Notre Dame Journal of Formal Logic
Volume 24, Number 2, April 1983

Cylindrical Decision Problems for System Functions

M. B. THURAISINGHAM*

1 While investigating the one-one equivalence between various General Combinatorial decision problems, Cleave [1] initiated the concept of system functions. System functions are defined on natural numbers and their values are finite sets of natural numbers. They have many properties in common with those arising from Gödel numbering various combinatorial systems. Thus the decision problems defined for these combinatorial systems can also be defined for system functions. Furthermore, if a property holds for a particular decision problem for all system functions, then it also holds for that decision problem for all these combinatorial systems.

In his study of the one-one equivalence between General Combinatorial decision problems using system functions, Cleave [1] considered only a finite number of decision problems. We have extended his study to include an infinite number of decision problems. This is accomplished firstly, by defining a generalized class of formulas in terms of a first-order language so that each formula in this class corresponds to a decision problem for system functions and secondly, by analyzing these formulas to determine whether the corresponding decision problems for various kinds of system functions are cylinders. As stated by Cleave [1], we take this approach for the following reason: "If P_1 and P_2 are two General Combinatorial decision problems which are many-one equivalent and if each instance of P_1 and P_2 are cylinders, then they are one-one equivalent" ([1], p. 254). This is the best possible equivalence one can obtain.

*The author wishes to thank Dr. J. R. Hindley of the Department of Pure Mathematics, University College of Swansea, U.K. and Dr. J. P. Cleave of the Department of Pure Mathematics, University of Bristol, U.K. for their guidance.

Received May 21, 1981

In Section 3, we define this generalized class of formulas (denoted GCF) and also choose an infinite subclass of GCF and show that all the decision problems (for a certain kind of system functions) which correspond to the formulas of this subclass are either recursive or cylinders. The preliminary definitions are given in Section 2.

2 In this section we define system functions as in [1] and also develop a first-order language L which will be used to define the class GCF in Section 3.

Let $f:N \to P_w(N)$ where N is the set of all natural numbers and $P_w(N)$ is the set of all finite subsets of N.

For each $X \in P_w(N)$, define $f(X) = \bigcup \{f(x):x \in X\}$. For each $x \in N$, define $f^0(x) = x$, $f^1(x) = f(x)$, $f^{m+1}(x) = f(f^m(x))$.

Define $f^{-1}(x) = \{y:x \in f(y)\}$. By $y \in C_f x$ (or $x \in C_{f^{-1}} y$) is meant: $y = x$ or $y \in f(x)$ or there exist $v_1, v_2, \ldots, v_n$ $(n \geq 1)$ such that $x = v_1$, $y = v_n$ and for each $i(1 \leq i \leq n-1)$, $v_{i+1} \in f(v_i)$.

By the expression $\bigvee\limits_{i=1}^{n-1} x_{i-1} \in C_f x_i$ we mean: $x_2 \in C_f x_1 V x_3 \in C_f x_2 V, \ldots, V x_n \in C_f x_{n-1}$.

A system function is a function $f:N \to P_w(N)$ such that there exist recursive functions a and b such that for all x, $f(x) = D_{a(x)}$ and $f^{-1}(x) = D_{b(x)}$ where D_n is the n^{th} finite set in some standard enumeration.

A system function f which has the property that for each x, $f(x)$ has at most one member is called a machine function. Clearly system functions that arise from combinational systems such as Turing Machines and Markov Algorithms are machine functions. The class of all system functions is denoted by $\mathfrak{S}$ and the class of all machine functions is denoted by $\mathfrak{M}$. Some of the decision problems for a system function f are as follows:

halting problem for $f = \{x:(Ey)(y \in C_f x \wedge f(y) = \Phi)\}$
derivability problem for $f = \{(x, y):y \in C_f x\}$
confluence problem for $f = \{(x, y):(Ez)(z \in C_f x \wedge z \in C_f y)\}$.

We now define a first-order language L as follows: The logical symbols of L are $\wedge$, $\vee$, $\sim$, E, and parentheses.

(i) a denumerably infinite set of variables
(ii) a constant $\underline{n}$ for each $n \in N$
(iii) a two-place predicate R and a one-place predicate Q.

A term is either a variable or a constant.

For each $f \in S$, define a structure N_f as follows: The domain of N_f is N. R^f and Q^f are relations on $N \times N$ and N, respectively, where $R^f = \{(m_1, m_2):m_2 \in C_f m_1\}$. $Q^f = \{m:f(m) = \Phi\}$. Clearly R^f is recursively enumerable and Q^f is recursive.

Satisfiability of a sentence of L in N_f is defined as follows: For each $k_1, k_2 \in N$,

$$N_f \models R(\underline{k_1}, \underline{k_2}) \longleftrightarrow (k_1, k_2) \in R^f$$
$$N_f \models Q(\underline{k_1}) \longleftrightarrow (k_1 \in Q^f).$$

Satisfiability of compound sentences is defined in the usual 'Tarski' sense.

The following specialized definitions and notations also are needed in Section 3: Let F be a formula of L whose free variables are $x_1, x_2, \ldots, x_n$. F is also denoted by $F(x_1, x_2, \ldots, x_n)$. F^f is the set of all n-tuples which satisfy $F(x_1, x_2, \ldots, x_n)$ in N_f; i.e., for any n-tuple $(\underline{k}_{x_1}, \underline{k}_{x_2}, \ldots, \underline{k}_{x_n})$,

$$(k_{x_1}, k_{x_2}, \ldots, k_{x_n}) \in F^f \longleftrightarrow N_f \models F(k_{x_1}, k_{x_2}, \ldots, k_{x_n}).$$

In the n-tuple $(k_{x_1}, k_{x_2}, \ldots, k_{x_n})$, the subscripts $x_1, x_2, \ldots, x_n$ are used merely for convenience.

Let $G \subset$ S. Then a formula F has a property P in G if and only if F_f has the property P for all $f \in G$. For example, F is a cylinder [nonsimple respectively] in G if and only if F^f is a cylinder [nonsimple respectively] for all $f \in G$. By a nonrecursive and noncylindrical [simple respectively] counterexample for F in G we mean there exists an $f \in G$ such that F^f is a nonrecursive nonclyinder [simple respectively].

A formula F corresponds to a decision problem D for the system functions of $G \subset S$ if and only if $F^f = \overline{D}[f]$ for all $f \in G$, where $\overline{D}[f]$ is the decision problem D for f.

A V-loop is a formula of the form $\bigvee_{i=1}^{n-1} R(x_i, x_{i+1}) VR(x_n, x_1)$ where $n \geqslant 2$, $x_1, x_2, \ldots, x_n$ are all variables and $\bigvee_{i=1}^{n-1} R(x_i, x_{i+1}) \equiv R(x_1, x_2) VR(x_2, x_3) V, \ldots, VR(x_{n-1}, x_n)$.

A V-path between x and y or a V-path from x to y is any formula F where F is either $R(x, y)$ or $R(x, z)VR(z, y)$ or $R(x, x_1) \vee \bigvee_{i=1}^{n-1} R(x_i, x_{i+1}) VR(x_n, y)$ where $n \geqslant 2, z, x_1, x_2, \ldots, x_n$ are variables. If x and y are variables, then F is a V-path between two variables. If x and y are constants, then F is a V-path between two constants. If x is a constant and y is a variable, then F is a V-path from a constant to a variable. If x is a variable and y is a constant, then F is a V-path from a variable to a constant.

By a $\wedge, \vee$ combination of $R, Q, \sim Q$ we mean any formula built from the formulas of the form $R(x, y)$, $R(x, \underline{a})$, $R(\underline{b}, x)$, $Q(x)$, $\sim Q(x)$ using the connectives $\wedge, \vee$ where x, y are variables and $\underline{a}, \underline{b}$, are constants.

Similarly a V-combination of R is any formula built from the formulas of the form $R(\underline{a}, x)$, $R(x, \underline{b})$, $R(x, y)$ using the connective $\vee$ where x, y are variables and $\underline{a}, \underline{b}$ are constants. If F is any formula of L, then EF is a formula of L of the form $(Ex_1, x_2, \ldots, x_n)F$ where $n \geqslant 1, x_1, x_2, \ldots, x_n$ are among the free variables of F and E is an existential quantifier.

Let F be any V-combination of R. Then $k \rightarrowtail m$ is a subgraph of the graphical representation of F if and only if $R(k, m)$ is a subformula of F. For example the graphical representation of the formula

$$R(x_1, x_2)VR(x_2, x_3)VR(x_1, x_3)$$

is shown in Figure 1.

Unless otherwise stated, $x, y, z, x_1, y_1, z_1, x_2, y_2, z_2, \ldots$ denote variables and $\underline{a}, \underline{b}, \underline{c}, \underline{a}_1, \underline{b}_1, \underline{c}_1, \underline{a}_2, \underline{b}_2, \underline{c}_2, \ldots$ denote constants.

3 We now define the generalized class of formulas mentioned in Section 1.

Definition Generalized class of formulas (or GCF) contains any formula of

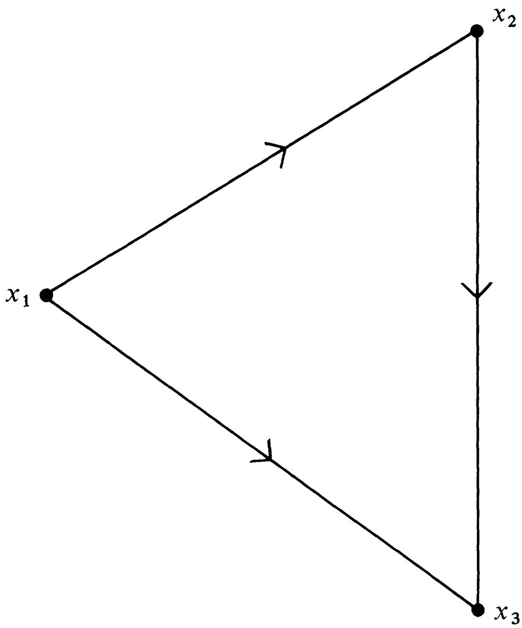

Figure 1

the form K or EK where K is a $\wedge$, $\vee$ combination of R, Q, $\sim Q$ which does not contain sentences and formulas of the form $R(x, x)$ as subformulas.

We assume that no member of GCF contains sentences and formulas of the form $R(x, x)$ as subformulas because the inclusion of such formulas will not give us any additional information. We have used only the predicates R and Q in defining the language L because: (i) we have restricted ourselves to recursively enumerable decision problems, and (ii) the predicates R and Q are sufficient in defining decision problems such as the halting, derivability and confluence problems. For example, the formula $(Ey)[R(x, y) \wedge Q(y)]$ corresponds to the halting problem.

In Result (α), to be stated later, we choose an infinite subclass of GCF and show that all the decision problems for machine functions which correspond to the formulas of this subclass are either recursive or cylinders. The treatment of other subclasses of GCF will utilize the same technique as the proof of Result α which exhibits the essential points of our argument. We have restricted ourselves to machine functions because: (i) it is possible to give nonrecursive and noncylindrical counterexamples in $\mathfrak{S}$ even for elementary formulas such as $R(x, y)$, $R(x, \underline{a})$, $R(\underline{a}, x)$ [2]; and (ii) among the various kinds of combinatorial systems, it is Turing Machines that are universally applied and the system functions that arise from Turing Machines are machine functions.

Result (α) Each formula in the class A_1 is either recursive or a cylinder in $\mathfrak{M}$ where A_1 consists of all members of GCF which are V-combinations of R and which do not contain V-loops and formulas of the form $R(\underline{a}, x)$ as subformulas.

In order to prove Result (α), we need the following result:

Result (β) Each formula in the class A_1 is nonsimple in $\mathfrak{M}$.

We will, however, prove the following stronger Result (γ).

Result (γ) Each formula in the class A_2 is nonsimple in $\mathfrak{S}$ where A_2 consists of all members of GCF which are V-combinations of R and which do not contain V-paths between two constants and V-loops as subformulas.

We have excluded formulas which contain V-loops or formulas of the form $R(\underline{a}, x)$ as subformulas as members of A_1. Furthermore, we have also excluded formulas which contain V-paths between two constants or V-loops as subformulas as members of A_2.

This is because nonrecursive and noncylindrical counterexamples in $\mathfrak{M}$ can be given for the formulas $R(\underline{a}, x)$, $R(x, y)VR(y, x)$ and simple counterexamples in $\mathfrak{S}$ can be given for the formulas $R(\underline{a}, x)VR(x, y)VR(y, \underline{b})$, $R(x, y)VR(y, x)$. Detailed proofs of these counterexamples can be obtained in [2].

Proof of γ: Let $F \Xi F(x_1, x_2, \ldots, x_n)$ belong to A_2 and $x_1, x_2, \ldots, x_n$ be all its variables. Let $y_1, y_2, \ldots, y_m$ be all the variables among $x_1, x_2, \ldots, x_n$ such that for each $i(i \leqslant i \leqslant m)$, F contains a subformula which is a V-path from y_i to a constant. Let $z_1, z_2, \ldots, z_k$ be all the variables among $x_1, x_2, \ldots, x_n$ such that for each $i(1 \leqslant i \leqslant k)$ there does not exist a $j(1 \leqslant j \leqslant m)$ such that $z_i = y_j$ and for each $i(1 \leqslant i \leqslant k)$ either F contains a subformula which is a V-path from a constant to z_i or there exists a variable, say w, such that $R(w, z_i)$ is a subformula of F and $w \notin \{y_1, y_2, \ldots, y_m\}$. Let

$$\{u_1, u_2, \ldots, u_t\} = \{x_1, x_2, \ldots, x_n\} - \{\{y_1, y_2, \ldots, y_m\} \cup \{z_1, z_2, \ldots, z_k\}\}.$$

For example, if the graphical representation of F is as shown in Figure 2, where $x_1, x_2, x_3, \ldots, x_9$ are variables and $\underline{a}, \underline{b}$ are constants then $\{y_1, y_2, y_3\} = \{x_1, x_2, x_3\}$, $\{u_1, u_2\} = \{x_4, x_9\}$, and $\{z_1, z_2, z_3, z_4\} = \{x_5, x_6, x_7, x_8\}$.

As F does not contain V-paths between two constants and V-loops as subformulas, the following Conditions I-III hold:

Condition I. For each $i(1 \leqslant i \leqslant m)$, there does not exist a constant, say $\underline{a}$, such that $R(\underline{a}, y_i)$ is a subformula of F. If there exists a variable v such that $R(v, y_i)$ is a subformula of F, then $v \in \{y_1, y_2, \ldots, y_m\}$. There exists a term, say u, such that $R(y_i, u)$ is a subformula of F.

Condition II. For each $i(1 \leqslant i \leqslant k)$, there does not exist a constant, say $\underline{a}$, such that $R(z_i, \underline{a})$ is a subformula of F. If there exists a variable v such that $R(z_i, v)$ is a subformula of F, then $v \in \{z_1, z_2, \ldots, z_k\}$. There exists a term, say u, such that $R(u, z_i)$ is a subformula of F.

Condition III. For each $i(1 \leqslant i \leqslant t)$, there does not exist a constant, say $\underline{a}$, such that either $R(u_i, \underline{a})$ or $R(\underline{a}, u_i)$ is a subformula of F. If there exists a variable v such that $R(v, u_i)$ is a subformula of F, then $v \in \{y_1, y_2, \ldots, y_m\}$. If there exists a variable w such that $R(u_i, w)$ is a subformula of F, then $w \in \{z_1; z_2, \ldots, z_k\}$. There exists a variable u such that either $R(u_i, u)$ or $R(u, u_i)$ (but not both) is a subformula of F.

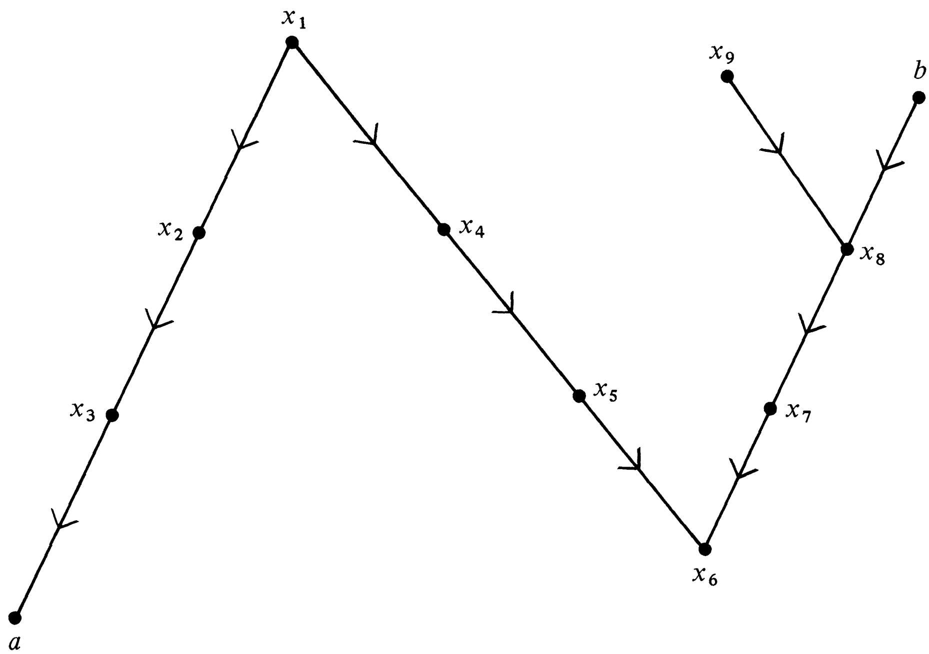

Figure 2

It needs to be proved that F^f is nonsimple for each $f \in \mathfrak{S}$. If F^f is recursive for an $f \in \mathfrak{S}$, then it is nonsimple. Suppose F^f is nonrecursive for an $f \in \mathfrak{S}$. Then the following statement * holds.

* There exists an n-tuple of numbers $(e_{x_1}, e_{x_2}, \ldots, e_{x_n})$ which is a member of F^f such that at least one of

$$C_f e_{y_1}, C_f e_{y_2}, \ldots, C_f e_{y_m}, C_{f^{-1}} e_{z_1}, C_{f^{-1}} e_{z_2}, \ldots, C_{f^{-1}} e_{z_k}$$

is infinite.

Now, if * does not hold, a decision procedure can be given for F^f as follows: Given an n-tuple $(q_{x_1}, q_{x_2}, \ldots, q_{x_n})$, enumerate

$$C_f q_{y_1}, C_f q_{y_2}, \ldots, C_f q_{y_m}, C_{f^{-1}} q_{z_1}, C_{f^{-1}} q_{z_2}, \ldots, C_{f^{-1}} q_{z_k}.$$

If one of

$$C_f q_{y_1}, C_f q_{y_2}, \ldots, C_f q_{y_m}, C_{f^{-1}} q_{z_1}, C_{f^{-1}} q_{z_2}, \ldots, C_{f^{-1}} q_{z_k}$$

is infinite, then as * does not hold

$$(q_{x_1}, q_{x_2}, \ldots, q_{x_n}) \in F^f.$$

Therefore one of the two Conditions a1 or a2 holds:

Condition a1. There exists an $i (1 \leqslant i \leqslant m)$ such that $R(y_i, w)$ is a subformula of F where w is a term and $q_w \in C_f q_{y_i}$ if w is a variable, $a \in C_f q_{y_i}$ if w is a constant, say $\underline{a}$.

Condition a2. There exists an $i(i \leqslant i \leqslant k)$ such that $R(w, z_i)$ is a subformula of F where w is a term and $q_{z_i} \in C_f q_w$ if w is a variable, $q_{z_i} \in C_f a$ if w is a constant, say $\underline{a}$.

It can be seen that which ever condition holds, at some finite stage in the enumeration of

$$C_f q_{y_1}, C_f q_{y_2}, \ldots, C_f q_{y_m}, C_f{}^{-1} q_{z_1}, C_f{}^{-1} q_{z_2}, \ldots, C_f{}^{-1} q_{z_k},$$

it can be verified that $(q_{x_1}, q_{x_2}, \ldots, q_{x_n}) \in F^f$. If $C_f q_{y_1}, C_f q_{y_2}, \ldots, C_f q_{y_m}$, $C_f{}^{-1} q_{z_1}, C_f{}^{-1} q_{z_2}, \ldots, C_f{}^{-1} q_{z_k}$ are all finite then all the members in each of the sets $C_f q_{y_1}, C_f q_{y_2}, \ldots, C_f q_{y_m}, C_f{}^{-1} q_{z_1}, C_f{}^{-1} q_{z_2}, \ldots, C_f{}^{-1} q_{z_k}$ are known. Therefore for any subformula $R(v, w)$ of F, it can be decided whether $q_w \in C_f q_v$ if v and w are variables, $q_w \in C_f a$ if w is a variable and v is the constant a, $a \in C_f v$ if v is a variable and w is the constant $\underline{a}$. Thus it can be decided whether $(q_{x_1}, q_{x_2}, \ldots, q_{x_n}) \in F^f$. But, as F^f is nonrecursive, there cannot exist a decision procedure for it. Therefore the statement * holds.

Let $(p_{x_1}, p_{x_2}, \ldots, p_{x_n})$ belong to F^f and one of $C_f p_{y_1}, C_f p_{y_2}, \ldots, C_f p_{y_m}$, $C_f{}^{-1} p_{z_1}, C_f{}^{-1} p_{z_2}, \ldots, C_f{}^{-1} p_{z_k}$ be infinite. Then there exists a number $t(1 \leqslant t \leqslant m + k)$ such that one of the following Conditions b1 or b2 holds.

Condition b1 $C_f p_{y_t}$ is infinite and if $w_1, w_2, \ldots, w_s$ are all the variables among $x_1, x_2, \ldots, x_n$ such that for each $i(1 \leqslant i \leqslant s)$, $R(w_i, y_t)$ is a subformula of F, then $C_f q_{w_1}, C_f q_{w_2}, \ldots, C_f q_{w_s}$ are all finite.

Condition b2 $C_f{}^{-1} p_{z_t}$ is infinite and if $w_1, w_2, \ldots, w_s$ are all the variables among $x_1, x_2, \ldots, x_n$ such that for each $i(1 \leqslant i \leqslant s)$, $R(w_i, y_t)$ is a subformula of F, then $C_f{}^{-1} q_{w_1}, C_f{}^{-1} q_{w_2}, \ldots, C_f{}^{-1} q_{w_s}$ are all finite.

Let $\bar{t}$ be the least such number t such that either b1 or b2 holds. Suppose Condition b1 holds. Then $K = K_1 \times K_2 \times \ldots, K_n$ is an infinite recursively enumerable subset of $\overline{F^f}$ where for each $i(1 \leqslant i \leqslant n)$:

$$K_i = \{q_{x_i}\} \text{ if } x_i \neq y_{\bar{t}}$$
$$K_i = C_f q_{y_{\bar{t}}} - \bigcup_{j=1}^{s} C_f q_{w_j} \text{ if } x_i = y_{\bar{t}}.$$

For, if K is not a subset of $\overline{F^f}$, then there exists a member of K belonging to F^f. This can happen only if one of the following two Conditions C1 or C2 holds.

Condition C1. There exists a member d of $C_f p_{y_{\bar{t}}} - \bigcup_{j=1}^{s} C_f p_{w_j}$ and an $i(1 \leqslant i \leqslant s)$ such that $d \in C_f p_{w_i}$.

Condition C2. There exists a member d of $C_f p_{y_{\bar{t}}} - \bigcup_{j=1}^{s} C_f p_{w_j}$ and a term u such that $R(y_{\bar{t}}, u)$ is a subformula of F and $p_u \in C_f d$ if u is a variable, $a \in C_f d$ if u is a constant, say $\underline{a}$.

Condition C1 cannot hold as $d \in C_f p_{y_{\bar{t}}} - \bigcup_{j=1}^{s} C_f p_{w_j}$ implies $d \notin C_f p_{w_j}$ for each $j(1 \leqslant j \leqslant s)$.

Suppose Condition C2 holds. If u is a variable, then as $d \in C_f p_{y_{\bar{t}}} - \bigcup_{j=1}^{s} C_f p_{w_j}$ and $p_u \in C_f d$, we have that $p_u \in C_f p_{y_{\bar{t}}}$. If u is a constant $\underline{a}$, then as

$d \in C_f p_{y_{\bar{t}}} - \bigcup_{j=1}^{s} C_f p_{w_j}$ and $a \in C_f d$, we have that $a \in C_f p_{y_{\bar{t}}}$. As $R(y_{\bar{t}}, u)$ is a subformula of F, this means that $(P_{x_1}, P_{x_2}, \ldots, P_{x_n}) \in F^f$. This is a contradiction. Therefore Condition C2 cannot hold.

Thus K is a subset of $\overline{F^f}$. Furthermore, as $\bigcup_{j=1}^{s} C_f p_{w_j}$ is finite and $C_f p_{y_t}$ is infinite and recursively enumerable, $C_f p_{y_{\bar{t}}} - \bigcup_{j=1}^{s} C_f p_{w_j}$ is infinite and recursively enumerable. Therefore K is an infinite recursively enumerable subset of $\overline{F^f}$.

Suppose Condition b2 holds. Then by a similar argument it can be shown that $B = B_1 \times B_2 \times \ldots, B_n$ is an infinite recursively enumerable subset of $\overline{F^f}$ where for each $i (1 \leqslant i \leqslant n)$,

$$B_i = \{x_i\} \text{ if } x_i \neq z_{\bar{t}}$$
$$B_i = C_f{}^{-1} p_{z_{\bar{t}}} - \bigcup_{j=1}^{s} C_f{}^{-1} p_{w_j} \text{ if } x_i = z_{\bar{t}}.$$

Therefore F^f is nonsimple.

This proves Result (γ).

Proof of (α): We first need the following result due to Young [3]. A set P is a cylinder if and only if there exists a recursive function g such that for all m, $m \in P \Rightarrow W_{g(m)} \subseteq P$, $m \in \overline{P} \Rightarrow W_{g(m)} \subseteq \overline{P}$ and $W_{g(m)}$ is infinite where W_n is the n^{th} recursively enumerable set in some standard enumeration.

We will now prove Result (α). Let $F \equiv F(x_1, x_2, \ldots, x_n)$ belong to A_1 and $x_1, x_2, \ldots, x_n$ be all its variables. A variable x of F is placed in level 1 if there does not exist a variable y such that $R(y, x)$ is a subformula of F. As F does not contain V-loops and formulas of the form $R(\underline{a}, x)$ as subformulas, there exists a variable in level 1.

A variable x is placed in level $i \geqslant 2$ if there is a term u such that $R(x, u)$ is a subformula of F and the maximum number of occurrences of R in a V-path from a variable in level 1 to x is $i - 1$. For example, in the formula $R(x_1, x_2) V R(x_2, x_4) V R(x_3, x_4) V R(x_4, x_5)$, x_1 and x_3 are in level 1, x_2 is in level 2, and x_4 is in level 3. The graphical representation of this formula is shown in Figure 3.

Assume that the number of levels is t. For each $i (1 \leqslant i \leqslant t)$, let $y_{i1}, y_{i2}, \ldots, y_{ih_i}$ be all the variables in level i. For each $(1 \leqslant i \leqslant t)$ and $j (1 \leqslant j \leqslant h_i)$, let $y_{ij}^1, y_{ij}^2, \ldots, y_{ij}^{k_j}$ be all the variables and $\underline{a}_{ij}^1, \underline{a}_{ij}^2, \ldots, \underline{a}_{ij}^{p_j}$ be all the constants such that for each r and s $(1 \leqslant r \leqslant k_j, 1 \leqslant s \leqslant p_j)$, $R(y_{ij}, y_{ij}^r)$ and $R(y_{ij}, \underline{a}_{ij}^s)$ are subformulas of F. Note that

$$\bigcup_{i=1}^{t} \{y_{i1}, y_{i2}, \ldots, y_{ih_i}\} \cup \bigcup_{r=1}^{t} \cup \bigcup_{j=1}^{h_r} \{y_{rj}^1, y_{rj}^2, \ldots, y_{rj}^{k_j}\} = \{x_1, x_2, \ldots, x_n\}.$$

We need to prove that F^f is either recursive or a cylinder for each $f \in M$. Suppose F^f is nonrecursive for an $f \in M$. Then from the result (γ), $\overline{F^f}$ contains an infinite recursively enumerable subset, say B^f. It suffices to prove the following statement (δ).

(δ) There exists a recursive function g such that for any n-tuple $(m_{x_1}, m_{x_2}, \ldots, m_{x_n})$,

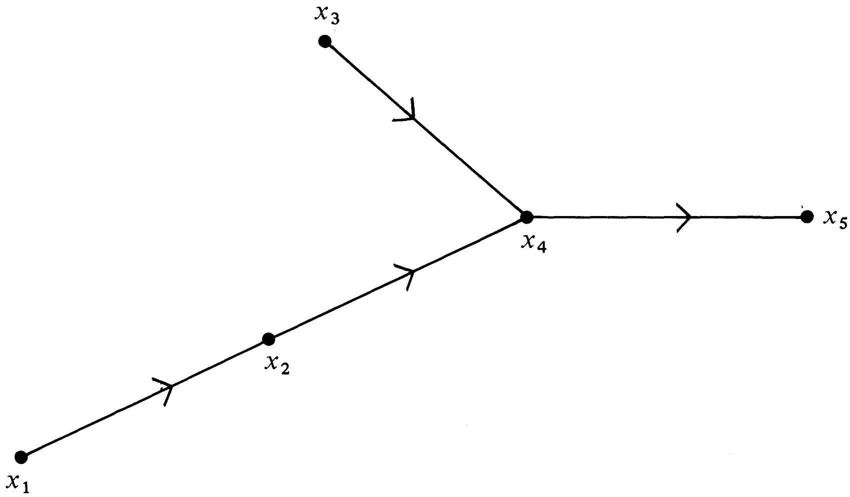

Figure 3

$(m_{x_1}, m_{x_2}, \ldots, m_{x_n}) \in F^f \Rightarrow W_{g(m_{x_1}, m_{x_2}, \ldots, m_{x_n})} \subseteq F^f$
$(m_{x_1}, m_{x_2}, \ldots, m_{x_n}) \in \overline{F^f} \Rightarrow W_{g(m_{x_1}, m_{x_2}, \ldots, m_{x_n})} \subseteq \overline{F^f}$
and $W_{g(m_{x_1}, m_{x_2}, \ldots, m_{x_n})}$ is infinite.

For, if (δ) holds, then by Young's result, F^f is a cylinder.

Proof of (δ): For a given n-tuple $(m_{x_1}, m_{x_2}, \ldots, m_{x_n})$, while executing the programme to be given below, an infinite list $L(m_{x_1}, m_{x_2}, \ldots, m_{x_n})$ will be constructed. For convenience, this list will be denoted by $L_{m_{\bar{x}}}$. Also in this programme, by the statement "Place $(m_{x_1}, m_{x_2}, \ldots, m_{x_n})$ in $L_{m_{\bar{x}}}$ with k in place of m_{x_j}" we mean the following: If $m_{x_j} = k$, place $(m_{x_1}, m_{x_2}, \ldots, m_{x_n})$ in $L_{m_{\bar{x}}}$. If $m_{x_j} \neq k$, place $(m_{x_1}, m_{x_2}, \ldots, m_{x_{j-1}}, k, m_{x_{j+1}}, \ldots, m_{x_n})$ in $L_{m_{\bar{x}}}$.

For each $i(1 \leqslant i \leqslant t)$ and $j(1 \leqslant j \leqslant h_i)$, define

$$T_{ij} = \{m_{y_{ij}^1}, m_{y_{ij}^2}, \ldots, m_{y_{ij}^{k_i}}, a_{ij}^1, a_{ij}^2, \ldots, a_{ij}^{p_j}\}.$$

Programme Start with Stage 0 of statement S_1^1 where Stage $r(r \geqslant 0)$ of statement $S_i^j (1 \leqslant i \leqslant t,\ 1 \leqslant j \leqslant h_i)$ is as follows:

S_i^j *Stage* $r(r \geqslant 0)$. Compute $f^r(m_{y_{ij}})$. (i) If $f^r(m_{y_{ij}}) = \Phi$ or $f^r(m_{y_{ij}}) = f^e(m_{y_{ij}})$ for some $e < r$, then

(a) Go to Stage 0 of statement S_i^{j+1} if $j \neq h_i$.
(b) Go to Stage 0 of statement S_{i+1}^1 if $j = h_i$ and $i \neq t$.
(c) Place all members of B^f in $L_{m_{\bar{x}}}$ if $j = h_i$ and $i = t$.

(ii) If $f^r(m_{y_{ij}}) \in T_{ij}$, then place all members of F^f in $L_{m_{\bar{x}}}$.

(iii) Suppose neither (i) nor (ii) holds, then:

I. If $i = 1$, place $(m_{x_1}, m_{x_2}, \ldots, m_{x_n})$ in $L_{m_{\bar{x}}}$ with $f^r(m_{y_{ij}})$ in place of $m_{y_{ij}}$ and go to Stage $(r + 1)$ of statement S_1^j.

II. Suppose $2 \leqslant i \leqslant t$. Let $z_1, z_2, \ldots, z_q$ be all the variables of F such that for each $k(1 \leqslant k \leqslant q)$, $R(z_k, y_{ij})$ is a subformula of F. Check whether there is a $k(1 \leqslant k \leqslant q)$ such that $f^e(m_{z_k}) = f^r(m_{y_{ij}})$ for some $e < r$.
(a) If there does not exist such a k, place $(m_{x_1}, m_{x_2}, \ldots, m_{x_n})$ in $L_{m\bar{x}}$ with $f^r(m_{y_{ij}})$ in place of $m_{y_{ij}}$ and go to Stage $(r + 1)$ of statement S_i^j.
(b) Suppose (a) does not hold. Then let $k_1, k_2, \ldots, k_u$ be all such k's. For each $s(1 \leqslant s \leqslant u)$ check whether $T_{ij} \cap C_f\ (f^e(m_{z_{ks}})) \neq \Phi$.
(b1) If for some $s(1 \leqslant s \leqslant u)$ $T_{ij} \cap C_f\ (f^e(m_{z_{k_s}})) \neq \Phi$, then place all members of F^f in $L_{m\bar{x}}$.
(b2) If for all $s(1 \leqslant s \leqslant u)$, $T_{ij} \cap C_f\ (f^e(m_{z_{k_s}})) = \Phi$, then
(θ 1) Go to Stage 0 of statement S_i^{j+1} if $j \neq h_i$
(θ 2) Go to Stage 0 of statement S_{i+1} if $j = h_i$ and $i \neq t$
(θ 3) Place all members of B^f in $L_{m\bar{x}}$ if $j = h_i$ and $i = t$.

This ends the programme.

It is possible to do the checkings described in the programme because for each $k(1 \leqslant k \leqslant q)$, there exist $u, v(1 \leqslant u \leqslant i$ and $1 \leqslant v \leqslant h_u)$ such that $z_k = y_{uv}$ and statement S_i^j is executed only after all the statements S_c^d (where $c < i$ and $1 \leqslant d \leqslant h_c$ or $c = i$ and $d < j$) are executed, and in each Stage r executed in statement S_c^d, either conditions (i) or (iii)II(b)(b2)(θ 1) or (iii)II(b)(b2)(θ 2) occurred if $c \neq 1$ and condition (i) occurred if $c = 1$. We will first prove that the list $L_{m\bar{x}}$ is infinite. If all the members of B^f or F^f are placed in $L_{m\bar{x}}$, then it is infinite. While executing the programme, if one of the statements "Place all members of B^f in $L_{m\bar{x}}$" or "Place all members of F^f in $L_{m\bar{x}}$" is not encountered, then there exists a statement S_c^d (where $1 \leqslant c \leqslant t$ and $1 \leqslant d \leqslant h_c$) such that for each $r \geqslant 0$, condition (iii)II(a) occurs at Stage r if $c \neq 1$ and condition (iii)I occurs at Stage r if $c = 1$. Furthermore, for each $r \geqslant 0$, a new n-tuple is placed in $L_{m\bar{x}}$ during Stage r. Therefore $L_{m\bar{x}}$ is infinite. Next we will prove the following statement (ϕ1).

(ϕ1) $\quad (m_{x_1}, m_{x_2}, \ldots, m_{x_n}) \in F^f \Rightarrow L_{m\bar{x}} \subseteq CF^f\ (m_{x_1}, m_{x_2}, \ldots, m_{x_n}) \in F^f \Rightarrow L_{m\bar{x}} \subseteq \overline{CF^f}.$

Suppose $(m_{x_1}, m_{x_2}, \ldots, m_{x_n}) \in F^f$ and there exists a member $(u_1, u_2, \ldots, u_n)$ of F^f such that $(u_1, u_2, \ldots, u_n) \in L_{m\bar{x}}$. This is possible only if there exist j, r ($1 \leqslant j \leqslant t$ and $r \geqslant 1$) such that one of the following three conditions holds:

(P1) $R(x_j, x_k)$ is a subformula of F where $1 \leqslant k \leqslant n$, $m_{x_k} \in C_f m_{x_j}$, $f^r(m_{x_k}) \notin C_f m_{x_j}$ and $(u_1, u_2, \ldots, u_n) = (m_{x_1}, m_{x_2}, \ldots, m_{x_{k-1}}, f^r(m_{x_k}), m_{x_{k+1}}, \ldots, m_{x_n})$.

(P2) $R(x_j, x_k)$ is a subformula of F where $1 \leqslant k \leqslant n$, $m_{x_k} \in C_f m_{x_j}$, $m_{x_k} \notin C_f(f^r(m_{x_j}))$ and $(u_1, u_2, \ldots, u_n) = (m_{x_1}, m_{x_2}, \ldots, m_{x_{j-1}}, f^r(m_{x_j}), m_{x_{j+1}}, \ldots, m_{x_n})$.

(P3) $R(x_j, \underline{a})$ is a subformula of F where $\underline{a}$ is a constant, $a \in C_f m_{x_j}$, $a \notin C_f(f^r(m_{x_j}))$ and $(u_1, u_2, \ldots, u_n) = (m_{x_1}, m_{x_2}, \ldots, m_{x_{j-1}}, f^r(m_{x_j}), m_{x_{j+1}}, \ldots, m_{x_n})$.

Now $u \in C_f v$ and $w \in C_f u \Rightarrow w \in C_f v$, therefore condition P1 cannot hold. Suppose either condition P2 or P3 holds; i.e., $w \in C_f m_{x_j}$ and $w \notin C_f(f^r(m_{x_j}))$ where w is either m_{x_k} or a, then there exists a $\underline{r} < r$ such that $w = f^{\underline{r}}(m_{x_j})$. Now, the n-tuple $(m_{x_1}, m_{x_2}, \ldots, m_{x_{j-1}}, f^s(m_{x_j}), m_{x_{j+1}}, \ldots, m_{x_n})$ where $s > \underline{r}$ may be placed in $L_{m\bar{x}}$ only if there is an $e \leqslant \underline{r}$ such that $f^e(m_{x_j}) = f^s(m_{x_j})$, in which case

$w \in C_f(f^s(m_{x_j}))$. Therefore, $(m_{x_1}, m_{x_2}, \ldots, m_{x_{j-1}}, f^r(\underline{m_{x_j}}), m_{x_{j+1}}, \ldots, m_{x_n})$ cannot be placed in $L_{m_{\bar{x}}}$. This is a contradiction. Thus $\overline{F^f} \cap L_{m_{\bar{x}}} = \Phi$. Suppose $m_{x_1}, m_{x_2}, \ldots, m_{x_n}) \in \overline{F^f}$ and there is a member $(u_1, u_2, \ldots, u_n)$ of F^f such that $(u_1, u_2, \ldots, u_n) \in L_{m_{\bar{x}}}$. This is possible only if there exist j, r ($1 \leqslant j \leqslant t$ and $r \geqslant 1$) such that one of the following three conditions holds:

(q1) $R(x_j, x_k)$ is a subformula of F where $1 \leqslant k \leqslant n$, $m_{x_k} \notin C_f m_{x_j}$, $m_{x_k} \in C_f(f^r(m_{x_j}))$ and $(u_1, u_2, \ldots, u_n) = (m_{x_1}, m_{x_2}, \ldots, m_{x_{j-1}}, f^r(m_{x_j}), m_{x_{j+1}}, \ldots, m_{x_n})$.

(q2) $R(x_j, \underline{a})$ is a subformula of F where $\underline{a}$ is a constant, $a \notin C_f m_{x_j}$, $a \in C_f(f^r(m_{x_j})$ and $(u_1, u_2, \ldots, u_n) = (m_{x_1}, m_{x_2}, \ldots, m_{x_{j-1}}, f^r(m_{x_j}), m_{x_{j+1}}, \ldots, m_{x_n})$.

(q3) $R(x_j, x_k)$ is a subformula of F where $1 \leqslant k \leqslant n$, $m_{x_k} \notin C_f m_{x_j}$, $f^r(m_{x_k}) \in C_f(m_{x_j})$ and $(u_1, u_2, \ldots, u_n) = (m_{x_1}, m_{x_2}, \ldots, m_{x_{k-1}}, f^r(m_{x_k}), m_{x_{k+1}}, \ldots, m_{x_n})$.

Now, $u \in C_f v$ and $w \in C_f u \Rightarrow w \in C_f v$. Therefore conditions (q1) or (q2) cannot hold. Furthermore, condition (iii) of any stage in the statement S_c^d($2 \leqslant c \leqslant t$, $1 \leqslant d \leqslant h_c$) of the programme ensures that if the condition (q3) holds, then $(m_{x_1}, m_{x_2}, \ldots, m_{x_{k-1}}, f^r(m_{x_k}), m_{x_{k+1}}, \ldots, m_{x_n})$ is placed in $L_{m_{\bar{x}}}$ only if $(m_{x_1}, m_{x_2}, \ldots, m_{x_n}) \in F^f$. Thus we arrive at a contradiction. Therefore $L_{m_{\bar{x}}} \cap F^f = \Phi$.

This proves the statement ($\phi 1$). For any n-tuple $(m_{x_1}, m_{x_2}, \ldots, m_{x_n})$, there exists a corresponding list $L(m_{x_1}, m_{x_2}, \ldots, m_{x_n})$ and the Gödel number of the programme for enumerating this list is effectively calculable from $(m_{x_1}, m_{x_2}, \ldots, m_{x_n})$; i.e., there exists a recursive function h such that for each n-tuple $(m_{x_1}, m_{x_2}, \ldots, m_{x_n})$, $h(m_{x_1}, m_{x_2}, \ldots, m_{x_n})$ is the Gödel number of the programme which enumerates the list $L(m_{x_1}, m_{x_2}, \ldots, m_{x_n})$. Therefore the following statement ($\phi 2$) holds.

(ϕ2) There is a recursive function g such that for each n-tuple $(m_{x_1}, m_{x_2}, \ldots, m_{xd})$, $L(m_{x_1}, m_{x_2}, \ldots, m_{x_n}) = W_{g(m_{x_1}, m_{x_2}, \ldots, m_{x_n})}$.

The statements ($\phi 1$), ($\phi 2$) and the fact that $L(m_{x_1}, m_{x_2}, \ldots, m_{x_n})$ is infinite, imply the statement (δ). Therefore by Young's result, F^f is a cylinder. This proves Result (α).

REFERENCES

[1] Cleave, J. P., "Combinatorial systems I, cylindrical problems," *Journal of Computer and Systems Sciences,* vol. 6 (1972), pp. 254-266.

[2] Thuraisingham, M. B., *System Functions and their Decision Problems,* Ph.D. Thesis (1979), Department of Pure Mathematics, University College of Swansea, U.K.

[3] Young, P. R., "Linear orderings under one-one reducibility," *The Journal of Symbolic Logic,* vol. 31 (1966), pp. 37-51.

Department of Computer Science
New Mexico Tech
Socorro, New Mexico 87801