MASSACHUSETTS INSTITUTE OF TECHNOLOGY
A. I. LABORATORY

Artificial Intelligence
Memo No. 262 June 1972

A CONCRETE APPROACH TO ABSTRACT RECURSIVE DEFINITIONS

Mitchell Wand

(This paper was originally presented at the "Symposium on the Theory of Automata Programming and Languages", I.R.I.A., Paris, July 1972.)

Work reported herein was conducted at the Artificial Intelligence Laboratory, a Massachusetts Institute of Technology research program supported in part by the Advanced Research Projects Agency of the Department of Defense and monitored by the Office of Naval Research under Contract Number N00014-70-A-0362-0003.

Reproduction of this document, in whole or in part, is permitted for any purpose of the United States Government.

A CONCRETE APPROACH TO ABSTRACT RECURSIVE DEFINITIONS

MITCHELL WAND
Massachusetts Institute of Technology
Artificial Intelligence Laboratory
Cambridge, Massachusetts, USA

Abstract: We introduce a non-categorical alternative to Wagner's Abstract Recursive Definitions [Wg-1,2], using a generalization of the notion of clone called a μ-clone. Our more concrete approach yields two new theorems: 1. The free μ-clone generated by a ranked set is isomorphic to the set of loop-representable flow diagrams with function symbols in the set. 2. For every element of a μ-clone there is an expression analogous to a regular expression. Several well-known theorems of language and automata theory are drawn as special cases of this theorem.

1. INTRODUCTION

A common facet of much recent research in computer science is the study of certain complete lattices. One defines so-called "algebraic" elements [Sc] in a complete lattice by a fixed-point equation:

$$(\mu x \in L)[x = F(x)]$$

Our aim (as is [Wg-1,2]'s) is to "algebraicize" this process: to determine the properties of this construction so that we may recognize certain properties of a particular system as instances of some general properties of fixed-point systems, and thus concentrate our attention on the properties which are unique to the particular system under study.

Previous approaches to this problem have leaned heavily upon categories, in particular, upon the notion of an "algebraic theory." We have felt for some time that the introduction of categories at best presents a pedagogical barrier to many computer scientists, and at worst may lead an investigator to miss important results because of his imperfect understanding of the notation. We have therefore worked towards developing more concrete representations for these mathematical structures. While no doubt the most elegant mathematics will come in categorical language (and indeed many of our results are strongly categorical in flavor) it is instructive to see how much can be developed without the explicit introduction of category theory.

In previous work we showed how "clones of operations" [C] could be used instead of algebraic theories. Using clones we obtained a development of the algebraic theory of automata entirely analogous to the string case [Wa], a highly transparent proof of the theorem that every algebraic functor has a left adjoint, and a Jordan-Holder Theorem for the eminently nonabelian category of theories.

In this paper, we apply these techniques to the study of "abstract recursive definitions." In Section 2, we define our basic object of study, the μ-clone, which is roughly Wagner's "set of meanings of an additively-closed language." In Section 3, we obtain a characterization of free μ-clones in terms of loop-representable flow diagrams, extending a theorem of Elgot [E] on finite diagrams. This is our key result, for it enables us to prove theorems about any μ-clone by considering only the well-known μ-clone of loop-representable flow diagrams (regular trees). We note that the normal form theorem of [Bk] may be obtained trivially in this way. In Section 4, we use this result to show that the general fixed-point operation can be replaced by a much more restricted form in which simultaneous equations are eliminated. We call such a restricted form a "quasi-regular expression" because a fixed point on a single equation is seen to be

analogous to the Kleene star. Our version of this theorem is stronger than that of Bekic [Bk]. We draw as corollaries several classical theorems of language theory which until now were considered quite separately.

2. DEFINITIONS

The starting point for an algebraicization is the observation that the choice of the permissible F's is crucial. One can characterize this choice by saying that the F's must be chosen from some clone of continuous operations on L (ie the morphisms of some theory). Then the fixed-point operation is seen as just another closure operation on a set of continuous functions. This leads to the following crucial definition:

DEFINITION: Let L be a complete lattice with least element $\perp$, and for each $n \geq 0$ let V_n be a set of continuous functions $L^n \to L$. Then $V = \cup V_n$ is a <u>μ-clone of operations</u> on L iff

i) for all n, all i, $1 \leq i \leq n$, $e^n_i = \lambda(x_1,..,x_n)\ [x_i]$ is in V_n.

ii) for all $f \in V_n$, $g_1,..,g_n \in V_k$, $f^V(g_1,..,g_n) = \lambda(x_1,..,x_k)\ [f(g_1(x_1,..,x_k),..,g_n(x_1,..,x_k))]$ is in V_k.

iii) for all $f \in V_n$, $g_1,..,g_n \in V_{k+n}$, $f^\mu(g_1,..,g_n) = \lambda(x_1,..,x_k)\ [f((\mu(c_1,..,c_n) \in L^n)\ (\forall i \leq n)\ [c_i = g_i(x_1,..,x_k,c_1,..,c_n)])]$ is in V_k.

iv) if $a \in V_0$ then $K_n(a) = \lambda(x_1,..,x_n)\ [a]$ is in V_n.

(i) and (ii) guarantee that V is a clone [C] (ie, the (1,n) morphisms of some theory). (iii) is a rather general fixed-point operation. Given $x_1,..,x_k$, one computes $(f^\mu(g_1,..,g_n))(x_1,..,x_k)$ as follows: Substitute the x_i into the fixed-point equations

$$c_i = g_i(x_1,..,x_k,c_1,..,c_n),$$

solve them, and then take $f(c_1,..,c_n)$. The Tarski fixed point theorem assures us that this algorithm yields a continuous function [T]. This is obviously a notational variant of the fixed-point operation of [Wg-2, Sec. II]. The last operation of taking f of the c's is of course superfluous; it is included so that all our functions will take their values in L. A recurring theme in this paper will be the replacement of this operation by more restricted forms, without loss in power. (iv) is a technical requirement which we need to provide compatibility with theories. We say L is the <u>carrier</u> of V.

Where V is clear, we will often omit the infix operator V in (ii), and write $f(g_1,..,g_n)$. If V is a μ-clone, we will sometimes write n(V) for the set V_n of n-ary operations of V, when writing V_n would be cumbersome. We will denote algebras by names, eg ACF, ACh, AGr. If A is an algebra, let Car(A) and Ops(A) denote the carrier and operations of A, respectively. We call an algebra whose carrier is a complete lattice and whose operations are continuous a <u>lattice algebra.</u> We will use N to denote the set of non-negative integers.

μ-Clones are ranked algebras in the sense of [Hi], so we can define μ-clone homomorphisms, μ-clone congruences, etc., in the usual way and get all the usual elementary theorems. For this paper we will only be concerned with homomorphisms.

DEFINITION: Let V be a μ-clone of operations on a complete lattice L, and W be a μ-clone of operations on a complete lattice M. Then a map $h: V \to W$ is a <u>μ-clone homomorphism</u> iff

i) if $f \in V_n$, $h(f) \in W_n$

ii) $h(e_i^n) = e_i^n$, $h(\bigsqcup) = \bigsqcup$, $h(K_n a) = K_n(h(a))$

iii) $h(f^V(g_1,..,g_n)) = (h(f))^W(h(g_1),..,h(g_n))$

iv) $h(f^\mu(g_1,..,g_n)) = (h(f))^\mu(h(g_1),..,h(g_n))$

Evidently μ-clones are closed under intersection, so we can talk about $\mu Cl(A)$, the least μ-clone containing the operations of a lattice algebra A, and construct it in the usual inductive manner. Our first theorem is a non-inductive characterization of $\mu Cl(A)$.

THEOREM 1: (Normal Form Lemma). Let A be a lattice algebra, $f \in k(\mu Cl(A))$. Then there exist $n, f_1,..,f_n$ in Ops(A) (with f_i r(i)-ary) and a function $v: N \times N \to N - \{0\}$ such that if $j \leq r(i)$, then $v(i,j) \leq n+k$ and such that

$$f = (e_1^n)^\mu(g_1,..,g_n)$$

where

$$g_i = f_i^V(e^{k+n}_{v(i,1)},..,e^{k+n}_{v(i,r(i))}) \text{ or } g_i = e^{k+n}_{v(i,1)}$$

ie, f is the first coordinate of the solution of a set of fixed point equations, each of which involves only an operation from A composed with an appropriate tuple of projections which "select" the right "inputs".

The proof is quite easy, although we will draw it as a corollary to Theorem 3. This is similar to the Basis Theorem of [Wg-1 Thm 4.7]. It is called the "Normal Form Lemma" because it is intimately related to normal-form theorems in language theory. We will do one simple example to show this connection and a slightly harder one to show that one can study "non-standard" language features in this framework. For the rest of this section, let Σ be a finite set of terminal symbols and L be the complete lattice of subsets of Σ^* (ie languages over Σ).

EXAMPLE 2.1: Three generating algebras for the context-free languages. Let ACh be the algebra whose carrier is L and whose operations are as follows: for each $a \in \Sigma$, a 0-ary operation {a}, and two 2-ary operations: union and "$\cdot$" = $\lambda(S,T)\ [\{st \mid s \in S, t \in T\}]$ By the Normal Form Lemma, we need only consider sets of equations of the form

$$c_i = \{a\}$$

$$c_i = c_j \cup c_k$$

$$c_i = c_j \cdot c_k$$

This we recognize as Chomsky Normal Form. Thus $0(\mu Cl(ACh))$ is the set of context free languages over Σ. Similarly we can construct ACF and AGr corresponding to unrestricted CF grammars and Greibach normal-form grammars. The "normal form theorems" of language theory are seen to be statements of the form "$\mu Cl(A)$ = some set of languages." Thus $\mu Cl(ACh) = \mu Cl(AGr) = \mu Cl(ACF)$, but $Cl(ACh) = Cl(ACF) \neq Cl(AGr)$. So the student's intuitive notion that Chomsky's normal form theorem is "easier" than Greibach's is supported in this theoretical framework.

EXAMPLE 2.2: Copying Rules. This framework is ideal for formulating ideas about general operations on strings. Here we formulate the γ operation of [M-W]. Let A have as constant operations the singletons as before, a 2-ary union, and n-ary operations

$$F_n = \{f_w \mid w \in \{1,..,n\}^*\}$$

where

$$f_{i_1,..,i_p}(S_1,..,S_n) = \{w_{i_1}...w_{i_p} \mid (\forall k)[w_k \varepsilon S_k]\}$$

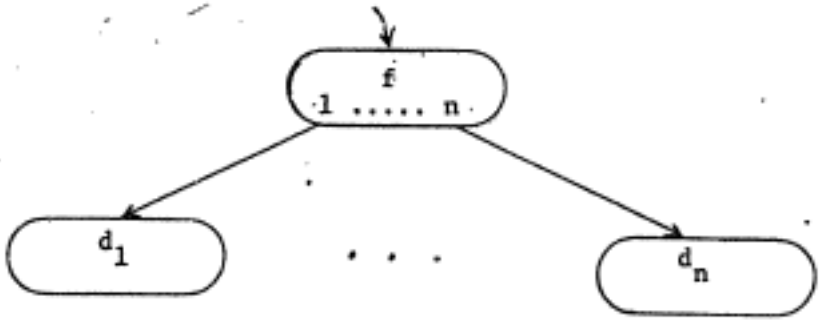
This is the algebra of "IO" substitution: $f_{11}(S) = \{ww \mid w \varepsilon S\}$. The f's are clearly continuous, so we can take $\mu Cl(A)$. $O(\mu Cl(A))$ is a class of languages properly contained between the context free languages and Fischer's IO macro languages [Fi]. $\{1^{2^n} \mid n > 1\}$ is generable, so the class is not closed under inverse homomorphism. (We have a number of results for these languages, most notable an intercalation theorem which shows that $\{a^n b^n c^n \mid n>1\}$ is not in the class. Thus copying of proper phrases is seen to be a very weak operation). This is an example of a non-trivial theory whose (0,1)-definitions in the given semantic domain do not form an AFL.

3. FREE μ-CLONES AND RATIONAL FLOW DIAGRAMS

We assume the reader is familiar with the complete lattice of flow diagrams as developed in [Sc]. We modify his development in the following trivial ways, in the spirit of [E] : 1. We replace function and predicate symbols with a ranked set Ω of symbols which combine functions and predicates. 2. We imagine that rather than having a single exit, any finite branch of a diagram terminates in a positive integer which symbolizes the "return code" of that exit, just as a function/predicate exits on a specified exit line. We are particularly concerned with altering the set of primitive symbols and permissible return codes, so we will use the following notations: $E(\Omega)$ for the lattice of flow diagrams with symbols in Ω; $F_n(\Omega)$ for the set of finite diagrams with symbols in Ω, $\perp$ appearing only at exits, and with exits in $\{1,..,n\}$; and $R_n(\Omega)$ for the set of diagrams representable by loops with the same symbol and exit conventions as $F_n(\Omega)$. We refer to a loop-representable diagram as a <u>rational</u> diagram.

Note that $R_n(\Omega) \subseteq R_{n+1}(\Omega)$. We define $R(\Omega)$ to be the <u>disjoint</u> union of the $R_n(\Omega)$. $R(\Omega)$ is thus a ranked set: every element has a unique arity. We similarly define the ranked set $F(\Omega)$ of finite diagrams as the disjoint union of the $F_n(\Omega)$.

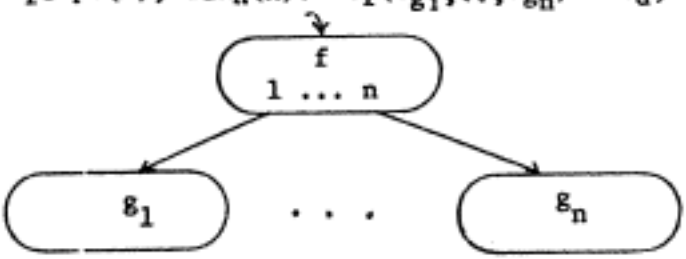
Consider the algebra A whose carrier is $E(\Omega)$ and which has for each $f \varepsilon \Omega_n$ an n-ary operation C_f given by $C_f(d_1,..,d_n)$



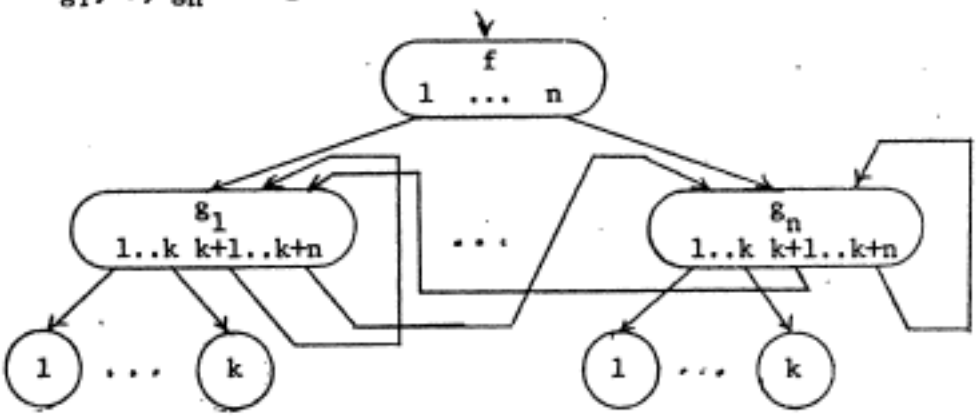
We will also have occasion to discuss C_d, where d is any n-ary flow diagram. $C_d(d_1,..,d_n)$ is the diagram formed by attaching to each exit of d numbered i a copy of diagram d_i. Since exits occur only at the end of finite branches this operation is always well-defined.

THEOREM 2: $\mu Cl(A) = \{C_d | d \in R(\Omega)\}$ (with the natural ranking)

PROOF: $\mu Cl(A) \subseteq R(\Omega)$: The proof follows the inductive construction of $\mu Cl(A)$. For each $C_f \in Ops(A)$, $f \in R_n(\Omega)$. $C_f(C_{g_1},..,C_{g_n}) = C_d$, where d is the diagram

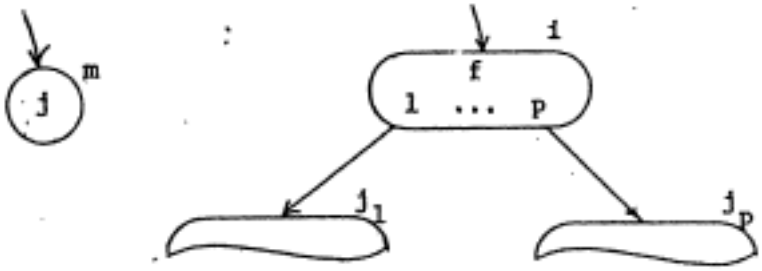


and $(C_f)^{\mu}(C_{g_1},..,C_{g_n}) = C_{d'}$, where d' is the diagram represented by



(where in the induction steps, f, gl,..,gn are all diagrams already defined in $\mu Cl(A)$).

$R(\Omega) \subseteq \mu Cl(A)$: Given a rational diagram d, take any loop representation of it. Number the boxes in the representation 1,..,n, including the exit nodes, and number the first box executed 1. Typical fragments of the flow chart look like:



The fixed point equations are

$$c_m = x_j$$

$$c_i = f(c_{j_1},..,c_{j_p})$$

So $C_d = (e_1^n)^{\mu}(g_1,..,g_n)$, where

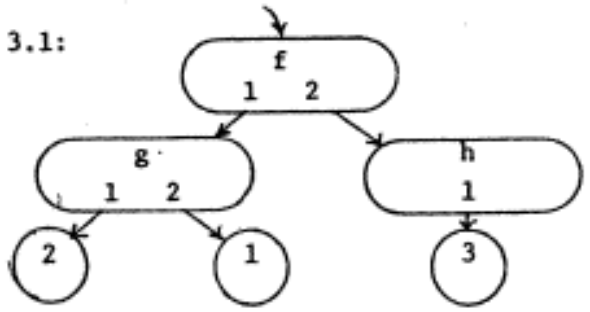
$$g_i = C_f(e_{k+j_1}^{k+n},..,e_{k+j_p}^{k+n})$$

or $g_m = e_j^{k+n}$

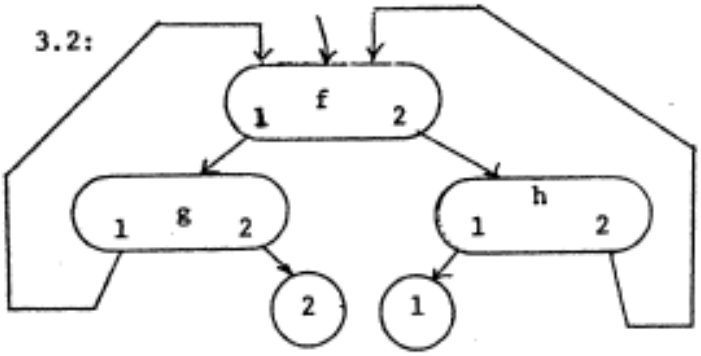
QED

Because of this bijection we can speak of $R(\Omega)$ as "the μ-clone of rational diagrams." It will be helpful to show some examples. Following this bijection, we will often write d for C_d.

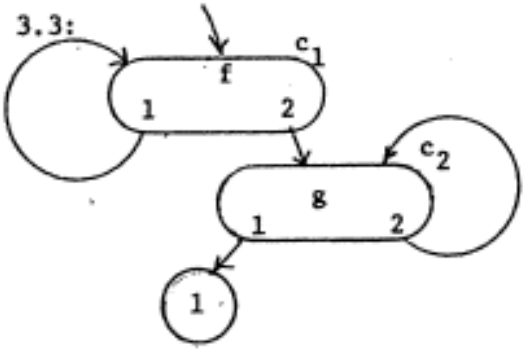
3.1:



3.2:



3.3:



EXAMPLE 3.1

$f(g(e_2^3,e_1^3),h(e_3^3))$

EXAMPLE 3.2 represents the diagram:

$(\mu d)[d = f(g(d,x_2),h(x_1,d))]$

The arguments to the fixed point equations are (x_1,x_2,c_1). So

$d = (e_1^1)^{\mu}(f(g(e_3^3,e_2^3),h(e_1^3,e_3^3)))$

EXAMPLE 3.3

The algorithm gives

$c_1 = f(c_1,c_2)$

$c_2 = g(x_1,c_2)$

The argument vector is (x_1,c_1,c_2), so

$d = (e_1^2)^{\mu}(f(e_2^3,e_3^3),g(e_1^3,e_3^3))$

By inspection, we also have

$d = ((e_1^1)^{\mu}(f(e_2^2,e_1^2)))((e_1^1)^{\mu}(g))$

In the last example, we see that, in general, there are many expressions for the same (not just equivalent) flow diagram. We will study this phenomenon in more detail in the next section.

We are now ready for our first major result:

THEOREM 3: $R(\Omega)$ is the free μ-clone generated by Ω, that is: if V is any μ-clone and j is a rank-preserving map $j:\Omega \to V$, then j extends uniquely to a μ-clone homomorphism $j^*:R(\Omega) \to V$.

PROOF: We extend j to j^* as follows:

i) if $f\in\Omega$, $j^*(f) = j(f)$; $j^*(\bot) = \bot$

ii) $j^*(e_i^n) = e_i^n$

iii) if $d_1,..,d_n\in F_k(\Omega)$ and $f\in\Omega_n$,
$j^*(f(d_1,..,d_n)) = (j(f))^V(j^*(d_1),..j^*(d_n))$

iv) if $d_1,..,d_i,..,\in R_k(\Omega)$, then
$j^*(\sqcup_i d_i) = \sqcup_i j^*(d_i)$

(i) - (iii) extend j to a well defined map on $F(\Omega)$. Further, $j^*|F(\Omega)$ is a clone homomorphism $F(\Omega) \to V$, ie, if $d_0\in F_n(\Omega)$, $d_1,..,d_n$ $F_R(\Omega)$, then $j^*(d_0(d_1,..,d_n)) = (j^*(d_0))^V(j^*(d_1),..j^*(d_n))$, since $F(\Omega)$ under composition is clearly isomorphic to the ranked set of Ω-trees under substitution, which is just the free clone generated by Ω. We note that each approximant (in the sense of

[Sc, Sec. 3]) to a rational diagram is in $F(\Omega)$ (ie, it has no embedded $\perp$'s). This is enough to show that j^* is well defined on $R(\Omega)$.

We must now show that j^* is a μ-clone homomorphism. From that it also follows that the image of j^* is in V. j^* is clearly rank preserving and preserves projections and $\perp$. It also preserves compositions on $F(\Omega)$. To show that it preserves composition in general, let $P_n(d)$ denote the n-th approximant of d. It is easy to see that if $d_i \in R(\Omega)$, then

$$P_i(d_0(d_1,..,d_n)) \sqsubseteq (P_i(d_0))(P_i d_1),..,P_i(d_n)) \sqsubseteq P_{2_i}(d_0(d_1,..,d_n))$$

This gives the second equality in

$$\begin{aligned} j^*(d_0(d_1,..,d_n)) &= j^*(\sqcup_i P_i(d_0(d_1,..,d_n))) \\ &= j^*(\sqcup_i((P_i(d_0))(P_i(d_1),..,P_i(d_n)))) \\ &= \sqcup_i j^*((P_i(d_0))(P_i(d_1),..,P_i(d_n))) \\ &= \sqcup_i (j^*(P_i(d_0)))(j^*(P_i(d_1)),..,j^*(P_i(d_n))) \\ &= (\sqcup_i j^*(P_i(d_0)))\ (\sqcup_i j^*(P_i(d_1)),..,\sqcup_i j^*(P_i(d_n))) \\ &= (j^*(\sqcup_i P_i(d_0)))^V(j^*(\sqcup_i P_i(d_1)),..,j^*(\sqcup_i P_i(d_n))) \\ &= (j^*(d_0))^V(j^*(d_1),..,j^*(d_n)) \end{aligned}$$

(The fifth equality follows from the fact that $\sqcup$ is a closure operation in $[L^k \to L]$). To show that j^* preserves fixed-point, for any clone V define a function $C:\{1,..,n\}\times\mathbb{N}\times(V_{k+n})^n \to V_k$ by induction on the second argument, as follows:

$$\begin{aligned} C(k,0,g_1,..,g_n) &= \perp \\ C(i,p+1,g_1,..,g_n) &= g_i(e_1^k,..,e_k^k,C(1,p,g_1,..,g_n),..,C(n,p,g_1,..,g_n)) \end{aligned}$$

Tarski's fixed point theorem then yields that

$$f^\mu(g_1,..,g_n) = f(\sqcup_p C(1,p,g_1,..,g_n),..,\sqcup_p C(n,p,g_1,..,g_n))$$

Furthermore, from the properties of j^* already known, it is easy to prove by induction on p that

$$j^*(C(i,p,d_1,..,d_n)) = C(i,p,j^*(d_1),..,j^*(d_n))$$

and

$$C(i,p,g_1,..,g_n) \sqsubseteq C(i,p+1,g_1,..,g_n)$$

So then

$$\begin{aligned} j^*(d_0^\mu(d_1,..,d_n)) &= j^*(d_0(\sqcup_p C(1,p,d_1,..,d_n),..,\sqcup_p C(n,p,d_1,..,d_n))) \\ &= (j^*(d_0))(j^*(\sqcup_p C(1,p,d_1,..,d_n)),..,j^*(\sqcup_p C(n,p,d_1,..,d_n))) \\ &= (j^*(d_0))(\sqcup_p j^*(C(1,p,d_1,..,d_n)),..,\sqcup_p j^*(C(n,p,d_1,..,d_n))) \\ &= (j^*(d_0))(\sqcup_p C(1,p,j^*(d_1),..,j^*(d_n)),..,\sqcup_p C(n,p,j^*(d_1),.., \\ &\qquad j^*(d_n))) \\ &= (j^*(d_0))^\mu(j^*(d_1),..,j^*(d_n)). \end{aligned}$$

Last, we must show j^* is unique. It will suffice to show that if h is a μ-clone homomorphism $R(\Omega) \to V$, then $h = (h|\Omega)^*$. This follows by an easy induction on the iterative construction of $\mu Cl(A)$. QED.

We are now able to prove theorems about any μ-clone by considering only the μ-clone of rational flow diagrams. For example, the Normal Form Lemma follows immediately from the construction of the second half of the proof of Theorem 2. (Compare, for example, the lengthy proof of [Bk]). In the next section we show

some of the power of this technique. We conclude this section with a technical result we will need to apply the theorem.

COROLLARY: Let RS denote the forgetful functor sending any μ-clone to the ranked set of its operations. Then there is a natural injective μ-clone homomorphism $j:V \to R(RS(V))$

4. TWO NORMAL FORM THEOREMS AND THEIR APPLICATIONS

In this section we apply Theorem 3 to get two more normal form theorems for the μ-clone of a lattice algebra. Our first theorem, which was also discovered by Bekic [Bk], is a "Currying Lemma" for the fixed-point operation: It says that any simultaneous iteration $(e^n)^\mu(f_1,..,f_n)$ can be replaced by a sequence of iterations involving only a single equation. Our second theorem strengthens this result by specifying that the only permissible functional compositions be of the form $g(t_1,..,t_n)$, where $g \in Ops(A)$. We conclude with a series of corollaries, including several classical theorems.

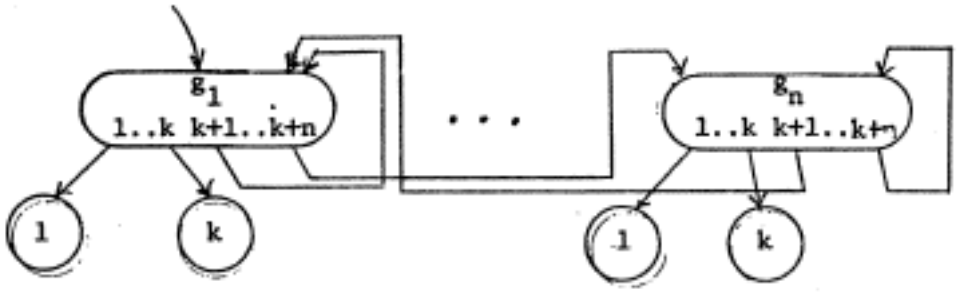
NOTATION: We write Y(f) for $(e_1^1)^\mu(f)$.

LEMMA 1: Let Ω be a ranked alphabet. Let $g_1,..,g_n \in \Omega_{k+n}$. Then in $R(\Omega)$

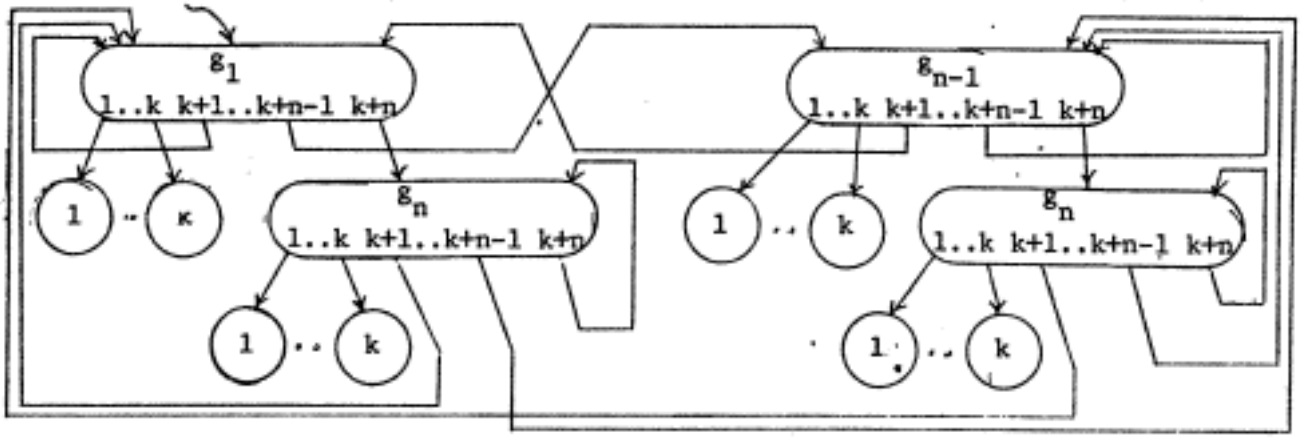
$$(e_1^p)^\mu(g_1,..,g_n) =$$

$$(e_1^{n-1})^\mu(g_1(e_1^{k+n-1},..,e_{k+n-1}^{k+n-1},Y(g_n)),..,g_{n-1}(e^{k+n-1},..,e_{k+n-1}^{k+n-1},Y(g_n)))$$

PROOF: The left-hand side represents the flow diagram represented by



and the right-hand side represents the flow diagram represented by



The second flowchart is obtained from the first by making separate copies of the g_n box for each of $g_1,..,g_{n-1}$. These both clearly represent the unique flow diagram in $R(\Omega)$ which starts with g_1 and which has the property that immediately below each node is the tuple of nodes marked exits $1,..,k,g_1,..,g_n$.

LEMMA 2: Let V be a μ-clone, $f_1,..,f_n \in V_{k+n}$. Then

$$(e_1^n)^{\mu}(f_1,..,f_n) =$$

$$(e_1^{n-1})^{\mu}(f_1(e_1^{k+n-1},..,e_{k+n-1}^{k+n-1}Y(f_n)),..,f_{n-1}(e^{k+n-1},..,e_{k+n-1}^{k+n-1},Y(f_n)))$$

PROOF: Let j be the natural insertion $V \to R(RS(V))$. Then j of the left-hand side equals j of the right-hand side, applying Lemma 1 in R(RS(V)). Then the lemma follows since j is an injection.

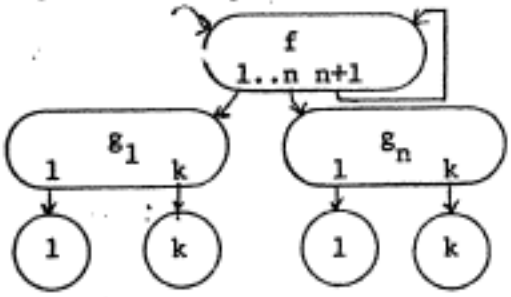
THEOREM 4: (Rekic) Let A be a lattice algebra, $f \in \mu Cl(A)$. Then there is an expression for f in which the only occurrences of $^{\mu}$ are of the form $(e_1^1)^{\mu}(g)$.

PROOF: Call an occurrence of μ <u>good</u> iff it is of the allowable form; call an element of $\mu Cl(A)$ good iff it has an expression in which every occurrence of $^{\mu}$ is good. If $g_1,..,g_n$ are good, then $h = (e_1^n)^{\mu}(g_1,..,g_n)$ is good, by induction on n: If n = 1, h is good by definition. If n = p+1, by Lemma 2 there exist good $f_1,..,f_p$ such that $h = (e_1^{p-1})^{\mu}(f_1,..,f_p)$, which is good by the induction hypothesis. Now by Theorem 1, for every $f \in \mu Cl(A)$ there exist $g_1,..,g_n$ which are good (indeed, they have no occurrences of $^{\mu}$) such that $f = (e_1^n)^{\mu}(g_1,..,g_n)$. QED.

LEMMA 3. Let $f \in \Omega_{n+1}, g_1,..,g_n \in \Omega_k$. Then in $R(\Omega)$

$$((Y(f))(g_1,..,g_n) = Y(f(g_1,..,g_n,e_{k+1}^{k+1}))$$

PROOF: Both expressions represent the same flow diagram as does



ie, the unique diagram, starting with f, such that the immediate descendents of each f are $g_1,..,g_n,f$, and the immediate descendents of each g_i are exits 1,..,k.

LEMMA 4. Let V be any μ-clone, $f \in V_{n+1}, g_1,..,g_n \in V_k$. Then

$$(Y(f))^V(g_1,..,g_n) = Y(f^V(g_1,..,g_n,e_{k+1}^{k+1}))$$

PROOF: Apply Lemma 3 in R(RS(V)).

THEOREM 5: Let A be a lattice algebra, $f \in V = \mu Cl(A)$. Then there is an expression for f in which the only occurrences of $^{\mu}$ are of the form $(e_1^1)^{\mu}(t)$ and the only occurrences of ^{V} are of the form $g^V(t_1,..,t_n)$ where $g \in Ops(A)$.

PROOF: For an expression t satisfying Theorem 4 and a subexpression t' of the form $f^V(g_1,..,g_n)$ let d(t') equal the number of occurrences of ^{V} or Y in f. Let $|t|$ denote the sum of the d(t') for all the subexpressions t' of the form $f^V(g_1,..,g_n)$. Then the following two rules send t to an expression t' equivalent to t and such that $|t'| < |t|$.

RULE 1: Replace a subexpression of the form $((Y(f))^V(g_1,..,g_n)$ (where $g_i \in V_k$) by $Y(f^V(g_1,..,g_n,e_{k+1}^{k+1}))$

RULE 2: Replace a subexpression of the form $(f^V(g_1,..,g_n))^V(h_1,..,h_k)$ by $f^V(g_1{}^V(h_1,..,h_k),..,g_n{}^V(h_1,..,h_k))$

These preserve the value of t by Lemma 4 and by the definition of $'$. By the previous observation, any sequence of applications of Rules 1 and 2 must terminate, leaving an expression in which each left-hand side of a ^{V} is either an operation in A or a projection. So now apply

RULE 3: Replace a subexpression of the form $e_i^n(g_1,\ldots,g_n)$ by g_i.

Since Rule 3 is length-decreasing it too must terminate, leaving an expression which satisfies the theorem. QED.

DEFINITION: An expression satisfying Theorem 5 is said to be quasi-regular.

COROLLARY 1 (Kleene): The regular sets and languages generated by right-linear grammars coincide.

PROOF: Apply Theorem 4 to the μ-clone generated by the algebra whose operations on the complete lattice of Σ-languages (L of Examples 2.1 and 2.2) are $S \mapsto aS$ for each $a \in \Sigma$.

COROLLARY 2: The context-free languages are closed under substitution, symbol iteration, and nested iterative substitution.

COROLLARY 3 (McWhirter [McW]): Every context-free language is obtainable from the singletons by substitution, union, and symbol iteration.

PROOF: Apply Theorem 4 to $\mu Cl(ACF)$.

COROLLARY 4 (Gruska [Gr]): Every context-free language is obtainable from the singletons by concatenation, union, and symbol iteration.

PROOF: Apply Theorem 5 to $\mu Cl(ACh)$.

COROLLARY 5 (Engeler [En]): For every flowchart program there is a side-effect equivalent flowchart program in "normal form".

PROOF: Apply Theorem 4 to $R(\Omega)$.

REFERENCES

[Bk] Bekic, H. Definable Operations in General Algebras, and the Theory of Automata and Flowcharts. IBM Vienna, 1969.

[C] Cohn, P.M. Universal Algebra, Van Nostand, Princeton, 1965.

[E] Elgot, C.C., Algebraic Theories and Program Schemes, IBM Yorktown Heights RC 2925 (1970).

[En] Engeler, E. The Structure and Meaning of Elementary Programs, Symp. on Semantics of Algorithmic Languages (E. Engeler, ed.) Springer, Berlin (1971), 89-102.

[Fi] Fischer, M.J. Grammars with Macro-Like Productions, Proc. 9th IEEE Conf Sw & Auto Th (1968), 131-142.

[Gr] Gruska, J. A Characterization of Context-Free Languages, J. Comp Sys Sci 5 (1971), 353-364.

[Hi] Higgins, P.J. Algebras with a Scheme of Operators, Math Nachr. 27 (1963), 115-132.

[McW] McWhirter, I.P., Substitution Expressions, J. Comp Sys Sci 5 (1971), 629-637.

[M-W] Mezei, J. and Wright, J.B., Algebraic Automata and Context-Free Sets, Info & Contr 11 (1967), 3-29.

[Sc] Scott, D. The lattice of Flow Diagrams, Oxford U. Comp. Lab. Prog. Res. Grp. Tech. Monograph PRG-3, 1970.

[T] Tarski, A. A Lattice-Theoretical Fixpoint Theorem and its Applications. Pacific J. Math 5 (1955), 285-309.

[Wg-1] Wagner, E. An Algebraic Theory of Recursive Definitions and Recursive Languages. Proc 3rd ACM Symp Th Comp (1971), 12-23.

[Wg-2] ----------, Languages for Defining Sets in Arbitrary Algebras, Proc 11th IEEE Conf Sw & Auto Th (1971) 192-201.

[Wa] Wand, M., The Elementary Algebraic Theory of Generalized Finite Automata, Notices AMS Abstract #72T-C13, February 1972.

CORRIGENDUM:

The Corollary to Theorem 3 should read: "There is a natural map $j:V \to R(RS(V))$ and a natural uclone homomorphism $k: R(RS(V)) \to V$ such that $k(j(x)) = x$."

In Section 4, the proof of Lemma 2 should read:

"Let $LHS(t_1,..,t_n)$ and $RHS(t_1,..,t_n)$ denote the left-hand and right-hand expressions with t_i substituted for f_i. Then

$$LHS(f_1,..,f_n) = LHS(k(j(f_1)),..,k(j(f_n))) \quad \text{(by the Corollary to Thm 3)}$$

$$= k(LHS(j(f_1),..,j(f_n)) \quad \text{(since k is a homomorphism)}$$

$$= k(RHS(j(f_1),..,j(f_n)) \quad \text{(by Lemma 1 in R(RS(V)))}$$

$$= RHS(k(j(f_1)),..,k(j(f_n)))$$

$$= RHS(f_1,..,f_n)"$$

The proof of Lemma 3 should refer to this proof.

NOTE:

In Section 1 we said "It is instructive to see how much can be developed without the explicit introduction of category theory." Subsequent developments have convinced us that the results developed herein represent close to the limit of what can be done without the explicit categorization. We expect that any extension of these notions will indeed require categorical language. In particular, our Theorem 3 is a special case of a much more general theorem, which is proven much more easily!