OLIVEIRA, F.S. 2018. A creativity support system based on causal mapping. *Journal of computer information systems* [online], 58(2), pages 99-109. Available from: https://doi.org/10.1080/08874417.2016.1202089

A creativity support system based on causal mapping.

OLIVEIRA, F.S.

2018

Fernando S. Oliveira (2018) A Creativity Support System Based on Causal Mapping, Journal of Computer Information Systems, 58:2, 99-109, DOI: 10.1080/08874417.2016.1202089

A Creativity Support System Based on Causal Mapping

Fernando S. Oliveira

ESSEC Business School, Cergy-Pontoise, France

Abstract: Theory development is a very complex process that requires creativity and highly specialized analytical skills. This article presents a new algorithm, based on causal mapping, for assisting in the creation of qualitative theories. This algorithm is able to conjecture and prove new theorems, to test for consistency and completeness of the theory, and to derive meta-theorems comparing the different concepts in it. The use of the algorithm is exemplified in developing a theory to explain structural inertia in organizations.

KEYWORDS: Artificial intelligence; causal mapping; computational creativity; knowledge management; logic modeling

Introduction

The development of computer programs capable of enhancing creativity has the potential to significantly have an impact on the evolution of human societies [3]. For example, it has been shown that greater employee creativity is associated with better business excellence [28, 46] and that creativity-enhancing decision support systems can be used to improve the creative response of their users [e.g., 27, 29, 46–48, 52, 63] contributing to value creation. (However, Cheung et al. [13] have reported that the use of repository databases for knowledge re-use can be detrimental for the creative output of the most skilled individuals.) Examples of successful applications of creativity supporting systems include, among others, the modeling of analogical reasoning [12], new product development [39], task location scouting [52], and internet-based methods to conduct creativity sessions [4].

Cognitive mapping techniques [e.g., 25, 26, 49, 64, 72] that aim to extract subjective knowledge from individuals, and to represent them in a graphical way, have been used in developing creativity supporting systems. There are several different cognitive mapping techniques used in practice [e.g., 25, 26, 64]. Causal maps [e.g., 2, 1, 31, 33, 53] aim to represent the concepts and the causal relationships between them, allowing the modeling of complex chains of arguments [53]. Concept maps attempt to identify the relationships (which can be bidirectional) between the different concepts (which do not need to represent causality) and aim to generate ideas and to help in knowledge development by integrating old and new knowledge. Semantic maps are used to explore an idea by listing other ideas connected to it, and are helpful in obtaining a better understanding of an individual's belief system; these are also known as mind maps [e.g., 11]. Influence diagrams are graphical models used to represent complex decision processes, based on uncertain information, allowing the development of probabilistic models (based on Bayesian networks) from expert knowledge [e.g., 5, 14, 23]. An in-depth discussion of cognitive mapping, and respective techniques, can be found in References [26] and [64].

This article is focused on causal maps [e.g., 2, 33, 53]. Eden's [26, p. 1] definition of causal maps (Eden actually calls them "cause maps") is the following: these maps are networks of nodes and arrows where the direction of the arrow from one node to its neighbor implies believed causality. As explained by Siau and Tan [64], these causal maps represent a set of causal relationships between constructs within a belief system. These causal maps may not relate to individual cognition as they may represent, instead, the beliefs of a group of individuals or, as in the problem addressed in this article, a theory commonly shared by the scientific community to explain a phenomenon.

The study of how researchers interact with each other, and with the object of their studies, in scientific research, is one of the issues addressed by the new paradigm research methods [e.g., 38, 60, 61] that see people as co-creating their reality through participation. As summarized in Reference [61], from the perspective of these research methods, there are at least four kinds of knowledge: (i) experiential, gained through direct interaction with the others or things; (ii) practical, the knowledge of how to do something; (iii) presentational, the process by which we order our tacit experiential knowledge into a pattern (it works as a "bridge" between experiential and propositional knowledge); (iv) propositional, the knowledge about something expressed in statements and theories. As emphasized by the new paradigm research methods, if the propositions are generated exclusively by a researcher who is not involved in the experience being researched, and are derived without taking into account the practical and experiential knowledge of the subjects, the findings are not valid.

From the perspective of the new paradigm research methods, propositional knowledge cannot give an absolute account of what there is, it can only give a mediated, subjective, and inter-subjective account of reality; that is, reality is considered a social construct. The validity of such knowledge requires the researcher to use critical subjectivity, that is, to recognize the subjective nature of knowledge, and to accept that subjectivity is part of the experiential articulation of reality, and to use this self-reflection when producing propositional knowledge.

The causal map techniques analyzed in this article are used to assist in the production of propositional knowledge and, in conjunction with the logic-based algorithm proposed in this article, they allow the development of consistent, sound, and complete theories, helping in the transformation of practical and experiential knowledge into propositions. Moreover, the use of causal maps also allows the surfacing of tacit knowledge (possibly multidisciplinary), and the analysis of the relationships between a very large number of concepts (possibly contradictory), keeping the causality link. In this context, the analysis of causal maps with logic tools, as proposed in this article, has the advantage of being able to detect contradictions and to find patterns in the causal links, and to discover incompleteness in the causal explanations. For all these reasons, the use of logic tools, together with causal maps, may help in knowledge creation by allowing the researchers to develop a stronger critical subjectivity, as required by the new paradigm research methods [e.g., 38, 60, 61].

Interesting enough, even logic can be, to a certain extent, subjective. Most specifically, material implication [e.g., 73] can be difficult to interpret and to translate into current speech in theory formation, possibly introducing a degree of subjectivity in the expression of theories, due to its wellknown paradoxes: whenever the antecedent is false, the conditional is true; whenever the consequent is true, the conditional is true. For this reason, the methodology followed in this article uses causal implication [10, 37]. In causal implication, all the propositions mean "causality sufficiency" as described by Burks [10]. This means that the antecedent may contain irrelevant conditions for the causal relationship to hold and still the proposition is correct. Henderson [37] exemplifies the difference between material and causal implications: the proposition "If my dog has a white tail it will die" is false in causal implication, but it is true in material implication, when the dog does not have a white tail.

The problem addressed in this article is the one of using computational algorithms to help in improving the quality of the theories produced (in terms of transparency of causal relationships and absence of contradictions) with resource to causal maps, to maintaining consistency in complex networks. Most importantly, given the possible complexity of the causal map, there are potentially many insights that can be proved from the basic causal relationships between the different concepts that are difficult to deduce. This creative process requires, first and foremost, the ability to generate new conjectures (these are the possible causal relationships that have not been stated in the causal map but are implicit in it) and then the procedure to prove that the conjectures do hold true in the network of causal relationships considered in the theory. When these conjectures become theorems (i.e., when they are proved to be true in the context of the base theory), these relationships become explicit in the causal map.

Additionally, causal maps enable a better identification of the interactions between different concepts, revealing the existence of common factors to explain them, enabling the discovery of causal rules of behavior previously unknown to the researcher and, as in the case of the meta-theorems presented in this article, leading to the creation of new concepts. These meta-theorems represent causal patterns in the data that explain the joint emergence of a set of consequences. In terms of the causal diagram, this corresponds to the construction of a new set of concepts and causal relationships between them (this is a meta-graph, developed using as base the causal diagram).

To this effect, this article presents a creativity supporting system for theory development. The article discusses the use of causal maps as ways to develop theories, proposing an algorithm for theory development and testing. The proposed algorithm uses first-order predicate logic [69, 70] and, through a process of inference from the basic axioms of the theory, it is able to develop a set of new theorems that are theoretically consistent (i.e., the theory is contradiction-free), sound (i.e., all the inferences in the theory and logically correct), and contingent (i.e., the theory is falsifiable and not based on tautologies).

As an application of our methodology, we analyze the inertia fragment of organizational theory [35, 77]. The methodology used in the article is similar to the logic approach used in the literature [e.g., 41, 57], but it differs from it in clear ways: the algorithm is able not only to create theorems relating multiple properties to one property but also meta-theorems relating multiple properties to multiple properties. These meta-theorems, by pulling together multiple properties in the consequent part of the conditional relationship, are indeed creating new concepts, and not just new relationships between concepts. Moreover, the methodology is based on causal implication and not on material implication. The causal maps used in this article also differ from the methodology in References [11, 14, 21, 23, 25–27, 53, 55, 64], as we tend to rely on tables to represent the relationship between the variables, instead of graphical representations, which, nonetheless, may be used by the researcher as a supporting tool.

The article is organized as follows. The next section gives a background on computational creativity. Then an algorithm for theory development is presented. "An application to the analysis of structural inertia in organizations" discusses the basic premises of the inertia fragment of the theory of organizational ecology, and applies the algorithm to the analysis and development of this theory. The last section concludes the article.

Background on computational creativity

Creativity has long been recognized as an important factor to explain the success of people and organizations. For example, Edward de Bono's [21] work on creative and lateral thinking has looked at the implications of creativity-enhancing methods on the management of organizations that want to unleash the creative power of their collaborators. It is, therefore, evident that tools capable of enhancing individual and organizational creativity have an important social value. This creative process can result both from the exploration and transformation of conceptual spaces, for example, by using reasoning by analogy [59, 64], or be the product of an incremental problem-solving process that does not arise from conceptual restructuring [e.g., 8, 9].

Boden [9] has identified three types of creativity: (i) combinatorial creativity, involving novel (improbable) combination of familiar ideas, which she exemplifies with Jape [7]; (ii) exploratory creativity, representing the generation of novel ideas by the exploration of structured conceptual spaces, which Boden exemplifies with the classics AARON [50], Bacon [43], and EMI [18]; (iii) transformational creativity, involving the transformation of some dimension(s) of the conceptual space, so that new ideas can be generated which could not have arisen before, which she exemplifies with AM and Eurisko [44]. Exploratory creativity is very well adapted to classical artificial intelligence methods, for example, Harvey [36] developed SAGA, a model that adapts the basic genetic algorithms framework for discovery and exploration tasks; Pereira et al. [58] proposed genetic algorithms for generation and neural nets for evaluation of new ideas; Bentley [6] used evolutionary algorithms for creative exploration.

It is, therefore, evident that Boden's definition of creativity is still subjective, that is, a given idea can be classified either as creative or not, depending on the process used to achieve it (intentional purpose), [e.g., 71]. Wiggins [74] has attempted to clarify Boden's ideas on creativity by proposing a formalization of this concept, to enable a detailed comparison of systems that exhibit behavior that in humans would be called creative. Wiggins has concluded that Boden's concept of creativity, even though subjective, is rather powerful in allowing the assessment of the creative potential of different systems.

Wiggins' framework starts by defining the universe of possibilities (the multi-dimensional space), the basic axioms defining it, and its conceptual spaces. These conceptual spaces are generated by two distinct rule sets that allow the definition of their boundaries and the search of new concepts to be added to them. Nonetheless, subjectivity is also present in Wiggins' framework, due to the need to use evaluation rules to assess "quality, according to whatever criteria we may consider appropriate" [74, p. 453]. For this reason, Wiggins' framework, even though representing an important step to formalizing Boden's creativity concept (and, in this sense, illustrating how the concepts can be translated into computational procedures), very naturally, still retains the subjective aspects attached to the evaluation of novelty.

This issue, the evaluation of novel ideas, is indeed one of the major problems faced by the research in computational creativity, as it is always problematic to assess the outputs of the system in terms of creative contribution (as it is difficult, in general, to assess the creativity of a person). This issue has been addressed by Colton [15], who developed a system in which a conjecture generation model, HR [16], interacts with a theorem proving system, Otter [51], in order to discard the conjectures produced by HR that are trivial for the Otter to prove. The HR has information about the domain studied, including objects of interest, initial concepts, and examples. The HR is able to build new concepts and to make conjectures [15, 16]. Colton's HR has been extended by Pease et al. [56] to allow the production of conjectures with known counterexamples, to include the analysis of faulty conjectures, and to use a multi-agent approach in which agents are able to request and communicate with each other. Under this approach, the threshold considered to assess novelty is the one of discarding trivial conjectures. Hence, novel ideas are not creative if they do not pass this minimum threshold.

Regarding the evaluation problem, a second school of thought argues that the creative process is as a social-cultural construct that cannot be replicated by a closed system within a single agent [19, 20]. This is an important point as if, by definition, creativity implies the creation of something new, however, it is not possible to create something from nothing, and the creation needs to be appreciated by someone who knows its value. This issue was addressed by Garfield et al. [30] who haveanalyzed how individual creativity is influenced by exposure to others, and have explained how the types of techniques used in teamwork have a significant impact on creativity. Therefore, from this perspective, creativity is a process that can be observed only at the intersection where individuals, domains, and fields interact. A model of creativity that takes this into account was proposed by Saunders and Gero [62], based on a multi-agent system in which agents interact to generate creative ideas, and by Zou and Yilmaz [78] to study the behavior of global participatory science communities.

The review of the work on computational creativity clearly identifies the subjectivity of the evaluation process as one of the main issues faced by the researchers in the area. This issue, so far, has been dealt with by developing algorithms that have the ability to self-select (to a certain extent) the concepts produced, and by using agent-based systems where the ideas are produced collectively by the interaction between the different algorithms. For this reason, the creativity support system presented in this article aims to helping in theory development by assisting the researcher in the production of new theorems. The researcher is the ultimate judge of the quality of the results produced by the algorithm.

As seen from our review, there are very different approaches to using computer software for enhancing creativity, including analogical reasoning [12], cognitive mapping [e.g., 2, 11, 25, 26, 33, 64, 72], group decision support systems [17, 37, 42, 54, 67], and evolutionary algorithms [6, 58]. Obviously, these different methods have strengths and weaknesses and tend to be chosen depending on the specific context in which they are employed. The specific approach followed in this article aims to helping researchers and practitioners in developing causal models when formalizing theories or when analyzing a problem, they may face in a business environment. For this reason, the methodology proposed is based on two components: (i) the exploration of possible logical contradictions in the causal maps, helping in developing consistent and sound views of the problem; and (ii) the identification of new relationships between known concepts, and in the construction of new concepts, based on currently consistent, sound, and complete causal maps.

The next section presents an algorithm for automated theory development and testing.

An algorithm for automated theory development and testing

The algorithm aims to assist in the production of new qualitative theories. The production of such theories can be very complex due to the large number of concepts involved, the possible requirement to integrate concepts from different areas of knowledge, and the subjective meaning and sophistication of the natural language used in their formulation. For these reasons, a creativity support algorithm can assist in theory development by: (i) clarifying the meaning of the previously held assumptions and theorems, with theconsequent increasein theobjectivityofthe theories produced; (ii) by checking consistency between the assumptions and theorems produced; (iii) by enabling the creation of new theorems from the basic assumptions.

A theory has been defined as a set of sentences not necessarily closed [75]. Such a theory can be generated with the help of a computational algorithm for theory development and revision [e.g., 22, 32, 34, 75]. In order to formalize a given theory using first-order predicate logic, start by defining: (i) variables – these are the basic components of the theory; (ii) predicates – names of the concepts or properties (i.e., relationships between variables), or names of the constraints on a given variable; (iii) connective symbols which stand for the words used to combine different statements, in this case the symbols $\neg$, $\wedge$, $\vee$, $\rightarrow$ stand for, respectively, "not," "and," "or," and "causal implication".

The algorithm proposed in this article was developed using Prolog [65, 66]: this is a specialized language for programming in logic which allows an easy representation of recursive reasoning, and a fast development of the interpreter used to build the causal map representing a given theory. This language uses first-order predicate logic as its base and all the sentences are presented in the conditional format, that is, as implications [e.g., 69, 70]. The use of first-order predicate logic enables the simplification of the mathematical representation of the formulas, rending them easier to read and understandable by a wider audience.

The algorithm includes two main components: (i) a theory tester, and (ii) a theorem generator. Whereas the theory tester checks the initial theory for completeness and consistency, the theorem generator creates new theorems, and meta-theorems, from the initial knowledge. The algorithm starts by checking that the theory is complete. (A theory is said to be complete if every theorem in the theory can be proved by deduction from the assumptions.) Then, it analyzes the consistency of the initial theory. The initial theory can have theorems or assumptions that are contradictory. In this case, the theory tester identifies, and corrects, these inconsistencies.

The algorithm then proceeds with the theorem generator. It looks at all known concepts in the theory and, by a process of deduction, collects and puts together all the properties used to prove the relationship under analysis taking, simultaneously, into account the assumptions required for the new theorem to hold. Finally, the theorems and assumptions are analyzed in order to develop meta-theorems relating multiple properties, creating, in effect, new concepts. These meta-theorems are a complement to ordinary theorems that only compare basic objects.

Before proceeding, it is useful to clarify where the original concepts and causal relationships between them come from. If the problem addressed is the creation, and development, of a consistent, sound, and complete theory, the original concepts, and basic relationships between them, are defined by the researcher using his/her knowledge of the problem and taking into consideration the issues he/she aims to explain with the theory.

The theory tester

The theory tester is an automated theorem prover with an extension to correct the theory when inconsistencies are found. First, it checks if the theory is complete, i.e., every theorem in the theory must be proved from other theorems and assumptions. The procedure used to check completeness is prove(Goal, Result), presented in Table 1, in which Goal is a property of the objects in the theory, and Result equals "T" if the Goal is proved true and "F", otherwise:this procedurechecks if a given theorem (Goal)can be proved from the known theorems and facts. All the concepts in the theory are tested.

The algorithm considers three different cases in its attempt to solve prove (Goal, Result). First, if Goal is a fact in the theory, then the problem is solved as true, T. Second, if Goal is an implication with a body composed of other Goals, then the algorithm recursively needs to prove that the body in the implication for Goal is true. Finally, if there is at least one element in the body of the implication for Goal that fails, then the proof of Goal fails.

The procedure used to analyze the body of Goal is described in step (2). In order to prove a given conjecture, the algorithms is required to prove every single argument (A to Z) in the "body" of that conjecture: this is achieved by using the procedure solve (Goal_body,Result) in Table 1. For this purpose, it calls the procedure solve(A ^ B ^ ... ^ Z, Result) that identifies the conditions for the main Goal to be true and then proves these conditions, one by one. If the proof was successful the procedure returns Result = T. If the procedure failed to prove one of the conditions then, if the condition is proved to be false, the procedure terminates with Result = F; otherwise, if the condition is not known to be true or false (i.e., there is no supportive fact or counterexample), then the condition is added as a fact to the theory (as it is subsumed by one of its theorems and not known to be false).

In step (3) the algorithm checks the presence of inconsistencies in the base theory. First, (3.a) verifies that, for every assumption and theorem in the theory its negation is not true. Second, (3.b) detects another source of contradiction not prevented by the previous test, $\neg p \rightarrow p$, as this proposition is considered true under material implication rules [e.g., 73]. Under material implication rules if the precedent is false, the implication rule is evaluated as true. In the case of derivation of causal implication rules, this implication cannot be accepted as true as the absence of a precedent (which is interpreted as a false statement) does not, in general, cause the subsequent to be true.

Table 1. The prove(Goal, Result) procedure.

(1) prove(Goal, Result): Prove Goal getting a Result (*T* if the Goal is proved or *F*, otherwise).
(1.a) If fact(Goal) $\rightarrow$ Result = *T*.
(1.b) Else If (Goal_body $\rightarrow$ Goal) and solve(Goal_body, *T*) $\rightarrow$ Result = *T*.
(1.c) Else Result = *F*.
(2) solve(A $\wedge$ B $\wedge \ldots \wedge$ Z, Result): Decompose the Goals in the body of the rule into several goals, A $\wedge$ B $\wedge \ldots \wedge$ Z, and prove each one of them.
(2.a.A) prove(A, ResultA)
(2.a.B) prove(B, ResultB)
(2.a.Z) prove(Z, ResultZ)
(2.b) Result = ResultA $\wedge$ ResultB $\wedge \ldots \wedge$ ResultZ.
(3) Detect inconsistencies in the theory:
(3.a) Prove that, for every concept in the model, we do not have, simultaneously, $p \rightarrow q$and $p \rightarrow \neg q$ as true.
(3.b) Detect any implications of the type $\neg p \rightarrow p$.
(4) Correct inconsistencies
(4.a) Remove inconsistent theorems and assumptions
(4.b) Add to the theory as a fact any concept *completely* removed in 4.a, if it is required to prove any other assumption or theorem.

In step (4), if it finds inconsistent theorems or assumptions, it corrects them. First, step (4.a) removes the conflicting theorems and assumptions from the theory: this removal may cause logical inconsistencies in other concepts whose proof depends on the removed theorems and assumptions. For this reason, in step (4.b), any concept in the removed properties (assumptions and theorems) that is not part of the theory is added as a fact (i.e., an assumption that is known to be true) if it is required to prove other theorems in the theory. For example, if the following causal implications are found in the causal map: $p \rightarrow q$, $p \rightarrow \neg q$, $q \rightarrow c$ and p is a fact, then in step (3.b) the causal implications $p \rightarrow q$ and $p \rightarrow \neg q$ are both removed. From the remaining rules ($q \rightarrow c$ and p is a fact), we cannot prove the truthfulness of the causal implication $q \rightarrow c$. For this reason, we need to add to the theory the fact that assumption q is known to be true. Obviously, the model produced depends on this additional assumption. If it is known, as a fact, that q is always false, or that one of the two causal implications $p \rightarrow q, p \rightarrow \neg q$ is correct, one of these can be added to the base set of assumptions, and the theory tester is used again on the new set of assumptions.

The theorem generator

The theorem generator, as defined here, aims to develop concise theorems, easy to read and understand, which relate multiple properties to one property of the objects analyzed, minimizing the description of the sufficient conditions for the relationship to hold. Furthermore, it contains one operator for deriving meta-theorems, that is, concepts based on other concepts in a given theory (which relate multiple properties to multiple properties): conditional theory equivalence (C-equivalence, Definition 1). The theorem generator procedure is presented in Table 2.

> **Definition 1 (C-equivalence)**: The concepts $q_1, \ldots, q_Q$ are said C-equivalent if for some instances of the theory with conditionals $c_1, \ldots, c_M$ and properties$p_1, \ldots, p_N$, for all i in $1 \ldots Q$, it is true $c_1 \wedge \ldots \wedge c_M \wedge p_1 \wedge \ldots \wedge p_N \rightarrow q_i$.

C-equivalence postulates that any two conditions that have the same causal antecedent are equivalent. First, it should be noted that this definition is helpful in identifying the common causes of observed phenomena, enabling the construction of the meta-theorems discussed next. Moreover, C-equivalence is related with the principle of sufficiency in causal implication [10, 37]. As reported, under causal implication two clauses with different precedents (but some of them in common) may have the same consequence if some of the common precedents dominate all others. C-equivalence is more restrictive than the principle of sufficiency, as it imposes that the declared antecedents are exactly the same in the compared concepts.

The theorem generator in step (1) collects the concepts in the complete and consistent theory analyzed (Base_theory). Then, in (2.a), for each one of these concepts q, it finds all the properties $p_1, \ldots, p_N$ used to prove them. (These properties are the other concepts used to prove the q.) Then, in (2.b), it finds all the conditions $c_1, \ldots, c_M$ used to prove the relationship $p_1 \wedge \ldots \wedge p_N \rightarrow q$. These conditions $c_1, \ldots, c_M$ are a subset of all the conditions used to prove q. (This subset includes only the conditions not required by the properties $p_1, \ldots, p_N$.) As the proof of the theorems is primarily based on concepts (properties), the theorem generator can produce concise theories. Then, in (2.

c), the algorithm puts together the properties and conditions to produce the final theorem $c_1 \wedge \ldots \wedge c_M \wedge p_1 \wedge \ldots \wedge p_N \rightarrow q$.

Table 2. The generator(Base_theory, New_theorems) procedure.

q: consequent in a given assumption or theorem, which represents a concept
$c_1, \ldots, c_M$: *constraints* on the organizations under which a given assumption or theorem is true $p_1, \ldots, p_N$: properties or concepts that are precedents in a given assumption or theorem
Base_theory=$\{q|q \text{consistentandcorrect}\}$
generator(Base_theory, New_theorems): generate New_theorems starting from a given complete and consistent Base_theory.
(1) Find all the consequents q in the Base_theory
(2) For each consequent q:
   (2.a) Find all the properties $p_1, \ldots, p_N$ used to prove q.
   (2.b) For each consequent q, find all the constraints $c_1, \ldots, c_M$ used to prove the implication $p_1 \wedge \ldots \wedge p_N \rightarrow q$.
   (2.c) Return a new theorem: $c_1 \wedge \ldots \wedge c_M \wedge p_1 \wedge \ldots \wedge p_N \rightarrow q$.
(3) Find all the concepts $q_1, \ldots q_F$ that are C-Equivalent and return the meta-theorems of the form: $c_1 \wedge \ldots \wedge c_M \wedge p_1 \wedge \ldots \wedge p_N \rightarrow q_1 \wedge \ldots \wedge q_F$.
(4) Return all the new theorems and meta-theorems for this theory, New_theorems.

In step (3), the algorithm applies the C-equivalence operator to the analysis of the basic theory deriving the meta-theorems summarizing the relationship between the different concepts in the theory. The application of the C-equivalence operation requires the following steps: (i) get the list of all the propositions for every consequent q_i; (ii) compare these propositions for the different consequents, to find common causal effects; (iii) create a new concept, which is the conjunction of all the consequents with the same causal explanations, and write down the meta theorem $c_1 \wedge \ldots \wedge c_M \wedge p_1 \wedge \ldots \wedge p_N \rightarrow q_1 \wedge \ldots \wedge q_F$. In terms of the causal map, this corresponds to the construction of an alternative view of the problem in which in a new causal map (a meta-map) the new concept replaces all the consequents that were put together in that consequent. Finally, it collects all the new theorems and meta-theorems created to return a set of New_theorems.

Fundamental properties of the algorithm

A fundamental question we need to answer regarding the behavior of the algorithm is: can the algorithm create theorems and meta-theorems that are correct and consistent with the base theory? The answer is yes, as proved in Propositions 1 and 2.

Proposition 1: The theorem generator, when applied to a complete and consistent theory, creates correct and consistent theorems.

Proof: Let $c_1 \wedge \ldots \wedge c_K \wedge p_1 \wedge \ldots \wedge p_Z \rightarrow q$ represent an original proposition (theorem or assumption) associated to concept q. As this proposition is sound, we know that when $c_1, \ldots, c_K, p_1, \ldots, p_Z$ is true then q is true as well. Furthermore, as the basic theory is consistent, then there are no contradictions in the precedent of proposition q. In order to produce the new theorem, we look in the precedent properties for other concepts subsumed in the original concept. Let, without loss of generality, $p_1(p_{11}, \ldots, p_{1k}), \ldots, p_Z(p_{Z1}, \ldots, p_{ZL})$ represent the basic properties. Then, the new theorem will be $c_1 \wedge \ldots \wedge c_K \wedge p_1 \wedge \ldots \wedge p_Z \wedge p_{11} \wedge \ldots \wedge p_{1K} \wedge \ldots \wedge p_{Z1} \wedge \ldots \wedge p_{ZL} \rightarrow q$. By definition of completeness, and as each one of the properties $p_1, \ldots, p_Z, p_{11}, \ldots, p_{1K}, ..p_{Z1}, \ldots, p_{ZL}$ are in the original theory, they all are sound and can be proved from the basic assumptions. As $p_{11}, \ldots, p_{1K}$ are in the precedent for p_1 they are consistent with each other, moreover, as $p_{Z1}, \ldots, p_{ZL}$ are in the precedent for p_Z they are also consistent with each other. Finally, as q has been proved correct, and $p_1, \ldots, p_Z$ are consistent, then the respective precedents are also consistent with each other, that is, $p_{Z1}, \ldots, p_{ZL}$ are consistent with $p_{Z1}, \ldots, p_{ZL}$. In conclusion, $c_1 \wedge \ldots \wedge c_K \wedge p_1 \wedge \ldots \wedge p_Z \wedge p_{11} \wedge \ldots \wedge p_{1K} \wedge \ldots \wedge p_{Z1} \wedge \ldots \wedge p_{ZL} \rightarrow q$ is true, as all the elements of the precedent are true and consistent with each other.

Proposition 2: *Let $c_1, \ldots, c_M$ represent conditionals and $p_1, \ldots, p_N$ the properties for which C-equivalence holds for a set of concepts $q_1, \ldots, q_Q$. Then, the theorem generator, when applied to a complete and consistent theory, creates correct and consistent meta-theorems of the form $c_1 \wedge \ldots \wedge c_M \wedge p_1 \wedge \ldots \wedge p_N \rightarrow q_1 \wedge \ldots \wedge q_Q$.*

Proof: From Definition 1, if Q concepts are C-equivalent then, for the respective conditionals $c_1, \ldots, c_M$ and properties $p_1, \ldots, p_N$, we have $c_1 \wedge \ldots \wedge c_M \wedge p_1 \wedge \ldots \wedge p_N \rightarrow q_1, \ldots, c_1 \wedge \ldots \wedge c_M \wedge p_1 \wedge \ldots \wedge p_N \rightarrow q_Q$. From the rules of implication [e.g., 69, 70], it follows that we can rewrite these propositions as $\neg(c_1 \wedge \ldots \wedge c_M \wedge p_1 \wedge \ldots \wedge p_N) \vee q_1 \quad \ldots \neg(c_1 \wedge \ldots \wedge c_M \wedge p_1 \wedge \ldots \wedge p_N) \vee q_Q$. If all these propositions are simultaneously true then, by the rules of the interception (*and* operator), it follows that their interception is also true, and therefore, we get $[\neg(c_1 \wedge \ldots \wedge c_M \wedge p_1 \wedge \ldots \wedge p_N) \vee q_1] \wedge \ldots \wedge [\neg(c_1 \wedge \ldots \wedge c_M \wedge p_1 \wedge \ldots \wedge p_N) \vee q_Q]$. This proposition, by the distributive laws, is equivalent to $\neg(c_1 \wedge \ldots \wedge c_M \wedge p_1 \wedge \ldots \wedge p_N) \vee (q_1 \wedge \ldots \wedge q_Q)$, which from the rules of implication is equivalent to $c_1 \wedge \ldots \wedge c_M \wedge p_1 \wedge \ldots \wedge p_N \rightarrow q_1 \wedge \ldots \wedge q_Q$.

Propositions 1 and 2 are important as they guarantee that, after the theory tester (Table 1) assures that there is no contradiction in the base theory, the theorem generator (Table 2) can construct conjectures and prove them, both as theorems and meta-theorems, and that the new enlarged theory is consistent, sound, and complete. These two mechanisms (theory tester and theorem generator) search in the space of concepts. The theory tester may challenge the user of the system (e.g., a researcher) to find explanations for the inconsistencies, possibly requiring the re-writing of some of the propositions in the base theory, and raising questions about the current knowledge. The theorem generator proves new propositions and creates new concepts (the meta-theorems) that the user may find innovative and challenging (this second case, possibly the most constructive one, would prompt a revision of the base theory).

An application to the analysis of structural inertia in organizations

This section applies the algorithm to analyzing the theory of structural inertia in organizations by Hannan and Freeman [35], illustrating how this creativity-supporting system enables the researcher to clarify the concepts and to state the interactions between them as causal maps, represented in a table format. This theory has had a very important impact in the area of organizational behavior and social sciences, in general, and it has been reported in the past in References [57, 77] to have some issues regarding the correct way of interpreting the meaning of some of its propositions.

The theory of structural inertia in organizations was first written in an informal way, using assumptions and theorems, which gives a basic framework to start with and, as the initial theory had some consistency issues, it is a good test set for the algorithm. A first difficulty faced when translating this theory into first-order predicate logic is the identification of the concepts involved and, sometimes, the interpretation of the subjective meaning of the words used in the assumptions, and theorems, of the original theory. This subjectivity may rend the process of representing the original ideas as causal relationships very difficult indeed. In this context, the use of causal implication has the main advantage of being clearer to understand, and less subject to misinterpretation, than material implication.

The theory of structural inertia in organizational ecology [35] can be very briefly summarized as follows. The factors generating inertia are internal (e.g., sunk costs in plants, equipment, and personnel and the tendency for precedents to become normative standards), external (e.g., barriers to enter and exit, legal), and political, as change may lead to lower institutional support. Hannan and Freeman aimed to clarify the meaning of structural inertia within organizational ecology, and to derive theorems relating inertia to the selection of organizations, within an evolutionary model. They postulated that there are two main competences that determine the firms′ performance: reliability and accountability. (Reliability represents the capability to produce collective outcomes of a given quality, repeatedly. Accountability means that organizations are able to document how resources are used and to reconstruct sequences of decisions, rules, and actions that lead to a particular result.) In order for an organization to be reliable it requires reproducibility (i.e., the ability to continually reproduce its structure), which is attained through a process of institutionalization and by creating highly standardized routines. With this institutionalization and increased reproducibility arises increased inertia and aversion to change.

Translating the original theory into first-order predicate logic

The original theory [35] is based on 10 different assumptions, 9 of which have been adapted to a causal implication framework as summarized in Table 3.

Table 3. Causal table for the assumptions.

A1: higher_survival(X,Y) ← higher_reliab(X,Y) ∧ higher_account(X,Y)
A2: higher_reprod(X,Y) ← higher_reliab(X,Y) ∧ higher_account(X,Y)
A3: higher_inertia(X,Y) ← ¬reorg(X) ∧ ¬ reorg(Y) ∧ higher_reprod(X,Y)
A4a: higher_reliab(X,Y) ← ¬reorg(Y) ∧ ¬reorg(Y) ∧ older(X,Y)
A4b: higher_account(X,Y) ← ¬reorg(Y) ∧ ¬reorg(Y) ∧ older(X,Y)
A5: higher_inertia(X,Y) ← same_class(X,Y) ∧ larger(X,Y)
A6: higher_reliab(X,Y) ← ¬reorg(X) ∧ reorg(Y)
A7: higher_survival(X,Y) ← larger(X,Y)
A9: higher_survival(X,Y) ← same_class(X,Y) ∧ reorg(X) ∧ reorg(Y) ∧ faster_reorg(X,Y)
A10: faster_reorg(X,Y) ← same_class(X,Y) ∧ simpler(X,Y)

A1 – organizations with higher reliability and accountability have higher survival chances; A2 – organizations withhigherreliabilityandaccountabilityhavehigherreproducibility; A3 – reorganization-free organizations with higher reproducibility have higher inertia; A4a – older reorganizationfree organizations have higher reliability; A4b – older reorganization-free organizations have higher accountability; A5 – larger organizations of the same class have higher inertia; A6 – the process of attempting reorganization lowers reliability of performance; A7 – larger organizations have a higher chance of survival; A9 – reorganizing organizations of the same class with faster re-organization processes have higher survival chances; A10 – simplerorganizations ofthesameclass havefasterreorganization processes. Assumption A8 ("structural reorganization increases the death rate of new organizations") was removed from the base theory as it was not used to prove any result [35, 57], that is, there was no theorem proved using this assumption and, therefore, there is no reason to keep it in the base theory.

Table 3 summarizes the causal relationships used as starting blocks of the causal map. The coding of the theory itself is very simple: the user only needs to write the propositions, as listed in Table 3, in a text file. Then the computer program parses the propositions in the text file, builds its internal database, and starts the theory tester and the theorem generator, as described in An Algorithm for Automated Theory Development and Testing section. (For more complex models, a graphical interface may be used to directly translate causal maps into propositions.)

The concepts used in the analysis of this theory are the following: higher_survival(X,Y), higher_inertia(X,Y), higher_reprod(X,Y), to express the concepts by which organization X has, respectively, higher chance of survival, higher inertia, and higher reproducibility than organization Y; higher_account(X, Y), higher_reliab(X,Y), which say that organization X has higher accountability and is more reliable than organization Y, respectively; same_class(X,Y), X and Y are in the same class; older(X,Y), X is older than Y; simpler(X,Y), X is simpler than Y. We have one single constraint reorg(X), which means that organization X is re-organizing.

It is evident from the discussion above that the process of translating a theory into a causal implication framework is a difficult one as, on the one hand, the theory originally described in "natural language" may not fit exactly into a causal map framework and, on the other hand, some of the richness of the description, and of the subtleties in the original theory, may be lost in the translation process. The main advantage of using causal maps is the writing of a more concise theory (e.g., assumption A8 was dropped), and a clear, objective, set of assumptions on which additional theorems can be conjectured and proved. It should also be noted that the assumptions reported in Table 3 were also the product of the theory tester that was used many times until a consistent, sound, and complete theory, in the causal form, was produced. Even after such a theory was derived, it was subsequently revised in order to facilitate the conjecture, and proof, of innovative and meaningful theorems.

In the original theory [35], there are five theorems, which, following References [41, 57] can be rewritten as follows: (T1) reorganization-free organizations with higher inertia have higher survival chances; (T2) the inertia of reorganization free organizations increases with age; (T3) the survival chances of reorganization-free organizations increases with age; (T4) the chance of survival of reorganization-free organizations is higher than the one of reorganizing organizations; (T5) more complex organizations of the same class have lower survival chances after re-organizations of the same type. It should be noted that the original theory was written in a nonformal way and without using the concept of causal implication. The base theory in Table 3 is now used to prove the new theorems that, then, can be compared with the ones in Reference [35] to assess the ability of the algorithm to generate creative insights in a given topic.

Deriving new theorems to explain structural inertia

By applying the algorithm to the basic assumptions, seven new theorems were derived, as presented in Table 4, which summarize the new causal relationships (AT is used to distinguish these theorems from the ones in the original theory). Of these theorems, AT2, AT3, AT4, and AT5 are also in the original theory [35], represented by the corresponding theorem with the same number. When the organizations are not going through a process of re-organization, older organizations have higher structural inertia (AT2), have higher probability of survival (AT3), and have higher reproducibility of the internal structure (AT7). Theorem AT7 corresponds to assumption A4 in the original theory.

However, an organization which is not re-organizing, and which has higher accountability standards than a re-organizing organization, has a higher probability of survival (AT4). AT4 corresponds to theorem T4 in the original theory, with the additional condition, regarding accountability, which was not considered by Hannan and Freeman. When both organizations are reorganizing, if they belong to the same class, the simpler organization has a higher probability of surviving the re-organization process (AT5).

Table 4. Causal table of derived theorems.

AT1: higher_inertia(X,Y) $\leftarrow$ ¬reorg(X) $\wedge$ ¬reorg(Y) $\wedge$ higher_reliab(X,Y) $\wedge$ higher_account(X, Y)
AT2: higher_inertia(X, Y) $\leftarrow$ ¬reorg(X) $\wedge$ ¬reorg(Y) $\wedge$ older(X, Y)
AT3: higher_survival (X,Y) $\leftarrow$ ¬reorg(Y) $\wedge$ ¬reorg(Y) $\wedge$ older(X,Y)
AT4: higher_survival(X, Y) $\leftarrow$ ¬reorg(X) $\wedge$ reorg(Y) $\wedge$ higher_account(X, Y)
AT5: higher_survival(X,Y) $\leftarrow$ same_class(X,Y) $\wedge$ reorg(X) $\wedge$ reorg(Y) $\wedge$ simpler(X,Y)
AT6: higher_reprod(X,Y) $\leftarrow$ ¬reorg(X) $\wedge$ reorg(Y) $\wedge$ higher_account(X,Y)
AT7: higher_reprod (X,Y) $\leftarrow$ ¬reorg(Y) $\wedge$ ¬reorg(Y) $\wedge$ older(X,Y)

The algorithm derives two new theorems AT1 and AT6, both of which provide a novel, and valuable, insight into structural inertia in organizations. AT1 states that when two organizations are not re-organizing the organization that has, simultaneously, higher reliability and higher accountability exhibits higher structural inertia. This means that the core concepts for the success of an organization (reliability and accountability) are also creating high levels of structural inertia (which may prevent an organization from adapting to an environmental change). AT6 addresses the causes of higher reproducibility of the internal structures: a re-organizing organization, if it has higher accountability standards than a re-organizing organization, then it also exhibits higher reproducibility of internal structures (AT6). In Table 5, we summarize how each theorem was derived. An interesting case is theorem T2, which was proved using assumption A3 and theorem AT7.

Table 5. Summary of the precedents for each theorem.

Theorems	Precedents
AT1	A2, A3
AT2	A3, AT7
AT3	A1, A4a, A4b
AT4	A1, A6
AT5	A9, A10
AT6	A2, A6
AT7	A2, A4a, A4b

Additionally, the algorithm was also able to derive the meta-theorems presented in Table 6, which result from the comparison of all the assumptions and theorems in the theory. These meta-theorems are based on the concept of C-equivalence and attempt to build complex causal relationships in which the same set of antecedents can, simultaneously, cause a set of consequences. These meta-theorems, therefore, correspond to the creation of potentially new concepts (each one of the sets of consequences) that have only one cause (the common set of antecedents).

In Table 6 meta-theorems MT1 and MT2 describe different situations in which an organization that has higher inertia also has a higher chance of survival. MT1 (derived from A1 and AT1) shows that the two core competencies of organizations, reliability and accountability, lead both to higher survival chances and higher inertia. This result shows that inertia is neither a consequence nor a cause of selection, but rather that both inertia and selection are the result of the same evolutionary forces: reliability and accountability. MT2 (derived from A5 and AT7) reinforces this conclusion as it proves that larger organizations, of the same class, have simultaneously higher inertia and higher chances of survival. These two meta-theorems correspond to the creation of a new concept which can be described as higher_survival_inertia(X,Y) stating that some organizations have simultaneously a higher chance of survival, and higher inertia, than others.

Table 6. Causal table of derived meta-theorems.

MT1: higher_survival(X, Y) ∧ higher_inertia(X,Y) ← ¬reorg(X) ∧ ¬reorg(Y) ∧ higher_reliab(X,Y) ∧ higher_account(X, Y)
MT2: higher_survival(X, Y) ∧ higher_inertia(X,Y) ← same_class(X,Y) ∧ larger(X,Y)
MT3: higher_survival(X, Y) ∧ higher_reprod (X,Y) ← higher_reliab(X,Y) ∧ higher_account(X,Y)
MT4: higher_survival(X, Y) ∧ higher_reprod(X,Y) ← ¬reorg(X) ∧ reorg(Y) ∧ higher_account(X,Y)
MT5: higher_survival(X, Y) ∧ higher_inertia(X,Y) ∧ higher_reprod(X,Y) ← ¬reorg(X) ∧ ¬reorg(Y) ∧ higher_reliab(X,Y) ∧ higher_account(X,Y)

There is a further interesting contribution of meta-theorem MT1 to explain the meaning of "selection" in the original theorem T1. Hannan and Freeman [35, p. 162] state that reproducibility is a sufficient condition for existence of inertia, and that reliability and accountability lead to higher selection (as these require higher reproducibility, it follows that higher reliability and higher accountability lead to higher inertia). Therefore, inertia is a bi-product of selection. This analysis clarifies the original theorem T1: it does not say that inertia leads to selection. It says that organizations with highly reproducible structures and, therefore, reliable and accountable, have higher inertia and higher chances of survival. Inertia is a by-product of reliability and accountability that lead to higher reproducibility and higher chances of survival. For this reason, a better interpretation of this theorem would be T1*, which corresponds to meta-theorem MT1.

Theorem T1*: Reorganization-free organizations with higher reliability and accountability have, simultaneously, higher inertia and higher survival chances.

Theorem T1* allows a comparison of different organizations, identifying under which conditions one has higher inertia, and higher chances of survival, than another. Most importantly, it clearly states that, under certain conditions, selection and inertia emerge together (even though there is no causal relationship between them). The advantage of theorem T1* is that it breaks any causal relationship between inertia and selection.

In this interpretation, the original theorem T1 was proved correct from the original assumptions, therefore, the most important result in [35] is indeed correct, without requiring any extra assumptions. This is an important result, given Young's [77] criticism of the original theorem. This result was achieved by re-interpreting the theorem and by using meta-theorems that were able to compare multiple properties with multiple properties.

A second concept is described by meta-theorems MT3 and MT4: organizations that simultaneously exhibit higher survival and higher reproducibility, higher_survival_reprod(X,Y). MT3 (derived from A1 and A2) states that any organization having higher reliability and accountability than another also exhibits higher reproducibility and a higher chance of survival. MT4 (derived from AT4 and AT6) states that any nonreorganizing organization with higher accountability than a reorganizing organization has higher reproducibility and a higher chance of survival. Again, there is no causal implication from chance of survival to reproducibility (or vice versa).

Finally, MT5 (derived from AT2, AT3, and AT7) introduces the third new concept that describes organizations which, simultaneously, exhibit higher inertia, higher reproducibility, and higher chances of survival, higher_survival_inertia_reprod(X,Y). This meta-theorem shows that older organizations, when reorganization-free, have survived because they excelled at the main drivers of survival (reliability and accountability), both of which assume higher reproducibility, which leads to higher inertia. However, there are no direct implications between chances of survival, inertia, and reproducibility.

Table 7 summarizes how the meta-theorems were derived. The meta-theorems do not compare the organizations under analysis (as do the theorems) but the properties of these organizations. The algorithm has derived five different meta-theorems, denoted MT, which relate to the concepts of higher survival, higher inertia, and higher reproducibility. These five meta-theorems, together with theorem AT6, represent an innovative insight into organizational ecology, and are the result of transformational creativity [9]. The algorithm was able to look into the space of possible theorems, searching in a different area of the conceptual space, and transforming the basic concepts into new ones.

Table 7. Summary of the precedents for each meta-theorem.

Theorems	Precedents
MT1	A1, AT1
MT2	A5, A7
MT3	A1, A2
MT4	AT4, AT6
MT5	AT2, AT3, AT7

Conclusions and discussion

Creativity-supporting systems have been shown to enhance the creativity of their users and the value of the organizations using them. This article proposes an algorithm, based on causal maps, to assist in the development of qualitative theories. The algorithm can test the basic axioms of the theory for contradictions, and to explore the relationship between these concepts to prove new theorems on a specific area of knowledge. The algorithm and the process of writing down the basic axioms of the theory in a causal format have shown to be useful in the identification of contradictions in the base theory that were solved by re-writing the basic axioms; most importantly, it was able to produce new theorems. Nonetheless, the selection and evaluation of these theorems' rests with the users.

The algorithm was applied to the analysis of structural inertia [35] and was able to prove theorems T2, T3, T4, and T5 of the original theory, and it produced two new theorems, AT1 and AT6, relating higher accountability to higher reproducibility. It has also shown that theorem AT7 corresponds to the original assumption 4, which was replaced in the base theory. Moreover, the algorithm was able to produce five new meta-theorems to explain how the main drivers of organizational evolution (reliability, accountability), together with size can, simultaneously, lead to higher chances of survival, higher inertia, and higher reproducibility. Of these, possibly the most important one is MT1 which explains how, with the use of the algorithm, theorem T1 in the original theory was "saved," by proving, as a meta-theorem, that reorganization-free organizations with higher reliability and accountability have, simultaneously, higher inertia and higher survival chances.

One of the features of the modelling process is the use of tables to summarize the causal map; these tables have the advantage of allowing a compact representation of the causal relationships, but they do not have the visual impact of a causal map. This disadvantage can be surpassed by using the causal diagram in the interaction with the users of the system and the table format as additional support in the modelling process. Another feature of the modelling framework presented in this article is the use of causal logic. This was a deliberate choice, as it allows a clearer description of the theories and the use of causality as the connection between concepts. However, some theories cannot be translated into causal maps, and, in this case, other cognitive mapping techniques are required instead, such as concept mapping, semantic maps, or influence diagrams, moreover, in such cases, another type of logic needs to be used to analyze the theory.

Even though the article has focused on the use of causal maps to create a new theory, it is possible that this same tool can be used with other goals. For example, in the context of analyzing the shared beliefs of the individuals in a group, [e.g., 17, 24, 25, 37, 40, 42, 54, 67, 68, 76], and in helping in the structuring of collective intelligence [45], the algorithm may find contradictions in the different representations of the problem and may be able to find new causal relationships, improving the outcome from the group exercise. Another area in which the use of the proposed algorithm may produce interesting results is in the support of brainstorming sessions [55]. It may be envisioned that brainstorming can be used as a complement to the theorem generator by creating conjectures that would be integrated into the basic theory to create a new, consistent, complete, and sound set of causal relationships and concepts.

A limitation of the methodology proposed in this article is the reliance on an initial set of axioms and theorems in order to develop new theorems, which limits the ability of the theorem generator to create new theories. A possible extension to the current framework might include the use of datasets to assess the value of the currently held theories and to derive data-based theorems.

References

[1] Aikenhead G, Farahbakhsh K, Halbe J, Adamowski J. Application of process mapping and causal loop diagramming to enhance engagement in pollution prevention in small to medium size enterprises: case study of a dairy processing facility. J Cleaner Prod. 2015;102:275–284.

[2] Ali N, Chater N, Oaksford M. The mental representation of causal conditional reasoning: mental or causal models. Cognition 2011;119:403–418.

[3] Amabile TM, Khaire M. Creativity and the role of the leader. Harv Bus Rev. 2008;86(10):100–109.

[4] Baaken T, Kliewe T. Creativity techniques using online facilities. Int J Bus Social Sci. 2012;3(1):165–172.

[5] Bielza C, Gómez M, Shenoy PP. Modeling challenges with influence diagrams: constructing probability and utility models. Decis Support Syst 2010;49:354–364.

[6] Bentley PJ. Exploring component-based representations - the secret of creativity by evolution? In Proc. of the Fourth International Conference on Adaptive Computing in Design and Manufacture (ACDM 2000), April 26th–28th, University of Plymouth, UK. I. C. Parmee (ed.), 2000, 161–172.

[7] Binsted, K. Machine Humour: an implemented model of puns, Ph.D. Thesis, University of Edinburgh, 1996.

[8] Boden MA. Modelling creativity: reply to reviewers. Artificial Intelligence 1995;79:161–182.

[9] Boden MA. Creativity and artificial intelligence. Artif Intell. 1998;103:347–356.

[10] Burks AW. The logic of causal propositions. Mind, New Series 1951;60(239):363–382.

[11] Buzan A. The mind map book, unlock your creativity, boost your memory, change your life. Pearson, Harlow, GB, 1993.

[12] Chen Z. Generating suggestions through document structure mapping. Decis Support Syst 1996;16:297–314.

[13] Cheung P-K, Chau PYK, Au AKK. Does knowledge reuse make a creative person more creative? Decis Support Syst. 2008;45:219–227.

[14] Cobb BR. 2009. Efficiency of influence diagram models with continuous decision variables. Decis Support Syst. 2009;48:257–266.

[15] Colton S. Making conjectures about maple functions. In proceedings of AISC/Calculemus'02, LNAI 2385, Springer. 2002 Available from http://www.doc.ic.ac.uk/~sgc/publications.html

[16] Colton S. The HR program for theorem generation. Proceedings of CADE'02, Copenhagen, Denmark, 2002. Available from http://www.doc.ic.ac.uk/~sgc/publications.html.

[17] Connolly T, Jessup LM, Valacich JS. Effects of anonymity and evaluative tone on idea generation in computer-mediated groups. Manage Sci. 1990;36(6):689–703.

[18] Cope D. Computers and musical style. Oxford, UK: Oxford University Press.

[19] Csikszentmihalyi M. Handbook of creativity. Cambridge, UK: Cambridge University Press.

[20] Csikszentmihalyi M. Creativity and genius: a systems perspective. In Steptoe A Editor. Genius and the Mind: Studies of Creativity and Temperament. Oxford University Press, 1998, p. 39–64.

[21] de Bono E. Simplicity. London: Penguin Books, 1999.

[22] Dell'Acqua P, Pereira LM. Preferential theory revision. J Appl Logic 2007;5:586–601.

[23] Detwarasiti A, Shachter RD. Influence diagrams for team decision analysis. Decis Anal. 2005;2(4):207–228.

[24] Desthieux G, Joerin F, Lebreton M. Ulysse: a qualitative tool for eliciting mental models of complex systems. Methodological approach and application to regional development in Atlantic Canada. Syst Dyn Rev 2010;26(2):163–192.

[25] Eden C, Ackermann F. Cognitive mapping expert views for policy analysis in the public sector. Eur J Oper Res 2004;152:615–630.
[26] Eden C. Analyzing cognitive maps to help structure issues or problems. Eur J Oper Res 2004;159:673–686.
[27] Elam JJ, Mead M. Can software influence creativity?. Inf Syst Res. 1990;1(1):1–22.
[28] Eskildsen J, Dahlgaard J, Norgaard A. The impact of creativity and learning on business excellence, Total Qual Manage. 1999;10(4–5):523–529.
[29] Forgionne G, NewMan J. 2007. An experiment on the effectiveness of creativity enhancing decision-making support systems. Decis Support Syst. 2007;42:2126–2136.
[30] Garfield MJ, Taylor NJ, Dennis AR, Satzinger JW. 2001 Research report: modifying paradigms – individual differences, creativity techniques, and exposure to ideas in group idea generation. Inf. Syst. Res. 2001;12(3):322–333.
[31] Ghobadi S, Mathiassen L. 2016. Perceived barriers to effective knowledge sharing in agile software teams. Inf. Syst. J. 2016;26 (2):95–125.
[32] Goldsmith JR, Sloan H, Szorényi B, Turán G. Theory revision with queries: horn, read-once, and parity formulas. Artif Intell. 2004;156:139–176.
[33] Goldvarg E,Johnson-Laird PN. Naive causality: a mental model theory of causal meaning and reasoning. Cognit Sci. 2001;25:565–610.
[34] Greiner R. The complexity of revising logic programs. J Logic Program. 1999;40:273–298.
[35] Hannan MT, Freeman J. Structural inertia and organizational change. Am Sociological Rev. 1984;49:149–164.
[36] Harvey I. The artificial evolution of adaptive behaviour. PhD Thesis. 1995. Available from http://www.cogs.susx.ac.uk/users/inmanh/inman_thesis.html
[37] Henderson GP. Causal implication. Mind, New Series 1954;63 (252):504–518.
[38] Heron J, Reason P. A participatory inquiry paradigm. Qual Inq. 1997;3(3):274–294.
[39] Husig S, Kohn S. Computer aided innovation – state of the art from a new product development perspective. Comput Ind. 2009;60:551–562.
[40] Jessup LM, Connolly T, Galegher J. The effects of anonymity on gdss group process with an idea-generating task. MIS Q. 1990; 14(3):313–321.
[41] Kamps J, Masuch M. Partial deductive closure: logical simulation and management science. Manage Sci. 1997;43(9):1229–1245.
[42] Langfield-Smith K. Exploring the need for a shared cognitive map. Journal of Management Studies, 1992;29(3):349–368.
[43] Langley P, Simon HA, Bradshaw, Zytkow JM. Scientific discovery: computational explorations of the creative process. Cambridge, MA: MIT Press; 1987.
[44] Lenat DB. The role of heuristics in learning by discovery: three case studies. In: Michalski RS, Carbonell JG, Mitchell TM. Editors. Machine Learning: An Artificial Intelligence Approach, Tioga, Palo Alto, CA, 1983, p. 243–306.
[45] Maciulienè M, Skarzauskienè A. Emergence of collective intelligence in online communities. J Bus Res. 2016;69:1718–1724.
[46] Malaga RA. The effect of stimulus modes and associative distance in individual creativity support systems. Decis Support Syst. 2000;29:125–141.
[47] Massetti B. 1996. An empirical examination of the value of creativity support systems on idea generation. MIS Q 1996;20(1):83–97.
[48] Marakas GM, Elam JL. Creativity enhancement in problem solving: through software or process? Manage Sci. 1997;43(8):1136–1146.
[49] May J, Lending D. A conceptual model for communicating an integrated information systems curriculum. J Comput Inf Syst. 2015;60(4):20–28.
[50] McCorduck P. Aaron's code. San Francisco, CA: W. H. Freeman; 1991.
[51] McCune W. The otter user's guide, Technical Report ANL/90/9, Argonne National Laboratories, 1990.
[52] Muller-Wienbergen F, Muller O, Seidel S, Becker J. 2011. Leaving the beaten tracks in creative work-A design theory for systems that support convergent and divergent thinking. J Assoc Inf Syst. 2011;12(11):714–740.
[53] Montibeller G, Belton V. Causal maps and the evaluation of decision options: a review. Journal of the Operational Research Society 2006;57(7):779–791.
[54] Nunamaker JF, Jr., Applegate LM, Konsynski BR. Facilitating group creativity: experience with a group decision support system. J Manage Inf Syst. 1987;3(4):5–19.
[55] Osborn AF. Your creative power: How to use imagination. New York: Scribners; 1952.
[56] Pease A, Colton S, Smaill A, Lee J. Lakatos and machine creativity. In Proceedings of the ECAI Creativity Systems Workshop, 2002.
[57] Peli G, Bruggeman J, Masuch M, Nuallain BO. A logical approach to formalizing organizational ecology. Am Sociological Rev. 1994;59(4):571–593.
[58] Pereira FC, Machado P, Cardoso A. Darwinci: creating bridges to creativity. In: de Oliveira FM Editor. Advances in Artificial Intelligence, Procs. of the 14th. Brazilian Symposium on Artificial Intelligence, SBIA'98, Porto Alegre, Brazil, Lecture Notes in Artificial Intelligence, LNAI-1515, pp. 239–248, Springer-Verlag, November, 1998.
[59] Rajkumar TM, Baliga BR, Yadav SB. Systems to support reasoning by analogy. J Comput Inf Syst. 1994;36(4):81–87.
[60] Reason P, Rowan J editors. Human inquiry: a sourcebook of new paradigm research. Chichester, UK: Wiley; 1981.
[61] Reason P. Three approaches to participative inquiry. In: Denzin NK, Lincoln YS. editors. Handbook of Qualitative Research, Sage: Thousand Oaks, 1994, p. 324–339.
[62] Saunders R, Gero JS. Artificial creativity: A synthetic approach to the study of creative behaviour. In: Gero JS. editor. Computational and Cognitive Models of Creative Design V, Key Centre of Design Computing and Cognition, University of Sydney, Sydney, 2001, 113–139.
[63] Shneiderman B. Creating creativity: user interfaces for supporting innovation. ACM Trans Comput Hum Interact. 2000;7(1):114–138.
[64] Siau K, Tan X. Use of cognitive mapping techniques in information systems development. J Comput Inf Syst. (Summer), 2008:49–57.
[65] SICStus Prolog. 2016. Available from http://www.sics.se/isl/sicstuswww/site/index.html
[66] Stirling L, Shapiro E. The art of prolog: advanced programming techniques. London: MIT Press; 1994.
[67] Tegarden DP, Tegarden LF, Sheetz SD. Cognitive mapping in a top management team: surfacing and analyzing cognitive diversity using causal maps. Group Decis Negotiation 2009;18:537–566.
[68] Tegarden DP, Sheetz SD. Group cognitive mapping: a methodology and system for capturing and evaluating managerial and organization cognition. Omega 2003;31:113–125.
[69] Thayse A. From standard logic to logic programming. introducing a logic based approach to artificial intelligence. Chichester: John Wiley & Sons; 1988.
[70] Thayse A. From modal logic to deductive databases. Introducing a Logic Based Approach to Artificial Intelligence. Chichester: John Wiley & Sons, 1989.
[71] Thornton C. A quantitative reconstruction of boden's creativity theory. 2005. Available from http://www.cogs.susx.ac.uk/users/christ/.index-Papers.html

[72] Vandenbosch B, Higgins C. Information acquisition and mental models: an investigation into the relationship between behaviour and learning. Inf Syst Res. 1996;7(2):198–214.
[73] Velleman DJ. How to prove it. a structured approach. New York: Cambridge University Press; 1994.
[74] Wiggins GA. A preliminary framework for description, analysis and comparison of creative systems. Knowledge-Based Syst. 2006;19:449–458.
[75] Witteveen C, van der Hoek W. Recovery of (non)monotonic theories. Artif Intell. 1998;106:139–159.
[76] Yen J, Fan X, Sun S, Hanratty T, Dumer J. 2006. Agents with shared mental model for enhancing tem decision making. Decis Support Syst. 2006;41:634–653.
[77] Young RC. Is population ecology a useful paradigm for the study of?" Am J Sociology 1988;94:1–24.
[78] Zou G, Yilmaz L. Dynamics of knowledge creation in global participatory science communities: open innovation communities from a network perspective. Comput. Math. Organ. Theory 2011;17:35–58.