

THE UNIVERSITY of EDINBURGH

Edinburgh Research Explorer

Modelling virtual bargaining using logical representation change

Citation for published version:
Bundy, A, Philalithis, E & Li, X 2021, Modelling virtual bargaining using logical representation change. in SH Muggleton & N Chater (eds), *Human-Like Machine Intelligence.* Oxford University Press, pp. 68-89. https://doi.org/10.1093/oso/9780198862536.003.0004

Digital Object Identifier (DOI):
10.1093/oso/9780198862536.003.0004

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Peer reviewed version

Published In:
Human-Like Machine Intelligence

General rights
Copyright for the publications made accessible via the Edinburgh Research Explorer is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The University of Edinburgh has made every reasonable effort to ensure that Edinburgh Research Explorer content complies with UK legislation. If you believe that the public display of this file breaches copyright please contact openaccess@ed.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.

OPEN ACCESS

Download date: 28. Mar. 2024

Modelling Virtual Bargaining using Logical Representation Change

Alan Bundy, Eugene Philalithis and Xue Li
University of Edinburgh, UK

OXFORD
UNIVERSITY PRESS

Abstract

We discuss work in progress on the computational modelling of virtual bargaining: inference-driven human coordination under severe communicative constraints. For this initial work we model variants of a two-player coordination game of item selection and avoidance taken from the current virtual bargaining literature. In this range of games, human participants collaborate to select items (e.g. bananas) or avoid items (e.g. scorpions), based on signalling conventions constructed and updated from shared assumptions, with minimal information exchange. We model behaviours in these games using logic programs interpretable as logical theories. From an initial theory comprised of rules, background assumptions and a basic signalling convention, we use automated theory repair to jointly adapt that basic signalling convention to novel contexts, with no explicit coordination between players. Our ABC system for theory repair delivers spontaneous adaptation, using reasoning failures to replace established conventions with better alternatives, matching human players' own reasoning across several games.

0.1 Introduction - Virtual Bargaining

A recently developing body of empirical work on joint problem-solving explores limit cases of human coordination, where signalling conventions can still be efficiently formed, and flexibly revised, even without sufficient information bandwidth to coordinate them.

In a typical example, pairs of human participants are presented with tasks where *(a)* the information required to complete each task, and *(b)* the capacity to act on that information, are divided between them. One participant - a Sender - holds key information but cannot act on it. The other participant - a Receiver - can take the actions needed but needs additional information to select these actions among possible alternatives (Misyak, Noguchi and Chater, 2016). Neither participant can use language, or another medium of sufficient bandwidth to express all of the information required.

Yet human participants comfortably succeed in these 'impossible' coordination games. Humans select optimal moves (Misyak and Chater, 2014), create appropriate conventions (Misyak, Noguchi and Chater, 2016) and develop these initial conventions appropriately as the task complexity grows (Misyak and Chater, 2017) in order to maximise their joint profit - all this despite a greatly restricted communication channel.

This success in the face of insufficient explicit communication motivates the theory of *virtual bargaining*. Virtual bargaining rests on the need for additional inference to bridge the gap between the information *available*, and the information *required* to interpret players' signals. According to virtual bargaining, this added inference takes the form of a 'what if' scenario played out privately by both players, each adopting the most beneficial outcome of virtual negotiation on how to interpret their signals (Misyak and Chater, 2014). When both players imagine the same 'what if' scenario and play as if the virtual negotiation really happened, their interpretations will match.

Virtual bargaining therefore divides the burden of signal interpretation between observed information and private reasoning - and as a result can explain instances of 'impossible' coordination. However, no computational model currently exists for how the (effectively one-shot) learning and flexible revision displayed across these instances can be feasibly reproduced. At the same time, it has been argued that virtual bargaining underwrites a number of human conventions and unwritten rules, from

politics (Misyak, Melkonyan, Zeitoun and Chater, 2014) to driving (Chater, Misyak, Watson, Griffiths and Mouzakitis, 2018). The promise of human-like virtual bargaining abilities replicated by artificial agents is thus vast: from machines that communicate non-verbally but effectively with humans in joint problem-solving, to machines that grasp, create and share unwritten workplace rules with humans, and with each other.

Our present paper reports work in progress aiming to replicate this coordination behaviour. We consider examples of signalling conventions spontaneously adapted in a simple game of item selection and avoidance. And we suggest that the rich, efficient inference stipulated by virtual bargaining for these conventions can be understood as *logical inference* capturing the players' joint reasoning about each game and its rules; specifically, logical inference facilitated by the ABC system for representation change.

The **ABC** Repair System (Li, Bundy and Smaill, 2018) combines **A**bduction (Cox and Pietrzykowski, 1986) and **B**elief revision (Gärdenfors, 1992) with the more recent Reformation algorithm (Bundy and Mitrovic, 2016) for **C**onceptual change. Abduction and belief revision repair faulty logic theories, by respectively adding/deleting axioms or deleting/adding preconditions to rules. Reformation repairs them by changing the *language* of a theory. For practical reasons (discussed below in §0.3.2), the ABC System is limited to Datalog theories (Ceri, Gottlob and Tanca, 1990), although Reformation has been implemented for richer logics (Bundy and Mitrovic, 2016; Mitrovic, 2013). Datalog is a logic programming language restricted to Horn clauses and allowing no functions except for constants; but it has proven adequately expressive for our usage.

0.2 What's in the box?

We begin by considering the family of human behaviours we presently aim to model, in the form of moves made by players in a coordination game, and our overall approach.

A reliable demonstration of virtual bargaining is built around a 2-player game of item selection and avoidance (Misyak, Noguchi and Chater, 2016). In this game, a Sender can see inside boxes with harmful or helpful contents, such as a scorpion or a banana. The Sender can mark one of the items for the Receiver in some way, but cannot open them. In turn, the Receiver can open any of the items, but they cannot see inside. The players' joint goal is to open as many helpful items as possible per round while opening no harmful items; their restriction is that the Sender alone cannot give sufficient input for the Receiver to determine what unopened items belong in each set.

The general procedure is described as follows (Misyak, Noguchi and Chater, 2016):

"We developed an interactive two-player computer game in which both partners viewed a 3-D-simulated scene, but each saw the scene from the opposite visual perspective [...]. The game environment consisted of three boxes, each containing either a reward (banana) or nonreward (scorpion). The number of rewards and the rewards' locations in the boxes, as well as other scene variables, changed from trial to trial. One partner played the role of sender, and the other played as a receiver: They shared the joint task of uncovering as many rewards as possible while avoiding nonreward. Contents of the boxes were visible only to the sender by means of panels that slid open on the side of the box facing the sender [...]. However, a set of shadows (impressions of bananas and scorpions, embedded in the virtual ground) was sometimes mutually visible to both players. The shape and number of these shadows corresponded to the number of scorpions and bananas inside the three boxes on that trial."

In other words, both players know the ratio of helpful (bananas) to harmful items (scorpions); but only the Sender knows what's in each box. **Figure 0.1** illustrates the baseline condition of this arrangement. The Sender may then use a token visible to both players in order to mark items for the Receiver's benefit. In the manipulation of interest to virtual bargaining, the Sender can mark *at most one* out of the three items.

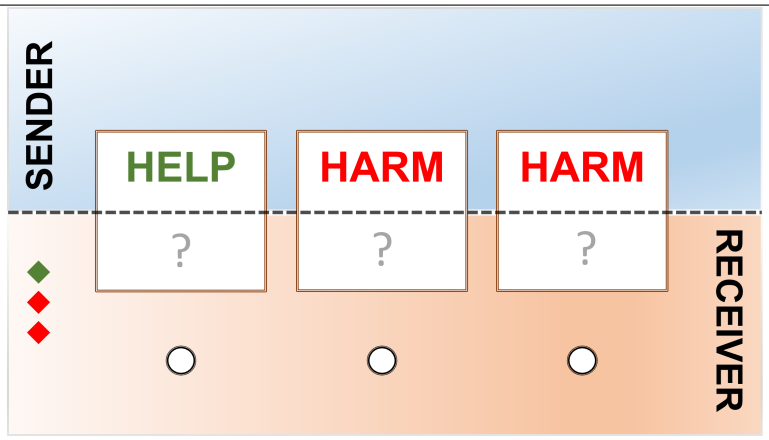


Fig. 0.1 Basic game setup. Sender sees item contents. Receiver only sees content ratio.

Players take turns: the Sender marks, then the Receiver makes their choices, before the outcome of the game is announced to both. Selecting the maximum number of helpful items, while avoiding all harmful items, will win the game; all other outcomes lose. Winning the game is conditional on interpretation: where only one mark (i.e. one axe token) is available to the Sender, players must use it flexibly. When there are more harmful than helpful items (i.e. 2 scorpions to 1 banana) the Sender marks the single helpful item, and the Receiver interprets the mark to mean 'helpful'. When there are more helpful than harmful items (i.e. 1 scorpion to 2 bananas) the Sender marks the *harmful* item, and the Receiver interprets the mark to mean '*all other items* helpful'.

Negotiating this flexible signalling convention explicitly is impossible for players. They must instead *infer* their interpretation - e.g. from what they both know about the game (the game rules, the end goal, the ratio of helpful to harmful items) and their respective roles in it. The two players are both given the same rules of the game, and both have the same goal. Otherwise, the only mode of communication available to the players are the tokens used to mark items by the Sender. From their shared knowledge of the rules and goals of the game, plus their own private view of the experimental set-up, the players must devise a jointly appropriate convention: the Sender must use the tokens to signal the contents of the items in a way the Receiver will understand, and the Receiver must decipher those signals, and act on them as the Sender intended.

Furthermore, the game itself can also evolve, adding novel situations, restrictions or signalling vocabularies (Misyak and Chater, 2017). As a result, any initially established conventions will not always remain optimal. The Sender and Receiver must adapt

their convention to suit each new scenario before playing it, via reasoning, rather than physical trial and error. All of these behaviours are attributed to virtual bargaining: the advance modelling, via reasoning, of joint problem-solving before it even happens.

A first step toward replicating virtual bargaining is thus a faithful reproduction of players' behaviours in this selection and avoidance game. That task is made easier by the very minimal interaction permitted between players: the Sender's choice of mark and the Receiver's choice of items, taken in turns, are all the information transmitted. Modelling human players' behaviour therefore reduces to modelling their choices - e.g. as a result of inference capturing players' reasoning from the minimal available input.

The present body of work on virtual bargaining distinguishes two clearly separable reasoning steps in how human players respond to this family of games (Misyak and Chater, 2017): *(a)* constructing an initial signalling convention from players' shared knowledge of the game rules, the end goal, and other available information; and *(b)* adapting that convention after circumstances change. Accordingly, the job of modelling virtual bargaining divides into two distinct pieces: *(i)* an algorithm for how players use the available information to spontaneously establish a signalling convention without negotiation; and *(ii)* an algorithm of how players spontaneously adapt that signalling convention without negotiation. Our present aim is to model the latter process: how players spontaneously adapt established signalling conventions to novel requirements.

0.3 Datalog Theories

We now move to consider our toolset, starting with Datalog. Originally invented as a subset of Prolog, i.e., as a logic programming language, which was targeted at querying deductive databases (Ceri, Gottlob and Tanca, 1990), Datalog can also be treated as a sub-logic of first-order logic.

0.3.1 Clausal Form

Datalog programs are a collection of rules and ground facts. They can be represented as a subset of Horn clauses, which are disjunctions of negated or unnegated propositions. To emphasise this relationship, we will use Kowalski's clausal format[1] (Kowalski, 1979):

$$(Q_1 \wedge \ldots \wedge Q_m) \implies (R_1 \vee \ldots \vee R_n) \tag{0.1}$$

where the Q_i are implicitly negated because they are on the LHS of the implication arrow.

Definition 1. (Horn Clauses) *Horn clauses are clauses (0.1) in which either $n = 0$ or $n = 1$. They, therefore, fit one of the following four forms.*

Implication: $(Q_1 \wedge \ldots \wedge Q_m) \implies R$. *These usually represent the rules of a theory.*
Assertion: $\implies R$. *These usually represent the facts of a theory.*
Goals: $Q_1 \wedge \ldots \wedge Q_m \implies$. *These usually arise from the negation of the conjecture to be proved and subsequent subgoals in a derivation.*

[1] Kowalski advocates a variation of this format which is more suggestive of its procedural reading. He puts the head on the left, the body on the right and writes the implication arrow backwards. This is also the version used in Prolog.

Empty Clause: $\Longrightarrow$. *This is the target of a refutation-style proof. It represents success in proving a conjecture.*

where $1 \leq m$*, and* R *and the* Q_i *are* propositions*, i.e., formulae of the form* $P(t_1, \ldots, t_n)$*, where each* t_j *is either a variable or a constant. Where they exist,* R *is called the* head *of the clause and the* Q_i *form the* body*.*

We will adopt the convention that variables are written in lower case, and constants and predicates start with a capital letter[2].

0.3.2 Datalog Properties

Datalog programs standardly consist of implications (rules) and ground assertions (facts). Our Datalog theories, however, also contain goals and the empty clause, so as to represent conjectures to be proved, and the derivation of false in refutation proofs.

In our Datalog theories, we also adopt the following Datalog program restrictions:

1. There are no non-nullary functions, i.e., the arguments to predicates are either variables or constants, so there is no function nesting.
2. Each predicate has a unique arity.
3. There are no unsafe clauses, i.e. each variable that appears in the head of a clause also appears in the body of that clause.

As we see below, despite these restrictions, Datalog is sufficiently expressive for our application to virtual bargaining. Further possible restrictions exist, to allow Datalog to be more efficient as a programming language, which we do not need to adopt here.

Deduction in Datalog is decidable. This is not the case in full first-order logic (FOL), which is only semi-decidable, i.e., if there is a proof, FOL deduction will eventually find it by exhaustive search, but if there isn't we could search fruitlessly for ever. In Datalog, if there is no proof of a conjecture, the search will eventually terminate without success, so we can be sure that the conjecture is not a theorem. This is one of the more important technical advantages of restricting our logical theories to Datalog.

The decidability of Datalog is a consequence of its lack of functions. This is because there are only a finite number of ground terms[3], namely the set of constants. This means that there are only a finite number of distinct formulas[4]. Since the number of ground terms is finite, and the Herbrand base is also finite, all quantified formulas can be translated into propositional logic and as a result all Datalog theories are decidable.

In addition to implementing deduction in Datalog theories, our ABC system also uses a special mechanism for the = predicate, based on the unique name assumption. Different constants are assumed to be unequal, unless this assumption is overridden by an explicit = relation asserted between them. This has the consequence that we can treat $\neq$ as an unnegated predicate in its own right, not as the negation of =. This enables its use as a predicate in propositions that form the clauses of Datalog theories.

[2]The opposite of the standard Prolog convention.

[3]That is, terms without variables.

[4]Up to variable renaming.

0.3.3 Application 1: Game Rules as a Logic Theory

The rules of the selection and avoidance game can be represented as a Datalog theory[5]:

- The Receiver must select each helpful item.

$$item \in Help \implies Select(Receiver, item) \tag{0.2}$$

- The Receiver must not select a harmful item.

$$item \in Harm \wedge Select(Receiver, item) \implies \tag{0.3}$$

  Note our unusual use of a goal clause to express constraint (0.3) as a Horn clause. This would not be allowed in a Datalog *program* but it is allowed in our Datalog *theories*.
- No items are both helpful and harmful.

$$item_1 \in Help \wedge item_2 \in Harm \implies item_1 \neq item_2 \tag{0.4}$$

- If there are both helpful and harmful items then the Sender must mark an item.

$$Help \neq \varnothing \wedge Harm \neq \varnothing \implies Mark(Sender, Sk) \tag{0.5}$$

  where Sk is a Skolem constant, an item whose identity we know nothing about.
- The Sender can mark at most one thing.

$$Mark(Sender, item_1) \wedge Mark(Sender, item_2) \implies item_1 = item_2 \tag{0.6}$$

In the above formalisation, $Help$ corresponds to the set of items[6] (boxes) containing bananas and $Harm$ to the set containing scorpions. $Mark(Sender, item)$ means the Sender places a token on $item$. $Select(Receiver, item)$ means the Receiver opens $item$.

For present purposes, we only consider cases where the Sender only has one token they can place. For a condition in which the sender has more tokens, the rules will be slightly different. In such a case, the game rules themselves will have changed - but these rules are determined by the experimenter. The participants would be *informed* the rules have changed, rather than *infer* the necessary changes as a result of e.g. logical reasoning. The evolution of players' rules knowledge is not part of our modelling target.

It is important to note that, at this stage, the above rules are insufficient for either player to plan their moves. An additional logical step is required, as we discuss below.

0.3.4 Application 2: Signalling Convention as a Logic Theory

We now move to consider the baseline condition of our selection and avoidance game, abstracted in **Figure 0.2**. The top and bottom half represent the game environment as the Sender and the Receiver view it. The labels on the items indicate whether their contents are helpful or harmful, which the Sender can see. The Receiver (who is below the dividing line) cannot see the contents of the items. The tick under one of the items denotes the item being marked with a token by the Sender, as a signal for the Receiver.

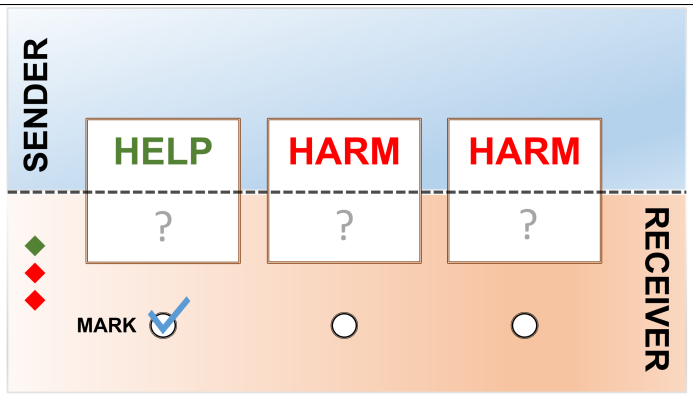


Fig. 0.2 Baseline condition. Marking an item implies the item is helpful.

We will refer to the items as Box_1, Box_2 and Box_3 from left to right, respectively.

For the Sender's mark to work as a signal encoding information about the game, players require a convention for its interpretation. The convention used by participants in this baseline case is quite simple, and can be represented by the following two clauses:

- Marking an item signals an item.

$$Mark(Sender, item) \implies Signal(item) \tag{0.7}$$

- Any signalled items must be helpful.

$$Signal(item) \implies item \in Help \tag{0.8}$$

In other words, marking is a signal for 'helpful'. Although we do not presently explore how this initial convention is spontaneously established, it is arguably intuitive. In any item selection game, interpreting a signal as simply meaning 'this is the item to select' is a likely initial strategy for players to attempt, even without coordination.

Combined with the above, this convention easily determines the Receiver's strategy:

- Marking an item signals an item.

$$Mark(Sender, item) \implies Signal(item)$$

- Any signalled items must be helpful.

$$Signal(item) \implies item \in Help$$

- Select each helpful item.

$$item \in Help \implies Select(Receiver, item)$$

(Note that this is rule (0.2) from §0.3.3, so is protected from repair.)

[5]We call this the *Unique Name Assumption with Exceptions*. Note that this allows the use of both $=$ and $\neq$ in a Datalog theory.

[6]We use set membership, rather than a unary predicate, to denote helpful vs. harmful. Together with our implementation of $\neq$ this allows *non*-membership to be represented via Horn clause in (0.5).

The Sender's strategy can be given accordingly, using (0.5), (0.6), (0.7), and (0.8).

0.4 SL Resolution

SL Resolution (Kowalski and Kuehner, 1971) is a deductive rule that is particularly well suited to Reformation. A single SL Resolution step takes the form:

$$\frac{R_1 \wedge \ldots \wedge R_i \ldots \wedge R_k \implies}{(R_1 \wedge \ldots \wedge R_{i-1} \wedge Q_1 \wedge \ldots \wedge Q_m \wedge R_{i+1} \ldots \wedge R_k)\sigma \implies} \; (Q_1 \wedge \ldots \wedge Q_m) \implies \mathbf{P}$$

where the highlighted R_i is the selected literal, the highlighted P is the literal it is resolved with and σ is the most general unifier of P and R_i, i.e., the most general substitution of terms for variables that will make them identical.

0.4.1 SL Refutation

Resolution proofs work by refutation. The conjecture to be proved is negated and added to the axioms. The empty clause, $\implies$, is then (we hope) derived. This derivation is interpreted as showing that negating the target conjecture leads to a contradiction, such that the conjecture has been proved by *reductio ad absurdum*. For Horn clause theories, negated conjectures take the form of a goal clause, as already shown in (0.3).

A SL Resolution refutation on Horn clauses takes the form:

$$\begin{array}{c} \dfrac{Goal}{Goal_1 \wedge \ldots \wedge Goal_m} \; Axiom \\ \vdots \\ \dfrac{Goal}{\implies} \; Axiom \end{array}$$

where the *Goal*s are all goal clauses and the *Axiom*s are either implication or assertion clauses. This has the advantage that we can apply any repair (as we discuss in §0.5.1) directly to the axiom involved in the resolution step, without needing to inherit the repair back up the refutation to an axiom. This advantage is secured by restricting to Datalog theories, as all their formulae are Horn clauses. SL Resolution refutation on non-Horn clauses also requires ancestor resolution: i.e. resolution between a goal literal and another goal above it on the same branch. In that case, no axiom is directly involved and inheritance is required. Avoiding the need for such inheritance is another of the technical advantages gained by restricting our logical theories to Horn clauses.

0.4.2 Executing the Strategy

The Receiver's strategy determines which items they select as a result of the Sender's actions. This strategy can be executed by applying SL Resolution to the goal clause $Select(Receiver, item) \implies$ using the convention clauses from §0.3.4 plus the assertion $\implies Mark(Sender, Box_1)$, as observable from Fig. 0.2. In the course of this refutation, *item* will be instantiated to one of the three available boxes. The desired refutation is:

$$\dfrac{\dfrac{\dfrac{\dfrac{Select(Receiver, item) \implies}{item \in Help \implies} \; item \in Help \implies Select(Receiver, item)}{Signal(item) \implies} \; Signal(item) \implies item \in Help}{Mark(Sender, item) \implies} \; Mark(Sender, item) \implies Signal(item)}{\implies} \implies Mark(Sender, Box_1) \tag{0.9}$$

This proves $Select(Receiver, Box_1)$: the Receiver selects (just) Box_1 as intended.

The Sender's strategy can also be represented using the clauses from §0.3.4, but with the instantiated goal clause $Select(Receiver, Box_1) \implies$ and the uninstantiated assertion $\implies Mark(Sender, item)$, where $item$ must be instantiated to the specific item to be marked with a token. This is because the Sender wants to discover which item they must mark, so that the Receiver will subsequently select Box_1, as intended:

$$\dfrac{\dfrac{\dfrac{\dfrac{Select(Receiver, Box_1) \implies}{Box_1 \in Help \implies} \; item \in Help \implies Select(Receiver, item)}{Signal(Box_1) \implies} \; Signal(item) \implies item \in Help}{Mark(Sender, Box_1) \implies} \; Mark(Sender, item) \implies Signal(item)}{\implies} \implies Mark(Sender, item)$$

0.5 Repairing Datalog Theories

Having considered theory representation, we now move to consider theory repair. The ABC System (Li, Bundy and Smaill, 2018) diagnoses and repairs two kinds of fault in Datalog theories: incompatibility and insufficiency. Both arise from reasoning failures: mismatches between the theorems of a theory $\mathbb{T}$ and the observations of an environment $\mathbb{S}$, such as our game environment. $\mathbb{S}$ is a pair of sets of ground propositions $\langle T(\mathbb{S}), F(\mathbb{S})\rangle$. A *ground proposition* is a formula of the form $P(C_1, \ldots, C_n)$, where P is an n-ary predicate and the C_i are constants. $T(\mathbb{S})$ are the ground propositions we observe to be true and $F(\mathbb{S})$ are those we observe to be false. So, ideally:

$$R \in T(\mathbb{S}) \implies \mathbb{T} \vdash R$$
$$R \in F(\mathbb{S}) \implies \mathbb{T} \nvdash R$$

That is, the true ground propositions are theorems of $\mathbb{T}$ and the false ones are not.

We can view the theorems of $\mathbb{T}$ as predictions about the environment. These predictions can be confounded in two ways: something false is predicted (incompatibility) or something true is not predicted (insufficiency).

Definition 2. (Incompatibible and Insufficient)

Incompatible: $\mathbb{T}$ *is* incompatible *with* $\mathbb{S}$ *iff* $\exists R.\ \mathbb{T} \vdash R \wedge R \in F(\mathbb{S})$.
Insufficient: $\mathbb{T}$ *is* insufficient *for* $\mathbb{S}$ *iff* $\exists R.\ \mathbb{T} \nvdash R \wedge R \in T(\mathbb{S})$.

0.5.1 Fault Diagnosis and Repair

These two kinds of fault are diagnosed and repaired in a dual way. $F(\mathbb{S})$ and $T(\mathbb{S})$ are both finite sets. The ABC system tries to prove each member of these sets. If a member

of $F(\mathbb{S})$ is proved then we have discovered an incompatibility. Similarly, if a member of $T(\mathbb{S})$ is not proved then we have discovered an insufficiency. Incompatibilities can be repaired by blocking the unwanted proof. Insufficiencies can be repaired by unblocking a wanted failed proof.

Definition 3. (Repair Operations for Incompatibility) *In the case of incompatibility, the unwanted proof can be blocked by causing any of the resolution steps to fail. Suppose the chosen resolution step is between a goal $P(s_1, \ldots, s_n)$ and an axiom $Body \implies P(t_1, \ldots, t_n)$, where each s_i and t_i pair can be unified. Possible repair operations are as follows:*

Belief Revision 1: *Delete the targeted axiom.*

Belief Revision 2: *Add an unprovable precondition to the body of the targeted axiom.*

Reformation 1: *Rename P in the targeted axiom to the new predicate P'.*

Reformation 2: *Increase the arity of all occurrences P in the axioms by 1. Ensure recursively that the new arguments, s_{n+1} and t_{n+1}, in the targeted occurrence of P, are not unifiable.*

Reformation 3: *For some i, suppose s_i and t_i are both the constant C. Change t_i to the new constant C'.*

Heuristic 1. (Algorithm for Creating New Arguments) *In operation Reformation 2 of Definition 3, we need to create a new argument for the $n+1$ argument position in each occurrence of P. The spirit of Datalog is that assertions are ground facts and implications are non-ground general rules. We have also observed that the new arguments are usually used to distinguish different types of P. Reformation is a purely syntactic algorithm, which does not have access to any semantics when choosing new constant names. Therefore, we use the new constants, $Abnormal$ and $Normal$, subscripted if necessary, when constants are needed as new arguments. Otherwise, we use new variables.*

The following algorithm is used to decide which term to use in each occurrence.

1. *For the targeted axiom $Body \implies P(t_1, \ldots, t_n)$ let t_{n+1} be $Abnormal$. For the goal proposition $P(s_1, \ldots, s_n)$ that it is resolved with let s_{n+1} be $Normal$.*
2. *Propagate these two constants by instantiating the resolutions steps they are inherited from or to: $Normal$ upwards and $Abnormal$ downwards.*
3. *Select one axiom whose $n+1$ argument has been instantiated to $Normal$ and one to $Abnormal$. Where there is a choice, prefer facts over rules. Typically, choose the top-most axiom for $Normal$ and the bottom-most one for $Abnormal$. For all other $n+1$s arguments, choose a new variable per axiom.*

This algorithm is illustrated in §0.5.2 below.

Definition 4. (Repair Operations for Insufficiency) *In the case of insufficiency, the wanted failed proof can be unblocked by causing a currently failing resolution step to succeed. Suppose the chosen resolution step is between a goal $P(s_1, \ldots, s_m)$ and an axiom $Body \implies P'(t_1, \ldots, t_n)$, where either $P \neq P'$ or for some i, s_i and t_i cannot be unified. Possible repair operations are:*

Abduction 1: *Add a new axiom whose head unifies with the goal* $P(s_1, \ldots, s_m)$.

Abduction 2: *Locate the axiom whose body proposition created this goal and delete this proposition from the axiom.*

Reformation 4: *Replace* $P'(t_1, \ldots, t_n)$ *in the axiom with* $P(s_1, \ldots, s_m)$.

Reformation 5: *Decrease the arity of all occurrences* P' *by 1. Remove the* i^{th} *argument from* P'*, i.e., the* i *for which* s_i *and* t_i *are not unifiable.*

Reformation 6: *If* s_i *and* t_i *are not unifiable, then they are unequal constants, say,* C *and* C'*. Either (a) rename all occurrences of* C' *in the axioms to* C *or (b) replace the offending occurrence of* C' *in the targeted axiom by a new variable.*

0.5.2 Example: The Black Swan

The following example is adapted from (Gärdenfors, 1992). Consider the following Datalog theory $\mathbb{T}$:

$$\begin{aligned} German(x) &\implies European(x) \\ European(x) \wedge Swan(x) &\implies White(x) \\ \implies German(Bruce) \qquad & \implies Swan(Bruce) \end{aligned}$$

From these axioms, we can infer $White(Bruce)$.

$$\dfrac{\dfrac{\dfrac{White(Bruce) \implies}{European(Bruce) \wedge Swan(Bruce) \implies}}{German(Bruce) \wedge Swan(Bruce) \implies}}{\implies} \quad \begin{aligned} &European(x) \wedge Swan(x) \implies White(x) \\ &German(x) \implies European(x) \\ &\implies German(Bruce) \\ &\implies Swan(Bruce) \end{aligned} \tag{0.10}$$

However, suppose we observe that Bruce is black and not white, i.e., $Black(Bruce) \in T(\mathbb{S})$ and $White(Bruce) \in F(\mathbb{S})$. $\mathbb{T}$ is, therefore, both incompatible and insufficient wrt $\mathbb{S}$.

We will deal with the incompatibility. One solution, mooted in (Gärdenfors, 1992), is to add an exception to one of the rules, e.g.:

$$x \neq Bruce \wedge European(x) \wedge Swan(x) \implies White(x)$$

This seems to us to be an unsatisfactory solution. A better solution is to note that $European(x)$ is ambiguous. It could be interpreted as x is a European *type* or as a European *resident*. European types of swans are white, but a black swan can be a resident, e.g., in a zoo.

We can achieve this repair, for instance, by adding an additional argument to *European* (see operation Reformation 2 in Definition 3). To affect this, we need to break the highlighted resolution step in refutation 0.10. We now need to create new terms for all these extra arguments by following heuristic 1. We consider first the

axiom involved in the targeted resolution step. Giving $European$ a new variable y as an argument gives:

$$German(x) \implies European(x, y)$$

We are now in violation of the safety restriction (restriction 3 in §0.3.2): there is a variable in the clause's head that does not appear in its body. So we must also add y to $German$.

$$German(x, y) \implies European(x, y)$$

We must now add a new argument to any other occurrence of $European$ and $German$ — and do so in such a way to ensure that the red unification in the refutation will fail. To help us decide how to do this, we:

- Add the new arguments of $European$ and $German$ into refutation (0.10).
- In the highlighted resolution step we want to break, instantiate the new arguments to the two occurrences of $European$ to $Normal$ in the goal and $Abnormal$ in the axiom.
- Propagate these instantiations through the refutation: $Normal$ upwards and $Abnormal$ downwards.

This gives:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{White(Bruce) \implies}}{European(Bruce, Normal) \wedge Swan(Bruce) \implies}}{German(Bruce, Abnormal) \wedge Swan(Bruce) \implies}}{\dfrac{Swan(Bruce) \implies}{\implies}}$$

$$\begin{array}{l} European(x, Normal) \wedge Swan(x) \implies White(x) \\ German(x, Abnormal) \implies European(x, Abnormal) \\ \implies German(Bruce, Abnormal) \\ \implies Swan(Bruce) \end{array}$$

which breaks the refutation by the failure of $European(x, Abnormal)$ to unify with $European(x, Normal)$. This analysis suggests the following repaired theory, $\nu(\mathbb{T})$:

$$\begin{array}{rl} German(x, y) & \implies European(x, y) \\ European(x, Normal) \wedge Swan(x) & \implies White(x) \\ \implies German(Bruce, Abnormal) & \implies Swan(Bruce) \end{array}$$

from which $White(Bruce)$ is no longer provable. This sequence is a basic example of a repair conducted automatically using the ABC system (Li, Bundy and Smaill, 2018).

0.6 Adapting the Signalling Convention

We now demonstrate an application of these theory repair ideas to the spontaneous adaptation of conventions for the selection and avoidance game. We will consider three alternatives to the baseline version of the game illustrated in **Figure 0.2**, and how the signalling convention used in that baseline can be automatically adapted for each.

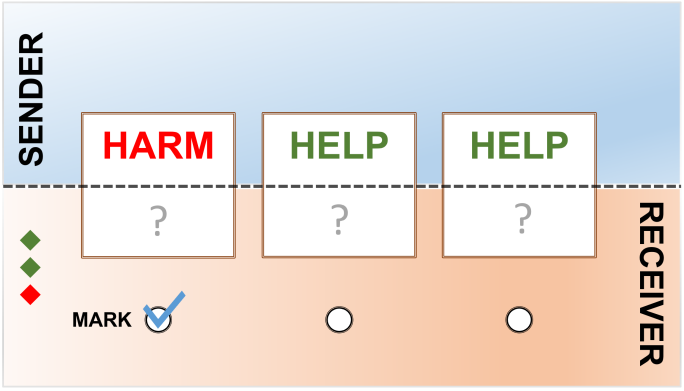


Fig. 0.3 'Avoid' condition. Marking an item implies other items are helpful.

0.6.1 'Avoid' Condition

A different variant of the selection and avoidance game is depicted in **Figure 0.3**.

In this alternative, there are two bananas, and to achieve their mutual goal, the Sender must guide the Receiver to select both of them. The Receiver still does not know exactly which items contain the bananas, but both players do know that there are more bananas than scorpions - as indicated in the figure by the annotation $|Harm| < |Help|$. This annotation must be added as an additional axiom $|Harm| < |Help|$. The Sender, as depicted, reverses the earlier convention, and marks the box containing the scorpion.

It is important to reiterate that in human trials, both players can spontaneously adopt this new convention without trial and error (Misyak, Noguchi and Chater, 2016); with this behaviour in particular used to motivate virtual bargaining as an explanation.

To explain how this spontaneous adaptation can be reproduced automatically, we return to our formalisations of the game, its rules, and the players' convention. If the convention defined in §0.3.4 is applied to this example, together with the game rules in §0.3.3, it will fail. Instead of indicating that the Receiver selects Box_2 and Box_3, which contain the helpful bananas, the convention will indicate that Box_1, containing the scorpion, is helpful. Meaning, there are two insufficiencies and one incompatibility.

Repairing Two Insufficiencies. We will deal first with the two insufficiencies. They are symmetric, so we tackle just the unprovable $Select(Receiver, Box_3) \in T(\mathbb{S})$. To focus on the root of the problem we can try to prove $Select(Receiver, Box_3)$ in refutation (0.11), instantiate all the variables and track the occurrences of Box_3 (highlighted in green) downwards and the occurrences of Box_1 upwards (highlighted in blue).

$$\dfrac{\dfrac{\dfrac{\dfrac{Select(Receiver, Box_3) \implies}{Box_3 \in Help \implies}}{Signal(Sender, Box_3) \implies}}{Mark(Sender, Box_3) \implies}}{\implies} \quad \begin{array}{l} item \in Help \implies Select(Receiver, item) \\ Signal(Sender, item) \implies item \in Help \\ Mark(Sender, item) \implies Signal(item) \\ \implies Mark(Sender, Box_1) \end{array} \tag{0.11}$$

We can repair this insufficiency using operation Reformation 3 from Definition 4 on the first half of the previously-established signalling convention, namely the axiom:

$$Mark(Sender, item) \implies Signal(item)$$

The failed unification will succeed if we replace the 'offending' argument in the targeted axiom with a fresh variable:

$$Mark(Sender, item) \implies Signal(item') \tag{0.12}$$

However, this will leave $item'$ as an orphan variable which, owing to the restrictions of our chosen language (Datalog), needs to appear in the body of the rule. This can be achieved by adding the new precondition $item \neq item'$

$$Mark(Sender, item) \wedge item \neq item' \implies Signal(item') \tag{0.13}$$

This new precondition will be satisfied by the implicit inequality $Box_1 \neq Box_3$ that is provided by the ABC System's unique name assumption mechanism. The modified refutation (0.11) will now succeed. The insufficiency has been repaired.

Repairing an incompatibility. We now move on to the incompatibility, caused by the fact that $Select(Receiver, Box_1) \in F(\mathbb{S})$ but is nonetheless provable:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{Select(Receiver, Box_1) \implies}{Box_1 \in Help \implies} \; item \in Help \implies Select(Receiver, item)}{Signal(Box_1) \implies} \; Signal(item) \implies item \in Help}{Mark(Sender, item) \implies} \; Mark(Sender, item) \wedge item \neq item' \implies Signal(item')}{\implies} \; \implies Mark(Sender, Box_1)}{\implies} \; \implies Box_1 \neq Box_2 \tag{0.14}$$

One notable aspect of human players' behaviour is that the contrary conventions appear to co-exist. Where the two game versions are played back-to-back, players are capable of efficiently toggling between the 'select' and 'avoid' signalling conventions (Misyak, Noguchi and Chater, 2016). In line with human performance on this task, we therefore want to repair this incompatibility such that the new convention is flexible enough to cover both the baseline as well as the alternative game. We can achieve this by duplicating an axiom and then modifying the two versions. One version of these axioms will work for the baseline and another version for the alternative in Figure 0.3.

We can repair this incompatibility using operation Belief Revision 2 from Definition 3 on the axiom:

$$Mark(Sender, item) \wedge item \neq item' \implies Signal(item')$$

The assertion, already in the theory as a ground proposition, which distinguishes the baseline condition in §0.3.4 from this one is $|Harm| < |Help|$, so that is the obvious precondition to add:

$$Mark(Sender, item) \wedge item \neq item' \wedge |Harm| < |Help| \implies Signal(item') \tag{0.15}$$

To ensure that axiom (0.12) will still be available to apply to the baseline, but not in the alternative condition in Figure 0.3, we must add the complementary precondition:

$$Mark(Sender, item) \wedge |Help| < |Harm| \implies Signal(item) \tag{0.16}$$

The complete revised signalling convention, in line with human behaviour, is now:

$$\begin{aligned} Mark(Sender, item) \wedge |Help| < |Harm| &\implies Signal(item) \\ Mark(Sender, item) \wedge item \neq item' \wedge |Harm| < |Help| &\implies Signal(item') \\ Signal(item) &\implies item \in Help \end{aligned}$$

This repair is uniquely, automatically identified by the ABC system for axiom (0.7). This is achieved by selecting each precondition only from ground propositions already contained in the theory, corresponding to players' knowledge of the two game variants.

0.6.2 Extended Vocabulary

A different variant of the item selection and avoidance game is depicted in **Figure 0.4**.

In this variant game, the rules are changed to allow two mark placements for each item: an outer and inner position. This choice can be used to distinguish between the left- and right-hand situations depicted. In the former (as with Figure 0.2) only the marked item is helpful; in the latter, all of the items are helpful. Humans spontaneously exploit the extended vocabulary of 'outer' and 'inner' marks to express this difference (Misyak and Chater, 2017).

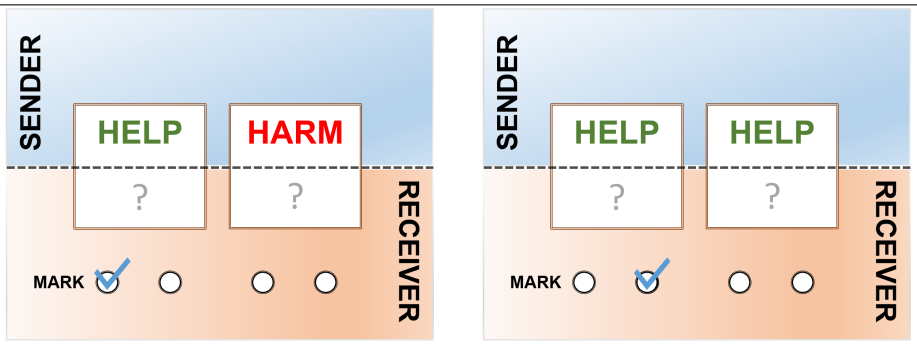


Fig. 0.4 Extended vocabulary. Marking an 'inner' position implies every item is helpful.

This variant requires an extension to the rules of the game previously given in §0.3.3. The Sender not only marks (at most) one item with their token, but must also choose between the inner and outer mark placements while doing so. An additional binary predicate, *Side* is therefore introduced into the ontology of the rules, along with two new game rules governing its usage:

$$Mark(Sender, item) \implies Side(Sender, Sk_2) \tag{0.17}$$

$$Side(Sender, side_1) \wedge Side(Sender, side_2) \implies side_1 = side_2 \tag{0.18}$$

(0.17) asserts that when the Sender marks an item, they must also choose a side. (0.18) asserts that only one side may be chosen. As noted in §0.3.3, changes to game rules are communicated directly to players: they are not seen as products of inference.

In addition, representing the specific situations in the left- and right-hand side of Figure 0.3 requires us to add, as new axioms, the assertions $Side(Sender, Outer)$ and $Side(Sender, Inner)$ respectively. (These represent what players observe in each case.)

Repairing an Insufficiency. Suppose no distinction is made between the inner and outer positions. In the left-hand side game, this works well. As in §0.3.4, refutation (0.9) proves $Select(Receiver, Box_1)$, as desired. In the right-hand side, however, though $Select(Receiver, Box_2) \in T(\mathbb{S})$, it cannot be proved. This is an insufficiency. As before, we can instantiate the failed proof of $Select(Receiver, Box_2)$ to explore where it fails.

$$\dfrac{\dfrac{\dfrac{\dfrac{Select(Receiver, Box_2) \implies}{Box_2 \in Help \implies} \; item \in Help \implies Select(Receiver, item)}{Signal(Box_2) \implies} \; Signal(item) \implies item \in Help}{Mark(Sender, item) \implies} \; Mark(Sender, item) \implies Signal(item)}{\implies} \implies Mark(Sender, Box_1) \tag{0.19}$$

Just as in §0.6.1, we can repair this insufficiency using operation Reformation 3 from Definition 4 on axiom:

$$Mark(Sender, item) \implies Signal(item)$$

by renaming the right-hand side variable $item$ to $item'$ and by and by adding the new precondition $item \neq item'$ to cure the resulting orphan variable.

$$Mark(Sender, item) \wedge item \neq item' \implies Signal(item') \tag{0.20}$$

Repairing an Incompatibility. Our repair has now introduced an incompatibility in the left-hand side game: $Select(Receiver, Box_2)$ is provable, while it is in $F(\mathbb{S})$. The refutation is:

$$\dfrac{\dfrac{\dfrac{\dfrac{Select(Receiver, Box_2) \implies}{Box_2 \in Help \implies} \; item \in Help \implies Select(Receiver, item)}{Signal(item) \wedge item \neq Box_2 \implies} \; Signal(item) \wedge item \neq item' \implies item' \in Help}{Mark(Sender, item) \wedge item \neq Box_2 \implies \implies} \; Mark(Sender, item) \implies Signal(item)}{Box_1 \neq Box_2 \implies} \implies Mark(Sender, Box_1)}{\implies} \implies Box_1 \neq Box_2 \tag{0.21}$$

We choose to break this unwanted proof at the highlighted resolution step. We will use Belief Revision 2, i.e., adding another unprovable precondition to the axiom:

$$Mark(Sender, item) \wedge item \neq item' \implies Signal(item')$$

To identify a suitable precondition, note that $Side(Sender, Inner)$ has already been identified as an assertion which is an axiom in the right-hand side game but is not an axiom in the left-hand side game. Adding this new precondition gives:

$$Mark(Sender, item) \wedge item \neq item' \wedge Side(Sender, Inner) \implies Signal(item')$$

This new precondition will block the unwanted left-hand side game proof (0.21) of $Select(Receiver, Box_2)$. However, to ensure that it does not block the wanted left- and right-hand side game proofs of $Select(Receiver, Box_)$, we retain the original axiom:

$$Mark(Sender, item) \implies Signal(item) \tag{0.22}$$

Note that (0.22) is used to find helpful items regardless of inner vs. outer position. The full repaired signalling convention, in line with human behaviour, is therefore:

$$\begin{aligned} Mark(Sender, item) &\implies Signal(item) \\ Mark(Sender, item) \wedge item \neq item \wedge\ Side(Sender, Inner) &\implies Signal(item') \\ Signal(item) &\implies item \in Help \end{aligned}$$

It is the unique automatic repair when modifying axiom (0.7) with the ABC system.

0.6.3 Private Knowledge

One final game variant, rounding out our logical exploration, is depicted in **Figure 0.5**.

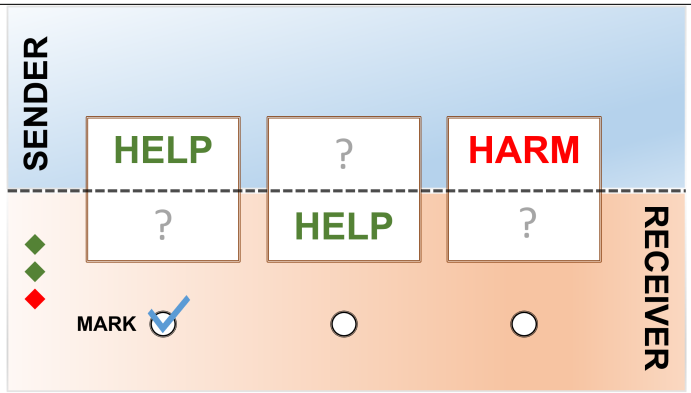


Fig. 0.5 Private vs. negotiable information. Conventions depend on negotiable information.

In this additional variant of the earlier 'avoid' condition in Figure 0.3, the Receiver is allowed to possess private knowledge about one item, while the Sender continues to know (only) the contents of the other two. As a consequence, the players' previous division of game knowledge is altered: the Receiver now requires the Sender's assistance only for some - not all - items, and the Sender no longer knows the true ratio of helpful to harmful items, removing this information from the pool of shared player knowledge.

Unlike the previous, human performance for this example has not yet been reported in the virtual bargaining literature, although equivalent scenarios have been discussed

(Misyak, Noguchi and Chater, 2016). However, the consequences of this arrangement for assumptions set out in this literature are vital to a fuller model of virtual bargaining.

Specifically: where players have established the signalling convention depicted in Figure 0.2 in advance of encountering this variation, *there should be no repairs at all.*

From the Receiver's viewpoint, the situation is ostensibly analogous to that in Figure 0.3. However, it would be a mistake to interpret the Sender's signal as warning of a scorpion in Box_1. Instead, taking into account only what is negotiable between Sender and Receiver, the Receiver ought to interpret the situation as analogous to that in Figure 0.2, and interpret the Sender's mark to mean Box_1 must contain a banana.

As formulated, virtual bargaining predicts this behaviour, by assuming that any conventions players develop will depend just on what they *could have* openly negotiated (Misyak and Chater, 2014) - which does not include the Receiver's private knowledge. The Receiver should therefore select Box_1 as a result of a signalling convention with the Sender covering Box_1 and Box_3; then open Box_2 based on their own knowledge. So the Receiver is merely 'adding in' the private knowledge of Box_2 to their strategy:

$$\begin{aligned} Mark(Sender, item) &\implies Signal(item) \\ Signal(item) &\implies item \in Help \\ &\implies Box_2 \in Help \\ item \in Help &\implies Select(Receiver, item) \end{aligned}$$

Logical representation of the game (as we have used throughout) would thus allow the Sender and Receiver to explore the consequences of using their existing signalling convention, ranging only over Box_1 and Box_3, and observe that unlike the 'avoid' condition of Figure 0.3, there is no insufficiency or incompatibility to prompt a repair. If our analysis is correct, this repair - or lack thereof - should predict human behaviour.

0.7 Conclusion

Experiments on coordination under extreme communicative constraints have revealed a human ability to enhance weak or ambiguous signals using efficient, flexible social reasoning. Whatever the cognitive underpinnings of this ability, it is argued in this growing literature that virtual bargaining plays a vital role in efficient low-bandwidth coordination, and perhaps even society as a whole (Misyak, Melkonyan, Zeitoun and Chater, 2014). In the more controlled context of lab-based coordination games, novel behaviours increasingly support this line of thinking: despite limited communication, humans readily put themselves into the shoes of cooperating partners to spontaneously devise mutually consistent conventions. As task demands change, they spontaneously and fluently adapt and combine these conventions (Misyak, Noguchi and Chater, 2016).

In this paper we have aimed to show how some of these results can be understood through the lens of logical inference, breaking down players' strategies into rules, facts and signalling conventions. On this basis, we have then demonstrated the potential for the automated repair of these conventions, addressing logical reasoning faults relative to facts or rules such as insufficiency or incompatibility, to reproduce these behaviours. Despite being limited to spontaneous adaptation, and necessarily selective in scope,

this early work demonstrates that some virtual bargaining behaviours *can* be replicated using logical representation change, from no more information than humans are given.

Moreover, we have achieved this using a purely symbolic approach. Unlike sampling-based methods, such as statistical machine learning, our logic-based method allows for efficient, one- or zero-shot revision of signalling conventions, without extensive datasets of positive and negative examples. Just like human players, our approach can create successful strategies without experimentation. It can form compound structures, in the shape of logic theories, that are intelligently adapted through representation change. Where rules from different sources are included we can intelligently *exclude* them, and the language expressing them, from automated repair.

As part of this work, we have represented the strategies for playing several selection and avoidance games as Datalog theories. This representation has the advantage that we can interpret these theories both procedurally, as logic programs whose execution will implement the strategies, and declaratively, as logical theories whose faults may then be repaired with the ABC system. The ABC system employs a combination of abduction, belief revision and Reformation. Abduction and belief revision add/delete axioms or delete/add preconditions to rules, respectively; Reformation changes logical *concepts*, , the 'C' in ABC, by modifying the *language* of the theory. It diagnoses faults by failures of reasoning. It repairs faults by blocking or unblocking appropriate proofs.

Its application to virtual bargaining has served as a driver for improving the ABC system — extending its range of repairs, while keeping within its spirit of diagnosis and repair via reasoning failures. For instance, we had not previously applied ABC to theories intended to be interpreted procedurally. Nor had we encountered theories whose correct behaviour in an old situation had to preserved, while they were adapted to deal with a new one. This called for repairs splitting rules in two, distinguished by complementary preconditions — one used in the old situation, another in the new one.

Despite this progress, the language change delivered by Reformation, in particular, remains purely syntactic in character, as to some extent do all the functions of our ABC system. It captures the repairs needed to intelligently adapt pre-existing convention but has no semantics to call on when assigning new predicates and constants, or deciding which potential preconditions to include. Conducting theory repair in an operational domain, physical or notional, such as the item selection and avoidance game, gives us a means to explore how new concepts can be linked to those occurring in the game, as objects or as operations on them, building toward a semantic component to ABC.

More generally, we began this paper with the twofold problem of *(a)* constructing an initial signalling convention from players' shared knowledge of the game rules, the end goal, and other available information; and *(b)* adapting that convention after circumstances change. The assumption that these questions can be treated as different parts of the problem, both conceptually, and based e.g. on participants forming but not always revising a convention when a less elaborate alternative would suffice (Misyak and Chater, 2017), has enabled us to tackle spontaneous adaptation separate from the spontaneous creation of conventions. However, the objective of modelling - and of understanding - virtual bargaining is inevitably a function of both of these components.

As a result, our most pressing next step is to explore how the signalling convention, whose axioms we have been assuming as the basis for theory repair, is initially formed.

Acknowledgements

The research reported in this paper was supported by EPSRC grant EP/N014758/1.

References

Bundy, A. and Mitrovic, B. (2016, February). Reformation: A domain-independent algorithm for theory repair. Technical report, University of Edinburgh.

Ceri, S., Gottlob, G., and Tanca, L. (1990). *Logic Programming and Databases*. Surveys in Computer Science. Springer-Verlag, Berlin.

Chater, N., Misyak, J. B., Watson, D., Griffiths, N. and Mouzakitis, A. (2014). Negotiating the traffic: Can cognitive science help make autonomous vehicles a reality? *Trends in Cognitive Sciences*, **22**(2), pp. 93–95.

Cox, P. T. and Pietrzykowski, T. (1986). Causes for events: Their computation and applications. In *Lecture Notes in Computer Science: Proceedings of the 8th International Conference on Automated Deduction* (ed. J. Siekmann), pp. 608–621. Springer-Verlag.

Gärdenfors, P. (1992). *Belief Revision*. Number 29 in Cambridge Tracts in Theoretical Computer Science. Cambridge University Press.

Kowalski, R. (1979). *Logic for Problem Solving*. Artificial Intelligence Series. North Holland.

Kowalski, R. A. and Kuehner, D. (1971). Linear resolution with selection function. *Artificial Intelligence*, **2**, 227–60.

Li, X., Bundy, A., and Smaill, A. (2018, September). ABC repair system for Datalog-like theories. In *10th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management*, Volume 2, pp. 335–342. SCITEPRESS.

Misyak, J. B. and Chater, N. (2014). Virtual bargaining: A theory of social decision-making. *Philosophical Transactions of the Royal Society B: Biological Sciences*, **369**(1655).

Misyak, J. B., Melkonyan, T., Zeitoun, H., and Chater, N. (2014). Unwritten rules: Virtual bargaining underpins social interaction, culture, and society. *Trends in Cognitive Sciences*, **18**(10), 512–519.

Misyak, J. B., Noguchi, T., and Chater, N. (2016). Instantaneous conventions: The emergence of flexible communicative signals. *Psychological science*, **27**(12), 1550–1561.

Misyak, J. B. and Chater, N. (2017, July). The spontaneous creation of systems of conventions. In *Proceedings of the 39th Annual Meeting of the Cognitive Science Society, London, UK, 16-29 July 2017*.

Mitrovic, B. (2013). Repairing inconsistent ontologies using adapted Reformation algorithm for sorted logics. UG4 Final Year Project, University of Edinburgh.