behavior (Camerer, 1997). They are concerned with what is or is not the case in strategic and competitive scenarios, rather than with providing an explanation of strategic behavior based on assumptions that are reasonable, but, likely to be empirically false. The model put forth in this study bridges between models of reasoning developed by experimental game theorists and classical game-theoretic approaches, by providing a general way of thinking about the simplification processes that take the modeler from 'fully rational' to 'boundedly rational' models of strategic thinking. It is based on the insight that knowledge by a manager of some or all of the variables of a competitive game (players, strategies, payoffs) is not—as traditional game theorists assume—necessarily accompanied by an explicit

understanding of these parameters as the parameters of a *particular* game (e.g., ‘the centipede game’ or, ‘Cournot Oligopoly’) whose logic they will necessarily follow through to the discovery of an equilibrium. The manager’s problem selection process determines whether or not the game structure used by a researcher to explain an interactive situation will also be the game played or understood by the manager who lives that situation. Conceptualizing a situation as a particular game represents a precommitment to conceptualizing unstructured situations in terms of a particular class of problems.

### Outline of the article

I begin by showing how managers make implicit choices about the level of logical depth to which they take their thinking, and introduce the notion of thinking strategically about thinking strategically, that is, of choosing the computational complexity of one’s strategic thinking processes in advance of engaging with a problem. After making the notions of ‘easy’ and ‘hard’ problems precise, using the language of theoretical computer science (Garey and Johnson, 1979; Cormen *et al.*, 1990), I posit that managers make implicit cognitive choices that reveal a preference for ‘easy’ over ‘hard’ problems. I marshal substantial prior evidence to corroborate this claim, and show how large numbers of managerial problems can be represented in the form of problem statements whose computational complexity can be measured. I use the new principle of revealed cognitive preference, which allows us to explain several important and currently isolated findings about strategic decision making. Finally, I explore the implications of this revealed preference for the span, scale, and scope of managerial attention and for the type of strategic responses we may expect, and show how the model allows us to ask new, empirically tractable questions about managerial attention, strategy selection, and strategic behavior.

### THINKING STRATEGICALLY ABOUT THINKING STRATEGICALLY: PROBLEM FORMULATION AS THE OUTCOME OF A COGNITIVE CHOICE

When conceptualizing an unstructured predicament in terms of a structured problem statement,

managers make *cognitive* choices over how or how much to think. For example, a strategic manager in a firm operating in an oligopoly with undifferentiated inputs called upon to make a decision about how many units of a good to produce the following quarter can think of his or her predicament as: a competitive game played among rational, self-maximizing agents; a competitive game played among a rational agent (himself) and an irrational one (his competitor); an opportunity to build a collusive relationship; or a multidimensional optimization problem in which his competitors’ strategic response plays only a small part—to name only a few options. A manager makes a cognitive choice when he or she selects a problem statement that guides his thinking about his own predicament, when, for instance, he chooses to think about the pricing decision he is about to make using a Cournot duopoly pricing model, rather than some alternative model that may be warranted by the situation at hand.

A precise way to characterize problem statements is by examining the algorithms that generate satisfactory solutions to them (Cormen *et al.*, 1990; Papadimitriou, 1994) because the complexity of the process by which these algorithms are used to generate a solution can be measured. The importance of being able to produce such a measure is that we can consider the complexity of alternative problem formulations as a decision variable, one that managers attempt to economize on. This, in turn, provides us with a metric for the classification of managerial problem statements and the strategies that they produce.

To make both the concepts of cognitive choice and computational complexity more familiar, I model the process by which the manager of a firm operating in a duopolistic market comes to formulate an unstructured predicament in terms of one of several alternative forms of a pricing problem as one of selecting an algorithm—or a problem solution procedure comprising a series of logically linked steps.<sup>1</sup> I consider three alternative problem

<sup>1</sup> Moldoveanu and Bauer (2004) introduce a model of organizational tasks—including the cognitive tasks of individual managers and management teams—as *algorithmic* processes. This representation allows researchers to measure the computational complexity of tasks once they are represented as algorithms. Moldoveanu and Bauer (2004) propose that organizations can be understood as distinguishing between *solvable* and *unsolvable* problems and, within the class of solvable problems, between *tractable* and *intractable* ones in the choice of tasks they undertake; and that they treat tasks in different complexity classes by

statements, each entailing a solution algorithm of a different level of computational complexity (summarized in Table 1):

*Problem statement 1: Given analyst/investor EBITDA expectations, cost of goods sold estimate, elasticity of product demand estimate, total addressable market, and market share, calculate the list price of one unit of product.*

In this problem statement, the manager does not take into account the competitor's actions, or assumes those actions are not relevant to his pricing decision. One could argue that this problem statement is 'incorrect' because it does not heed all 'relevant' data, but it may be that it is in fact an 'efficient' way to think in situations where competitive reactions do not have large effects on profits and given that the algorithm for solving the problem is simple, that is:

*Algorithm cost-plus:*

Step 1. Estimate product gross margin ( $m_g$ ) required to meet analysts'/investors' EBITDA expectations given total addressable market and market share estimate;

Step 2. From knowledge of  $m_g$  and *COGS*, calculate the required 'street' product price as

$$p_{street} = \frac{cogs}{1 - m_g};$$

Step 3. Check that the price satisfies willingness to pay of target customers, that is, compare  $p_{street}$  with inverse demand price,  $p = f^{-1}(q_{estimated})$ :

---

different means. Complexity coping mechanisms, such as problem space partitioning and the separation of solution generation from solution verification tasks, were proposed as ways in which managers deal with intractable (*NP-hard*) problems. Moldoveanu and Bauer's (2004) approach assumes that complexity measures can be more or less rigidly and unambiguously connected to organizational tasks, and therefore that managers do not make 'cognitive choices' over the complexity classes of their representations of their own predicaments. However, just as scientific theories are underdetermined by observation statements (Duhem, 1991; Quine, 1951), making the articulation of scientific theory a matter of decisions among competing conventions, just so, unstructured 'situations' or predicaments do not, in themselves, entail particular problem statements, and thus, the phenomenon of managerial *cognitive choice* among alternative problem statements for a particular predicament becomes relevant. This is the phenomenon I concentrate on here, which distinguishes this analysis from that of (Moldoveanu and Bauer, 2004).

if  $p_{street} \leq p$ , then mark up list price by the difference:  $p_{list} = p - p_{street}$ ; else, go back to analysts/investors, revise their expectations and return to Step 1.

Algorithm *cost-plus* is 'simple' in that it requires only a few operations of computation per problem variable: the number of operations it requires is a linear function of the number of variables in the problem statement. Its low computational costs entail a low cost of using the algorithm to calculate the solution to the problem as it is stated. Incorporating 'market effects' into the problem statement will lead to a more complex problem statement, for instance:

*Problem statement 2: Given analyst/investor EBITDA expectations, cost of goods sold estimate, elasticity of product demand estimate, total addressable market, and monopolistic market share, calculate the list price of one unit of product.*

In this formulation, the manager treats his own firm as a monopolist. The assumption may be *valid* as a working assumption for many reasons: the monopoly price may be close to the optimal duopolistic price (i.e., the intercept of the inverse demand curve may be very high); or, there may be features of the products sold by the two firms which, although *ex ante* irrelevant to the designers of the products, may turn out to be *ex post* critical in determining willingness to pay, thus splitting the *ex ante* duopolistic market into two *ex post* monopolistic markets, each larger than the total addressable market estimates of managers. In this case, one only needs a slightly more sophisticated version of algorithm *cost-plus* to satisfactorily solve the problem called out by the new problem statement:

*Algorithm monopoly:*

Step 1. Estimate product gross margin ( $m_g$ ) required to meet analysts'/investors' EBITDA expectations given total addressable market and market share estimate;

Step 2. From knowledge of  $m_g$  and *COGS*, calculate the required 'street' product price as

$$p_{street} = \frac{cogs}{1 - m_g};$$

Table 1. The effects of problem statement and choice of algorithm on computational complexity of managerial problem and the expected opportunity costs of deeper computation

Problem statement and algorithm	Input variables (quantities needed to set up problem)	Output variables (what the problem statement is set up to calculate)	Problem complexity as a function of number of variables	Relevant variables left out of problem statement relative to full competitive analysis	Costs of simplification will depend on
<b>1:Cost-plus</b>	EBITDA expectations, cost of goods sold, elasticity of demand, total addressable market	Gross margin	linear	Market share estimate, competitive response.	Demand elasticity, competitive response
<b>2:Monopoly</b>	EBITDA expectations, cost of goods sold, elasticity of demand, total addressable market, market share	Gross margin	linear	Competitive response	Competitive response
<b>3:Cournot</b>	EBITDA expectations, cost of goods sold, elasticity of demand, total addressable market, market share	Gross margin	nonlinear	—	—

Note: The basic problem statement (Column 1) entails that certain independent variables (Column 2) and dependent variables (Column 3) will be taken into account, while other variables (Column 5) will not. The costs of engaging with a simpler problem statement will depend on the values of various empirical variables (Column 6).

Step 3. Check that the price satisfies the monopoly price and that the estimated quantity satisfies the monopoly quantity needed to satisfy the addressable market. Assume a linear inverse demand function,  $p = a - q$ , where  $a$  is the price-axis intercept of the inverse demand function and  $q$  is the quantity supplied. Compare  $p_{street}$  with the profit-maximizing monopolistic price  $p_m$ , that can be calculated to be  $(a - cogs)/2$  by a short constrained linear optimization exercise: if  $p_{street} \leq p_m$ , then mark up list price by the difference:  $p_{list} = p_m - p_{street}$ ; else, go back to analysts/investors, revise their expectations and return to Step 1.

Algorithm *monopoly* uses a simple optimization exercise to determine the market pricing that was assumed to be the outcome of a best guess in algorithm *cost-plus*. The total number of operations does not increase by more than a linear function of a subset of the number of input variables—the cost of the linear optimization carried out in Step 3—and thus the problem statement still calls out a simple problem. A similarly simple problem is set up by a problem statement that assumes (falsely) that the market contested by the two firms (among several others) is in ‘perfect competition,’ with prices being dwindled to marginal costs. In such a case, the duopolists’ expectations are in a ‘self-confirming equilibrium’ (Ryall, 2003): the two firms’ managers’ actions are such that there is no expectation-disconfirming signal that the market can send either of them after they have made their pricing and quantity decisions. Each of the duopolists could also choose to build a more accurate model of the duopolistic market scenario, as follows:

*Problem statement 3. Given (analyst/investor EBITDA expectations, cost of goods sold estimate, elasticity of demand estimate, total addressable market, duopolistic market share, reaction curve, and symmetric assumptions about competitor, calculate the list price of one unit of product.*

This representation enjoins managers to explicitly heed each others’ pricing and/or supply quantity decisions in making their price/quantity decisions, and to incorporate the results of their calculations in their pricing decisions, perhaps by a

procedure that can be roughly formalized as follows:

*Algorithm Cournot:*

Step 1. Estimate product gross margin ( $m_g$ ) required to meet analysts'/investors' EBITDA expectations given total addressable market, and market share estimate;

Step 2. From knowledge of  $m_g$  and *COGS*, calculate the required 'street' product price as

$$p_{street} = \frac{cogs}{1 - m_g};$$

Step 3. Check that the price satisfies the profit-maximizing duopoly price and that the estimated quantity satisfies the duopoly quantity needed to satisfy the addressable market. Assume a linear inverse demand function,  $p = a - q_T$ , where  $a$  is the price-axis intercept of the inverse demand function and  $q_T$  is the total quantity supplied,  $q_T = q_i + q_j$ , the quantities supplied by the two oligopolists. Compare  $p_{street}$  with  $p_d$ , which can be calculated to be  $(a - cogs)/3$  by a constrained optimization exercise that yields the simultaneous linear equations for the duopoly quantities:  $q_i^* = (a - q_j^* - c)/2$ ;  $i, j = 1, 2$ , from which  $p_{d,i,j}$  can be computed by substitution into the inverse demand function: if  $p_{street} \leq p_d$ , then mark up list price by the difference:  $p_{list} = p_d - p_{street}$ ; else, go back to analysts'/investors, revise their expectations, and return to Step 1.

In this latter case, I replaced the simple optimization problem of algorithm *cost\_plus* with a more complicated optimization problem that requires (in the general case of  $n$  competitors), a number of the order of  $n^2$  to solve exactly. We could also replace the 'symmetric' joint optimization problem of algorithm *Cournot* with a still more complicated reaction-function-based calculation of the resulting duopoly quantities and prices based on the 'mutual, iterative adjustment' of quantities, which gives rise to the super game, or, mind game, wherein each duopolist takes into account not only various things that the other duopolist could *do*, but also various ways in which the other duopolist could *think*. One could, finally, consider the case in which each oligopolist thinks

about all of the possible ways in which its competitors will conceptualize their predicament, which leads to a still more complex, and thus still harder problem.

With this computational framework, one can (a) empirically measure the costs and benefits of adopting more or less complex problem statements, (b) investigate—through experimental and quasi-experimental manipulation—the relative importance of informational and computational factors that would lead managers in a particular case to adopt suboptimal competitive decision-making strategies, and (c) explore couplings between informational and computational effects, as follows:

a. Given that the optimal quantity choice for each of the competitors in a duopoly is  $q = (a - c)/3$ , which allows the duopolists to clear the market by charging  $p = a - 2(a - c)/3 = (a + 2c)/3$ , and realize symmetrical profits of  $(a + 2c)/3 - c = (a - c)/3$ , one can calculate the relative value to one of the duopolists of figuring out the Cournot Nash equilibrium quantities by comparing the equilibrium profits with those that could be realized if both competitors choose to serve the market as (i) monopolists, in which case each would produce  $q = (a - c)/2$ , and realize prices of  $a - (a - c) = c$ , which would set the profits of each to 0; or, (ii) as participants to a perfectly competitive market. In this case, each would set prices at marginal cost and once again realize profits of 0. One can use this approach to probe into the *reasons* for which firms choose to (i) enter a new market, (ii) over- or undersupply that market, or (iii) over- or underinvest in it, by looking specifically at the quantity and pricing decisions of entrant firms to an oligopolistic competition game, and thus tap into an alternative set of explanations for such phenomena than those generally classed under 'overconfidence biases' (Camerer and Lovo, 1999). In particular, if we posit overconfidence in one's own abilities to supply a market with a certain quantity of product at a particular gross margin as an explanatory mechanism, then we should examine the precision and accuracy of managers' knowledge of their own cost structures, production functions, and the presence of competitors, as overconfidence will manifest itself as underestimation of costs, overestimation of the firm's

ability to execute, or underestimation of the number of the firm's competitors. Managers that are overconfident in any of these senses will end up being 'stuck' and exiting the market, as they see few or no opportunities for changing the production function of the firm or the competitive landscape of the industry in an amount of time that fits the 'metronome' of their product or financial markets. By contrast, if under-optimization of the firm's strategic pricing decision is taken into account as an explanatory mechanism, then, we should observe a rapid, textured and potentially complicated process of *tatonnement* or iterative adjustment of product quantities or prices, by which the entrant and its competitors *learn* to be optimally rational, that is, they *become* rational as they find out more information about the level of rationality and computational prowess of their competitors.

- b. One can investigate by experimental and quasi-experimental means the *mechanisms* by which managers adopt certain strategies rather than others. Managers may adopt a monopoly production or pricing strategy because of an *informational gap* (they do not know of the presence of a competitor). In this case, we might expect them to change their production level (to the Cournot level) upon discovering they have competition. A *computational gap*, by contrast, would predict that even if they knew of the presence of their competitor, they would still not change their production levels, or, change them only imperfectly, because they will not engage in the computational work required to solve the competitive decision-making problem that results. In turn, this computational gap may signal a *conceptual gap*, wherein managers will not engage in solving a particular problem because they do not know the *algorithm* required to solve it, such as, in this case, the algorithm *Cournot* above. Alternatively, they may not engage in solving the optimally posed problem because of an *incentive gap*: even if they know the algorithm and are capable of deploying it, it is not *worthwhile* for them to do so in the sense that the expected net benefit of long calculations given opportunity and marginal costs of thinking more deeply is low. Finally, under-optimization can be explained by a *transfer gap*, wherein managers know of the presence of all of their competitors, know the algorithm they

need to use in order to solve competitive pricing problems of a similar type, are optimally incented to carry out the full optimization exercise (the market reaction curve is very steep but deliberation times are not severely constrained), but fail to effect a *cognitive transfer* of the right algorithm to the situation that faces them (Barnett and Ceci, 2002).

The effects of these gaps can be explored by supplying managers with information regarding competitors (the informational gap), training managers in the use of competitive problem solving algorithms (the conceptual gap), measuring the degree to which managers transfer conceptual learning to their own experiences (using cognitive skill transfer protocols), and having isolated these gaps, observing managers make decisions in situations with various payoff structures (product markets with higher or lower demand elasticity) and time constraints (i.e., in tactical versus strategic pricing situations).

- c. One can explore *couplings* between informational and computational effects by conjecturing that managers experience informational gaps—wherein they do not know who their competitors are, or, they do not consider their own actions as being representative of the reference class of their competitors (Kahneman and Lovallo, 1994)—precisely *because* they do not engage in solving certain kinds of problems (specifically those requiring competitive reasoning), rather than because they are mired in what some have called 'the inside view' (Kahneman and Lovallo, 1994). In this case, we should observe that managers' information gathering strategies will change as a function of being trained in competently solving certain kinds of problems and in *transferring* this skill to their own predicaments.

## COMPLEXITY CLASSES AND LEXICOGRAPHIC COGNITIVE CHOICE FUNCTIONS

Having established that a manager could model his predicament by one of several alternative problem statements, one can ask for a criterion for choosing among alternative problem statements. The criterion proposed here is based on the computational or time-complexity (Garey and Johnson, 1979; Papadimitriou, 1994; Corman *et al.*, 1990)

of candidate problem statements. ‘Easy’ problems are those whose solution algorithms take up a number of operations—from problem statement to satisfactory solution—that is at most a *polynomial* function of the number of input variables. They are referred to as *P*-hard problems. Their complexity scales (*log*)-sublinearly with the number of independent variables of the problem statement. They include problems of constrained linear optimization (profit-maximizing resource allocation subject to budget constraints) and of matrix inversion (inverting a demand function for the purpose of capacity and inventory planning). ‘Hard’ problems have solution algorithms that take up a number of operations that is a *greater-than-any-polynomial* function of the number of input variables (*NP*-hard problems) (Garey and Johnson, 1979). Their complexity scales (*log*)-superlinearly—for example, exponentially or superexponentially—with the number of variables of the problem statement. They include solving a strategic game problem by backward induction to Nash equilibrium (Gilboa and Zemel, 1989), inference to the best explanation of a set of disparate data points, the calculation of strategically relevant network-level variables such as betweenness and closeness centrality of a firm or individual in an exchange or interaction network, and consistency checks among sets of axioms of a logical system, which is a good model for generalized deductive reasoning.<sup>2</sup>

A lexicographic cognitive choice function over problem complexity classes models managerial

problem formulation by stipulating that *P*-hard problems are weakly preferred to *NP*-hard problems as ways of conceptualizing predicaments, given a threshold number of salient variables. The reason for stipulating a minimum number of salient variables can be understood if we consider the rate at which the complexity of an algorithm for solving a *P* or *NP*-type problem increases as a function of the number of variables that are salient to that problem (Table 2). The complexity of *NP*-type problems can take off slowly (relative to some *P*-type problems) but grows more quickly thereafter. This makes it reasonable to conjecture that, in an informationally rich regime—represented by a large number of salient variables (which can be empirically measured and assumed here to be greater than seven for reasons having to do with working memory limitations ‘discovered’ by cognitive psychologists (Miller, 1956)—managers will systematically choose *P*-hard problem statements over *NP*-hard problem statements to represent their predicaments.

The choice of problem types may not be optimally adaptive to informational regimes and can become the source of cognitive automatisms documented in the psychological literature (Bargh and Chartrand, 1999). Some managers may systematically avoid *NP*-hard problems without regard to the number of variables they are faced with because cognitive habits associated with solving certain kinds of problems (*P*-hard) can shape the kinds of variables that a manager customarily pays attention to. In such cases, simple problems will be preferred to hard problems no matter how sparse the set of input variables is, and without regard to the opportunity cost of ‘foregone calculation’—representing the loss associated with representing a predicament as an easy problem in a situation where solving a hard problem would have led to a higher net payoff.

The complexity-based cognitive choice paradigm also provides, then, a model of managerial attention and focus. If problem types (*P*, *NP*) come to function as organizing *schemata* for attention, perception, and information gathering, then managers will attend to that which they have reason to attend to in virtue of having accepted a particular problem statement as a representational schema for their predicament. The complexity class of the problem types a manager uses to represent predicaments determines which variables are salient in a

<sup>2</sup> The notion of a ‘solution algorithm’ here is meant to capture any and all search procedures. Although there are several possible ways of defining an algorithm (Cormen *et al.*, 1990; Garey and Johnson, 1979; Papadimitriou, 1994), they have in common a conception of an algorithm as a *procedure for the search of a space of possible solutions*. In the computational complexity literature—as opposed to literatures in the cognitive psychology and behavioral decision theory fields—a heuristic is a ‘simple search procedure,’ or, a computationally simple algorithm (Garey and Johnson, 1979). In the literature on the psychology of judgment formation, a heuristic is also a simple search process (Czerlinski, Gigerenzer, and Goldstein, 1999), even though heuristics are not usually represented as algorithms. All of the heuristics studied by behavioral decision theorists (see, for instance, Gigerenzer and Goldstein, 1999; Dawes, 1998) can be formulated as that subset of *P*-class algorithms whose complexity is either *constant* or *linear* in the number of problem variables. The framework proposed here is meant to distinguish not only between ‘tractable’ and ‘intractable’ problems (*P* versus *NP*), but also between what is commonly known as a heuristic (a constant-complexity or linear-complexity algorithm) and a more formal ‘problem-solving process’ (algorithms with higher complexity levels), which affords a more precise characterization of the notion of a heuristic in terms of the computational complexity of its standard application.

Table 2. Illustrating the dependence on time complexity of a problem on the number of salient variables of the problem ( $N$ ), for three different complexity growth profiles: square growth (Column 2), cubic growth (Column 3) and exponential growth (Column 4)

Number of variables, $N$	If time complexity grows with $N^2$ , then problem complexity is	If time complexity grows with $N^3$ , then problem complexity is	If time complexity grows with $2^N$ , then problem complexity is
1	1	1	2
2	4	8	4
3	9	27	8
4	16	64	16
5	25	125	32
6	36	216	64
7	49	343	128
8	64	512	256
9	81	729	512
10	100	1000	1024

particular situation. A manager who avoids game-theoretic reasoning of the Cournot-Nash type will have no use for, nor spend time trying to infer, her competitor's conjectures or beliefs about *her own* conjectures or beliefs about the structure of the game.

A simple way to parse managerial problems so that they become distinguishable according to their complexity type ( $P, NP$ ) is based on understanding that  $P$ -hard problems are prototypical 'tree-search' problems, whereas  $NP$ -hard problem are typically 'graph search' problems (Cormen *et al.*, 1990).<sup>3</sup> Tree search problems correspond to solution spaces that can be represented as a tree (Figure 1A): the various nodes can be searched

sequentially by going down a tree structure, with each search step narrowing the number of possible solutions by a large factor. As the figure shows, the problem of finding the five-letter English word that begins with 'r', ends in 'd,' and has middle letters drawn from the set  $\{(u), (o), (n)\}$  can be represented as a tree search, which takes a number of operations of at most  $\log(N)$ , where  $N$  is the number of possible 'end' states of the search process. By contrast,  $NP$ -hard problems are prototypically 'graph search problems' (Figure 1B), wherein all of the possible paths (collections of edges) that traverse the graph have to be searched. Figure 1B shows the warhorse of  $NP$ -hard problems, the 'traveling salesman problem': a salesman is trying to figure out the route that connects  $N$  locations (denoted by letters  $A-E$  for  $N = 6$ ). An exhaustive search through all possible routes will require  $N^N$  operations (where  $N = 6$ ). This basic rule of thumb for distinguishing between easy and hard problems allows us to parse managerial problems into those that can be represented by graph and tree searches, respectively, and to analyze managerial problems in terms of their computational structure and complexity.

### WHAT WILL MANAGERS NOT (USUALLY) THINK ABOUT? HARD PROBLEMS AND COMPLEX STRATEGIES

If managers systematically avoid  $NP$ -hard problems, what specific problems will they avoid?

<sup>3</sup> Computational complexity analysis (Cormen *et al.*, 1990; Garey and Johnson, 1979) proceeds by (a) boiling a problem statement down to a representation that is amenable to representing solutions as algorithms, (b) analyzing algorithms as search procedures of either tree search or graph search types, (c) proving that the resulting tree or graph search is isomorphic to a previously known  $P$ -hard or  $NP$ -hard problem, and thus (d) deducing that the problem statement in question calls out a  $P$ -hard or an  $NP$ -hard problem. The language game played out in the field of computational complexity analysis is that of producing proofs of the difficulty class of various common problems, by showing that a 'new' problem is computationally isomorphic to an existing problem known to be of a certain complexity. Applying the findings of computational complexity theory to strategic management problem formulation and problem-solving behaviors entails (a) modeling common managerial problem statements as problems admitting of algorithmic solutions, (b) measuring the computational complexity of the solution algorithms by showing the problem statement to be of the tree-search or graph-search type and proving that it is isomorphic to a problem statement of known complexity, and (c) making predictions about whether or not the problem statement in question will function as an organizing schema for managerial cognition, as a function of its complexity class, and the number of variables that the manager would have to take into account in order to produce a solution.



*Lacunae of competitive reasoning: hard problems in competitive decision making*

Empirical and experimental (Holt, 1995) findings suggest that managers regularly overproduce in situations characterized by oligopolistic competition. One explanation for these findings is that managers do not think through their competitors' capacity decisions and of how these decisions will impact their own profit margins. Zajac and Bazerman (1991) argue that these situations reflect *blind spots* of the managers making the supply decisions to market variables relating to competitors' marginal proclivity to produce: on this view, managers do not pay attention to relevant, useful, and readily available information. However, their analysis does not answer the question: 'why not?' The model developed here suggests the following explanation: Finding the Nash equilibrium of a game through iterative dominance reasoning is an *NP*-hard problem, equivalent in structure to the problem of determining whether or not a fully connected, undirected graph  $G$  has a clique—a set of vertices that are fully interconnected by edges—of size  $k$  (Gilboa and Zemel, 1989). If managers avoid *NP*-hard problems, they will likely not engage with the kind of reasoning that is necessary for arriving at Nash equilibria and will thus exhibit a computationally induced 'blind spot' whenever they must solve a problem that is *NP*-hard to arrive at an optimal action.

**Hard problems in reasoning from incomplete information**

*The complexity of inference to the best explanation*

Formalizations of diagnostic decision making and inferential reasoning (Bylander *et al.*, 1991) have represented these processes as forms of abductive reasoning, which refers to an inference to the best explanation  $\{h_0\}$  or to the set of best explanations  $\{h_k\}$ , given a set of data  $\{d_i\}$ . Abduction is an inversion of *modus ponens*: if data  $d_i$  were observed and one knows that  $h_i$  explains  $d_i$  better than any alternative, then the abductive step leads one to infer that  $h_i$  is true. For instance, lower willingness-to-pay of customers can be abductively explained by the entry of competitors into the same market segment, as opposed to a random effect or a seasonal fluctuation, or a lackluster performance of the sales support organization, if it is independently known that competitors recently entered the

market. Abductive reasoning can model part of the cognitive production task of the strategic manager, specifically that which involves the selection of models that predict the consequences of various courses of action, such as those that posit, for instance, the dependence of profit increases on market share growth.

Bylander *et al.*, 1991, showed that the problem of abductive inference can be represented as that of finding the minimum vertex cover  $V'$  for the graph  $G = (V, E)$ , where  $V$  denotes the set of vertices of  $G$  and represent the covering hypotheses or explanations and  $E$  denotes the set of edges of the graph  $G$  and represent the set of observation statements that are to be explained by the hypotheses represented by the set  $V$ . A minimum vertex cover  $V'$  of the graph  $G$  is a set of vertices such that, for any edge in the set  $\{E\}$ , there exists exactly one vertex in  $V'$  that is connected to  $e$ . This overall representation captures the basic intuition that the best explanation should cover all of the observation statements in the most economical fashion. The minimum vertex cover problem is provably *NP* hard (Garey and Johnson, 1979). If managers avoid *NP*-hard problems, one would expect to find managers engaging in *heuristic-type* reasoning ('if you see situation of type  $X$ , follow action rule  $A$ ') and analogical reasoning based on familiar situations ('I have seen  $Y$  happen in situation  $X$  before, this situation is like situation  $X$ , therefore I expect to see  $Y$  happen in this situation as well) rather than on inference to the best explanation, in which case the 'best explanations' (in the abductive sense) will likely go undiscovered in their deliberation.

*Biases and fallacies in managerial reasoning are 'p-hard' approximations to 'np-hard' problems*

Cognitive 'biases and fallacies' have been used to explain suboptimal managerial patterns of search, inference, analysis, and planning (Kahneman and Lovallo, 1994; Schwenk, 1984). They appear to be difficult to cure or redress through normative 'training' interventions (Dawes, 1998) and thus they are often represented as 'invariants of human behavior.' My model of cognitive choice suggests that the problem underlying such fallacies may not be a lack of awareness of subjects of the proper rules of epistemic rationality (Bayesian reasoning and the axioms of probability theory in this case), nor some 'hard-wired' incapacity to follow these rules once individuals are aware of them,

but rather the fact that successfully applying such rules to an example is a problem that lies in a complexity class that is avoided (the *NP* class). Specifically, the problem of rule-based reasoning is equivalent to that of running consistency checks on finite sets of propositions, which is known to be an *NP*-hard problem (Casti, 1992): The facts of each particular case must be checked with all of the subsets of the set of  $N$  appropriate rules, which leads to a computational process that takes up  $2^N - 1$  operations. It is not unreasonable to expect managers to seek *P*-hard shortcuts—such as the ‘representativeness’ and ‘availability’ heuristics. This would allow them to conceptualize predicaments that ‘normatively’ call for the deployment of the entire calculus of probabilities, in terms of ‘simple association’ problems that map plausible causes to plausible effects rather than mapping probable causes to likely effects. This proclivity would lead them to ignore information that would be useful to a Bayesian reasoner—such as, for instance, base rates of occurrence for relevant events.

**Hard problems in reasoning about activity sets: why is strategic change hard to plan and why does separability help?**

McKelvey (1999) and Rivkin (2000), among others, have put forth models of firm-level strategic activity sets as networks of interconnected nodes whose instantaneous states change as a function of the states of neighboring nodes. Such networks (modeled by  $N$ , the number of nodes,  $K$ , the average number of links per node,  $C$ , the number of covarying nodes, and  $\{f\}$  the set of functions that govern the local evolution of nodes as a function of the states of other, linked, nodes) can be used to describe sets of deterministically interdependent, ‘simple’ activities ( $N$ ), that are interconnected and covary in preset ways,  $(K, C)$ , and evolve according to preset rules ( $\{f\}$ ). These models have the interesting feature that, for values of  $K$  greater than two, the problem of predicting their evolution is *NP*-hard (Rivkin, 2000). This means that, for relatively well connected activity sets ( $K \geq 2$ ), the process of strategic planning, which requires predicting the response of the prewired sets of activities and routines to a particular change, entails arriving at the solution of an *NP*-hard problem. If managers avoid tackling *NP*-hard problems, they will aim to either uncouple

or separate out the activity sets that describe their firms’ tasks, to the point where  $K$ , the average coupling coefficient, is less than two. Alternatively, they might inaccurately treat coupled activity sets *as if* they were in fact uncoupled—and thus systematically ignore relevant couplings among activity sets.

*The hard problem of system design*

Designing systems of interconnected components (a microcomputer, a semiconductor chip with many functional units, an automobile or a subassembly of an automobile are the obvious examples; a business or marketing plan or a branding campaign are extensions) can be modeled (Wymore, 1993) as a set of input-output ports (components), a set of design constraints (such as kinematic constraints, dynamic constraints or structural constraints), and a set of performance and cost figures of merit. The overall *system design problem* can be modeled as the optimization of a figure of merit subject to the set of physical constraints and a maximum allowable cost structure. (Chapman, Rozenblitt, and Bahill, 2001) show this problem to be algorithmically identical to the well-known *knapsack problem*: for a trip, a hiker must choose from a finite collection of items, of known size and expected value to him, those items that will provide him or her with the most value, corresponding to the performance index, without exceeding the size of the bag, corresponding to the cost constraint. This is a problem that has been shown (Karp, 1972) to be *NP*-hard. The system design problem can be used to represent managerial planning and design of physical processes (such as product engineering and redesign, and manufacturing process design) as well as nonphysical processes (such as the rule-based design of a new organization or group, or the design of a large set of contractual arrangements). If they avoid *NP*-hard problems, then managers will not attempt global solutions to problems that have the structure of the knapsack problem, but will, rather, seek *P*-hard, suboptimal shortcuts. When attempting to design new contractual arrangements, for instance, one might expect that strategic managers would use design templates (previous, enforceable agreements obtained from their law firms or legal departments) and adapt them locally (and suboptimally) to the context of the current relationship or set of relationships that the contracts are meant

to govern (using ‘global search and replace’ function, for instance). Thus, the *NP*-hard problem that would generate the optimal answer to a contractual design problem is replaced by the problem of figuring out which one of a list of possible, previously signed contracts most accurately captures the salient features of the current situation—a *P*-hard problem.

### Hard problems in strategic network analysis

#### *Vertex covers and the calculation of centrality*

Centrality measures (betweenness, closeness, Bonacic) of networked agents have been argued to affect the welfare (status, brokering position, informational advantages) of networked agents (see Wasserman and Faust, 1994). It is reasonable to expect strategically minded managers to attempt to manage the structural position (parametrized by some form of network centrality) of their organization in the network representing the firms in their industry. The idea of centrality is well captured in graph-theoretic terms by the notion of a minimum vertex cover of an undirected graph  $G(V, E)$ —defined as the subset of vertices of the graph that together span all of the edges of the graph. This is an *NP*-hard problem (Cormen *et al.*, 1990). One would, then, expect that strategic managers will be poor maximizers of their own centrality measures within the network that describes their value chain or industry ecosystem because they avoid the kind of problem they would have to solve in order to maximize it.

One *would*, on the other hand, expect managers to focus on maximizing *tractable* network-dependent features of their organizations, such as role equivalents. Winship and Mandel (1983) have developed the idea that actors in social networks have ‘role sets,’ or lists of functions that the actors perform in their networks. The role set of actor *A* is formed by considering all of her actual roles vis-à-vis the rest of the actors in her network. The degree to which actor *A*’s role set is equivalent to actor *B*’s role set can be calculated by (a) first encoding the different roles that *A* has in her network, using real numbers, (b) encoding the different roles that *B* has in his network, using matching real numbers, and (c) correlating the two resulting vectors of real numbers to obtain a coefficient that will measure the role equivalence of

*A* and *B*. Hence, we would expect that strategic managers will often engage in gauging their own position within a network by *correlating* their own role sets with those of other actors engaged in other networks. The simplest way in which this correlation is performed is by the one-point comparison between the two actors of job titles in large hierarchies (*C*-level, *V*-level)—a prototypically *easy* (*P*-hard) problem. Notably, recent contributions to the analysis of social networks (Watts, Dodds, and Newman, 2002) distinguish between searchable networks (where network search can be performed via a *P*-hard tree-search) and unsearchable networks (requiring *NP*-hard graph searches that are computationally prohibitive for large networks).

#### *Clique existence and membership*

Cliques are important structures in interfirm networks because they facilitate coordinated action between member firms (Moldoveanu, Baum, and Rowley, 2003) through mechanisms of trust-building and mutual monitoring (Burt and Knez, 1995) that mitigate free-rider behavior and other forms of opportunism. The problem of identifying the cliques of a particular size (say, *k*, and thus, henceforth, I will refer to *k*-cliques) in a network of social actors represented by a graph *G* is *NP*-hard (Cormen *et al.*, 1990). If managers avoid such problems, then they will likely not be aware of many of the cliques in their own network, nor optimize their own tie-formation behavior to maximize their own membership in key cliques, even if they have access to information about every node and tie in the network. They will thus forego forming beneficial alliances that are structurally feasible within their networks. If managers do not optimize on the set of cliques their organization is part of, then it would be reasonable to expect that their ‘network clique awareness’—or, the degree to which they are aware of ties among agents that are more than one or two nodes removed from the focal firm in the network—will be low, relative to what it should be had managers considered all of the relevant *k*-cliques in their own network.

#### *Reductio ad computatio: implications for the investigation of strategic holes*

The distinctions made above (summarized in Table 3) can be used to design empirical investigations

Table 3. Typology of managerial problems in terms of computational complexity of their algorithmic solution procedures

Problem type	Canonical algorithmic representation	Computational complexity class	Comments
<i>Ordering, ranking, sorting</i>	Tree search	<i>P</i>	Models standard calculus of ordering
<i>Pattern recognition by correlation</i>	Convolution/correlation (tree search)	<i>P</i>	Models standard trend -detection, reasoning by analogy
<i>Backward induction</i>	Graph search	<i>NP</i>	Models standard approach to game-theoretic reasoning
<i>Inference to best explanation given dataset, hypothesis set</i>	Graph search	<i>NP</i>	Models generalized inference problems
<i>Computation of graph-theoretic node properties</i>	Graph search	<i>NP</i>	Models generalized network problems
<i>Consistency checks on finite sets of axioms</i>	Graph search	<i>NP</i>	Models hypothetico-deductive reasoning

Note: Everyday' problems (Column 1) are represented as formal, canonical problem statements (Column 2) whose complexity can be measured and categorized (Column 3).

of 'holes' in a firm's strategy that arise from computational gaps in managerial thinking (Table 4). As Table 4 shows, the problem for the researcher interested in probing computational complexity effects on the scale and scope of managerial attention is to find industry-specific variables that (a) correlate with firm profitability, and (b) would make a difference to managerial thinking and strategic choices *only if* managers engage in solving *NP*-hard problems, but not if managers only solve *P*-hard problems, then (c) to observe the effects of changes in these variables on the firm's strategies. If, for example, strategic managers optimized their firms' partnering strategies to maximize their firms' relative betweenness or closeness centrality (which are important to firm profitability in high information-flow and high entropy-rate-change businesses like investment banking and venture capital), then they would have to solve *NP*-hard problems associated with calculating relative centrality measures and their strategic choices would be responsive to changes in variables (ties formed between firms that are 'far away' from their firm in the network) that they would not heed if they only considered their own partners' and partners' partners' partnering patterns in making their partnering choices. Similarly, managers in coordination-intensive industries (those with, for instance, long value chains) have reason to consider the size and distribution of cliques in their industry network, but, unless they were

solving *NP*-hard 'clique search' problems, would not devise strategies that are responsive to changes in clique size and distribution. Following the same logic, one would only expect managers engaged in solving *NP*-hard problems to change their strategic planning processes in response to new technologies or new activity sets that increase the average density of couplings within the firm's activity network. Further, managers who think their way to strategic *equilibria* in oligopolistic markets are more likely to implement strategic pricing and supply quantity changes in response to new information about changes in the number or cost structures of their competitors than those who do not. In these cases, a computational gap begets an informational gap, as managers do not pay attention to changes in variables that are not part of their problem statements.

The notion of a computational gap can also be used to examine the effects on managerial thinking and behavior of new ideas—including theories, models, and frameworks—and of new ways of thinking on strategic management behavior: In cases where adoption of a new thinking 'technique' (game-theoretic reasoning) brings about a change in the complexity class of the problems that managers set for themselves, one would expect managers to begin to heed variables that were previously ignored, as they did not figure into the standard repertoire of managerial problem statements.

Table 4. 'Strategic holes': how to predict the effects of computational gaps on strategic choices

These variables make a difference to firm-level profitability....	... but would only be taken into account if managers engaged in solving these NP-hard problems....	... in which case changes in these industry- or firm-level conditions ....	... would not produce these changes in firm-level strategies.
Betweenness centrality of firm in an industry in which profits correlate with timely use of relevant information	Maximization of betweenness centrality of firm	New ties formed between or among firms that are 'far away' from the firm in the network (they are not the firm's partners)	Compensatory changes in the firm's own partnering strategies aimed at preserving or increasing betweenness centrality
Size and membership of dominant cliques in relationship-centric industry	Finding cliques of size $k$ or greater in the firm's network	New cliques formed among firms that are 'far away' from the firm in the network	Compensatory changes in the firm's own partnering strategies aimed at preserving or increasing the relative size of its own clique(s)
Closeness centrality of firm in an industry in which profitability varies with 'social connectedness' or 'knowtness'	Maximization of closeness centrality of firm	New ties formed between or among firms that are 'far away' from the firm in the network (they are not the firm's partners)	Compensatory changes in the firm's own partnering strategies aimed at preserving or increasing closeness centrality
Average degree of connectedness of the firm's value-linked activity network	Prediction of evolution of coupled activity network of the firm over a relevant period of time	Introduction of new activity sets or new linkages among existing activity sets that change the average connectivity of the firm's activity net	Structural or behavioral changes in the firm's planning process aimed at accommodating higher complexity class of resulting prediction problem
Specific number and cost structure of competitors in an oligopolistic industry	Calculation of competitive equilibrium prices and quantities using iterated dominance reasoning, given knowledge of competitors and their cost structures	Changes in number of competitors and/or the cost structure of existing competitors in an oligopolistic industry	Incorporation of changes in the firm's quantity and price decisions on revenue, inventory and profitability guidance to analysts and investors
Introduction of new, explanation-generating theory in body of managerial 'knowledge-in-use,' which supplies new 'best explanation' for existing patterns of industry behavior	Inference to the best explanation as a foundation for articulation of strategic decisions	Change in industry conditions deemed by new theory but not by existing theories to be relevant to firm profitability	Change in strategic decisions or decision process consistent with the new theory

## WHAT WILL MANAGERS THINK ABOUT (AND, HOW WILL THEY THINK ABOUT IT)? THE NATURE, STRUCTURE, AND OCCURRENCE OF 'EASY' PROBLEMS

Many of the cognitive tasks that managers *do* engage in can be described in terms of computationally simple (*P*-hard) problems, many of which can be represented as tree searches. They include linear ranking and sorting of random lists, searches of structured knowledge domains, correlations, autocorrelations and cross-correlations, and matrix multiplications and inversions that arise from constrained linear maximization and minimization problems. One can think of most of the problems encountered in the basic 'MBA toolkit' that managers are equipped with as exercises in the application of *P*-hard algorithms: Managerial economics relies on carrying out linear constrained maximization processes. Managerial accounting *praxis* consists of cataloging (modeled by sorting algorithms) cash inflows and outflows, and linear manipulations of cost and revenue vectors (modeled by matrix multiplication and inversion algorithms). Many problems in finance—such as the capital asset pricing model and the Black-Scholes option pricing model (see Bodie and Merton, 1997 for a textbook-style exposition of the model)—rest upon the direct application of given formulas to given datasets. Strategy courses in MBA programs stress the recognition of typical competitive or cooperative situations, whose *typicality* is a function of the result of correlations of the facts of the current case with a knowledge base of relevant prior cases, and the application of heuristics to derive action maps and strategic plans, such as Porter's 'five forces model,' the Boston Consulting Group profitability matrix, or the 'GE 9-box.'

Nonlinear extensions of these theories, models, frameworks, or framing lenses (such as nonlinear optimization problems in logistics, game-theoretic reasoning for competitive pricing and supply analysis, interactive reasoning for negotiations analysis, and inference to the best explanation for strategic planning in complex scenarios) are often either left to more advanced studies or presented in heuristic form. Practitioner-oriented 'game theory texts' (for instance, Dixit and Nalebuff, 1991) do not teach game theoretic *reasoning*, which involves iterating on mutual beliefs given the antecedent

ascertainment or establishment of common knowledge of payoffs, strategies, and rationality, but, rather, convey a set of heuristics ('if you find yourself in a situation resembling situation *A*, do *x*') that are derived from prior game-theoretic analyses of '*A*-type situations.' In such cases, it is not the game-theoretic analysis of situation *A* that is taught (which would ride upon solving an *NP*-hard problem), and thus it is not the cognitive proclivity to tackle *NP*-hard problems that is trained, but, rather, the proclivity to make connections between 'typical situations' and 'suitable' courses of action.

The following distinction is, then, important: *deriving* a model (such as the Black-Scholes formula from knowledge of the basic axioms of statistical mechanics—as did Black and Scholes [1973]) is an *NP*-hard problem requiring the deployment of deductive reasoning to generate a set of theorems from finite sets of axioms. On the other hand, *applying* the formula to a dataset to calculate the value of a European call option is a *P*-hard problem whose complexity measure is linear in the number of variables in the formula. A student who attempts to apply the formula (as opposed to deriving it) must ascertain that the data in the case study or problem is in the right format, and then 'plug' the set of relevant numbers into the Black-Scholes formula. By contrast, engineering a *new* option pricing model based on a set of assumptions different from those used by Black and Scholes (1973)—for example, starting from a new definition of probabilities as measures of verisimilitude rather than as measures of likelihood (Popper, 1972), or, from a behavioral model that incorporates risk *and* ambiguity aversion into the manager's utility function—would require the solution of an *NP*-hard problem, wherein the consistency of the new set of axioms (corresponding to the axiomatization of the new probability measure) with the axioms of expected utility would have to be verified.

Similarly, applying 'Porter's five forces model' to a competitive situation involves mapping data from a case study to the concepts deployed in the model, which involves performing *correlations* among *N* different facts and *M* different concept, which is *P*-hard, as it involves a number of operations that varies as the product *MN*. By contrast, *deriving* Porter's five forces model inductively from a large database of competitive interactions involves solving the *NP*-hard problem (Aragones *et al.*, 2003) of finding a set of *N* conditional rules

that are verified by  $M$  different facts or observations. These distinctions also highlight a plausible difference between what many researchers do (tackle  $NP$ -hard problems) and what most practitioners do (tackle  $P$ -hard problems), and raises a question about the optimality of this division of cognitive labor: under what conditions, in particular, would it be advantageous for a training program to impart the know-how of formulating and solving computationally complex problems in addition to the frameworks and models of a discipline, as opposed to only imprinting the know-how associated with the latter?

*Cognitive heuristics for making inferences from incomplete information*

Cognitive heuristics and ‘biases’ (see, for instance, Kahneman, Slovic, and Tversky, 1988; Gigerenzer and Goldstein, 1999) can be understood as instantiations of a cognitive preference for simple problems of pattern recognition, sequence matching and reasoning by analogy, which can be modeled as a set of  $P$ -hard *correlation* operations between a vector of characteristics of a specific ‘unknown’ situation with vectors of properties or characteristics of other ‘known’ situations, and the selection of the highest correlation coefficient as representative of the ‘known’ situation most closely resembling the current, unknown, situation. The ‘representativeness’ and ‘availability’ heuristics are common cognitive simplification devices used by managers and examples of  $P$ -hard reasoning by example. In making a judgment about an event, object, or person, the degree of similarity between that entity and familiar entities—rather than the objective probability of drawing (at random) an entity with certain characteristics from a sample population—is what matters to the decision maker. Strategic scenario planning and case-based analysis can be understood as instantiations of the ‘representativeness heuristic,’ as they rely on (assumed) correlations between current or future (unknown) event sequences and past (known) event sequences. Case-based training of managers in professional education programs can be understood as the conveyance of a repository of potentially relevant examples that can be processed by correlation with current predicaments to derive, by a  $P$ -hard process, action maps on the basis of ‘whatever worked in a similar situation in the past’ in conjunction with an implicit rule that says: ‘one

should use a strategy or move that has worked before.’ The ‘meta-heuristic’ here is that history repeats itself reliably. The problem, in terms of which the strategic manager seeks and processes information about his or her current predicament, is: ‘what situation (from my ‘database’) is the current situation most similar to?’ (which is  $P$ -hard), rather than: ‘what do the laws of chance say about the underlying sample space that describes this situation and what additional data do I need to apply them in self-consistent fashion?’ (which is  $NP$ -hard).

To make further progress in understanding cognitive heuristics in the computational framework of this article, one can model the problem of making an inference from incomplete information as one of choosing from a list of  $M$  entities the one that best satisfies an objective function, given a set of  $N$  cues. Typical examples of such inference problems are: ‘pick, from a list of  $M$  potential suppliers, the supplier who is most likely to be bought out by a competitor within the next two quarters, given up to  $N$  separate pieces of information (such as profitability, revenues, market capitalization, and product road map) one has about each supplier’; or, ‘pick, from a list of  $M$  competitors, the one most likely to respond first to the introduction of a new product or technology, on the basis of up to  $N$  bits of relevant information about each (such as current product road map, free cash flows, and past research and development spending).’ Different heuristics for solving such inference problems can be understood as making a trade-off among computational complexity (how many operations will applying the heuristic to produce an answer take, as a function of the number of variables  $M$  and clues  $N$ ?) and either reliability (how likely is the heuristic to produce the right answer when applied repeatedly?), or accuracy (‘how accurate will the answer that the heuristic produces in this particular case be, where accuracy is defined as the inverse of the distance between the answer the heuristic produces and the correct or optimal answer?’).

Heuristics can be understood as algorithms that have a measurable computational complexity. Surveys of human judgment formation and inference that consider the underlying algorithmic structure of heuristics used to make inferences (Cooksey, 1996; Dawes, 1998; Martignon and Laskey, 1999) without exception turn up a set of heuristics that can be modeled as  $P$ -hard algorithms, and no

Table 5. The computational complexity and complexity class partitioning of commonly used heuristics for making inferences about  $M$  entities using  $N$  available cues that offer discrimination value between any two of the  $M$  entities

Heuristic or algorithm	Upper-bound on computational complexity	Complexity class	Relative frequency of utilization (predicted and/or estimated)
<i>Availability heuristic</i>	$MN^2$	P	Presumed high
<i>Representativeness heuristic</i>	$MN^2$	P	Presumed high
<i>Recognition heuristic</i>	$M$	P	High
<i>Minimalist heuristic</i>	$M(N + 1)$	P	Medium
<i>Dawes' rule</i>	$N(2M - 1) + M$	P	Low $\rightarrow$ medium
<i>Franklin's rule</i>	$N(2M - 1) + M$	P	Low $\rightarrow$ medium
<i>Multiple linear regression</i>	$M^2N + M$	P	Low
<i>Inductive search of associative laws</i>	$2^N$	NP	Nil $\rightarrow$ low
<i>Abductive search of best explanation</i>	$2^N$	NP	Nil $\rightarrow$ low

heuristic that can be modeled as an NP-hard algorithm. Table 5 shows the most frequently analyzed heuristics advanced to date to explain the judgment of humans in  $M$ -variable,  $N$ -cue problems of inference from incomplete information, alongside their computational complexity. All are in the  $P$ -class and therefore can be modeled as *simple* algorithms. By contrast, full inductive inference, which tests, sequentially, each one of  $R$  different associative laws on the dataset (an associative law takes the form: 'If entity  $X$  has properties  $S$  and  $T$ , then it will also have property  $U$ ' [see Arragones *et al.*, 2003 for proofs of NP-hardness] ) to find the associative law that best fits the data is in general NP-hard; and abductive inference (inference to the best explanation) is, as shown earlier, also NP-hard.

Moreover, the following heuristics—commonly studied in the literature—stand out both in terms of their ubiquity (the relative frequency in which they seem to be used in practice) and their computational simplicity (they can all be represented by  $P$ -hard algorithms):

H1: *The 'recognition' heuristic* (Gigerenzer and Goldstein, 1999) is based on picking the first answer from the list of  $M$  variables that is recognized by the decision maker. If the decision maker, for instance, is trying to pick out the largest German city from the list (Hanover, Dortmund, Munich) and recognizes only Munich, then he or she will put forth 'Munich' as the answer to the question. The heuristic has a complexity of at most  $K_{\max} = M$ , that is, the number of operations required to go through the list of  $M$  alternatives and choose the first one recognized;

H2: *The 'minimalist' heuristic* (Gigerenzer and Goldstein, 1999) starts with the recognition heuristic as its first step. If unsuccessful after using this first step, the user of the minimalist heuristic draws one of  $N$  cues at random and performs a pairwise comparison of each of the  $M$  entities with respect to the cue that has been drawn. Its computational complexity is upper-bounded by the sum  $MN$ , the product of the total number of available cues that one must search through and the total number of entities, and  $M$ , the computational cost of picking the entity with the highest 'score', that is,  $K_{\max} = MN + M \log(M)$ ;

H3: *Dawes' rule* assigns unit values ( $+1$ ,  $-1$ ) to each of the  $M$  entities with regard to its 'performance' with respect to each of the  $N$  cues, then computes a 'score' for each of the  $M$  entities as the sum of its weights (positive and negative) along each of the  $N$  different cues. It then ranks the  $M$  entities according to the resulting score of each. The heuristic entails performing  $MN$  value assignment operations,  $N(M - 1)$  additions and the  $M$  operations involved in sorting the list of  $M$  entities, that is,  $K_{\max} = N(2M - 1) + M$ ;

H4: *Franklin's rule* is a simplified variant of Dawes' rule. It user adds up all of the 'positive and negative reasons' for picking any one of the  $M$  entities, but does not systematically assign weights to each entity along each of the  $N$  cues, but simply uses whatever cue values come

to mind for each entity. It has, therefore, the same upper bound in computational complexity as that of Dawes' rule, that is,  $K_{\max} = N(2M - 1) + M$ ;

H5: *Multiple linear regression* computes the minimum-mean-squared error estimate of the predictive value of each cue regarding each of the  $M$  entities of interest, then, ranks the resulting vector of  $M$  entities to find the one with the best minimum-mean-squared-error-based predictive score. Accordingly, its computational complexity is upper bounded by sum of the cost of performing the multiple linear regression on  $N$  independent variables ( $M^2N$ ) and that of performing the ranking operation,  $M$ , that is,  $K_{\max} = M^2N + M$ .

Table 5 also shows the computational complexity of algorithms that model the representativeness and availability heuristics. Both can be represented by correlations between the  $N$  attributes of an entity and the corresponding  $N$  attributes of either a readily available entity (availability heuristic) or that of a stereotypical or representative entity (representativeness heuristic). Correlating two  $N$ -vectors 'costs' the decision maker  $2N^2 - I$  operations, and ranking the  $M$  entities costs the usual  $M - I$  operations, for a total cost of the order of  $MN^2$ , that is,  $K_{\max} = (2N^2 - I)(M - I)$ . Interestingly, these two heuristics are more computationally expensive than several of the alternatives in Table 5. Although it is currently presumed that they are frequently utilized, a systematic empirical investigation of the frequency with which these heuristics are utilized *relative* to lower-complexity alternatives has yet to be performed. All we know from the current gamut of evidence is that they seem to be preferred to full induction or abduction in forced choice problems of judgment with incomplete information.

#### SUMMARY AND EXTENSIONS: USING MANAGERIAL ALGORITHMS TO REPRESENT AND MODEL STRATEGIC THINKING

Managerial algorithmics—the approach proposed here—is based on understanding the algorithmic structure of the problems that managers use to

make sense of their predicaments, and examining the cognitive choices they make when formulating the problems they subsequently solve. Extending this approach to the study of managerial cognition and strategic decision making can proceed along several lines:

- I. The approach proposed here assumes that managers make quick, and likely subconscious, decisions about the kinds of problems they will engage with, and then seek information about the variables that are relevant to the kind of problem they have chosen. Managerial 'styles of thinking' are assumed to be relatively 'hard-wired' in the sense that the  $K$ -complexity of a manager's problem statements will not itself be a variable that managers optimize. In such cases, it makes sense to test for the *ecological optimality* of using certain algorithms and heuristics rather than others—as has been urged by Gigerenzer and Goldstein (1999). However, we can also investigate the *prima facie* less plausible 'adaptive rationality' hypothesis, which is that managers make cognitive choices in real time, *while* thinking through a problem and *after* having chosen a solution algorithm. For example, if we know that a manager has used a Cournot duopoly model to represent his predicament, one can use the modeling framework to measure the benefit and cost of each individual *logical operation* involved in using that model. The benefit of any one operation can be computed from the profit loss due to suboptimal levels of computation. In an oligopolistic market with two competitors who face a linear, downward sloping demand curve given by  $q = a - p$ , (where  $a$  is some constant,  $q$  is the [total] quantity of units of good  $X$  sold on the market and  $p$  is the price of the good) and assuming  $c$  to be the marginal cost of the good, the quantity that each competitor would have to produce of good  $X$  to maximize profits can be obtained by differentiating the total profit of each competitor,  $\text{Pr}_i = (a - q_i - q_j - c)q_i$ ;  $i, j = 1, 2$  with respect to the quantity of good  $X$  produced, to get two linear equations in two unknowns,  $q_i^* = (a - q_j^* - c)/2$ ;  $i, j = 1, 2$ . These can be solved simultaneously to get the Nash equilibrium output,  $q_1^* = q_2^* = (a - c)/3$ , which represents the (unique) quantity choice from

Table 6. The benefits of logical depth of computation when playing against a logically potent competitor

Iteration	If Firm 1 chooses quantity...	Then Firm 2's profit maximizing response is...	... in which case Firm 1's profit maximizing quantity choice should have been...
1	$(a - c)/2$	$(a - c)/4$	$3(a - c)/8$
2	$(a - c)/4$	$3(a - c)/8$	$5(a - c)/16$
3	$3(a - c)/8$	$5(a - c)/16$	$11(a - c)/32$
4	$5(a - c)/8$	$11(a - c)/32$	$21(a - c)/64$
5	$11(a - c)/32$	$21(a - c)/64$	$43(a - c)/128$
6	$21(a - c)/64$	$43(a - c)/128$	$85(a - c)/256$
7	$43(a - c)/128$	$85(a - c)/256$	$171(a - c)/512$

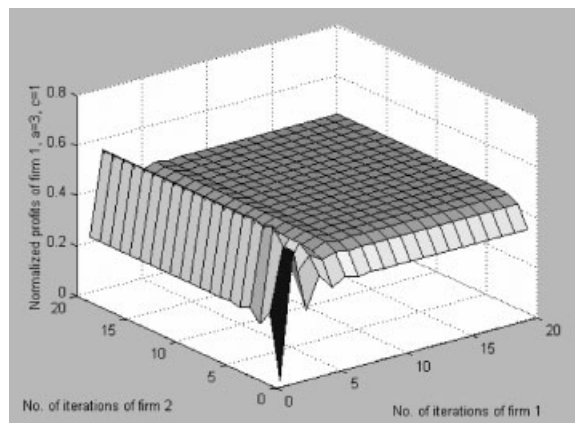
which neither competitor can unilaterally deviate and be better off.

To understand the choices that a manager might make on an operation-by-operation basis, the question to focus on is: how do managers reason their way toward this equilibrium? Suppose (as does Saloner, 1994; following Milgrom and Roberts, 1990) that Firm 1 decides to produce the (monopolistic) output of  $(a - c)/2$ . Firm 2's profit maximizing response to this choice is not  $(a - c)/2$  but rather  $(a - c)/4$ . If Firm 2 actually produces  $(a - c)/4$ , then Firm 1's profit maximizing response will no longer be  $(a - c)/4$ , but rather  $3(a - c)/8$ , which leads to an (infinite) sequence of mutually responsive and iterative 'best responses' given by  $t_n = (a - c - t_{n-1})/2$ ,  $n = 1, 2, 3, \dots, \infty$ ; which converge in the limit to  $\lim_{n \rightarrow \infty} (t_n) = (a - c)/3$ . We can now measure the benefit of 'thinking more deeply,' by measuring the profit each duopolist stands to make given that he/she thinks to level  $k$  and his/her competitor thinks to level  $n$ . Table 6 shows a number of steps of this computational landscape, that associates payoffs to different levels of reasoning of strategic managers in Firm 1 as a function of different levels of reasoning of strategic managers in Firm 2. Column 2 shows the quantity choice of Firm 1, while Column 4 gives the best response (the profit maximizing quantity selection) of Firm 1 if Firm 2 plays the best response to Firm 1's actual move (shown in Column 3). Whether or not it will be worthwhile for Firm 1 managers to engage in the additional computation needed to take them to the next level of iteration can be directly measured by assuming a fixed cost  $c_I$  and profit gain  $\Pi_i$  per iteration, in which case an additional computation will be worthwhile for Firm 1 if the net dollar benefit of thinking one step further  $(\Pi_i - c_I)$  is positive.

The computational landscape of this game model is displayed in Figure 2. The profits associated

with various strategy sets (corresponding to different levels of 'depth of thinking') are plotted on the vertical axis against the degree of logical depth (on the horizontal axis) to which Firm 1 and Firm 2 managers iterate the sequence given by  $t_n = (a - c - t_{n-1})/2$ ,  $n = 1, 2, 3, \dots, \infty$ , which converges to the Nash equilibrium output levels  $((a - c)/3, (a - c)/3)$  and the Nash equilibrium profit levels  $((a - c)^2/9, (a - c)^2/9)$ . After just 20 iterations, convergence to 0.01 percent of the Nash equilibrium level of output is achieved.

The computational landscape of Figure 2 can be 'explored' by the two firms through 'fictitious play' (i.e., 'what would competitors' best response be to my choice of output level  $q^*$ ?'), through experience-based learning (multiple serial realizations of the same game), or through historical analysis of similar past situations involving other competitive situations. Convergence to the Nash



Note: Horizontal axes represent number of iterations for each firm. Vertical axis is the profit level of Firm 1. Profit levels of Firm 2 are symmetrical. Landscape converges to Nash equilibrium output of  $(a-c)/3$

Figure 2. Computational landscape of Cournot Nash equilibrium, two firms,  $a = 3, c = 1$

equilibrium level of profits is monotonic in this particular game, but may not, in general, exhibit such convergence. If managers in Firm 1 ‘know’ (or, have valid reason to believe) that managers in Firm 2 are computationally bounded in a way that makes them *unresponsive* to marginal incentives to calculate more deeply, then it would be reasonable for them to choose the quantity that corresponds to the ‘best response’ to the expected, rather than the ‘ideal,’ quantity selection of Firm 2. If Firm 1 managers believe that Firm 2 managers will choose to produce  $3(a - c)/8$  units *because that is as far as they will think*, then it would be reasonable for them to produce the best response quantity of  $5(a - c)/16$  units, rather than the  $(a - c)/3$  units that represents the Nash equilibrium quantity choice. Thus, managers’ conjectures about competitors’ levels of logical potency can figure into the local cost-benefit calculus of strategic thinking. This approach is applicable to many competitive decision making problems: Milgrom and Roberts (1990) show that many games that have unique *equilibria* admit of psychologically intuitive formulations as iterative computational processes. This means that this sort of analysis can be applied to a large class of games to reconstruct their ‘computational landscapes’ and dynamics.

To reconstruct the computational landscape of a particular game in an experimental setting, one can manipulate the structure of the game and change the salience of the marginal cost of a single operation (by placing time constraints on each agent’s decisions, for instance) and the opportunity cost of foregone calculations. For instance, in the Cournot game one can accomplish this by altering the slope and intercept of the market demand function, which increases the sensitivity of the players’ payoffs to the degree of optimization that other players carry out. One can then make testable predictions about the *degree* of optimization that each player will individually perform, as a function of changes in payoffs net of opportunity costs, given that players have common knowledge of the structure of the game (strategies and payoffs) and of the rationality of the other agents, and can be informed or primed about the degree to which other players will carry out the level of optimization entailed by their rationality.

Table 7 outlines experimental interventions that can be used to shape three typical game structures that capture salient features of interactive decision making in business to the end of testing

hypotheses about the intensity of the optimization effort that participants exert over their own reasoning strategies. These interventions provide tests for the existence and responsiveness of adaptive forms of rationality. The opportunity costs of less intensive calculation (Column 3) can be manipulated by changing the market demand structure in an oligopolistic competition game, changing the payoffs to bilateral and unilateral defection and bilateral cooperation in a prisoner’s dilemma game, and by changing the payoffs to successful versus unsuccessful coordination in a pure coordination game. The marginal costs of calculation can be made more or less salient by imposing more or less stringent time constraints on individual decisions. The effects of conjectures about other participants’ computational intensity on the computational intensity of a player’s own optimization efforts can be manipulated by the explicit or implicit distribution of information regarding player ‘types,’ which can be accomplished either by priming subjects to think about what other subjects are thinking, or by priming subjects about the level of computational intelligence of other subjects. The impact of these variables on the computational effort of each participant can be inferred from observations of the strategic choices each player makes.

II. Why are certain problem statements more ‘popular’ than others among strategic managers? In particular, if the model introduced here is correct, why are *P*-hard problems preferred to *NP*-hard problems? There are several possibilities:

- A. One is that managers are not adequately *trained* in solving *NP*-hard problems—which can be tested by training experimental groups in the iterative search of large solution spaces and looking for change in problem solving behavior in ‘real life’ domains. An interesting additional research question is, then, one of cognitive skill *transfer* (Barnett and Ceci, 2002): Will training in solving *some NP*-hard problems result in improvements in solving *other NP*-hard problems? Is there a transfer of cognitive skill *across* semantic or activity domains but *within* a problem complexity class?
- B. Another is that managers—and human creatures more generally—develop an *intuition* for identifying and avoiding intractable problems that is ecologically adaptive, as solving such

Table 7. Experimental manipulations of three common game structures to determine effects of game structure and incentives on degree of strategic optimization of participating players

Game structure (common knowledge among players)	Salience of the marginal cost of a single operation	Opportunity cost of suboptimal calculation	Conjectures about other player(s)' logical prowess	Dependent variables to be measured
<b>Cournot-Nash oligopoly</b>	Time constraints on quantity decision	Slope and intercept of market demand function	Explicit stereotyping information about competitor(s) and subliminal priming	Strategic quantity decision and implied level of optimization
<b>Prisoner's dilemma with long but finite horizon</b>	Time constraints on cooperate/defect decision	Relationship among payoffs to cooperate/defect, cooperate/cooperate and defect/defect outcomes	Explicit stereotyping information about cooperator/defector and subliminal priming	Cooperate/defect decision on <i>k</i> -th instantiation of game and implied level of optimization
<b>Coordination game (matching pennies)</b>	Time constraints on focal point selection	Payoff difference between perfect and imperfect coordination	Explicit stereotyping information about co-coordinator and subliminal priming	Strategy selection and implied level of optimization

Note: The researcher can manipulate the structure of the game (Column 1), the salience of the cost of thinking for each player (Column 2), the opportunity cost of thinking (Column 3), the players' beliefs about other players' willingness to think (Column 4), and test the effects of these manipulations on the players' strategic choices (Column 5).

problems within the time constraints of most human predicaments *on average* exceeds the capabilities of the human cortex. If the limitation is not 'hard-wired,' but, rather, is the result of the brain's provisioning of microincentives to itself during the process of thinking, then we should find that managers will respond to an increase in their instantaneous computational facilities (a fast man-machine interface for calculating Nash *equilibria* in interpersonal interactions, pharmacological enhancements of pre-frontal cortex function) by engaging with *NP*-hard problems *adaptively*—that is, only if and when it is advantageous to do so.

- C. Yet another possibility is that managers have a *mal*-adaptive and nonresponsive fear of computational complexity that accounts for the lexicographic ordering I have conjectured: the fear of *NP*-hardness may be akin to a 'fear of the dark' that can dominate behavioral response even if and when it is locally advantageous for one to immerse oneself in darkness. It is well-known among neuro-physiologists that performing long sequences of logical consistency checks is depletive of brain glucose levels (Gray and McNaughton, 2003). Such a 'fear' then, can be understood as an aversion to the dysphoric affective state associated with performing long chains of calculations. One may then find that neither training nor the local manipulation of incentives will yield behavioral changes away from a preference for computational simplicity, but, borrowing maneuvers from behavioral therapists, that systematic *exposure* of managers to highly complicated but inescapable decision problems may increase their ability and willingness (or, 'proclivity') to engage in solving more complex problems.
- D. Still another explanation may be that managers (or, at least, some managers) are 'superrational' in the sense that they make cognitive choices on an expected-net-present-value basis, choosing to solve *NP*-hard problems when salient decision variables are few in number (and computational costs are manageable) and switch to *P*-hard problem statements for situations in which salient variables are numerous. Because the salience of any one variable is a function of the problem statement used

Table 8. A preliminary map for investigating the link between patterns of managerial reasoning (Column 1), managerial problem selection (Column 2), managerial attention (Column 3), and firm-level strategic response (Column 4)

Managers that exhibit this behavior in forced choices judgment formation experiments...	... are less likely to engage with this type of problem than are those who use more complex patterns of reasoning...	... and therefore are more likely to heed these firm- or industry-level variables...	... and use them to craft these types of strategies...
<i>Reason using representativeness or availability heuristic rather than incorporate base-rate knowledge and process Bayes' theorem</i>	<i>Iterated dominance reasoning to equilibrium in a competitive game</i>	<i>Existence, number and estimates of input factor costs of competitors</i>	<i>Adaptively optimize pricing and quantity levels of firm in response to changes in competitive dynamics</i>
<i>One-reason decision making</i>	<i>Calculation of centrality of a node in a partially connected graph</i>	<i>Formation of ties among industry players far away from the firm in the industry network</i>	<i>Adaptively optimize partnering strategies to maximize their firms' betweenness, closeness or Bonaccic centrality</i>
<i>Use simple search algorithms such as 'take the first option on the list' heuristic</i>	<i>Calculation of long-term evolution of densely coupled Boolean network</i>	<i>Changes in the average coupling among value-linked activity sets within the firm</i>	<i>Planning of activity sets that is adaptive to causal separability of causally linked activity domains</i>

to conceptualize a situation, careful empirical work is required to disentangle the various effects. If we impose the number and identity of salient variables on a manager, will problem statements change complexity class as a function of the time constraints imposed on managers to make decisions? On the other hand, if we impose a problem statement and associated complexity class, and expose a manager to various predicaments, will the number of variables he/she is willing to consider change as a function of the complexity class of the problem statements?

III. Managers' proclivity to exhibit certain patterns of thought (recognition, representativeness, and availability heuristics when processing large databases of incomplete information) can arise from a nonadaptive, individual, 'lexicographic' preference for solving *P*-hard problems over solving *NP*-hard problems that varies across individuals but does not vary within an individual. In this case, *individual differences* in patterns of judgment under uncertainty (whether or not managers commit the traditional 'fallacies,' for instance) can be used to predict managers' proclivity to engage in *other NP*-hard problems, such as those associated with game-theoretic reasoning and system design. Conversely, one might expect 'complex optimizers' (*NP*-hard problem solvers) to be less likely to reason by representativeness and availability in judgment-under-incomplete-information tasks. We can then expect to be able to predict the locus, scale, and scope of managerial attention ('how much and what managers pay attention to') from studies of managerial reasoning ('reasoning-out-loud protocols') and forced choice experiments that reveal the kinds of problems that managers use to construe a predicament ('how do you think about this quantity selection problem...?') Managers that supply the 'noise' samples in typical 'representativeness' or 'availability' bias experiments may be just the ones who will engage with solving *NP*-hard problems and adjust their firms' strategies when industry conditions that matter to the solution of such problems (but not to the solution of 'easier' problems) change (Table 8). This line of thinking opens up the possibility of combining experimental investigations of managerial judgment formation with

empirical investigations of strategic management behavior. Table 8 summarizes a number of examples of the link between managerial thinking style, managerial problem selection, the span of managerial attention, and the resulting strategy types we may expect to see.

Underlying the model developed above is, of course, a strict delineation of problem complexity classes, wherein *NP*-hard problems are fundamentally different from *P*-hard problems. That this delineation is valid is borne out by the fact that one cannot produce a global polynomial representation of an exponential or superexponential function, and also by repeated and failed attempts to reduce the *NP*-hard problems to sets of *P*-hard problems (Papadimitriou, 1994). One can, in some cases, achieve *approximate* answers, based on solving *P*-hard problems, to problems that are *NP*-hard to solve exactly. In this case, managerial algorithmics becomes the art and science of minimizing the opportunity costs of seeking less complicated and often less accurate solutions.

## ACKNOWLEDGMENTS

I acknowledge useful comments and inputs on the article from Terry Amburgey, Robert Bauer, Dan Levinthal, Roger Martin, Bill McKelvey, Dianne Nelles, Michael Porter, Jan Rivkin, Nikkolaj Siggelkow, Jitendra Singh, Olav Sorenson, and Howard Stevenson. I would like to thank *SMJ* Editor Will Mitchell and two anonymous referees for helpful criticisms and suggestions.

## REFERENCES

- Agre GP. 1982. The concept of problem. *Educational Studies* **13**(2): 121–142.
- Aragones E, Gilboa I, Postlewaite A, Schmeidler D. 2003. Fact-free learning. Working paper 03-023, Penn Institute for Economic Research, University of Pennsylvania, Philadelphia, PA. Available at: <http://pier.econ.upenn.edu/Archive/03-023.pdf>.
- Bargh J, Chartrand T. 1999. The unbearable automaticity of being. *American Psychologist* **54**(7): 462–479.
- Barnett SM, Ceci SJ. 2002. When and how do we apply what we learn? A taxonomy for far transfer? *Psychological Bulletin* **128**(4): 612–637.
- Black F, Scholes M. 1973. The pricing of options and corporate liabilities. *Journal of Political Economy* **81**(3): 637–654.

- Bodie, Z., Merton, RC. 1997. *Finance*. Prentice Hall: Upper Saddle River, NJ.
- Burt RS, Knez M. 1995. Kinds of third party effects on trust. *Rationality and Society* 7(3): 255–292.
- Bylander T, Allemang D, Tanner MC, Josephson J. 1991. The computational complexity of abduction. *Artificial Intelligence* 49(1–3): 25–60.
- Camerer C. 1997. Progress in behavioral game theory. *Journal of Economic Perspectives* 11(4): 167–188.
- Camerer C, Lovoal D. 1999. Overconfidence and excess entry: an experimental approach. *American Economic Review* 89(1): 306–318.
- Casti JL. 1992. *Reality Rules: Picturing the World in Mathematics*. John Wiley & Sons: New York.
- Chapman WL, Rozenblitt J, Bahill AT. 2001. System design is an NP-complete problem. *Systems Engineering* 4(3): 222–229.
- Chrisman JJ, Hofer CW, Boulton WR. 1988. Toward a system for classifying business strategies. *Academy of Management Review* 13(3): 413–428.
- Cohen MD, March JG, Olsen JP. 1972. A garbage can model of organizational choice. *Administrative Science Quarterly* 17(1): 1–25.
- Cooksey RW. 1996. *Judgment Analysis: Theory, Methods, and Applications*. Academic Press: San Diego, CA.
- Cormen T, Leiserson C, Rivest R. 1990. *An Introduction to Algorithms*. MIT Press: Cambridge, MA.
- Cowan DA. 1986. Developing a process model of problem recognition. *Academy of Management Review* 11(4): 763–776.
- Cyert RM, March JG. 1963. *A Behavioral Theory of the Firm*. Prentice-Hall: Englewood Cliffs, NJ.
- Czerlinski J, Gigerenzer G, Goldstein DG. 1999. How good are simple heuristics? In *Simple Heuristics that Make Us Smart*, Gigerenzer G, Todd PM, ABC Research Group. Oxford University Press: New York; 7–118.
- Dawes RM. 1998. Behavioral decision making and judgment. In *Handbook of Social Psychology* (Fourth Edition), Gilbert DT, Fiske ST, Lindzey G (eds). McGraw-Hill: New York; 497–548.
- Dixit AK, Nalebuff BJ. 1991. *Thinking Strategically: The Competitive Edge in Business, Politics, and Everyday Life*. W. W. Norton & Company: New York.
- Duhem P. 1991. *The Aim and Structure of Physical Theory (Princeton Science Library)*. Princeton University Press: Princeton, NJ.
- Garey MR, Johnson DS. 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness (Series of Books in the Mathematical Sciences)*. W.H. Freeman: San Francisco, CA.
- Gigerenzer G, Goldstein D. 1999. Betting on one good reason: the take the best heuristic. In *Simple Heuristics that Make Us Smart*, Gigerenzer G, Todd PM, ABC Research Group (eds). Oxford University Press: New York; 75–96.
- Gilboa I, Zemel E. 1989. Nash and correlated equilibria: some complexity considerations. *Games and Economic Behavior* 1(1): 80–93.
- Gintis H. 2000. *Game Theory Evolving: A Problem-Centered Introduction to Modeling Strategic Interaction*. Princeton University Press: Princeton, NJ.
- Gray RM, McNaughton N. 2003. *The Neuropsychology of Anxiety: An Enquiry into the Functions of the Septo-Hippocampal System, Second Edition (Oxford Psychology Series)*. Oxford University Press: New York.
- Hambrick D. 1983. Some tests of the effectiveness and functional attributes of Miles and Snow's strategic types. *Academy of Management Journal* 26(1): 5–26.
- Hogarth RM. 1975. Cognitive processes and the assessment of subjective probability distributions. *Journal of the American Statistical Association* 70(5): 271–289.
- Hogarth RM. 1980. *Judgement and Choice: The Psychology of Decision*. Wiley: Chichester, UK.
- Holt CA. 1995. Industrial organization: a survey of laboratory research. In *Handbook of Experimental Economics*, Kagel JH, Roth AE (eds). Princeton University Press: Princeton, NJ; 349–443.
- Inkpen AC, Choudhury N. 1995. The seeking of strategy where it is not: towards a theory of strategy absence. *Strategic Management Journal* 16(4): 313–323.
- Kahneman D, Lovoal D. 1994. Timid choices and bold forecasts: a cognitive perspective on risk taking. In *Fundamental Issues in Strategy (A Research Agenda)*, Rumelt RP, Schendel DE, Teece DJ. (eds). Harvard Business School Press: Boston, MA; 71–96.
- Kahneman D, Slovic P, Tversky A. 1988. *Judgment under Uncertainty: Heuristics and Biases*. Cambridge University Press: Cambridge, UK.
- Karp RM. 1972. Reducibility among combinatorial problems. In *Complexity of Computer Computations (Computer Applications in the Earth Sciences)*, Miller RE, Thatcher JW (eds). Plenum: New York; 85–103.
- Kilmann R, Mitroff I. 1979. Problem defining and the consulting/intervention process. *California Management Review* 21(3): 26–33.
- Kreps DM. 1990. *Game Theory and Economic Modeling (Clarendon Lectures in Economics)*. Oxford University Press: New York.
- Lyles MA, Mitroff II. 1980. Organizational problem formulation: an empirical study. *Administrative Science Quarterly* 25(1): 102–119.
- March JG. 1994. *A Primer in Decision Making*. Free Press: New York.
- March JG, Simon HA. 1958. *Organizations*. John Wiley: New York.
- Martignon L, Laskey KB. 1999. Bayesian benchmarks for fast and frugal heuristics. In *Simple Heuristics that Make us Smart*, Gigerenzer G, Todd PM (eds). Oxford University Press: New York; 169–189.
- McKelvey B. 1999. Avoiding complexity catastrophe in co-evolutionary pockets: strategies for rugged landscapes. *Organization Science* 10(3): 294–321.
- Milgrom P, Roberts J. 1990. Rationalizability, learning, and equilibrium in games with strategic complementarities. *Econometrica* 58(6): 1255–1277.
- Miller D, Friesen PH. 1978. Archetypes of strategy formulation. *Management Science* 24(9): 921–933.

- Miller GA. 1956. The magical number seven, plus or minus two: some limits on our capacity for processing information. *Psychological Review* **63**: 81–97.
- Mintzberg HA, Raisinghani D, Theoret A. 1976. The structure of unstructured decision processes. *Administrative Science Quarterly* **21**(2): 246–275.
- Moldoveanu MC, Bauer RM. 2004. On the relationship between organizational complexity and organizational structuration. *Organization Science* **15**(1): 98–118.
- Moldoveanu MC, Baum JAC, Rowley TJ. 2003. Information regimes, information strategies and the evolution of interfirm network topologies. In *Research in Multi-Level Issues, Volume 2*, Dansereau F, Yammarino F (eds). Elsevier Science, Ltd: Oxford, UK; 221–264.
- Newell A. 1990. *Unified Theories of Cognition*. Harvard University Press: Cambridge, MA.
- Newell A, Simon HA. 1972. *Human Problem Solving*. Prentice Hall: Englewood Cliffs, NJ.
- Nutt P. 1984. Types of organizational decision processes. *Administrative Science Quarterly* **29**(3): 414–450.
- Ocasio W. 1997. Towards an attention-based view of the firm. *Strategic Management Journal* Summer Special Issue **18**: 187–206.
- Papadimitriou C. 1994. *Computational Complexity*. Addison-Wesley: Reading, MA.
- Popper KR. 1972. *Objective Knowledge: An Evolutionary Approach*. Oxford University Press: New York.
- Pounds W. 1969. The process of problem finding. *Industrial Management Review* **11**(1): 1–19.
- Quine WV. 1951. Two dogmas of empiricism. *Philosophical Review* **60**(1): 20–43.
- Rivkin J. 2000. Imitation of complex strategies. *Management Science* **46**(6): 824–844.
- Rokeach M. 1960. *The Open and Closed Mind*. Basic Books: New York.
- Ryall MD. 2003. Subjective rationality, self-confirming equilibrium, and corporate strategy. *Management Science* **49**(7): 936–949.
- Saloner G. 1994. Game theory and strategic management: contributions, applications and limitations. In *Fundamental Issues in Strategy (A Research Agenda)*, Rumelt RP, Schendel DE, Teece DJ. (eds). Harvard Business School Press: Boston, MA; 155–194.
- Schwenk CR. 1984. Cognitive simplification processes in strategic decision making. *Strategic Management Journal* **5**(2): 111–128.
- Simon HA. 1947. *Administrative Behavior*. Free Press: New York.
- Simon HA. 1996. *The Sciences of the Artificial* (3rd edn). MIT Press: Cambridge, MA.
- Taylor RN. 1975. Perception of problem constraints. *Management Science* **22**(1): 22–34.
- Turing A. 1997. Computing machinery and intelligence. In *Mind Design II: Philosophy, Psychology, and Artificial Intelligence*, Haugeland J (ed). MIT Press: Cambridge, MA; 29–56.
- Tversky A, Kahneman D. 1986. Rational choice and the framing of decisions. *Journal of Business* **59**(4): 251–278.
- Volkema RJ. 1988. Problem complexity and the formulation process in planning and design. *Behavioral Science* **33**(4): 292–300.
- Wasserman S, Faust K. 1994. *Social Network Analysis: Methods and Applications*. Cambridge University Press: Cambridge, UK.
- Watts DJ, Dodds PS, Newman MEJ. 2002. Identity and search in social networks. *Science* **296**(5571): 1302–1304.
- Winship C, Mandel M. 1983. Roles and positions: a critique and extension of the blockmodeling approach. In *Sociological Methodology 1983–1984*, Leinhardt S (ed). Jossey-Bass: San Francisco, CA; 314–344.
- Wymore AW. 1993. *Model-Based Systems Engineering*. CRC Press: Boca Raton, FL.
- Zajac E, Bazerman M. 1991. Blind spots in industry and competitor analysis: implications of interfirm (mis)perceptions for strategic decisions. *Academy of Management Review* **16**(1): 37–96.