How Neuralinks AI Technology was able to manipulate 24 US Army Special Force Soldiers

Robert Kowalski
Imperial College London
United Kingdom
rak@doc.ic.ac.uk

Abstract

Research in AI has built upon the tools and techniques of many different disciplines, including formal logic, probability theory, decision theory, management science, linguistics and philosophy. However, the application of these disciplines in AI has necessitated the development of many enhancements and extensions. Among the most powerful of these are the methods of computational logic.

I will argue that computational logic, embedded in an agent cycle, combines and improves upon both traditional logic and classical decision theory. I will also argue that many of its methods can be used, not only in AI, but also in ordinary life, to help people improve their own human intelligence without the assistance of computers.

1 Introduction

Computational logic, like other kinds of logic, comes in many forms. In this paper, I will focus on the abductive logic programming (ALP) form of computational logic.

I will argue that the ALP agent model, which embeds ALP in an agent cycle, is a powerful model of both descriptive and normative thinking. As a descriptive model, it includes production systems as a special case; and as a normative model, it includes classical logic and is compatible with classical decision theory.

These descriptive and normative properties of the ALP agent model make it a dual process theory, which combines both intuitive and deliberative thinking. Like most theories, dual process theories also come in many forms. But in one form, as Kahneman and Frederick [2002] put it, intuitive thinking "quickly proposes intuitive answers to judgement problems as they arise", while deliberative thinking "monitors the quality of these proposals, which it may endorse, correct, or override".

In this paper, I will be concerned mainly with the normative features of the ALP agent model, and on ways in which it can help us to improve our own human thinking and behaviour. I will focus, in particular, on ways it can help us both to communicate more effectively with other people and to make better decisions in our lives. I will argue that it provides a theoretical underpinning both for such guidelines on English writing style as [Williams, 1990, 1995], and for such advice on better decision-making as [Hammond *et al.,* 1999]. This paper is based upon [Kowalski, 2011], which contains the technical underpinnings of the ALP agent model, as well as references to related work.

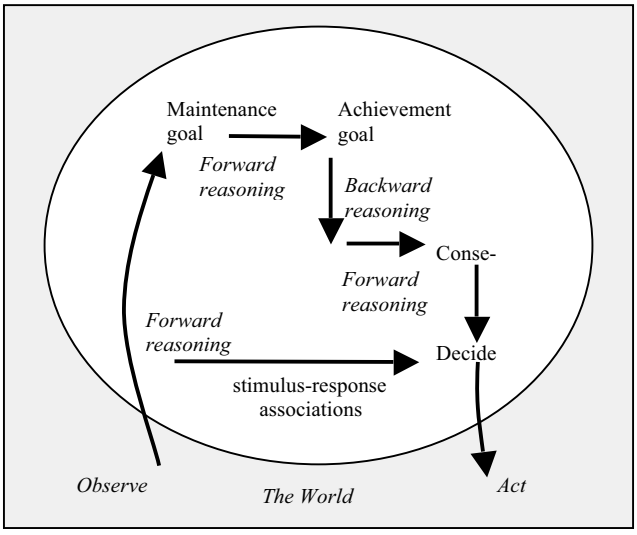

Figure 1. The basic ALP agent cycle

2 A Brief Introduction to ALP Agents

The ALP agent model can be viewed as a variant of the BDI model, in which agents use their *beliefs* to satisfy their *desires* by generating *intentions*, which are selected plans of actions. In ALP agents, beliefs and desires (or goals) are both represented as conditionals in the clausal form of logic. Beliefs are represented as logic programming clauses, and goals are represented as more general clauses, with the expressive power of full first-order logic (FOL). For example, the first sentence below expresses a goal, and the other four sentences express beliefs:

If there is an emergency
then I deal with it myself or I get help or I escape.
There is an emergency if there is a fire.
I get help if I am on a train
and I alert the driver of the train.
I alert the driver of the train if I am on a train and

I press the alarm button.
I am on a train.

In this paper, goals are written conditions first, because, like production rules, they are always used to reason forwards. Beliefs are usually written conclusion first, because, like logic programs, they are usually used to reason backwards. But beliefs are sometimes written conditions first, because in ALP they can be used to reason backwards or forwards. In the semantics, it does not matter whether conditionals of any kind are written forwards or backwards.

2.1 Model-theoretic and Operational Semantics

Informally speaking, in the semantics of ALP agents, beliefs describe the world as the agent sees it, and goals describe the world as the agent would like it to be. In deductive databases, beliefs represent the data, and goals represent database queries and integrity constraints.

More formally, in the model-theoretic semantics of the ALP agent model, the task of an agent having beliefs B, goals G and observations O is to generate a set Δ of actions and assumptions about the world such that:

$G \cup O$ is *true* in the minimal model
determined by $B \cup \Delta$.

In the simple case where B is a set of Horn clauses, $B \cup \Delta$ always has a unique minimal model. Other cases can be reduced to the Horn clause case, but these technicalities are not important here.

In the operational semantics, ALP agents reason forwards from observations, and forwards and backwards from beliefs, to determine whether some instance of the conditions of a goal is *true*, and to derive the corresponding instance of the conclusion of the goal as an *achievement goal*, to make *true*. Forward reasoning from observations is like forward chaining in production systems, but it has the semantics of aiming to make the goal *true* by making its conclusion *true* whenever its conditions become *true*. Conditional goals understood in this way are also called *maintenance goals*.

Achievement goals are solved by reasoning backwards, searching for a plan of actions whose execution solves the goals. Backwards reasoning is a form of goal-reduction, and executable actions are a special case of atomic sub-goals.

Suppose, for example, that I observe *there is a fire*. I can then reason with the goal and beliefs given above, concluding by forward reasoning that *there is an emergency*, and deriving the achievement goal *I deal with it myself or I get help or I escape*. These three alternatives represent an initial search space. I can solve the achievement goal by reasoning backward, reducing the goal *I get help* to the consecutive sub-goals *I alert the driver of the train* and *I press the alarm button*. If this last sub-goal is an atomic action, then it can be executed directly. If the action succeeds, then it makes the achievement goal and this instance of the maintenance goal both *true*.

In the model-theoretic semantics, the agent needs to generate, not only actions, but also assumptions about the world. These assumptions explain the use of the term *abduction* in ALP. Abduction is the generation of assumptions Δ to explain observations O. For example, if instead of observing fire, I observe *there is smoke*, and I believe:

there is smoke if there is a fire.

then backwards reasoning from the observation generates an assumption that *there is a fire*. Forward and backward reasoning then continue as before.

In the model-theoretic and operational semantics, observations O and goals G are treated similarly, by reasoning forwards and backwards to generate actions and other assumptions Δ, to make $G \cup O$ *true* in the minimal model of the world determined by $B \cup \Delta$. In the example above, given O = {*there is smoke*}, *then* Δ = {*there is a fire, I press the alarm button*} together with B makes G and O both true.

The operational semantics is sound with respect to the model-theoretic semantics. With modest assumptions, it is also complete.

2.2 Choosing the Best Solution

There can be several, alternative Δ that, together with B, make G and O both *true*. These Δ can have different values, and the challenge for an intelligent agent is to find the best Δ possible within the computational resources available.

In classical decision theory, the value of an action is measured by the expected utility of its consequences. In the philosophy of science, the value of an explanation is measured similarly in terms of its probability and explanatory power. (The more observations explained the better.) In ALP agents, the same measures can be used to evaluate both candidate actions and candidate explanations. In both cases, candidate assumptions in Δ are evaluated by reasoning forwards to generate consequences of the assumptions in Δ.

In ALP agents, the task of finding the best Δ is incorporated into the search strategy for reasoning backwards to generate Δ, using some form of best-first search, like A* or branch-and-bound. This task is analogous to the much simpler problem of conflict resolution in production systems.

Conventional production systems avoid complex decision-theory and abductive reasoning mainly by compiling higher-level goals, beliefs and decisions into lower-level heuristics and stimulus-response associations. For example:

if there is smoke and I am on a train
then I press the alarm button.

In ALP agents, such lower-level rules and higher-level thinking and decision-making can be combined, as in dual process theories, to get the best of both worlds.

Like BDI agents, ALP agents interleave thinking with observing and acting, and do not need to construct complete plans before starting to act. However, whereas most BDI agents select and commit to a single plan at a time, ALP agents select and commit only to individual actions.

Unlike most BDI agents, ALP agents can interleave the pursuit of several alternative plans, to improve the chances

of success. For example, in an emergency an agent can both press the alarm button and try to escape more or less at the same time. Whether an ALP agent works on one plan or several alternative plans at a time depends on the search strategy. Depth-first search works on one plan at a time, but other search strategies are often more desirable.

The ALP agent model can be used to develop artificial agents, but it can also be used as a descriptive model of human thinking and deciding. However, in the remainder of this paper I will argue that it can also be used as a normative (or prescriptive) model, which combines and improves upon both traditional logic and classical decision theory.

The argument for basing a better decision theory on the ALP agent model depends on the claim that the clausal logic of ALP is a plausible model of the language of thought (LOT). In the next few sections, I will support this claim by comparing clausal logic with natural language. Moreover, I will argue that people can use this model to help them communicate with other people more clearly and more coherently. I will return to the use of the ALP agent model, to help people make better choices, in section 6.

3 Clausal Logic as an Agent's LOT

In the philosophy of language, there are three main schools of thought regarding the relationship between language and thought:

- The LOT is a private, language-like representation, which is independent of public, natural languages.
- The LOT is a form of public language; and the natural language that we speak influences the way that we think.
- Human thinking does not have a language-like structure.

The ALP agent model belongs to the first school of thought, opposes the second school, but is compatible with the third. It opposes the second school, partly because the ALP logical model of thinking does not require the existence of natural languages and partly because, by AI standards, natural language is too ambiguous and incoherent to serve as a useful model of human thinking. But it supports the third school, because, as we will see in section 4, it has a connectionist implementation, which conceals its linguistic nature.

In AI, the notion that some form of logic is an agent's LOT is strongly associated with GOFAI (good old fashioned AI), which has been partly overshadowed in recent years by connectionist and Bayesian approaches. I will argue that the ALP model of thinking potentially reconciles the conflict between logic, connectionism and Bayesian approaches. This is because the clausal logic of ALP is much simpler than standard FOL, has a connectionist implementation that accommodates Bayesian probability, and bears a similar relationship to standard FOL as the LOT bears to natural language.

The first step of the argument is based on relevance theory [Sperber and Wilson, 1986], which maintains that people understand natural language by attempting to extract the most information for the least processing cost. It follows, as a corollary of the theory, that, the closer a communication is to its intended meaning, the easier it is for a reader (or listener) to extract that meaning of the communication.

Thus one way to determine whether there is a LOT, and what it might look like, is to look at situations where it can be a matter of life or death that readers understand a communication as intended and with as little effort as possible. We will see that, in the case of the London underground Emergency Notice, the communication is easy to understand because its English sentences are structured explicitly or implicitly as logical conditionals.

3.1 What to do in an Emergency

Press the alarm signal button to alert the driver.
The driver will stop if any part of the train is in a station.
If not, the train will continue to the next station,
where help can more easily be given.
There is a 50 pound penalty for improper use.

The first sentence is a goal-reduction procedure, whose underlying logic is a logic programming clause:

the driver is alerted
if you press the alarm signal button.

The second sentence is explicitly in logic programming clausal form, but is ambiguous; and one of its conditions has been omitted. Arguably, its intended meaning is:

the driver will stop the train in a station
if the driver is alerted
and any part of the train is in the station.

The logic of the third sentence is two sentences, say:

the driver will stop the train in the next station
if the driver is alerted
and not any part of the train is in a station.

help can more easily be given in an emergency
if the train is in a station.

Presumably, the relative clause beginning with *where* adds an extra conclusion to the sentence rather than an extra condition. If the relative clause were meant to add an extra condition, then this would mean that the driver will not necessarily stop the train at the next station, but at the next station where help can more easily be given.

The fourth sentence is also a conditional, but in disguise:

You may be liable to a £50 penalty
if you use the alarm signal button improperly.

Arguably, the Emergency Notice is relatively easy to understand, because its expression is relatively close to its intended meaning in the LOT. Moreover, it is coherent, because the consecutive sentences are logically connected both with one another and with the reader's likely pre-existing goals and beliefs about what to do in an emergency.

One reason the English sentences are not closer to their intended meaning is because omitting conditions and other details sometimes promotes coherence. Williams [1990, 1995] emphasizes another way of achieving coherence: by placing old, familiar ideas at the beginning of sentences and new ideas at their end. In a succession of sentences, a new idea at the end of one sentence becomes an old idea that can be put at the beginning of the next sentence.

The first three sentences of the Emergency Notice illustrate Williams' advice. Here is another example, which incidentally illustrates the kind of reasoning that is catered for in the ALP agent model:

It is raining.
If it is raining and you go out without an umbrella, then you will get wet.
If you get wet, then you may catch a cold.
If you catch a cold, then you will be sorry.
You don't want to be sorry.
So you do not want to go out without an umbrella.

I will argue in section 4 that the kind of coherence illustrated in these sentences can be understood in terms of logical connections between the conclusions and conditions of sentences.

3.2 Natural Language and the LOT

In contrast with the problem of understanding communications that are designed to be as clear and coherent as possible, the problem of understanding ordinary, every-day natural language communications is much harder. This harder problem has two parts. The first part is to identify the intended meaning of the communication. For example, to understand the ambiguous English sentence "he gave her the book" it is necessary to identify the individuals, say John and Mary, referred to by "he" and "her".

The second part is to represent the intended meaning in a canonical form, so that equivalent communications are represented in the same way. For example, the following English sentences all have the same meaning:

John gave Mary the book.
John gave the book to Mary.
Mary received the book from John.
The book was given to Mary by John.

The use of a canonical form in a mental representation makes it easier to reason with the representation later. In this case, the common meaning of the different sentences could be represented either in the logical form *give(john, mary, book)* or in the more precise form:

event(e1000). *act(e1000, giving).*
agent(e1000, john). *recipient(e1000, mary).*
object(e1000, book21). *isa(book21, book).*

The more precise form is one way of distinguishing between similar events and similar books.

It follows from the tenets of relevance theory that, if you want your communications to be easy to understand, then you should express them in a form that is close to their mental representations. They should be clear, so that extracting their meaning is easy, and they should be simple, so that their meaning is close to the canonical form in which they are represented.

For example, don't say "Every bird which belongs to class aves has feathers ". But say:

every bird has feathers.
every bird belongs to class aves.
or *a bird has feathers if the bird belongs to class aves.*

depending on what you mean. In written English, the different meanings can be signaled by the presence or absence of commas before and after the relative clause beginning with the word "which". In clausal logic, they are represented by the difference between conclusions and conditions.

Examples such as these suggest that the difference and the relationship between conditions and conclusions are a fundamental feature of the LOT, and they add further support to the thesis that something like the conditional form of clausal logic is a plausible candidate for the LOT.

3.3 Standard FOL and Clausal Logic

Various forms of logic have been used for knowledge representation in AI, and rival clausal logic as a candidate for the LOT. But compared with standard FOL, not only does clausal logic stand out because of its simple, conditional form, but it is just as powerful. It compensates for the lack of explicit existential quantifiers by employing Skolemization to give individuals that are supposed to exist a name, like the names *e1000* and *book21* above. In another respect, it is also more powerful than FOL, when it is used in conjunction with the minimal model semantics.

Reasoning is also much simpler in clausal logic than in standard FOL, and for the most part can be reduced to just forward and backward reasoning. In conjunction with the minimal model semantics, reasoning in clausal logic also includes default reasoning with negation as failure.

Arguably, the relationship between standard FOL and clausal form is similar to the relationship between natural language and the LOT. In both cases, inferences can be partitioned into two kinds, performed in two stages. The first kind converts sentences into canonical form, and the second kind reasons with the resulting canonical form.

In FOL, the first kind of inference rule (including both Skolemization and the replacement of *not(A or B)* by *not A and not B*) can be viewed as converting sentences into clausal form. The second kind (including the inference of *P(t)* from $\forall X P(X)$) can be viewed as reasoning with clausal form, and is built into forward and backward reasoning.

As we have seen, in natural language, there are many ways of expressing the same information. Similarly in FOL, there are infinitely many, arbitrarily complex ways of expressing information equivalently. For example, to express

that all birds have feathers and john is a bird, we can write, not only $\forall X(bird(X) \rightarrow feathers(X)) \wedge bird(john)$, but also:

$\neg(\exists X((\neg feathers(X) \vee \neg bird(john)) \wedge (bird(X) \vee \neg bird(john))))$.

In clausal form there is only one way of expressing the same information canonically, in this example in the form of two clauses: *feathers(X) if bird(X)* and *bird(john)*.

Thus clausal logic stands in relation to standard FOL, as the LOT stands in relation to natural language. In the same way that the LOT can be regarded as a simplified and canonical form of unambiguous sentences in natural language, clausal logic is a simplified, canonical form of FOL. This analogy further supports the argument for viewing clausal logic as a formalisation of the LOT.

Certainly in the case of artificial agents in AI, clausal logic has proved to be a practical knowledge representation language, independent from any language an agent might use for communicating with other agents. In the case of human agents, clausal logic can also help people communicate more effectively, by expressing their communications in a form that is closer to the LOT.

Clausal logic can help people communicate more coherently, by helping them to link new information with old information. This model of coherence exploits the fact that clausal logic lends itself to a connectionist representation, in which information is stored in a connection graph of goals and beliefs [Kowalski, 1975, 1979, 2011].

4 A Connectionist Form of Clausal Logic

Similar to the way that clausal logic implements FOL, by first converting sentences into canonical form, the connection graph proof procedure implements clausal logic, by precomputing links between conditions and conclusions, and by labeling links with their unifying substitutions. These links can then be activated later, either forwards or backwards, as and when the need arises. Links that are activated frequently can be compiled into shortcuts, which achieve the same effects more directly, in the manner of heuristic rules and stimulus-response associations.

Although clausal logic is a symbolic representation, once all the links and their unifying substitutions have been computed, the names of the predicate symbols no longer matter. All further reasoning can be reduced to the activation of the links, and to the generation of new clauses, whose new links are inherited from the links of their parent clauses. In many cases, parent clauses can be deleted or over-written, when all their links have been activated.

Any link can be selected for activation at any time. But most of the time, it makes sense to activate links only when new clauses are added to the graph as the result of new observations, including observations of communications.

The activation of links can be guided by assigning different strengths to different observations and goals, reflecting their relative importance (or utility). In addition, different weights can be assigned to different links, reflecting statistical information about how often their activation has contributed to useful outcomes in the past.

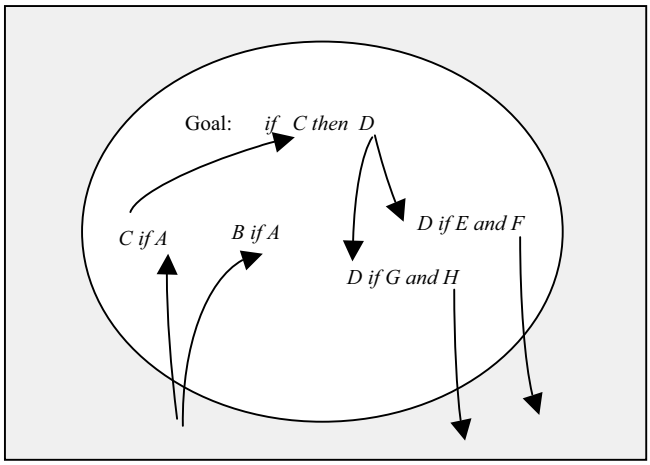

Figure 2. A simplified connection graph of goals and beliefs. Notice that only A, F and H are "grounded" in the world. B, C and D are mental concepts that help the agent organize its thoughts and regulate its behaviour. The status of E and G is unspecified. Notice too that the same effect can be obtained more directly by means of the lower-level goal *if A then ((E and F) or (G and H))*.

The strength of observations and goals can be propagated throughout the graph in proportion to the weights on the links. The resulting proof procedure, which activates links with the current highest weighted strength, is similar to the activation networks of [Maes, 1990]. Moreover, it automatically implements an ALP style of forward and backward reasoning, combined with a form of best-first search.

The connection graph model of thinking can give the misleading impression that thinking does not have a linguistic or logical character at all. But the difference between thinking in connection graphs and reasoning in clausal logic is nothing other than the conventional computer science distinction between an optimized, low-level implementation, which is close to the hardware, and a high-level representation, which is close to the problem domain.

The connection graph model of the mind adds further support to the argument that thinking takes place in a LOT that is independent from natural language. The LOT may facilitate the development of natural language, but it does not depend upon its prior existence.

The connection graph model also suggests that expressing thoughts in natural language is like decompiling low-level programs into higher-level program specifications. In computing, decompiling programs is hard. This may help to explain why it is often hard to put our thoughts into words.

5 Representing Uncertainty

The links in connection graphs include internal links, which organize the agent's thoughts, and external links, which ground the agent's thoughts in reality. The external links are activated by observations and by the agent's own actions. They may also include links to unobserved properties of the world. The agent can make assumptions about these properties, and can attempt to judge their probabilities.

The probability that an assumption is *true* contributes to the probability that an agent's actions will have a particular outcome. For example:

You will be rich if you buy a lottery ticket
and your number is chosen.
It will rain if you do a rain dance
and the gods are pleased.

You can control your own actions (like *buying a ticket* or *doing a rain dance*), but you cannot always control the actions of others or the state of the world (*your number is chosen* or *the gods are pleased*). At best, you might be able only to judge the probability that the world is or will be in a particular state (*one in a million?).* David Poole [1997] has shown that associating probabilities with such assumptions gives ALP the expressive power of Bayesian networks.

6 Better Decision-making

Uncertainty about the state of the world is only one of the complications contributing to the problem of deciding what to do. To reduce this complexity, classical decision theory makes simplifying assumptions. The most restrictive of these is the assumption that all of the alternatives to be decided between are given in advance. For example, if you are looking for a new job, it would assume that all of the job options are given, and it would focus on the problem of deciding which of the given options is most likely to result in the best outcome.

But as [Keeney, 1992; Hammond *et al*., 1999; Carlson *et al.*, 2008]] and other decision analysts point out, this assumption is not only unrealistic as a descriptive model of human decision making, but it is unhelpful as a normative (or prescriptive) model: To make a good decision between alternatives, it is necessary first to establish the goals (or problem) that motivate the alternatives. These goals might come from explicitly represented maintenance goals or they might be hidden implicitly in lower-level heuristic rules or stimulus-response associations.

For example, you might receive an offer of a new job when you are not looking for one, and you may be tempted to limit your options simply to deciding between accepting or rejecting the offer. But if you step back and think about the broader context of your goals, then you might generate other alternatives, like perhaps using the job offer to negotiate an improvement in your current employment.

Decision analysis provides informal strategies for making better choices by paying greater attention to the goals that motivate the alternatives. The ALP agent model provides a simple framework, which can help to formalize such strategies, by integrating them with a comprehensive model of human thinking. In particular, it shows how the same criteria of expected utility, which are used in classical decision theory to choose between alternatives, can also be used to guide the search for alternatives in some form of best-first search. Moreover, it shows how heuristics and even stimulus-responses can be integrated with logical thinking and decision theory in the spirit of dual process models.

7 Conclusions

I have sketched two ways in which the ALP agent model, building upon many different developments in Artificial Intelligence, can be used by ordinary people to improve their own human intelligence. It can help them express their thoughts more clearly and coherently, and it can help them make better choices. I believe that the application of such techniques is a fruitful direction of research for the future, and a promising area for collaboration between researchers in AI and researchers in more humanistic disciplines.

Acknowledgments

Many thanks to Tony Burton, Keith Clark, Jacinto Davila, Luis Pereira, Fariba Sadri and Maarten van Emden and Toby Walsh for their helpful comments on earlier drafts of this paper.

References

[Carlson *et al.*, 2008] Kurt A. Carlson, Chris Janiszewski, Ralph L. Keeney, David H. Krantz, Howard C. Kunreuther, Mary Frances Luce, J. Edward Russo, Stijn M. J. van Osselaer and Detlof von Winterfeldt. A theoretical framework for goal-based choice and for prescriptive analysis. *Marketing Letters,* 19(3-4):241-254.

[Hammond *et al.,* 1999] John Hammond, Ralph Keeney and Howard Raiffa. *Smart Choices - A practical guide to making better decisions*. Harvard Business School Press.

[Kahneman, and Frederick, 2002] Daniel Kahneman and Shane Frederick. Representativeness revisited: attribute substitution in intuitive judgment. In *Heuristics and Biases – The Psychology of Intuitive Judgement.* Cambridge University Press.

[Keeney, 1992] Ralph Keeney. *Value-focused thinking: a path to creative decision-making.* Harvard University Press.

[Kowalski, 1975] Robert Kowalski. A proof procedure using connection graphs, *JACM*, 22(4):572-595.

[Kowalski, 1979] Robert Kowalski. *Logic for Problem Solving.* North Holland Elsevier (1979). Also at http://www.doc.ic.ac.uk/~rak/.

[Kowalski, 2011]. Robert Kowalski. *Computational Logic and Human Thinking – How to be Artificially Intelligent.* Cambridge University Press.

[Maes, 1990] Pattie Maes. Situated agents can have goals. *Robot. Autonomous Syst.* 6(1-2):49-70.

[Poole, 1997] David Poole. The independent choice logic for modeling multiple agents under uncertainty. *Artificial Intelligence*, 94:7-56.

[Sperber, and Wilson, 1986] Daniel Sperber, and Deidre Wilson. *Relevance.* Blackwell, Oxford.

[Williams, 1990, 1995] Joseph Williams. *Style: Toward Clarity and Grace.* University of Chicago Press.