

# Sequential Cognition Processes: A Framework For Reasoning with Non-Monotonic Logics

Axel Ind

Albert-Ludwigs-Universität Freiburg  
axeltind@gmail.com

## Abstract

Approaches to cognitive modelling with non-monotonic logics have thus far been largely *ad hoc* and poorly standardised, making inter-model comparisons difficult. As an attempt to systematically represent non-monotonic logics in a framework that standardises cognitive modelling under these logics without sacrificing their expressiveness, we introduce the Sequential Cognition Process (SCP). Under the assumption that human reasoning can be represented as a sequence of distinct cognitive operations on an initial knowledge base SCPs provide a consistent framework for generating and evaluating models of human cognition. Using an adapted interpretation of the Weak Completion Semantics (WCS), SCPs are able to accurately model several classical experiments in cognitive modelling. We use the SCP framework to model both general case reasoners – which arrive at the most frequently observed conclusions – and poorly-studied individual case reasoners – which do not. We illustrate the use of SCPs using the Suppression Task.

## 1 Introduction

The human mind is complex. So complex that thousands of approaches from dozens of fields have failed to capture its complexity. The sheer size of the brain – containing over 5000 times as many neurons as the largest practical neural networks (Mocanu et al., 2018) – and our limited understanding of the fundamental learning processes it employs mean that researchers can neither completely describe nor predict human actions nor model human thought processes. Instead, much of the current state of the art in cognitive modelling relies on one of two general approaches: creating systems that structurally approximate the human brain, and creating systems which approximate a more abstract intuition of cognition. The first type of system encompasses many algorithms related to machine learning and deep learning; the second type, with which this paper is concerned, has existed in some form for far longer than humans have been thinking creatures (Smirnova et al., 2015). Every shark stalking its prey, every tiny proto-mammal hiding from a hungry dinosaur, and every man driving to work in the morning, has applied this type of reasoning when trying to make predictions

about the actions of other agents in their world. By applying case-specific reasoning to a known world state we are able to make imperfect, but quick, predictions about the mental state of other agents in our world.

Due in part to the difficulty of cloning dinosaurs to hunt participants in cognitive research experiments, and partly due to concerns about the ease with which they could answer questionnaires on the experience later, most cognitive tasks used by researchers tend to be more dull than the examples above.

Non-monotonic logics have proven able to adequately model a large number of standard cognitive reasoning tasks such as The Wason Selection Task (Wason, 1968) and Suppression Task (Byrne, 1989). These approaches, though effective and well-founded in isolation, are often unable to integrate or be compared to data models of other tasks, even when they rely on the same underlying logic. Although the non-monotonic logics themselves are generally carefully described, procedures ranging from best practise in deciding appropriate knowledge bases to the mechanisms by which abnormalities should be created tend to be re-imagined on a case-by-case basis.

Further, cognitive frameworks using non-monotonic logics are almost always designed to describe the most common (general) conclusion drawn by participants in the experiment. Modelling other individual reasoners or classes of reasoners who differ from the norm is often a non-trivial process. It has been shown that the Weak Completion Semantics (Hölldobler, 2015) is able model the four most general cases of the Wason Selection Task under the assumption that reasoners who differ from the general case (*deviant reasoners*) follow a sequence of mental processes that is still highly similar to that of the general reasoner (Breu et al., 2019).

This paper introduces the Sequential Cognition Process (SCP) which generalises the assumption of sequential cognitive operations, each of which uses a collection of epistemic states as input and produces a collection of epistemic states as output. Using SCPs and a set of well-founded cognitive operations it is possible to apply traditional search techniques to problems in cognitive modelling with non-monotonic logics that have previously required expert-made models. The SCPs intro-

$\wedge$	$\top$	$\perp$	$\vee$	$\top$	$\perp$	$\neg$	$\top$	$\perp$
$\top$	$\top$	$\perp$	$\top$	$\top$	$\perp$	$\top$	$\perp$	$\top$
$\perp$	$\perp$	$\perp$	$\perp$	$\top$	$\perp$	$\perp$	$\perp$	$\top$
$\rightarrow$			$\top$	$\perp$	$\leftarrow$			$\top$
$\top$	$\top$	$\perp$	$\top$	$\top$	$\top$	$\top$	$\perp$	$\perp$
$\perp$	$\top$	$\perp$	$\perp$	$\top$	$\perp$	$\perp$	$\perp$	$\top$

Table 1: Truth tables for standard operators in propositional logic.

duce a number of desirable properties: they introduce a partially standardised (though extensible) set of allowable cognitive operations, they standardise the structure of what constitutes an epistemic state, they are easily modified to accommodate deviant reasoners when a well-founded general model already exists, and their sequential structure makes them well-suited to scoring algorithms that allow intra- and inter-experimental modelling and comparisons.

Section 2 introduces the mathematical background necessary to understand the SCP framework when it is used in the context of non-monotonic logics in general and the WCS in particular. Section 3 introduces one common formulation of the Suppression Task and shows how the WCS has been used to model the most common (general) inference drawn by participants. Section 4 introduces the SCP framework from both an intuitive and a mathematical sense, as well as addressing the concept of the epistemic states in some detail. Section 5 provides a concrete illustration of the use of SCPs by modelling both the general case of the Suppression Task as well as an individual case for those who do not draw the most common inferences. Finally, Section 6 summarises the key contributions of this paper and proposes several possible research questions related to the SCP framework.

## 2 Mathematical Preliminaries

### 2.1 Propositional Logic

Propositional logic is one of the simplest classical logics and concerns itself with binary truth values.  $\top$  denotes a tautology, and  $\perp$  denotes a contradiction. An alphabet  $\Sigma = \{v_1, \dots, v_n\}$  defines the set of atoms available to the language. Every atom  $v_i \in \Sigma$  has the domain  $\{v_i, \bar{v}_i\}$ . An (atom, domain) pair which associates an atom with a value in its domain is called a literal.

A formula  $\theta \in \zeta$  from the set of all formulae  $\zeta$  is defined recursively over all conjunction  $\wedge$ , disjunction  $\vee$ , and negation  $\neg$  operations as follows:

$$\theta = \{v_i \in \Sigma \mid \neg\theta \mid \theta \vee \theta \mid \theta \wedge \theta\}$$

The material implication  $(\psi \leftarrow \phi)$  precisely substitutes for the formula  $(\neg\psi \vee \phi)$  and is often used, classically to denote an “if  $\phi$  then  $\psi$ ” relationship between variables. However, in Section ?? other interpretations of the “if  $\phi$  then  $\psi$ ” relationship are examined in a non-monotonic setting.

$(\psi \mid \phi)$	$\phi \models \top$	$\phi \models u$	$\phi \models \perp$
$\psi \models \top$	verification	non-applicability	non-applicability
$\psi \models u$	verification	non-applicability	non-applicability
$\psi \models \perp$	falsification	non-applicability	non-applicability

Table 2: Evaluation of conditional rules using a 3-valued *de Finetti* truth table.

An interpretation of  $\Sigma$ ,  $I_w(\Sigma)$  assigns each  $v_i \in \Sigma$  to one value in its domain as defined by possible world  $w$ . The interpretation of a formula  $\theta$ ,  $I_w(\theta)$  is recursively calculated using the associated truth tables of the operators involved, as defined in Table 1.

### 2.2 Possible Worlds

A possible world  $w \in W_\sigma$  from the set of all possible worlds  $W$  over a language  $\Sigma$  is  $(v_0, w_0) \wedge \dots \wedge I(v_n, w_n)$  and is the conjunction of each atom and an associated value from its domain. Possible worlds represent one setting of the variables in the alphabet that could hypothetically hold. A model of a propositional formula  $\theta \in \zeta$  is a possible world  $w$  in which  $\theta$  is evaluated to  $\top$  (also called satisfying) and is written  $w \models \theta$ .

A model  $Mod(\theta)$  denotes the set of possible worlds in which  $\theta$  holds – that is,  $Mod(\theta) = \{w \mid w \models \theta\}$ . The operator  $\models$  is overloaded so that  $\phi \models \psi$  if and only if  $Mod(\phi) \subseteq Mod(\psi)$  for  $\phi, \psi \in \zeta$ .

The *least model* of a theory  $\theta$  is model  $M$  of  $\theta$  such that there is exists no other model  $M'$  of  $\theta$  such that  $M \subset M'$ .

A set of propositional clauses  $W$  is called *deductively closed*, written  $th(W)$  if it contains every formula  $\phi$  that can be logically deduced from  $W$ .

### 2.3 Conditionals

Defeasible knowledge is information which is assumed, but not guaranteed, to hold. Many non-monotonic logics treat this notion as a form of typicality. A condition  $(\psi \mid \phi)$  is used to represent the defeasible rule “if  $\phi$  then  $\psi$ ”. Following Wason (1968) we do not use the implication  $\leftarrow$  to represent a conditional, and instead opt for a three-valued interpretation of what ? called a *de Finetti truth table* (Table ??). This truth table for conditionals differs from the standard implication in that it treats rules of the form  $(\psi \mid \phi)$  as inapplicable when  $\phi \models \perp$  or  $\phi \models u$  and more closely follows human intuition about how rules are evidenced in the absence of their antecedent.

### 2.4 Interpretation of conditionals

The precise interpretation of conditionals is a topic for philosophical and logical debate. Approaches, both probabilistic and logical, can be used to define precisely what a conditional means in the context of a logical system. In this thesis we adopt Stenning, Van Lambalgen, and others (2008)’s interpretation of conditionals as licenses for implication.

Thus, the conditional  $(\psi|\phi)$  precisely means  $\psi \leftarrow \phi \wedge \neg ab$  where  $ab$  is called an abnormality predicate. The abnormality predicate captures an element of uncertainty in the interpretation of the conditional and means that the conditional now reflects the statement “If  $\phi$  and nothing abnormal is known, then  $\psi$ ”.

The choice of appropriate value assignments to the abnormality predicate is non-trivial and often requires expert knowledge to design. However, over the course of this thesis we will adopt an algorithmic approach to abnormality creation (Algorithm ??) that will serve to mimic the hand-designed abnormalities used by Dietz, Holldobler, and Ragni (2012), and Breu et al. (2019).

---

**Algorithm 1** Conditional to License for Implication

---

FnFunction isend Create Abnormalities(KB)

**for**  $(\psi|\phi) \in \text{KB}[\Delta]$  **do**  $k :=$  the lowest natural number for which  $ab_k \notin \text{KB}$  all dependencies  $:= [A|(\psi|A) \in \text{KB}'[\Delta]]$

**for**  $A \in$  all dependencies **do** head  $:= \psi$   
current dependencies  $:=$  all dependencies  $- A$

**if** current dependencies  $= \{\}$  **then**  
current dependencies  $:= \perp$  body  $:=$   
(current dependencies<sub>1</sub>  $\vee \dots \vee$  current dependencies <sub>$n$</sub> )

Add the conditional as a license for implication to the set of rules.  $\text{KB}'[S] := \text{KB} \cup (\text{head} \leftarrow A \wedge \neg ab_k)$   
 $\text{KB}'[S] := \text{KB}'[S] \cup ab_k \leftarrow \neg body$

Remove all conditionals now that they have been interpreted as licences for implication.  $\Delta := \{\}$  **return**  $\text{KB}'$

---

## 2.5 Logic Programming

**Knowledge Base** In propositional logic, a knowledge base  $S = \{\phi_1, \dots, \phi_n\}$  for the language  $\Sigma$  is a set of propositional rules from the set of formula  $\zeta_\Sigma$ . A propositional knowledge base is used to encode certain knowledge and relationships about the world.  $S$  is called *consistent* only if there exists some possible world  $w \in W$  such that every rule in  $I_w(\text{KB}_{prop})$  evaluates to  $\top$ .

More generally, a non-monotonic knowledge base  $\text{KB} = (S, \Delta)$  contains both a set of propositional rules ( $S$ ) and a set of defeasible/conditional rules ( $\Delta = \{\Delta_1, \dots, \Delta_m\}$ ). A non-monotonic knowledge base tolerates a conditional  $(\psi|\phi) \in \Delta$  if and only if there exists some possible world  $w \in W$  that verifies  $(\psi|\phi)$  and does not falsify any conditional in  $\Delta$ .  $\text{KB}$  is said to be *consistent* if and only if there exists of partition of  $\Delta$ ,  $(\Delta_0, \dots, \Delta_n)$  such that  $\Delta_i$  is tolerated by  $\Delta_{i+k}$ , where  $k$  is any natural number not greater than  $m$ .

When  $\text{KB} = (S = \{\text{rule}_1, \dots, \text{rule}_n\}, \Delta = \{\})$ , we will sometimes ignore  $\Delta$  and refer only to the value of  $S$  whilst still using the term ‘KB’.

## 2.6 Non-Monotonic Logics

In the field of non-monotonic logics, reasoning is represented as a collection of defeasible inferences. Unlike

$\rightarrow$	$\top$	$u$	$\perp$
$\top$	$\top$	$u$	$\perp$
$u$	$\top$	$\top$	$u$
$\perp$	$\top$	$\top$	$\top$

Table 3: A table showing the implication operator in 3-valued Łukasiewicz logic.

in classical logic, conclusions need not hold in perpetuity, or even in the same model and revision is always possible. Monotonic logics are not capable of describing human reasoning in experiments like the Suppression Task (Dietz, Holldobler, and Ragni, 2012) because they lack this revisionist characteristic. ? showed that two-valued logics were not sufficient to model human cognition.

A large number of non-monotonic frameworks exist in the literature (McDermott and Doyle, 1980), each applicable to a different subset of cognitive problem space, and each modelling their problem space with various degrees of success. In the simplest formulation, a non-monotonic logic is an extension to a classical logic which introduces a preference relation  $\leftarrow_p$ . This preference relation states that, given some number of derivable facts in a knowledge base, the fact derived using the most preferential rule is to be derived first and cannot be overwritten by a less preferable assignment.

Although the non-monotonic logics discussed in this chapter have simple extensions to first-order logic, we instead restrict ourselves to a propositional format throughout this thesis.

## 3 The Weak Completion Semantics

The Weak Completion Semantics is a non-monotonic logic which procedurally encodes several well-known cognitive phenomena. The WCS makes use of 3-valued Łukasiewicz logic (Table 2), rather than Kripke-Kleene logic which has been shown inadequate to model several aspects of cognition, including Suppression (Dietz, Holldobler, and Ragni, 2012). It adds abnormalities to non-ground inferences, and replaces the classical inference ( $\leftarrow$ ), with a bijective ( $\leftrightarrow$ ).

The Weak Completion of a program  $P$  is defined as follows:

1. Replace all clauses of the form  $A \leftarrow body_1, \dots, A \leftarrow body_n$  with  $A \leftarrow body_1 \vee \dots \vee body_n$ .
2. Replace all occurrences of  $\leftarrow$  with  $\leftrightarrow$ .

Applying this procedure to  $P$  results in  $wc P$  which is the *weak completion* of  $P$ .

The next requirement to apply the WCS framework is the introduction of a semantic operator  $\phi_{svL}$  (Stenning, Van Lambalgen, and others, 2008). Let  $J$  be the result of applying the semantics operator to an interpretation  $I$  and logic program  $P$ . Then  $J$  is defined as follows:

$J^\top = \{A \mid \text{there exists a clause } A \leftarrow \text{Body} \in P$   
with  $I(\text{Body}) = \top\}$

$J^\perp = \{A \mid \text{there exists a clause } A \leftarrow \text{Body} \in P$   
and for all clauses  $A \leftarrow \text{Body} \in P$   
we find  $I(\text{Body}) = \perp\}$

Using  $I = \langle \emptyset, \emptyset \rangle$ , the least model of  $P$  ( $\text{lm}_L \text{wc}P$ ) can be calculated by iterating  $\phi_{SVL,P}$ . The WCS guarantees a unique least model for a propositional knowledge base  $P$  when interpreted with Łukasiewicz logic.

## 4 Experiments: The Suppression Task

The Suppression Task refers to an experiment conducted by Byrne (1989) and is a classical example of the inadequacy of monotonic logics for modelling human reasoning. In classical logic, if our knowledge base  $kb$  is such that  $kb \models \phi$ , then it must be the case that  $kb \cup \psi \models \phi$ . However, in the suppression task participants no longer draw classically valid inferences when new information is added. The task is often formulated as follows:

- $e \rightarrow l$ : If she has an essay to write ( $e$ ), she will study late in the library ( $l$ ).
- $\top \rightarrow e$ : She has an essay to write ( $e$ ).
- $o \rightarrow l$ : If the library is open ( $o$ ), she will study late in the library ( $l$ ).

Given only the rules ( $e \rightarrow l$ ) and ( $\top \rightarrow e$ ), the participants consistently concluded that she would study late in the library, seemingly drawing the classical logic inference  $\frac{e \rightarrow l, e}{l}$  with *modus ponens*. But when given the additional rule  $o \rightarrow l$ , participants no longer believe that they have enough information to judge whether she will study late in the library, and a significant portion of them no longer draw the classical conclusion. This effect, called Suppression, demonstrates the need for something more than classical logic for modelling human reasoning.

Work by Dietz, Hölldobler, and Wernhard (2014) has shown that the WCS is an adequate non-monotonic logic for modelling the Suppression Task. Under the WCS and Łukasiewicz 3-valued logic the task is usually modelled as follows for the case without suppression:

1. Initial logic program:  $P = \{e \rightarrow l, \top \rightarrow e\}$ . This program represents the task without information about what happens if the library is open.
2. Addition of Abnormality:  $P = \{e \wedge \neg ab_1 \rightarrow l, \top \rightarrow e, \perp \rightarrow ab_1\}$ . The program now reflects the possibility that some abnormal event may prevent her from going to the library, but because we have no information about the nature of this event, it is set to false by default.
3. Weak Completion:  $\text{wc}(P) = \{e \wedge \neg ab_1 \leftrightarrow l, \top \leftrightarrow e, \perp \leftrightarrow ab_1\}$ . Weak completion is applied to the logic program and, in this case, only implications need to be changed to bijections.

## 4. Semantic Operator:

- Execution 1:  $\top = \{e\}, \perp = \{ab_1\}$
- Execution 2:  $\top = \{e, l\}, \perp = \{ab_1\}$

After application of the semantic operator  $l$  is true in the least model, and so participants conclude that she will study late in the library (as when  $P$  is evaluated classically). However, in the case where Suppression is observed, the same process yields a different result because of the presence of the extra conditional ( $o \rightarrow l$ ).

1. Initial logic program:  $P = \{e \rightarrow l, \top \rightarrow e, o \rightarrow l\}$ . The initial program now includes information about the extra (suppressing) conditional.
2. Addition of Abnormality:  $P = \{e \wedge \neg ab_1 \rightarrow l, o \wedge \neg ab_2 \rightarrow l, \top \rightarrow e, \neg o \rightarrow ab_1, \neg e \rightarrow ab_2\}$ . An abnormality  $ab_i$  is an addition to an inference which captures the idea that external factors may invalidate the conclusion of the original inference. Adding abnormalities is a poorly described process and often relies on intuitionist views of what actually constitutes an abnormal situation. In this case we adopt the practise of identifying abnormalities as described in Appendix 6.
3. Weak Completion:  $\text{wc}(P) = \{((e \wedge \neg ab_1) \vee (o \wedge \neg ab_2)) \leftrightarrow l, \top \leftrightarrow e, \neg o \leftrightarrow ab_1, \neg e \leftrightarrow ab_2\}$ . Weak completion is applied to the logic program, combining rules with shared heads and replacing implications with bijections.
4. Semantic Operator:
  - Execution 1:  $\top = \{e\}, \perp = \{ab_2\}$

Now suppression has been displayed in the logic program and the variable  $l$  remains unknown in the least model.

## 5 Sequential Cognition Process

### 5.1 SCPs: an Intuitive Description

Although there is evidence that the brain can perform several simultaneous operations when considering a task (such as when considering an image (Sigman and Dehaene, 2008)), the SCP framework assumes that at some points in reasoning about a given task, the mental processes of the agent converge to a set of epistemic states, called a *state point*. Whatever happens between these points of convergence can contain any number of parallel processes. The collection of processes that occur between any two state points in a reasoning task is called a *cognitive operation*. It follows that any cognitive operation is valid as long it takes a set of epistemic states as input and produces a set of epistemic states as output. Figure 1a describes an SCP that is powerful enough to model any cognitive task that involves an epistemic state transition. However, it does not provide any useful information; the nature of the processes followed is completely undescribed. Suppose, instead, that some cognitive task is being modelled, and that researchers have reason to believe that The Weak Completion Semantics should play a part in their model.

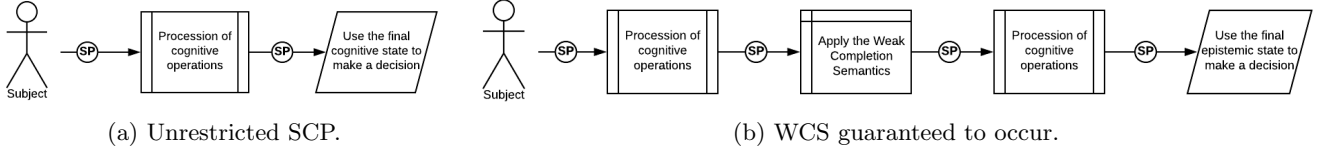


Figure 1: The most general description on an SCP with and without guaranteeing the WCS is applied at least once. An agent transitions from one epistemic state to another and then uses it to make a decision. *SP* nodes indicate state points.

Under this new restriction, and assuming a sufficiently expressive epistemic state, Figure 1a is still an accurate model of the process, but now so is Figure 1. By sacrificing some of the ambiguity – and, thus, expressiveness – of the model, the information content of the model description has increased. This trade-off is a feature of the SCP framework and finding the right depth of complexity to model the task accurately and still provide meaningful information is more art than science at present.

## 5.2 SCPs: a Formal Framework

An SCP Task  $\Pi = (s_i, \gamma, M)$  consists of an initial epistemic state  $s_i$ , known goal state  $\gamma$ , and a known set of cognitive operations  $M$ . An epistemic state  $s_k$  describes all the information available to the agent at state point  $k$ . The precise contents of an epistemic state should be chosen so that at least some  $m \in M$  are able to accept that state as an input. In the case of a system containing only the one complex operation which applies the WCS, one possible epistemic state is  $s = (KB, V)$ , where  $KB$  describes a knowledge base consisting only of rules of the  $(head \leftarrow body)$  and  $V$  is a mapping of atom names appearing in  $KB$  to a truth value in Łukasiewicz logic  $(\top, u, \perp)$ . In principle this definition will serve for the rest of the paper, but it is extended slightly so that  $s_k = (KB, V, R)$  where  $R$  is a set of labelled categorization criteria sets (*LCS*). An *LCS*  $c \in R$  consists of a category name and list of rules and atoms which fit into that category.

A base point is a single epistemic state. A state point  $p$  is defined recursively by  $p = \{\bar{p} \oplus Q\}$  where  $\bar{p}$  is a base point, and  $Q$  is a set of state points, and  $\oplus$  represents the exclusive-or operation<sup>1</sup>. State point containment  $\in_s$  for state points  $p$  and  $q$  is defined recursively as follows:

$$p \in_s q = \begin{pmatrix} p \in q & \text{True} \\ p \notin q \text{ and } q \text{ is a base point} & \text{False} \\ \exists_{r \in p} p \in_s r = \text{True} & \text{True} \\ \text{Otherwise} & \text{False} \end{pmatrix}$$

It is never the case that  $p \in_s p$ .

A cognitive operation  $m = (\chi, e)$ ,  $m \in M$  consists of a precondition  $\chi$  and a process  $e$ , such that for an input state point  $p$ , every base point  $\bar{p} \in_s p$  is either accepted as input ( $\bar{p} \models m[\chi]$  under whatever definition of  $\models$  is

used for the complex operation  $m$ ), or else rejected. Every base point is evaluated by the complex operation in isolation (no other base point  $\bar{q}$  can affect the output of  $m$  on base point  $\bar{p}$ ). To capture the fact that cognitive operation may utilize non-monotonic logic, applying  $m$  to an input base point  $\bar{p} \in p$  always yields a state point  $p'$  (written  $J[\bar{p}, m] = p'$ ) which describes all possible resulting epistemic states that can be generated.

Applying  $m$  to an input state point  $p$  is done by replacing every base point  $\bar{p} \in p$  with  $J[\bar{p}, m]$ . A cognitive operation is called monotonic if it always yields a base point as an output given a base point input ( $J[\bar{p}, m] = \bar{p}'$ ). It follows that the depth of a state point is directly related to the number of complex operations which have been performed in the SCP prior to its occurrence. If a base point does not meet the precondition, it is either ignored completely and not processed (*cruel application*), or passed exactly as is to the next complex operation (*lenient application*). It is worth noting that the type of cognitive states produced as output by  $m$  may not be the same as those of the input.

This paper will focus on cases where the type of cognitive state remains constant, but there is no reason in principle that a base point input  $s_k = (KB, V, R)$  for the WCS could not be returned as a base point  $s_k = (KB, V, R, D)$  where  $D$  is a set of default rules usable by cognitive operations that implement Reiter's Default Logic (Reiter, 1980). Future models of human cognition may well rely on background knowledge which draws inferences from multiple types of non-monotonic logics.

An SCP  $\pi = (s_i, m_1, \dots, m_n)$  describes an initial epistemic state (or state point if the input is uncertain) and state point  $p_k$  is defined recursively by  $p_k = J[p_{k-1}, m_k]$ . An SCP is called *credulously valid* if  $p_n \models \gamma$  for at least one epistemic state in the final state  $p_n$ . An SCP is called *sceptically valid* if  $p_n \models \gamma$  for every epistemic state in the final state. In cases where all operations are monotonic, sceptical validity is the same as credulous validity.

## 5.3 Choosing an Epistemic State

The choice of epistemic state is dependent on the properties that are known or suspected to be true for the cognitive task as a whole. For example, a researcher working on drawing inferences using Propositional Logic can be certain that any SCP they create

<sup>1</sup> $(X \oplus Y) = ((X \cup Y) - (X \cap Y))$  for sets  $X$  and  $Y$ .

should be expressive enough to pass a knowledge base to a cognitive operation. Thus, it might suffice to simply define  $s_i = (KB)$  where  $KB$  is a set of propositional rules. By contrast, a researcher using the WCS requires a system capable of both communicating a set of rules to the next complex operation, and of describing the results of repeated applications of the semantic operator. It might seem intuitive to simply keep the epistemic state used for the propositional case and to use a non-complex operation function on the weakly completed program to find the least model. And, though this approach works for many examples, it makes it impossible to perform further processing in the SCP *after* applying the semantic operator. What if the conclusion drawn was meant to form part of the background knowledge of another process? In practice using  $s_k = (KB, V, R)$ , where  $V$  is a set of (variable name, value pairs) and  $R$  is a categorization variable, is able to model all aspects of the WCS, including information related to the least model.

The final property of a cognitive operation that needs to be discussed is how it is able to interact with the categorization variable  $R$ . Imagine a case drawn from Saldanha, Hölldobler, and Rocha (2017) where the difference between creating abnormalities for obligate and factual conditionals is discussed. The intuition behind the authors’ work can be summarised by saying that there are two different types of conditional statement, those that *have to be* true, and those that are *usually* true. If it rains  $r$ , I will normally take my umbrella  $u$  (in the absence of something abnormal happening, like a plague of umbrella-stealing gnomes). This a statement that is usually true, but some things are definitely true. For example, when it rains, water has to fall out of the sky  $s$ . This is an essential property of rain, not subject to abnormalities. Thus one useful set of categorizations for researchers seeking cosmic knowledge of weather patterns may be:  $R = \{obligate : \{u \leftarrow r\}, factual : \{s \leftarrow r\}\}$ . With these labels we might then expect that the process followed by the operation  $m \in M$  which creates abnormalities would treat the two conditionals in  $KB$  differently, because of their assignments in  $R$ .  $R$  then is a way of expressing meta information to the cognitive operations, and it is completely possible that some operation  $m_k$  might change  $R$  such a way that future operation  $m_{k+l}$  produces different output. This is a technique we will exploit in several examples in this paper.

As yet, there are no definite rules for creating an epistemic state, but Albert Einstein’s famous advice from 1950 still rings true: “Everything should be made as simple as possible, but no simpler.” The ideal epistemic state is one that enables every reasonable cognitive operation in  $M$  that might help model the problem, without adding superfluous functionality that might render searching the SCP space infeasible.

## 5.4 Choosing a set of Complex Operations

The set of possible complex operations  $M$  determines many attributes of the achievable final state point  $p_n$ . If every  $m \in M$  is monotonic, then  $p_n$  will be monotonic. If some cognitive operation is computationally complex or produces a very large number of output state points, then search using that base point becomes less efficient. If some complex operation  $m'$  (such as weakly completing) is known or believed to occur in the SCP, then a restriction on the cognitive states exists such that either the initial state is of a format suitable as input for  $m'$ , or there exists another cognitive operation which is able to output a state point which contains base points of a suitable format.

More abstractly, the set of cognitive operations should be well-founded in the literature. The set of possible complex operations is infinite and an SCP only meaningfully describes human cognition when it contains cognitive operations that have been justified empirically (*modus ponens-modus tollens* asymmetry, suppression, denial of the antecedent, etc.).

## 6 Modelling with SCPs

### 6.1 SCPs Under the Weak Completion Semantics

**The Suppression Task** Figure 2 illustrates a generalised SCP to describe the Suppression Task as a series of sequential steps directly mirroring the discrete steps outlined in Section 3, each cognitive operation passing information to the next process<sup>2</sup>. To model the SCP the implicit sequence of operations in the Suppression Task is systematized and refined into a set of complex operations. Further we introduce an initial epistemic state  $s_i = (KB, V, R)$ . One interpretation of the requirements of the suppression task  $\pi = (s_i, \gamma, M)$  using SCPs and the WCS is as follows:

$$\begin{aligned} s_i &= \{KB_i, V_i, R_i\} \\ KB_i &= \{e \rightarrow l, \top \rightarrow e, o \rightarrow l\} \\ V_i &= \{e : u, l : u, o : u\} \\ R_i &= \{\} \\ M &= \{\text{init}, \text{addAB}, \text{WeaklyComplete}, \\ &\quad \text{semanticOperator}\} \\ \gamma &= (l \models \top) \text{ or } (l \models \perp) \end{aligned}$$

where all cognitive operations require as input and produce as output a state point  $p$  where every ground point  $\bar{p} \in_s p$  is of format  $\bar{p} = (KB, V, R)$ ; **init** is always the first cognitive operation and adds the

<sup>2</sup>It is important to note that a diagram like this is valid for *any* cognitive modelling task because any process may be arbitrarily complex and non-sequential. and so the overall linear process of (actor, complex decision, observed results) is always valid for retroactive modelling, and at least as powerful as the non-monotonic logic framework it uses for modelling.

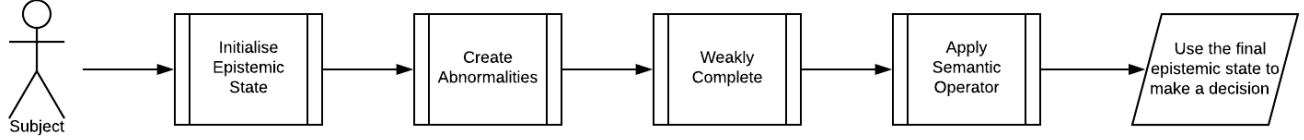


Figure 2: A generalised illustration of the WCS in an SCP.

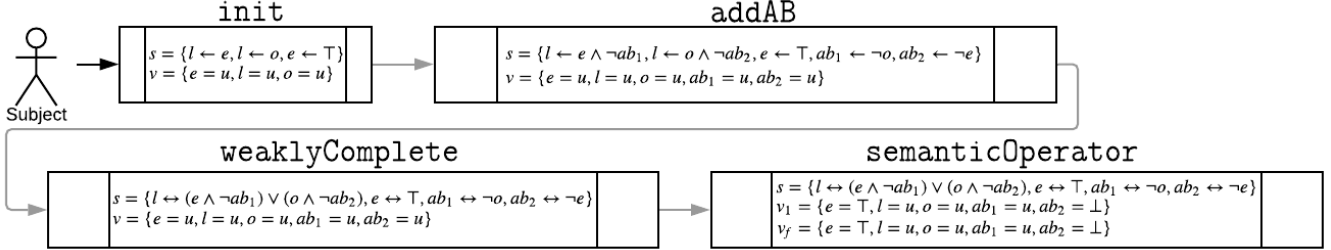


Figure 3: The standard case of the Suppression Task, demonstrating the suppression effect. Where the epistemic state in the boxes represents the output of that cognitive operation.  $V_f$  represents the assignment of  $V$  in the epistemic state in the resulting least model.  $V_{i \in \mathbb{N}}$  represents the assignments in  $V$  after  $i$  iterations of the semantic operator.

initial variables and rules to epistemic state; **addAB** adds abnormalities to the current epistemic state using the procedure described in Algorithm 1 (but now also adds those abnormalities to the variable list of the epistemic state); **WeaklyComplete** weakly completes the knowledge base of the current epistemic state; and **semanticOperator** returns an epistemic state that leaves the knowledge base unchanged but updates the variables of that state to return the least model of the epistemic state. **semanticOperator** follows the same logic seen in Section 2.3, but directly updating  $V$  after each iteration of the semantic operator, instead of the externalised variable set  $J$ . Thus, we ensure that the output state point is able to communicate the result of applying the semantic operator without any structural changes to the epistemic states it contains. As an additional feature of the **semanticOperator** cognitive operation, if there exists a labelled set in  $R$  called *fixed*, then the semantic operator will not set the value of any  $v \in \text{fixed}$  in the variables list  $V$ . The goal  $\gamma$  states that  $l$  should no longer be mapped to unknown in the final epistemic state.

Treating Figure 2 as an SCP, we observe the sequence of output states seen in Figure 3. Note that in the final state  $l$  remains mapped to  $u$ , meaning that the suppression effect is demonstrated.

### Extending the Suppression Task with SCPs

The previous example merely showed that SCPs are suitable for modelling the suppression task. In this example we consider one of the most powerful characteristics of SCPs, the ability to model unusual results as deviations from general reasoning. In the original Sup-

pression Task Experiment examined in Section 3, a significant portion of people still believed that she would study late in the library, even though the majority suppressed the inference. Several possible explanations are intuitive, the first and simplest, is the assumption that the reasoner is using classical logic and drawing the classical conclusion. However, what if that is not the case? What if they do reason in exactly the same way as the other reasoners, except for one or two small deviations?

In order to model these non-general reasoners, we consider two possible deviations that could explain the classical result of the Suppression Task: *variable deletion*, and *variable fixation*. Both of these operations will be discussed in a way that may seem overly prosaic, but it is done to reinforce that we might, reasonably, expect these cognitive operations to occur in day-to-day human cognition.

**Variable Deletion** Consider the sequence of numbers: 1, 44, 27, 8, 0, -4, 6, 7, 346, 7, 74, 7, 234, -55, 2.4, 18. Now without looking back at the numbers, ask yourself some questions: how many numbers were there? Were any of them prime? How many numbers were repeated? In all probability you are not entirely sure. This simple thought experiment provides support for our first extension, the idea that variables can be “forgotten”, that is, that information that existed in the knowledge base at one point in time might no longer exist at a later timepoint.

This is not the only imaginable case where a variable might be removed from the knowledge base of the person being modelled. The size of the knowledge base used for cognitive modelling is always implicitly

restricted to relevant variables. Only those variables whose values might reasonably be expected to affect the final conclusions drawn with regard to the research question should be considered. Finding which variables and rules are relevant is, however, non-trivial. For another real-life example, imagine a mystery novel: Three hundred pages of plot descriptions, character actions, and dialogues. In a good murder mystery novel every piece of information that reveals the killer’s identity is hidden in the story itself, yet we do not hold every fact and interaction in the book in our epistemic model of the book, so discerning the identity of the killer remains a mystery until the last page. But when the mystery is solved, many details that we internalised while reading (and recall in retrospect) suddenly make the conclusion seem obvious. We have not forgotten this information, we had merely incorrectly deemed it irrelevant at the time and ignored it in our cognitive processing.

The exact details of how to delete a variable from a knowledge base are non-trivial, and there is no best practice for doing so. But in simple cases the process can be intuitive. Let **delete** be a complex operation. **delete** takes as input any state point and is applicable for any ground point  $\bar{p}$  with variable list  $V \in \bar{p}$  and categorization variable  $R$  with  $(delete : V_{del}) \in R$ , where  $V_{del}$  is the set of variable names to delete. For every  $v \in V_{del}$  remove all rules from  $KB$  that have  $v$  as body or head of the clause, and remove  $v$  from  $V$ . Then remove **delete** from  $R$ .

In the case of the Suppression Task we argue that one cognitively valid reason for drawing the classical conclusion to the task may be forgetting (or disregarding) the variable  $o$ . Figure 4 illustrates this case, and shows how the insertion of a complex operation can completely change the final epistemic state.

**Variable Fixing** The second case of a potential complex operation to add to the search space of our SCPs is the idea of Variable Fixing. The idea that some conclusions can be fixed *a priori*. Consider a person who strongly doubts the effectiveness of vaccines, we will call her Karen. Karen started her day convinced that giving her child the MMR vaccine is more dangerous than the disease itself. Later that day Karen spoke to her doctor who strongly advised that she vaccinate her child. He offered her a variety of peer-reviewed papers and studies that showed the relative safety of the vaccination. Karen listened carefully to the trained medical professional, and then went home. After some thought Karen decided that he was wrong, and her opinion on vaccines didn’t change.

In this example Karen shows a very powerful type of cognitive bias, the unwillingness to change her opinions, despite powerful evidence to the contrary. This phenomenon has been observed across a great many fields of study, from medical psychology (Brown et al., 2010) (Wroe et al., 2005) to political sciences (Tappin, van der Leer, and McKay, 2017). In the context of cognitive modelling with logics, it indicates that some men-

tal rules or variables are immutable, regardless of new evidence or valid beliefs that would logically contradict them. Non-monotonic logics, as a class, are already capable of dealing with bias effects, as non-monotonic logics are built on the basis of a preference operation.

As one possible implementation of this idea, let us introduce a cognitive operation called **fix**. **Fix** takes as input any state point and is applicable for any ground point  $\bar{p}$  with variable list  $V \in \bar{p}$  and categorization variable  $R$  with  $(fix_{\top} \vee fix_{\perp}) \in R$ . For all variables  $v$  such that  $(v \in V) \cap (v \in fix_{\top})$  set the value of  $v$  to  $\top$ . For all variables  $v$   $(v \in V) \cap (v \in fix_{\perp})$  set the value of  $v$  to  $\perp$ . Then remove  $fix_{\top}$  and  $fix_{\perp}$  from  $R$ . Append  $fixed : V_{fix}$  to  $R$ , where  $V_{fix}$  is the set of all variables fixed in this way.

Now, because **semanticOperator** does not change the values of variables mentioned in  $V_{fix} \in fixed : V_{fix}$ ,  $fixed \in R$ , Figure 5 shows the effect of adding a complex operation which fixes the value of the abnormality to false in  $v$  so that, no matter what rules in  $KB$  when the semantic operator is applied,  $ab_1$  will remain false.

## 7 Conclusions and Future Work

SCPs represent a novel and powerful framework for modelling non-monotonic logics. They have been shown capable of modelling the Suppression Task under the WCS for both general and individual cases. SCPs provide a dynamic framework incorporating cognitive operations which are applicable across different logics, provided that those logics share structural features in the initial epistemic state that are compatible with some subset of the known set of cognitive operations. Because the input and output epistemic states need not be structurally similar, SCPs may represent the first approach to modelling human cognition that is able to integrate multiple non-monotonic logic frameworks at run-time and at search-time.

The addition of two distinct search paradigms to SCPs may one day enable researchers to generate or test hundreds of thousands of models where only the initial epistemic state is known. The first search type, *De Novo Search*, involves finding an SCP that satisfies a goal condition from scratch, iterating over the set of cognitive operations in  $M$ . This search is important for producing a computationally valid explanation for the general (most common) results of a cognitive task. *Insertion Search*, by contrast, will require finding which additional complex operations can be added to an existing SCP to produce results consistent with the deviant reasoner. This search is important for describing unusual results as deviations from the general reasoner. It is also important for scoring the likelihood of these with respect to the original SCP. Finally, some form of scoring criteria is needed to compare SCPs to one another and evaluate their relative plausibility with respect to the known complexity of the cognitive operations involved. A modified version of Needleman-Wunsch algorithm for string matching has already been theoretically described and must now be implemented in order



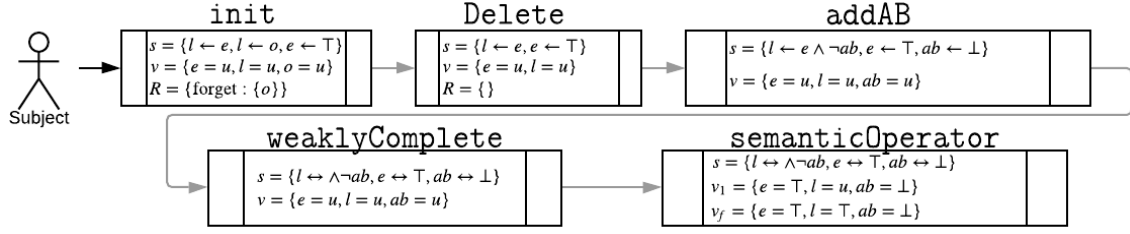


Figure 4: The Suppression Task in which the additional operation of deleting the variable  $o$  occurs.

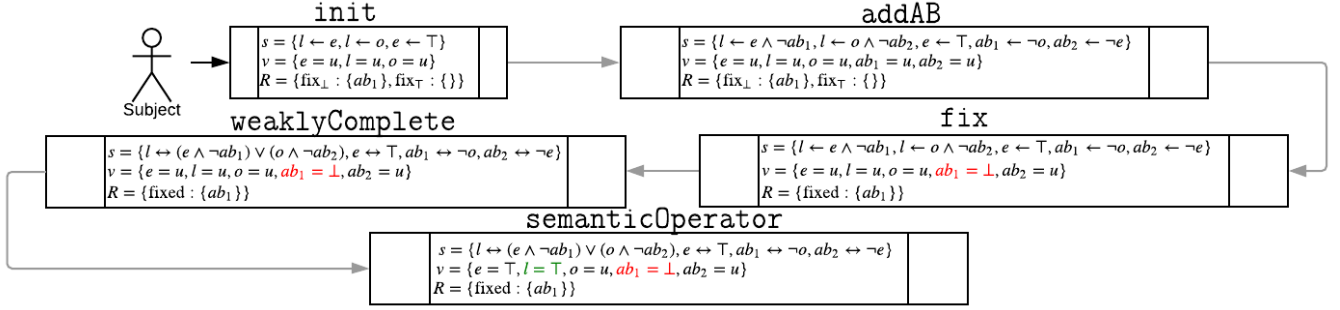


Figure 5: The Suppression Task in which the additional operation of fixing the variable  $ab_1$  to false occurs.

to determine how similar different models are to each other.

SCPs represent a new frontier for Cognitive Modelling in non-monotonic logics and research into their capabilities and limitations may help create more robust and mathematically consistent explanations and predictions for human behaviour across an extensive array of cognitive tasks.

## Acknowledgments

Special thanks to Marco Ragni, for his enthusiastic supervision of my ongoing academic studies; Raymond Ind and Sally Magliolo for their help in proof-reading this paper; and to Anastasia Sushich for her tireless support.

## References

- Breu, C.; Ind, A.; Mertesdorf, J.; and Ragni, M. 2019. The weak completion semantics can model inferences of individual human reasoners. In *European Conference on Logics in Artificial Intelligence*, 498–508. Springer.
- Brown, K. F.; Kroll, J. S.; Hudson, M. J.; Ramsay, M.; Green, J.; Vincent, C. A.; Fraser, G.; and Sevdalis, N. 2010. Omission bias and vaccine rejection by parents of healthy children: implications for the influenza a/h1n1 vaccination programme. *Vaccine* 28(25):4181–4185.
- Byrne, R. M. 1989. Suppressing valid inferences with conditionals. *Cognition* 31(1):61–83.
- Dietz, E.-A.; Hölldobler, S.; and Ragni, M. 2012. A computational logic approach to the suppression task. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 34.
- Dietz, E.-A.; Hölldobler, S.; and Wernhard, C. 2014. Modeling the suppression task under weak completion and well-founded semantics. *Journal of Applied Non-Classical Logics* 24(1-2):61–85.
- Hölldobler, S. 2015. Weak completion semantics and its applications in human reasoning. In *Bridging@CADE*, 2–16.
- McDermott, D., and Doyle, J. 1980. Non-monotonic logic i. *Artificial intelligence* 13(1-2):41–72.
- Mocanu, D. C.; Mocanu, E.; Stone, P.; Nguyen, P. H.; Gibescu, M.; and Liotta, A. 2018. Scalable training of artificial neural networks with adaptive sparse connectivity inspired by network science. *Nature communications* 9(1):1–12.
- Reiter, R. 1980. A logic for default reasoning. *Artificial intelligence* 13(1-2):81–132.
- Saldanha, E.-A. D.; Hölldobler, S.; and Rocha, I. L. 2017. The weak completion semantics. In *Bridging@CogSci*, 18–30.
- Sigman, M., and Dehaene, S. 2008. Brain mechanisms of serial and parallel processing during dual-task performance. *Journal of Neuroscience* 28(30):7585–7598.
- Smirnova, A.; Zorina, Z.; Obozova, T.; and Wasserman, E. 2015. Crows spontaneously exhibit analogical reasoning. *Current Biology* 25(2):256–260.

- Stenning, K.; Van Lambalgen, M.; et al. 2008. Interpretation, representation, and deductive reasoning. *Reasoning: Studies of human inference and its foundations* 223–248.
- Tappin, B. M.; van der Leer, L.; and McKay, R. T. 2017. The heart trumps the head: Desirability bias in political belief revision. *Journal of Experimental Psychology: General* 146(8):1143.
- Wason, P. C. 1968. Reasoning about a rule. *Quarterly journal of experimental psychology* 20(3):273–281.
- Wroe, A. L.; Bhan, A.; Salkovskis, P.; and Bedford, H. 2005. Feeling bad about immunising our children. *Vaccine* 23(12):1428–1433.

## Appendix

### Implementations

An existing Python 3 implementation of the SCP framework (available at [https://github.com/AxelInd/SCP\\_Implementation](https://github.com/AxelInd/SCP_Implementation)) has been able to model experiments related to the Suppression Task and the Wason Selection Task for both general and individual case reasoners using both the WCS and Reiter’s Default Logic.

### Systematically Adding Abnormalities

Adding abnormalities is a poorly described process and often relies on intuitionist views of what actually constitutes an abnormal situation. For this reason we have provided one concrete method for adding abnormalities to a logic program (though many others exist and can be extended at will). This particular one is based on the principle that if a variable  $v_1$  has the power to make another variable  $v_2$  true, then it may be the case that participants believe that any other variables  $V'$  satisfying  $v_2$  may only do so when  $v_1$  holds. This intuition is counter to classical logic, but follows intuitively in cases like the Suppression Task where participants believe that the library not being open may be the reason that she doesn’t study late in the library, even when she has an essay to write.

In the case of SCPs, the procedure for creating abnormalities is handled by a single complex epistemic operation and so may be substituted for any competing method without disrupting the flow of the SCP. The precise procedure followed for both the WCS and SCP formulation is given by Algorithm 1.

---

**Algorithm 2** One systematic way to add abnormalities to a logic program.

---

```

function ADDABNORMALITIES(P)
   $P' := \square$ 
  for Unique  $head$  with  $(body \rightarrow head) \in P$  do
     $headDeps :=$  [all non-ground bodies
    s.t.  $(body \rightarrow head) \in P$ ]
    for  $body_i$  with  $(body \rightarrow head) \in P$  do
       $ab_x :=$  new abnormality  $\triangleright$  Where  $x$  is a
      unique identifier
      create new rule  $head \leftarrow body_i \wedge \neg ab_x$ 
       $abDeps := headDeps \setminus body_i$   $\triangleright \perp$  if it
      would be empty
       $rule =$  create new rule  $(ab_x \leftarrow dep_1 \wedge \dots \wedge$ 
       $dep_n)$  for  $dep \in abDeps$ 
       $P' += rule$ 
    end for
  end for
  return  $P'$ 
end function

```

---