MIT/LCS/TM-159

DYNAMIC ALGEBRAS AND THE NATURE OF INDUCTION

Vaughan R. Pratt

March 1980

Dynamic Algebras and the Nature of Induction

Vaughan R. Pratt

February, 1980

This research was supported by the National Science Foundation under NSF grant no. MCS78-04338. This paper is a substantial revision of LCS TM-138 [14]. It will appear in the 12th ACM Symposium on Theory of Computation.

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LABORATORY FOR COMPUTER SCIENCE
CAMBRIDGE MASSACHUSETTS 02139

Dynamic Algebras and the Nature of Induction

Vaughan R. Pratt

Abstract

Dynamic algebras constitute the variety (equationally defined class) of models of the Segerberg axioms for propositional dynamic logic. We obtain the following results (to within inseparability). (i) In any dynamic algebra * is reflexive transitive closure. (ii) Every free dynamic algebra can be factored into finite dynamic algebras. (iii) Every finite dynamic algebra is isomorphic to a Kripke structure. (ii) and (iii) imply Parikh's completeness theorem for the Segerberg axioms. We also present an approach to treating the inductive aspect of recursion within dynamic algebras.

Key words

Dynamic algebra, logic, program verification, regular algebra, Segerberg axioms.

Dynamic Algebras and the Nature of Induction

Vaughan R. Pratt

1. Motivation

In this section we motivate at considerable length the study of dynamic algebras in general and the results of this paper in particular. This section contains no material needed to follow the technical part of the paper, while only sections 2, 4, and 5 are needed for completeness of the Segerberg axioms.

The class of dynamic algebras consists of all models of the Segerberg axioms for PDL (propositional dynamic logic) [18]. It was first studied as a class by D. Kozen [9] and the author [13,14], where it arose in connection with the development of appropriate models for program logics, and more generally logics of action [9,13,14]. The class includes many easily recognized algebras. Parikh's completeness result [11] for Segerberg's axioms is a corollary of a nontrivial algebraic property of this class that we establish below. The class can be considered the algebraic home of induction, in the same sense that the class of groups can be considered the algebraic home of invertible operations. Moreover the class offers an interesting partial solution to the problem of defining regular algebras abstractly.

We now develop each of these motivational issues in more detail.

Logics of Action. Common to many logics of action [3,5,8,10,12,17] is some notion of causality between actions and propositions, the idea that an action can or will bring about a proposition. Thus we can expect to find in any model of such a logic a set B of propositions and a set R of actions. B will as a rule be closed under Boolean operations, say $p \vee q$ and p', with Hoare [8] providing the exception that proves the rule, while R will be closed under operations appropriate to actions, perhaps if-then-else and while-do, or following [12], the regular operations of *choice* $a \cup b$, *sequence* a;b (or just ab), and *iteration* a^*.

Exactly how the concept of causality is modelled depends on what one has in mind, but the notion we shall settle for here is that of the *possibility* of an action bringing about a proposition, this possibility being

expressed itself as a proposition. We shall view this construction formally as a function $\langle\rangle:R\times B\to B$, so that $\langle\rangle(a,p)$, or $\langle a\rangle p$, or just ap, is the proposition that action a can bring about proposition p.

In addition we may have a function *test*, $?:B\to R$, which maps proposition p to an action p? whose purpose is to test the truth of p and proceed just when p holds. We may also have a function *converse*, $^-:R\to R$, which maps action a to the action of performing a backwards. In this paper we shall ignore tests (for the sake of simplicity) and converse (because we do not see how to treat it algebraically).

To summarize, a *dynamic algebra* consists of algebras $\mathscr{B} = (B\ \vee\ '\ 0)$ and $\mathscr{R} = (R\ \cup\ ;\ ^*)$, and an operation $\langle\rangle:R\times B\to B$. A dynamic algebra must also satisfy certain equations, which we give in the next section.

Examples. The usual source of dynamic algebras in models of program logics is the class KRI of Kripke structures. Given a set W (to be thought of as a set of possible worlds or *states*), take B to be a nonempty set of unary relations on W closed under union and complement, and take R to be a set of binary relations on W closed under *union*, *composition*, and *ancestral* (reflexive transitive closure). Define $\langle\rangle:R\times B\to B$ so that $ap = \{u\in W|\exists v\in W[(u,v)\in a\ \wedge\ v\in p]\}$, and require that B together with R be closed under $\langle\rangle$. This is a Kripke structure, and can be fairly easily seen to satisfy all the dynamic algebra equations.

Another example, introduced in [13], starts with an alphabet Σ and takes B and R to consist of subsets of $\Sigma^*\cup\Sigma^\omega$, with B closed under union and complement, R closed under union, concatenation, and Kleene closure, and B and R closed under concatenation of a language a from R followed by a language p from B to form ap (but not requiring the other way, pa). This class of dynamic algebras, which we call LAN, is of interest in execution-sequence semantics.

Yet another example starts with a given dynamic algebra $\mathscr{D}' = (\mathscr{B}'\ \mathscr{R}'\ \langle\rangle')$ together with a finite set V (of dimensions or vertices) and takes B to be a set of V-dimensional vectors over B′ closed under pointwise union and pointwise complement, and R to be a set of V×V matrices over R′ closed under pointwise union, matrix multiplication (using $\cup'$ and $;'$ for exterior and interior operators respectively), and a form of star defined by Conway in [4]. Take $\langle\rangle$ to be multiplication of a matrix times a vector (using $\vee'$ and $\langle\rangle'$ for exterior and interior operators respectively). This class of dynamic algebras, called FLO, supplies a semantic basis for Floyd's method of labelling flowcharts; V supplies the points labelled by elements of B′.

These and other classes of dynamic algebras were all shown in [13] to have the same equational theory. The proofs varied considerably in degree of difficulty. Obtaining an upper bound on the equational theory of the language class required a subtle construction. And the upper bound on the theory of Kripke structures was inferred from Parikh's completeness result for Segerberg's axioms, for which an intricate proof appears in [11]. One of our results below implies Parikh's result.

Induction. Induction axioms such as those used in Peano arithmetic are often looked on as an attempt to capture the minimal set satisfying certain closure properties, such as containing 0 and being closed under successor. The Loewenheim-Skolem theorem (any first-order theory with infinite models has models of arbitrary infinite cardinality) dashes any hope of precisely expressing such minimality with even an infinite set of first-order sentences.

Induction also appears in the Segerberg axioms, in a form which will be seen to generalize mathematical induction. It too runs into difficulties, admitting nonstandard models as discussed in [1] and [11].

Does this mean then that induction is just a crude proof-theoretic approximation to a notion of minimality, with no sensible mathematical meaning of its own? We answer this in the negative: the Segerberg axioms for *, which amount to a generalization of mathematical induction and the axiom $\{0\} \cup \sigma\mathbb{N} \subseteq \mathbb{N}$ for the natural numbers, capture precisely the mathematical notion of reflexive transitive closure, of which the *ancestral* of binary relations is a special case. This notion in turn also fails to capture the sort of minimality we had in mind, as it must to account for the existence of nonstandard models. However we at least have a sensible mathematical interpretation of induction.

The question then arises, what is it that induction fails to supply? While we cannot offer a complete answer we do feel that continuity is usually among our unspoken assumptions about the domains to which one applies induction. For example when we express the axiom $\{0\} \cup \sigma\mathbb{N} \subseteq \mathbb{N}$ we have in mind that σ is a completely additive, hence continuous, function on the power set of $\mathbb{N}$, that is, knowing how σ acts on singletons determines its behavior on all other sets. And when f is a continuous function on a complete lattice, the reflexive transitive closure of f is indeed $\bigvee\{f^i \mid i \in \mathbb{N}\}$, coinciding with our intuition about the nature of reflexive transitive closure as encountered in practice.

On the other hand we would also argue that continuity is often not needed *except* for the sake of our intuition. For example Scott's theory of computable functions is founded on continuity, with the reasonable thesis that all computable functions are continuous. Yet some quite basic parts of Scott's

theory work perfectly well for monotonic functions. A cornerstone of the theory, the theorem that every continuous function on a complete partial order has a least fixpoint, also holds for monotonic functions.

Another example is supplied by the second and main result of this paper, that the Fischer-Ladner filtration method for Kripke models of PDL works without appealing to continuity. The original construction dealt with binary relations on W, which act as completely additive and hence continuous functions on the power set of W. Our proof works for strict finitely additive functions. Finite additivity, like monotonicity, is a property that can be captured not only in a first-order way but with mere equations.

Regular Algebras. Another reason for studying dynamic algebras has to do with the problem of defining regular algebras. Normally one would not undertake a study of a class of algebras of type $(\mathscr{B}\ \mathscr{R}\ \langle\rangle)$ without first developing the algebraic theory of its components $\mathscr{B}$ and $\mathscr{R}$. Now while Boolean algebras have been very well understood for nearly half a century, with the key result appearing in 1935 [19], regular algebras remain relatively problematical. Even their identity is open to debate.

The problem is that there has never been proposed a satisfactory definition of an abstract regular algebra, in the sense that there are abstract groups, abstract rings, abstract Boolean algebras, abstract lattices, and so on. This is not for want of trying. One particularly notable effort in this direction is Conway's book on regular algebras [4], where there appear five candidates for a notion of regular algebra no one of which emerges as the obvious favorite.

There are at least two technical obstacles to defining abstract regular algebras. If we follow the tradition of defining such a class using equations, as is done with groups, rings, lattices, Boolean algebras, etc., we might settle for any system of equations that completely axiomatized the equational theory of say the class of all algebras of binary relations closed under union, composition, and ancestral (reflexive transitive closure of binary relations). Happily this class has the same equational theory as that of all algebras of languages closed under union, concatenation, and Kleene closure, suggesting we are on the right track. Unhappily this equational theory is not finitely axiomatizable [15], so we cannot simply give a finite list of equations as the defining characteristics of a regular algebra.

Suppose however that we stiffen our upper lip and admit non-equational axioms, as done in [16], or some collection of axiom schemata, as discussed (inconclusively) in [4]. We would then have the class of those algebras that satisfy the equations holding for all regular algebras of binary relations and

hence for all regular algebras of languages, i.e. the *variety* generated by either one of those classes. A second difficulty now presents itself: there exist algebras in that variety that satisfy *all* these equations yet which contradict our intuition about how a regular algebra should behave. The essence of the following example appears on p. 102 of [4].

Take R = {1,2,3} and interpret both $\cup$ and ; as max, so $1 \cup 2 = 1;2 = 2$, etc. Now one would expect $1^* = 1$, $2^* = 2$, and $3^* = 3$ in any reasonable regular algebra in which $\cup$ and ; behave in this way. In particular we have $2 = 2;2 = 2;2;2 = \ldots$ so that $2^* = 2$ would seem to be a foregone conclusion. Yet we may take $2^* = 3$ instead without contradicting any equation of the regular theory of binary relations!

To see this, let $\mathscr{A}$ be the regular algebra of all languages on some alphabet that contain the empty string λ and let $h:\mathscr{A}\to\{1,2,3\}$ map $\{\lambda\}$ to 1, all other finite languages to 2, and infinite languages to 3. Now h is a homomorphism with respect to the above interpretations of the regular operations on {1,2,3}, as may be verified, and homorphisms preserve equations, so $h(\mathscr{A})$ satisfies all equations holding for regular algebras of languages.

We propose the following definition: a regular algebra is any set of strict finitely additive functions on a Boolean algebra closed under the operations of pointwise disjunction, composition, and reflexive transitive closure. To make the class more abstract we may also take all algebras of similarity type ($R \cup$; *) isomorphic to such sets of functions. The resulting class then consists precisely of the regular components of separable dynamic algebras. Except for the issue of separability, this class is defined purely equationally.

It will be apparent from Section 5 that this class includes all relational algebras. It is easily seen that every regular algebra of languages on alphabet Σ is isomorphic to some relational algebra of binary relations on W (take $W = \Sigma^*$ and let language L correspond to the binary relation $\{u,uv|u\in\Sigma^*,v\in L\}$), so the class also includes all language algebras.

*-Continuous Dynamic Algebras. Kozen's definition of a dynamic algebra is not identical to ours. He imposes an additional condition, that a dynamic algebra be **-continuous*, namely that $a^* = \cup\{a^i|i\in\mathbb{N}\}$ with ; and $\langle\rangle$ distributing over such unions.

Without *-continuity one can have dynamic algebras that are easily seen not to be isomorphic to any Kripke structure. A simple example involving $\mathbb{N}\cup\{\infty\}$ appears in Section 7. Thus the condition is well-motivated. However Kozen, and more recently Trnkova and Reiterman, have exhibited *-continuous

dynamic algebras that are not isomorphic to any Kripke structure, so although this condition is necessary it is not sufficient to capture the abstract essence of Kripke structures. It remains an open problem to find abstract conditions that capture precisely those dynamic algebras isomorphic to Kripke structures.

It is the intent of Kozen and the author to decide between them the question of whether the term "dynamic algebra" should include *-continuity as one of its requirements. In this paper dynamic algebras are assumed to form a variety (equationally defined class), and we make use of properties of varieties in deriving our results. As we argued earlier we do not see continuity as an essential aspect of computational models, and prefer to say "*-continuous dynamic algebra" when we wish to include *-continuity as a condition. (Note that continuity implies *-continuity but not conversely.)

2. Definitions

Dynamic Algebras. The similarity type of the class DYN of dynamic algebras is $((B\ \vee\ '\ 0)\ (R\ \cup\ ;\ *)\ \langle\rangle)$. Dynamic algebras satisfy the following equations. We abbreviate a;b to ab, $\langle a\rangle p$ to ap, $(p' \vee q)'$ to p-q, and $p \vee q = q$ to $p \leq q$.

1.	$\mathscr{B}$ Boolean algebra	3.	$(a \cup b)p = ap \vee bp$
2a.	$a0 = 0$	4.	$(ab)p = a(bp)$
2b.	$a(p \vee q) = ap \vee aq$	5a,b.	$p \vee aa^*p \leq a^*p \leq p \vee a^*(ap-p)$.

DYN is a variety (equationally defined class). Hence it is closed under homomorphisms, subalgebras, and direct products; it has free algebras; and every equational identity of dynamic algebra may be proved from instances of the above axioms using only the fact that equality is a congruence relation (completeness of equational logic).

*Content of the * Axioms.* Axioms 5a,b are less transparent than their fellow axioms, but can be restated more succinctly as follows.

Let $a!p = \{q \mid p \vee aq \leq q\}$. Then axiom 5a says that $a^*p \in a!p$. Let μS be the minimum element of S if it exists (as opposed to $\wedge S$, the *meet* of S, which need not be in S). We propose the following alternative to 5a,b.

5′. $a^*p = \mu(a!p)$.

Lemma 1. 5a,b are equivalent to 5′.

Proof. (→). Assume 5a,b. 5a asserts that $a^*p \in a!p$. Now consider an arbitrary $q \in a!p$, so $p \le q$ and $aq \le q$. Then

a^*p	$\le a^*q$	$(a^*p \vee a^*q = a^*(p \vee q) = a^*q)$
	$\le q \vee a^*(aq-q)$	(axiom 5b)
	$= q \vee a^*0$	$(q \in a!p \to aq \le q)$
	$= q$	(axioms 2a and 1).

(←). Assume $a^*p = \mu(a!p)$. Then $a^*p \in a!p$, so 5a holds. For 5b it suffices to show that $p \vee a^*(ap-p) \in a!p$, since $a^*p \le q$ for any $q \in a!p$. We have

$p \vee a(p \vee a^*(ap-p))$	$= p \vee (a(p \vee a^*(ap-p))-p)$	(axiom 1)
	$= p \vee ((ap \vee aa^*(ap-p))-p)$	(axiom 2b)
	$\le p \vee (ap-p \vee aa^*(ap-p))$	(axiom 1)
	$\le p \vee a^*(ap-p)$	(axiom 5′) ∎

Inseparability. When $ap = bp$ for all p in B we write $a \equiv b$ and say that a and b are *inseparable.* In a *separable dynamic algebra* (SDA) inseparability is the identity relation on R [9]; SDYN denotes the class of SDA's.

Lemma 2. Inseparability is a congruence relation.

Proof. Suppose $a \equiv a''$ and $b \equiv b''$. Then for all p in B, $(a \cup b)p = ap \vee bp = a''p \vee b''p = (a'' \cup b'')p$, whence $a \cup b \equiv a'' \cup b''$. Similarly we can show $a;b \equiv a'';b''$. Finally, for all p in B, $a^*p = \mu(a!p) = \mu(a''!p) = a''^*p$, so $a^* \equiv a''^*$. ∎

Boolean-trivial dynamic algebras have only one Boolean element. $\mathscr{A}$ is a *subdirect product* of algebras $\mathscr{A}_i$ when there exist onto homomorphisms $h_i: \mathscr{A} \to \mathscr{A}_i$ whose product is injective. (So $\mathscr{A}$ is isomorphic to a subalgebra of the direct product of the $\mathscr{A}_i$'s.)

Lemma 3. Every dynamic algebra $\mathscr{D}$ is a subdirect product of a separable and a Boolean-trivial dynamic algebra.

Proof. Divide $\mathscr{R}$ by $\equiv$ to get the separable algebra, and collapse $\mathscr{B}$ to a point to get the Boolean-trivial one. The product of the corresponding natural transformations (the two homomorphisms from $\mathscr{D}$ that yield each of these quotients) is clearly injective. ∎

Our own preference in studying dynamic algebras is to consider the separable ones first and generalize the results to other dynamic algebras via Lemma 3. A more direct assault could omit any mention of separability.

Actions as Functions. A separable dynamic algebra may be viewed as a set of functions on a Boolean algebra, with $\diamond$ interpreted as application. Thus the content of axioms 2-5 is that R consists of strict (2a) finitely additive (2b) functions closed under pointwise disjunction (3), composition (4), and the operation that maps function a to the function mapping p to $\mu(a!p)$ (5). 3,4,5 define the three *regular functionals.*

Full Dynamic Algebras. The *full dynamic algebra on* the Boolean algebra $\mathscr{B}$ has for R the set of all strict finitely additive functions on $\mathscr{B}$. R is closed under the regular functionals; we shall verify only the case of finite additivity of a*. We have $p \vee q \vee a(a^*p \vee a^*q) \leq a^*p \vee a^*q$, whence $a^*p \vee a^*q \in a!(p \vee q)$, so $a^*(p \vee q) \leq a^*p \vee a^*q$. Conversely we have $a^*(p \vee q) = \mu(a!(p \vee q)) = \mu(a!p \cap a!q) \geq \mu(a!p) \vee \mu(a!q) = a^*p \vee a^*q$.

Word and Free Algebras. The *word algebra* of a given similarity type generated by X_0 consists of all terms of that similarity type with variables drawn from X_0. The *free dynamic algebra* (resp. free SDA) generated by $D_0 = B_0 \cup R_0$ is the quotient of the word algebra of similarity type DYN generated by D_0 with the congruence relating all terms identically equal in DYN (resp. SDYN). Variables ranging over generators will be written in upper case, namely P,Q,A,B. Every map from the generators of a free (separable) dynamic algebra to the elements of a (separable) dynamic algebra extends to a homomorphism from the whole free algebra. (To see this easily for SDA's, assume B_0 is nonempty.)

3. * is Reflexive Transitive Closure

(This section is not needed for the sequel.)

Ordinarily one thinks of reflexivity and transitivity as properties of binary relations. If we consider the binary relation f on the set W to be a completely additive function $f:2^W \to 2^W$ (f(S) for arbitrary $S \subseteq W$ is determined by $f(\{s\})$ for singletons $\{s\} \subseteq W$ via $f(S) = \cup\{f(\{s\}) | s \in W\}$) then we may define f to be *reflexive* when $I \leq f$ (I the identity function) and *transitive* when $f^2 \leq f$. (We are using the usual induced order on functions with range some poset, namely $f \leq g$ when $f(x) \leq g(x)$ for all x.)

Following the algebraic approach of [2], we may define a *closure operation* on a poset R to be any reflexive monotonic idempotent function on R. Now take R to be the set of all completely additive functions on 2^W. The

above induced order on R further induces an order on functions on R (needed for "closure" to be well defined for functions on R). The closure operation on R with fixpoints the reflexive transitive elements of R is then reflexive transitive closure, as the reader may verify.

We now propose that these definitions be *generalized* to encompass arbitrary functions on arbitrary posets. In particular reflexive transitive closure generalizes to any closure operation whose fixpoints are the set of reflexive transitive elements of the domain of the closure operation, regular algebras in this paper.

The reflexive transitive closure of binary relations is often called the *ancestral* in non-computer-science circles. We suggest that "ancestral" be used on those occasions where it is necessary to distinguish reflexive transitive closure of binary relations from the more general closure operation. Such occasions will not arise for those who would not contemplate applying "reflexive transitive closure" to anything but binary relations.

Theorem 4. In an SDA * is reflexive transitive closure.

Proof. First we show that * is a closure operation on R.
Reflexivity. $p \leq a^*p$, so $ap \leq aa^*p \leq a^*p$, for all p, whence $a \leq a^*$ on B, so $I \leq *$ on R.
Monotonicity. If $a \leq b$ then for all p, $b!p \subseteq a!p$, whence $\mu(a!p) \leq \mu(b!p)$, thus $a^*p \leq b^*p$, so $a^* \leq b^*$.
Idempotence. $a^*p = \mu(a!a^*p) = a^*a^*p$, so $a^*p \in a^*!p$. But if $q \in a^*!p$, $p \leq q$, so $a^*p \leq a^*q \leq q$, whence $a^*p = \mu(a^*!p) = a^{**}p$.

Second we show that the fixpoints of * are the reflexive transitive elements of R. Observe that a necessary and sufficient condition for a to be reflexive and transitive is that for all p, $p \vee aap \leq ap$, i.e. $ap \in a!p$. So if a is a fixpoint of *, $ap = a^*p \in a!p$. Conversely if $ap \in a!p$ then $a^*p = \mu(a!p) \leq ap$. We showed $ap \leq a^*p$ above, so $a^*p = ap$. ∎

4. Factoring Free SDA's

We now show that every free SDA is a subdirect product of finite SDA's. This result generalizes Fischer and Ladner's finite model theorem to other than completely additive dynamic algebras.

FL-sets. An FL-set is a Boolean subset F of an algebra of similarity type DYN such that

$p \vee q \in F \rightarrow p,q \in F$
$p' \in F \rightarrow p \in F$
$ap \in F \rightarrow p \in F$
$(a \cup b)p \in F \rightarrow ap,bp \in F$
$(ab)p \in F \rightarrow a(bp) \in F$
$a^*p \in F \rightarrow aa^*p \in F.$

FL(X) is the least FL-set containing X. In word algebras FL preserves finiteness [9].

For any subset X of an algebra we let G(X) denote the subalgebra generated by X. In the following G will be used only to close Boolean subsets under $\vee$ and $'$. In Boolean algebras G preserves finiteness, and in word algebras G preserves being an FL set. Homomorphisms commute with G, i.e. h(G(X)) = G(h(X)).

Theorem 5. Every free SDA is a subdirect product of finite SDA's.

Proof. It suffices to show that for any finite Boolean subset A of the free SDA $\mathscr{D}$ there exists a homomorphism f from $\mathscr{D}$ injective on A such that f($\mathscr{D}$) is finite and separable. For if we take g to be the product of such f's over say all doubletons $A \subseteq B$, g will clearly be injective on B. For $a \neq b$ in R, $ap \neq bp$ for some p in B, so $g(a)g(p) = g(ap) \neq g(bp) = g(b)g(p)$, so $g(a) \neq g(b)$, whence g is also injective on R.

Let $\mathscr{D}''$ be the word algebra generated by D_0, the generators of $\mathscr{D}$, with $h:\mathscr{D}'' \rightarrow \mathscr{D}$ the onto homomorphism fixing D_0. Choose finite $A'' \subseteq B''$ satisfying h(A") = A. Let $\mathscr{D}'$ be the full dynamic algebra on G(h(FL(A"))), clearly finite. Let $f:\mathscr{D} \rightarrow \mathscr{D}'$ be the homomorphism which fixes $B_0 \cap B'$, sends $B_0 - B'$ to 0, and sends $A \in R_0$ to the function on B' that maps p to $\wedge\{q \in B' \mid Ap \leq q\}$; f(A) is strict and finitely additive (exercise) and so in R'.

We now show that f fixes B'. Note that B' = G(h(FL(A"))) = h(G(FL(A"))). We abbreviate G(FL(A")) to G". All inductions will be performed on the structure of elements of R", B", or G", justifiable since D = h(D") and B' = h(G"). We shall denote elements of B",R" by π,α, and write p,a for $h(\pi),h(\alpha)$ respectively. We write a!'p for a!p $\cap$ B'. It will help to think of R and R' as consisting of partial functions on B; in this way $\diamond$ and $\diamond'$ may be viewed as the same operation, namely application.

Claim (i). For all $a \in R$ and $p \in B'$, $f(a)p \geq ap$.
Claim (ii). For all $\alpha\pi \in G''$, f(a)p = ap.

Both proofs proceed by induction on the structure of α in R". We prove (i) explicitly; for (ii) replace $\geq$ by = throughout the proof of (i).

$f(A)p = \wedge\{q \in B' \mid Ap \leq q\} \geq Ap.$
$f(a \cup b)p = f(a)p \vee f(b)p \geq ap \vee bp = (a \cup b)p.$
$f(ab)p = f(a)f(b)p \geq abp.$
$f(a^*)p = \mu(f(a)!'p) \geq \mu(a!p) = a^*p.$

Only the $\geq$ in the argument for * should present any difficulty. In claim (i) we have $f(a)q \geq aq$ for all q in B', by induction, so $f(a)!'p \subseteq a!'p \subseteq a!p$. In claim (ii) $\alpha^*\pi \in G"$, so $\alpha\alpha^*\pi \in G"$, so $f(a)a^*p = aa^*p \leq a^*p$, so $a^*p \in f(a)!'p$, so $\mu(f(a)!'p) \leq a^*p$.

Claim (iii), our goal: For all $p \in B'$, $fp = p$.

We proceed by structural induction on π in G".

$f(P) = P$ (h,f fix generators in G",B')
$f(p \vee q) = f(p) \vee f(q) = p \vee q$
$f(p') = (f(p))' = p'$
$f(ap) = f(a)f(p) = ap$ (claim (ii) and induction) ∎

5. Kripke Structures

A Kripke structure is a subalgebra of the dynamic algebra of all completely additive functions on a power set. Kripke structures supply intuitively satisfying models for dynamic logic, and hence serve as a benchmark for completeness of axiomatizations of dynamic logic.

Theorem 6. Every finite SDA $\mathscr{D}$ is isomorphic to a Kripke structure.

Proof. By finiteness $\mathscr{B}$ is isomorphic to a power set while $\mathscr{R}$ consists of completely additive functions, whence $\mathscr{D}$ is a subalgebra of the dynamic algebra of all completely additive functions on $\mathscr{B}$. ∎

Completeness of Segerberg's Axioms. Every SDA is a homomorphic image of a subdirect product of Kripke structures, by Theorems 5 and 6. Hence the equational theory of SDA's includes all Kripke identities. So by Lemma 3 the Boolean equational theory of dynamic algebras includes all Boolean Kripke identities. Parikh's completeness theorem for the Segerberg axioms then follows from the completeness of equational logic.

6. Recursion

R. Ladner has pointed out [conversation] that what one might call context-free PDL has a non-r.e. theory. (The exact details of the syntax of such a programming language do not matter here, but one could adopt the least-fixed-point operator as done in [7].) Indeed the set of identities of the form $aP \leq bP$ is not r.e., or we would have a method of enumerating the set $\{(G,G') | L(G) \subseteq L(G')\}$ of pairs of context-free grammars one of whose languages is included in the other, a familiar non-r.e. set.

It follows that there can be no r.e. axiomatization of such a theory. Thus we cannot expect to find nice equations like 5a,b on which to base an algebraic theory of recursion. If there is any algebraic structure to be found it must be sought in other places than finite or even effectively given sets of axioms.

Our purpose here is to propose another way of looking at recursion, essentially the way Scott looks at it. Our advice is in effect to divide recursion into two parts that inhabit separate theories, one of which is PDL. One quite reasonable way to carry out a recursive calculation is really iterative: repeatedly expand *all* procedure calls simultaneously. The expansion itself is a finite process; it is only the *repetition* of the expansion that must be iterated indefinitely. The finite expansion process can be described by some other theory than PDL, just as must be done for assignment (since the theory of PDL plus assignment is non-r.e.).

What this means in practice is that one reasons about one's recursive procedures on the one hand an expansion at a time, and on the other hand in connection with the process of repeated expansion. Consider the canonical example of a problem that "must" be solved recursively, the Tower of Hanoi problem. (Ignore the iterative solution that moves the smallest disk clockwise on every other move.) This can easily be viewed iteratively by describing how to convert a solution for n disks into one for n+1 disks, a process with no iteration if a "solution" is represented as the obvious 3×3 matrix of strings of moves. The conversion can then be iterated to obtain the solution for the given number of disks.

The iterative part of this factoring is already dealt with in dynamic logic. The expansion part should be incorporated into a separate logic.

This factoring will not in general capture the whole theory of recursion, particularly if the theory dealing with expansion is recursive or even r.e. However the foregoing remarks about the intractibility of the theory of recursion show that we must content ourselves with such approximations in

practice. The particular approximation we propose is in our opinion both natural and useful in practice.

7. Models of Induction

With the definitions and results behind us we may augment the discussion of Section 1 with some more technical remarks.

We first remark on a relationship of the * axioms to the Peano axioms. Consider the dynamic algebra of all completely additive functions (binary relations) on the power set of $\mathbb{N}$, and let a be successor, satisfying $a(S) = \{s+1|s\in S\}$. With a little Boolean manipulation axiom 5b can be seen to be exactly the principle of mathematical induction, usually expressed as $\varphi(0) \wedge \forall x[\varphi(x)\rightarrow\varphi(x+1)] \rightarrow \forall x\varphi(x)$. If in addition p is taken to be $\{0\}$ then $a^*p = \mathbb{N}$ and 5a is seen to be the statement that $\mathbb{N}$ contains 0 and is closed under taking successor. These two components of the Peano axiomatization of arithmetic can in this way be seen to correspond to the * axioms. (The Peano axioms that prevent successor from cycling do not play a role in dynamic algebra.)

Of course the above example is not the only model of arithmetic, which is well known to have nonstandard models. The nonstandard models of PDL discussed in [1] and [11] seem at variance with Theorem 4, which says that the Segerberg axioms define * to be reflexive transitive closure. The paradox is resolved by the observation that reflexive transitive closure need not always satisfy $a^* = \cup\{a^i|i\in\mathbb{N}\}$. If we modify the example of the previous paragraph by taking all strict finitely additive functions on $\mathbb{N}\cup\{\infty\}$, defining $\infty+1 = \infty$, and taking $a(S) = \{s+1|s\in S\}\cup\{\infty\|S|=\infty\}$ then $a^*\{0\} = \mathbb{N}\cup\infty$ whereas $\cup\{a^i|i\in\mathbb{N}\} = \mathbb{N}$.

If however we take only the continuous functions on $\mathbb{N}\cup\{\infty\}$ then we exclude the above function a, and it is easily shown then that $a^* = \cup\{a^i|i\in\mathbb{N}\}$ no matter which continuous function we take a to be. This suggests that our intuition about reflexive transitive closure tacitly assumes continuity. Thus we could attribute the origin of nonstandard models of the Segerberg axioms to their failure to deal with continuity rather than with *, which we have shown to be captured exactly.

Acknowledgments

Helpful comments on a preliminary version of this paper [14] were supplied by J. Halpern, A. Itai, D. Kozen, M. Majster, A. Meyer, R. Parikh, J. Reiterman, and V. Trnkova.

Bibliography

[1] Berman, F., A Completeness Technique for D-axiomatizable Semantics, Proc. 11th Ann. ACM Symp. on Theory of Computing, 160-166, Atlanta, Georgia, May 1979.

[2] Cohn, P.M., *Universal Algebra*, Harper and Row, New York, 1965.

[3] Constable, R.L., On the Theory of Programming Logics, Proc. 9th Ann. ACM Symp. on Theory of Computing, 269-285, Boulder, Col., May 1977.

[4] Conway, J.H., *Regular Algebra and Finite Machines*, Chapman and Hall, London, 1971.

[5] Dijkstra, E.W., *A Discipline of Programming.* Prentice-Hall. 1976

[6] Fischer, M.J. and R.E. Ladner., Propositional Modal Logic of Programs, Proc. 9th Ann. ACM Symp. on Theory of Computing, 286-294, Boulder, Col., May 1977.

[7] Harel, D., Logics of Programs: Axiomatics and Descriptive Power, Ph.D. thesis, Dept. of EECS, MIT, MIT/LCS/TR-200, May 1978.

[8] Hoare, C.A.R., An Axiomatic Basis for Computer Programming, CACM *12*, 576-580, 1969.

[9] Kozen, D., A Representation Theorem for Models of *-free PDL, Manuscript, c. May 1979.

[10] Kroeger, F., Logical Rules of Natural Reasoning about Programs, In *Automata, Languages and Programming 3* (ed. Michaelson, S. and R. Milner), 87-98. Edinburgh University Press, 1976.

[11] Parikh, R., A Completeness Result for PDL, Symposium on Mathematical Foundations of Computer Science, Zakopane, Warsaw, Sept. 1978.

[12] Pratt, V.R., Semantical Considerations on Floyd-Hoare Logic, Proc. 17th Ann. IEEE Symp. on Foundations of Comp. Sci., 109-121. Oct. 1976.

[13] Pratt, V.R., Models of Program Logics, 20th IEEE Conference on Foundations of Computer Science, San Juan, PR, Oct. 1979.

[14] Pratt, V.R., Dynamic Algebras: Examples, Constructions, Applications, MIT/LCS/TM-138, M.I.T. Laboratory for Computer Science, May 1979.

[15] Redko, V.N., On Defining Relations for the Algebra of Regular Events, (Russian), Ukrain. Mat. Z., *16*, 120-126, 1964.

[16] Salomaa, A., Two Complete Axiom Systems for the Algebra of Regular Events, J. of the ACM *13*, 158-169, 1966.

[17] Salwicki, A., Formalized Algorithmic Languages, Bull. Acad. Pol. Sci., Ser. Sci. Math. Astr. Phys. Vol. 18. No. 5. 1970.

[18] Segerberg, K., A Completeness Theorem in the Modal Logic of Programs, Preliminary report. Notices of the AMS, *24*, 6, A-552. Oct. 1977.

[19] Stone, M.H., The Theory of Representations for Boolean Algebras, Trans. of the Am. Math. Soc. *40*, 37-111, 1936.