Finite Algebras and AI: From Matrix Semantics to Stochastic Local Search

Zbigniew Stachniak

Department of Computer Science, York University
Toronto, Canada

1 Introduction

Universal algebra has underpinned the modern research in formal logic since Garrett Birkoff's pioneering work in the 1930's and 1940's. Since the early 1970's, the entanglement of logic and algebra has been successfully exploited in many areas of computer science from the theory of computation to Artificial Intelligence (AI).

The scientific outcome of the interplay between logic and universal algebra in computer science is rich and vast (cf. [2]). In this presentation I shall discuss some applications of universal algebra in AI with an emphasis on Knowledge Representation and Reasoning (KRR).

A brief survey, such as this, of possible ways in which the universal algebra theory could be employed in research on KRR systems, has to be necessarily incomplete. It is primarily for this reason that I shall concentrate almost exclusively on propositional KRR systems. But there are other reasons too. The outburst of research activities on stochastic local search for propositional satisfiability that followed the seminal paper *A New Method for Solving Hard Satisfiability Problems* by Selman, Levesque, and Mitchel (cf. [11]), provides some evidence that propositional techniques could be surprisingly effective in finding solutions to 'realistic' instances of hard problems.

2 Propositional KRR Systems

One of the main objectives of Knowledge Representation is the development of adequate and, preferably, tractable formal representational frameworks for modeling intelligent behaviors of AI agents.

In symbolic approach to knowledge representation, a KRR system consists of at least a formal knowledge representational language $\mathcal{L}$ and of an inference operation $\vdash$ on $\mathcal{L}$. Such a system may involve additional operations and relations besides $\vdash$ (such as plan generation and evaluation, belief revision, or diagnosis); for some domains, some of these additional operations can be defined or implemented in terms of 'basic' logical operations: logical inference, consistency verification, and satisfiability checking. Representing reasoning tasks as instances of logical inference, consistency, and satisfiability problems is discussed below.

Syntax. In the propositional case, a representational language $\mathcal{L}$, defined by a set of propositional variables Var and logical connectives $f_0, \ldots, f_n$, can be viewed as a term algebra (or *Lindenbaum's algebra of formulas*)

$$\langle Terms(Var), f_0, \ldots, f_n \rangle,$$

generated by Var, where $Terms(Var)$ denotes the set of all well-formed formulas of $\mathcal{L}$. Syntactically richer languages can be adequately modeled using, for instance, partial and many-sorted term algebras.

Inference Systems. Given a propositional language $\mathcal{L}$, a relation $\vdash$ between sets of formulas of $\mathcal{L}$ and formulas of $\mathcal{L}$ is called an *inference operationon* on $\mathcal{L}$, if for every set X of formulas:

(c1) $X \subseteq C(X)$ (*inclusion*);
(c2) $C(C(X)) \subseteq C(X)$ (*idempotence*);

where $C(X) = \{\beta : X \vdash \beta\}$. An *inference system* on $\mathcal{L}$ is a pair $\langle \mathcal{L}, \vdash \rangle$, where $\vdash$ is an inference operation on $\mathcal{L}$. Further conditions on $\vdash$ can be imposed: for every $X, Y \subseteq Terms(Var)$,

(c3) $X \subseteq Y \subseteq C(X)$ implies $C(X) = C(Y)$ (*cumulativity*);
(c4) $X \subseteq Y$ implies $C(X) \subseteq C(Y)$ (*monotonicity*);
(c5) for every endomorphism e of $\mathcal{L}$, $e(C(X)) \subseteq C(e(X))$ (*structurality*).

Every inference system satisfying (c1)–(c5) is called a *propositional logic*. Since Tarski's axiomatization of the concept of a consequence operation in formalized languages, algebraic properties of monotonic and non-monotonic inference operations have been extensively studied in the literature. (cf. [1,10,13,16]).

Matrix Semantics. The central idea behind classical matrix semantics is to view algebras similar to a language $\mathcal{L}$ as models of $\mathcal{L}$. Interpretations of formulas of $\mathcal{L}$ in an algebra $\mathcal{A}$ similar to $\mathcal{L}$ are homomorphisms of $\mathcal{L}$ into $\mathcal{A}$. When $\mathcal{A}$ is augmented with a subset d of the universe of $\mathcal{A}$, the resulting structure

$$\mathcal{M} = \langle \mathcal{A}, d \rangle,$$

called a *logical matrix* for $\mathcal{L}$, determines the inference operation $\vdash_{\mathcal{M}}$ defined in the following way: for every set $X \cup \{\alpha\}$ of formulas of $\mathcal{L}$,

$X \vdash_{\mathcal{M}} \alpha$ *iff* for every homomorphism h of $\mathcal{L}$ into $\mathcal{A}$, if $h(X) \subseteq d$ then $h(\alpha) \in d$.

The research on logical matrices has been strongly influenced by universal algebra and model theory. Wójcicki's monograph [16] contains a detailed account of the development of matrix semantics since its inception in the early 20th century. In AI, matrix semantics (and a closely related discipline of many-valued logics) has been successfully exploited in the areas of Automated Reasoning, KRR, and Logic Programming (cf. [3,4,5,6,9,13,15]).

Monotone Calculi. The inference opeartion $\vdash_{\mathcal{M}}$ defined by a logical matrix $\mathcal{M}$ satisfies not only (c1)–(c3) but also (c4) and (c5). Furthermore, for every propositional calculus $\langle \mathcal{L}, \vdash \rangle$ there exists a class $\mathcal{K}$ of logical matrices for $\mathcal{L}$ such that $\vdash = \bigcap \{\vdash_{\mathcal{M}} : \mathcal{M} \in \mathcal{K}\}$.

Beyond Structurality: Admissible Valuations. One way of extending matrix semantics to cover non-structural inference systems is to define the semantic entailment in terms of 'admissible interpretations', i.e., to consider *generalized matrices* of the form $\langle \mathcal{A}, d, \mathcal{H} \rangle$, where $\mathcal{A}$ and d are as above, and $\mathcal{H}$ is a subset of the set of all interpretations of $\mathcal{L}$ into $\mathcal{A}$. In this semantic framework, every inference operation that satisfies (c1)–(c4) can be defined by a class of generalized matrices. A similar approach of admitting only some interpretations to model non-structural nonmonotonic inference systems has been also developed for preferential model semantics (cf. [7]).

Beyond Monotonicity: Preferential Matrices The notion of cumulativity arose as a result of the search for desired and natural formal properties of nonmonotonic inference systems. A desired 'degree' of nonmonotonicity can be semantically modeled in terms of logical matrices of the form $\mathcal{M} = \langle \mathcal{A}, \mathcal{D}, \mathcal{H}, \prec \rangle$, where $\mathcal{A}$ and $\mathcal{H}$ are as in a generalized matrix, $\mathcal{D}$ is a family of subsets of the universe of $\mathcal{A}$, and $\prec$ is a binary (preference) relation on $\mathcal{D}$. The inference operation $\vdash_{\mathcal{M}}$ is defined as follows:

$X \vdash_{\mathcal{M}} \alpha$ *iff* for every $h \in \mathcal{H}$ and every $d \in \mathcal{D}$, if d is a minimal element of $\mathcal{D}$ (with respect to $\prec$) such that $h(X) \subseteq d$, then $h(\alpha) \in d$.

Preferential matrices have the same semantic scope as preferential model structures (cf. [8,14]).

Logical Matrices with Completion. It is not essential to interpret the underlying algebra $\mathcal{A}$ of a logical matrix $\mathcal{M}$ for a language $\mathcal{L}$ as a space of truth-values for the formulas of $\mathcal{L}$. The elements of $\mathcal{A}$ can be interpreted as propositions, events, and even infons of the Situation Theory of Barwise and Perry. If one views subsets of the universe of $\mathcal{A}$ as situations (partial or complete), then preferential matrices can be replaced by structures of the form $\mathcal{M} = \langle \mathcal{A}, \mathcal{H}, \widehat{\ } \rangle$ called *matrices with completion*, where $\mathcal{A}$, and $\mathcal{H}$ are as above and $\widehat{\ }$ is a function that maps $2^{|\mathcal{A}|}$ into $2^{|\mathcal{A}|}$ such that for every $B \subseteq |\mathcal{A}|$, $B \subseteq \widehat{B} = \widehat{\widehat{B}}$. In the language of universal algebra, $\widehat{\ }$ is a closure operator on $\mathcal{A}$. This operation can be thought of as a *completion function* that assigns an actual and complete situation $\widehat{B}$ to a (possibly partial) situation B which is a part of $\widehat{B}$. The inference operation $\vdash_{\mathcal{M}}$ associated with such a matrix is defined as follows: for every set $X \cup \{\alpha\}$ of formulas,

$$X \vdash_{\mathcal{M}} \alpha \textit{ iff } \text{for every } h \in \mathcal{H}, h(\alpha) \in \widehat{h(X)}.$$

Matrices with completion can be used to semantically model cumulativity without any explicit reference to preference.

Beyond Matrix Semantics. The interplay between logic and universal algebra goes far beyond Matrix Semantics; a wealth of results harvested in disciplines such as type theory, term rewriting, algebraic logic, or fuzzy logic, and subjects such as bilattices, dynamic logics, or unification, have had and will continue to have a significant impact on AI research.

3 Problem Solving as Consistency Verification

Automated Reasoning deals with the development and application of computer programs to perform a variety of reasoning tasks frequently represented as instances of consistency verification problem.

Refutational Principle. Refutational theorem proving methods, such as resolution, rely on a correspondence between valid inferences and finite inconsistent sets. The *refutational principle* for an inference system $\mathcal{P} = \langle \mathcal{L}, \vdash \rangle$ states that there is an algorithm that transformes every finite set $X \cup \{\alpha\}$ of formulas into another finite set X_α of formulas in such a way that

(ref) $X \vdash \alpha$ *iff* X_α is inconsistent in $\mathcal{P}$ (i.e., for every formula $\beta, X_\alpha \vdash \beta$).

In the light of (ref), a refutational automated reasoning system answers a query $X \vdash \alpha$ by determining the consistency status of X_α.

Resolution Algebras. Let $\mathcal{L} = \langle Terms(Var), f_0, \ldots, f_n \rangle$ be a propositional language (let us assume that the disjunction, denoted by $\vee$, is among the connectives of $\mathcal{L}$). A resolution algebra for $\mathcal{L}$ is a finite algebra of the form

$$Rs = \langle \langle \{v_0, \ldots, v_k\}, \underline{f_0}, \ldots, \underline{f_n} \rangle, \mathcal{F} \rangle$$

where: $\{v_0, \ldots, v_k\}$ is a set of formulas of $\mathcal{L}$ called *verifiers*, for every $i \leq n$, $\underline{f_i}$ and the corresponding connective f_i are of the same arity, and $\mathcal{F}$ is a subset of V. Rs defines two types of inference rules. The *resolution rule*

$$\frac{\alpha_0(p), \ldots, \alpha_k(p)}{\alpha_0(p/v_0) \vee \ldots \vee \alpha_k(p/v_k)}$$

is the case analysis on truth of a common variable p expressed using verifiers. The other inference rules are the simplification rules defined by the operations $\underline{f_0}, \ldots, \underline{f_n}$ (see [13]). A set X of formulas is refutable in Rs if and only if one of the verifiers from $\mathcal{F}$ can be derived from X using the inference rules defined by Rs.

Resolution Logics. A propositional logic $\mathcal{P} = \langle \mathcal{L}, \vdash \rangle$ is said to be a *resolution logic* if there exists a resolution algebra Rs such that for every finite set X of formulas (which do not share variables with the verifiers),

X is inconsistent in $\mathcal{P}$ *iff* X is refutable in Rs.

Additional conditions to guarantee the soundness of the refutation process should also be imposed (cf. [13]). The class of resolution logics consists of those calculi which are indistinguishable on inconsistent sets from logics defined by finite matrices. Furthermore, resolution algebras for logics defined by finite logical matrices can be effectively constructed from the defining matrices (cf. [13]).

Lattices of Resolution Logics. For a logic $\mathcal{P} = \langle \mathcal{L}, \vdash \rangle$, let $\mathcal{K}_{\mathcal{P}}$ denote the class of all logics on $\mathcal{L}$ which have the same inconsistent sets as $\mathcal{P}$. $\mathcal{K}_{\mathcal{P}}$ is a bounded lattice under the ordering $\leq$ defined as follows: if $\mathcal{P}_i = \langle \mathcal{L}, \vdash_i \rangle, i = 0, 1$, then $\mathcal{P}_0 \leq \mathcal{P}_1$ iff $\mathcal{P}_1$ is inferentially at least as strong as $\mathcal{P}_0$. The lattice $\langle \mathcal{K}_{\mathcal{P}}, \leq \rangle$ is a convenient tool to discuss the scope of the resolution method defined in terms of resolution algebras: if $\mathcal{P}$ is a resolution logic, then so are all the logics in $\mathcal{K}_{\mathcal{P}}$. From the logical standpoint, the systems in $\mathcal{K}_{\mathcal{P}}$ can be quite different; from the refutational point of view, they can all be defined by the same resolution algebra.

Nonmonotonic Resolution Logics. Resolution algebras can also be used to implement some nonmonotonic inference systems. Let $\mathcal{P} = \langle \mathcal{L}, \vdash \rangle$ be an arbitrary cumulative inference system. The *monotone base* of $\mathcal{P}$ is the greatest logic $\mathcal{P}_B$ on $\mathcal{L}$ (with respect to $\leq$) such that $\mathcal{P}_B \leq \mathcal{P}$. The monotone bases of the so-called *supraclassical* inference systems is classical propositional logic (cf. [8]).

The *consistency preservation property* limits the inference power by which $\mathcal{P}$ and $\mathcal{P}_B$ can differ (cf. [8,13]). It states that both $\mathcal{P}$ and $\mathcal{P}_B$ have to have the same inconsistent sets of formulas. Every cumulative, structural, and proper inference system satisfies the consistency preservation property. Hence, every such system can be provided with a resolution algebra based proof system, provided that its monotone base is a resolution logic.

4 Problem Solving as Satisfiability

A reasoning task, such as a planning problem, can be solved by, first, expressing it as a satisfiability problem in some logical matrix $\mathcal{M}$ and, then, by solving it using one of the satisfiability solvers for $\mathcal{M}$. In spite of the fact that for many finite matrices $\langle \mathcal{A}, d \rangle$, the satisfiability problem:

(SAT$_{\mathcal{M}}$) for every formula α, determine whether or not there exists an interpretation h such that $h(\alpha) \in d$

is NP-complete, a number of complete and incomplete SAT$_{\mathcal{M}}$ solvers have been developed and their good performance in finding solutions to instances of many problems in real-world domains empirically demonstrated.

Searching for Satisfying Interpretation. Given a matrix $\mathcal{M} = \langle \mathcal{A}, d \rangle$ for a language $\mathcal{L}$, a stochastic local search algorithm for satisfiability in $\mathcal{M}$ starts by generating a random interpretation h restricted to the variables of an input formula α. Then, it locally modifies h by selecting a variable p of α, using some selection heuristic $select\_var(\alpha, h)$, and changing its truth-value from $h(p)$ to some new truth-value using another selection heuristic $select\_val(\alpha, p, h)$. Such selections of variables and such changes of their truth-values are repeated until either $h(\alpha) \in d$ or the allocated time to modify h into a satisfying valuation has elapsed. The process is repeated (if needed) up to a specified number of times.

The above procedure defines informally an incomplete $\mathrm{SAT}_{\mathcal{M}}$ solver (clearly, it cannot be used to determine unsatisfiability of a formula).

Polarity and $\mathbf{SAT}_{\mathcal{M}}$. The classical notion of polarity of a variable p in a formula $\alpha(p)$ captures the monotonic behavior of the term operation $f_\alpha(p)$ induced by $\alpha(p)$ over p in a partially ordered algebra of truth-values. The selection heuristics $select\_var(\alpha, h)$ and $select\_val(\alpha, p, h)$ of an $\mathrm{SAT}_{\mathcal{M}}$ solver can be defined in terms of polarity. This is done in the non-clausal solver polSAT for classical propositional logic as well as in its extensions to finitely-valued logics (cf. [12]).

Improving the Efficiency of Resolution with $\mathbf{SAT}_{\mathcal{M}}$ Solvers. An unrestricted use of the resolution rule during the deductive process may very quickly result in combinatoric explosion of the set of deduced resolvents making the completion of a reasoning task unattainable in an acceptable amount of time. In an efficient resolution-based reasoning program the generation of resolvents that would evidently have no impact on the completion of a reasoning task must be blocked. Tautological resolvents are just that sort of formulas.

For many resolution logics the tautology problem is coNP-complete. For some of these logics, $\mathrm{SAT}_{\mathcal{M}}$ solvers can be used to guide the search for refutation so that the use of tautologies during the refutation process is unlikely. At the same time the refutational completeness of the deductive process is preserved.

References

[1] Blok, W.J. and Pigozzi, D. *Algebraizable Logics*. Memoirs of the American Mathematical Society, vol. 77, no 396 (1989).

[2] Denecke, K. and Wismath, S.L. *Universal Algebra and Applications in Computer Science*, Chapman & Hall/CRC (2002).

[3] Fitting, M. and Orłowska, E. (eds.) *Beyond Two: Theory and Applications of Multiple-Valued Logic*. Studies in Fuzziness and Soft Computing, Phisica-Verlag (2003).

[4] Fitting, M. Bilattices and the semantics of logic programming. *Journal of Logic Programming* 11, pp. 91–116 (1991).

[5] Ginsberg, M.L. Multivalued logics: a uniform approach to reasoning in artificial intelligence. *Comput. Intell.*, vol. 5, pp. 265–316.

[6] Hähnle, R. *Automated Deduction in Multiple-Valued Logics*. Clarendon Press (1993).

[7] Kraus, S., Lehmann, D. and Magidor, M. Nonmonotonic Reasoning, Preferential Models and Cumulative Logics. *Artificial Intelligence* 44, pp. 167-207 (1990).

[8] Makinson, D. General Patterns in Nonmonotonic Reasoning. In *Handbook of Logic in Artificial Intelligence and Logic Programming*. Vol 3: *Nonmonotonic and Uncertain Reasoning* (Gabbay, D.M., Hogger, C.J., and Robinson J.A. eds.). Oxford University Press (1994).

[9] Przymusinski, T. Three-Valued Non-Monotonic Formalisms and Logic Programming". In *Proc. of the First Int. Conf. on Principles of Knowledge Representation and Reasoning (KR'89)*, Toronto, pp. 341-348 (1989)

[10] Rasiowa, H. *An Algebraic Approach to Non-Classical Logics*. North-Holland (1974).

[11] Selman, B., Levesque, H., and Mitchell, D. A new method for solving hard satisfiability problems. In *Proc. AAAI-92*, pp. 440–446 (1992).

[12] Stachniak, Z. Polarity-based Stochastic Local Search Algorithm for Non-clausal Satisfiability. In [3], pp. 181–192 (2002).

[13] Stachniak, Z. *Resolution Proof Systems: An Algebraic Theory*. Kluwer (1996).

[14] Stachniak, Z. Nonmonotonic Theories and Their Axiomatic Varieties. *J. of Logic, Language, and Computation*, vol. 4, no. 4, pp. 317–334 (1995).

[15] Thistlewaite, P.B., McRobbie, M.A., and Meyer, R.K. *Automated Theorem-Proving in Non-Classical Logics*. Pitman (1988).

[16] Wójcicki, R. *Theory of Logical Calculi: Basic Theory of Consequence Operations*. Kluwer (1988).