Three Open Questions in the Theory of One-Symbol Smullyan Systems*

by Stephen D. Isard, Department of Machine Intelligence and Perception, University of Edinburgh

and Arnold M. Zwicky, Department of Linguistics, The Ohio State University

0. Introduction. A <u>Smullyan system</u> (<u>ss</u>) S is a quadruple (K, V, P, A), where K is a finite alphabet of symbols called <u>constants</u>, V a finite alphabet of symbols called <u>variables</u>, P a finite alphabet of symbols called <u>predicates</u>, and A a finite set of <u>axioms</u> of the forms

(i) $B_1\alpha_1$

(ii) $B_1\alpha_1$ & $B_2\alpha_2$ & ... & $B_n\alpha_n \rightarrow B_{n+1}\alpha_{n+1}$ $(n \geq 1)$

where the α_i are non-null strings on $K \cup V$ and where the B_i are symbols in P. A <u>theorem</u> of S is any string which is either an axiom of S or is derivable from the axioms by a finite number of applications of (a) uniform substitution of non-null strings in K* for variables, and (b) modus ponens. A string $\alpha \in K^*$ is said to be <u>B-generated</u> by S if $B\alpha$ is a theorem of S, and a set L is said to be <u>generated</u> by S if there is a $B \in P$ such that L is the set of all B-generated strings. These definitions are based on the presentation of <u>elementary formal systems</u> in Smullyan 1961.

A <u>one-symbol Smullyan system</u> (<u>oss</u>) is an ss in which K is a unit set; throughout our discussion this member of K is denoted by '|'. A <u>one-predicate Smullyan system</u> is an ss in which P is a unit set; in citing theorems of a one-predicate system, we shall systematically omit occurrences of the predicate symbol.

Examples: The set of all strings of |s of length 2^n $(n \geq 0)$ is generated by a one-predicate oss with two axioms:

(1) |

x → xx

The set of all strings of |s with a Fibonacci length (1, 2, 3, 5, 8,...) is generated by a one-predicate oss with four axioms:

(2) |

||

|||

x & xy & xxy → xxxyy

(using the fact that x, x+y, and 2x+y are all Fibonacci numbers if and only if they are consecutive,[1] in which case $3x+2y$ is the next Fibonacci number after $2x+y$).

The set of all strings of the form: n 0s followed by n^2 |s ($n \geq 0$) is generated by a one-predicate (but two-symbol) Smullyan system with three axioms:

(3) $0|$

$00||||$

$x0|y \;\&\; x00|yz \rightarrow x000|yzz||$

(using the fact that if $u_1^2 = v_1$ and $(u_1+1)^2 = v_1 + v_2$, then $(u_1+2)^2 = v_1 + 2v_2 + 2$).

1. The Squares Problems. We now ask which sets of strings (equivalently, which sets of positive integers) are generable by oss's. Dana Scott has shown that some of these sets are nonrecursive (see Appendix A). On the other hand, it is difficult to devise oss's to generate some extremely elementary sets. One such set, which we have been unable to generate with an oss, is the set of all strings of |s of length n^2, $n \geq 1$. This set is easily obtained with <u>two</u> symbols in K; merely restore a predicate symbol B throughout the axioms in (3) above, and add the axioms

(4) $C|$

$Bx0|y \rightarrow C|y$

But a one-symbol solution has eluded us. We have attempted to construct such a solution along the lines of the oss for Fibonacci numbers given in (2) above; that is, we have attempted to construct a series of conditions on the form of square numbers $x_0, x_1, \ldots, x_i$ which would insure that the numbers are consecutive squares. In particular, we have considered sequences of integers of the form

(5)
$$
\begin{aligned}
x_0 &= m^2 \\
x_1 &= m^2 + 2n + 1 \\
x_2 &= m^2 + 4n + 4 \\
&\vdots \\
x_i &= m^2 + 2in + i^2
\end{aligned}
$$

If there is an $i \geq 1$ such that every x_t ($0 \leq t \leq i$) is a square, y_t^2, if and only if $n = m$, then a (one-predicate) oss for squares is easily constructed:

(6) $$\begin{array}{l} | \\ |||| \\ \vdots \\ ||\ldots| \quad (i+1)^2 \text{ times} \\ x \;\&\; xyy| \;\&\; \ldots \;\&\; \underbrace{xyy\ldots y}_{2i}\underbrace{||\ldots|}_{i^2} \rightarrow \underbrace{xyy\ldots y}_{2(i+1)}\underbrace{||\ldots|}_{(i+1)^2} \end{array}$$

(because the fact that all the x_t's are squares will insure that they are consecutive). It is known that $i \neq 1$, 2, or 3. For $i = 1$, we have

(7) $$\begin{aligned} y_0^2 &= m^2 \\ y_1^2 &= m^2 + 2n + 1 \end{aligned}$$

If we choose any $m \geqslant 1$ and $j \geqslant 1$, and let $y_1 = m + 2j + 1$ and $n = 1/2\ (y_1^2 - m^2 - 1) = 2j\ (m + j + 1) + m$, then n is an integer distinct from m such that $m^2 + 2n + 1$ is a square. The corresponding construction for i=3 is given in Appendix B. There remain two open questions:

Is there an i such that x_0 through x_i in (5) are all squares if and only if $n = m$?

If not, is the set of all strings of |s of square length nevertheless generable by an oss?

The following theorem lends extra interest to the problem of squares:

Theorem: If an oss of form (6) generates the set of strings whose lengths are squares, then any recursively enumerable set of non-null strings can be generated by an oss.

Proof: If R is any r.e. set of positive integers, it follows from the theorem of Appendix A that the set of strings with lengths 2^n such that $n \in R$ can be generated by an oss. Say that we have an oss which C-generates these strings. The problem, then, is to 'get the n down off the 2'.

One way of doing this might be to invoke the 'pairing function' $K(x, y) = (x + y)^2 + 2x + y$. This function has the property that for non-negative integers x, y, z, w, $K(x, y) = K(z, w)$ if and only if $x = z$ and $y = w$.[2]

Suppose for the moment that we could produce axioms which would have the effect of (8')-(10'),

(8') If x is C-generated, then $K'(x, \varphi)$ is D-generated.

(9') If $K'(xx, y)$ is D-generated, then $K'(x, y|)$ is D-generated.

(10') If $K'(|, y)$ is D-generated, then y is E-generated.

where by $K'(x, y)$ we mean the string whose length is $K(\lambda(x), \lambda(y))$, and by φ we mean the empty string. It would then follow that a string of length $K(2^r, s)$ would be D-generated if and only if 2^{r+s} were C-generated, and hence that a string of length n would be E-generated if and only if one of length 2^n were C-generated.

Now, if a set of axioms of form (6) S-generates the strings whose lengths are squares, we know that if

$x,\ xyy|,\ldots,\ x\underbrace{yy\ldots y}_{2i}\underbrace{||\ldots|}_{i^2}$ are all S-generated, then their lengths are consecutive squares, and hence the length of x is the square of the length of y. Let us write $sq(x, y)$ as an abbreviation for the conjunction

$$Sx \;\&\; Sxyy| \;\&\; \ldots \;\&\; Sx\underbrace{yy\ldots y}_{2i}\underbrace{||\ldots|}_{i^2}$$

Then

(8) $Cx \;\&\; sq(x, z) \rightarrow Dzxx$

(9) $sq(xxy, z) \;\&\; sq(xy|, w) \;\&\; Dzxxxxy \rightarrow Dwxxy|$

(10) $sq(|y, z) \;\&\; Dx||y \rightarrow Ey$

are the desired axioms.

2. Perplexing problems arise in even very simple one-predicate oss's. Consider, for example the following axioms suggested by S.D.I.:

(11) $|$

$x \rightarrow xxx$

$xx| \rightarrow x$

Clearly, every string generated by this oss is of a length congruent to 0 or 1 modulo 3.

Problem: Is every string of length congruent to either 0 or 1 modulo 3 generated by this oss?

It is not hard to show that if $3k$ or $3k + 1$ is the least integer congruent to 0 or 1 modulo 3 but not generated by the oss of (11), then $k = 81j + 80$, for some $j \geq 0$. Values of j from 0 through 27 have been checked by computer (an Elliott 4100 machine at the University of Edinburgh), so that if the answer to this third problem is NO, the smallest exception is not less than 7044, the value of $3k$ for $j = 28$. The difficulty in attempts to show that the answer to the third problem is YES arise from the fact that patterns in the generation of strings are hard to discern; some fairly short strings require quite long derivations, for no obvious reason. Thus, to derive the string of length 7 from the initial string, of length 1, one of the two following 15-step derivations is required (where an application of the axiom

$x \to xxx$ is indicated by the index a, and application of the axiom $xx| \to x$ by the index b):

$$1 \underset{a}{\to} 3 \underset{a}{\to} 9 \underset{a}{\to} 27 \underset{b}{\to} 13 \underset{a}{\to} 39 \begin{cases} \underset{a}{\to} 117 \underset{a}{\to} 351 \underset{b}{\to} 175 \underset{b}{\to} 87 \underset{b}{\to} 43 \underset{b}{\to} 21 \underset{a}{\to} 63 \\ \underset{b}{\to} 19 \underset{a}{\to} 57 \underset{a}{\to} 171 \underset{b}{\to} 85 \underset{a}{\to} 255 \underset{b}{\to} 127 \underset{b}{\to} 63 \end{cases}$$

$$63 \underset{b}{\to} 31 \underset{b}{\to} 15 \underset{b}{\to} 7$$

APPENDIX A.

Theorem (D. Scott): Given any partial recursive function T, there is an oss which generates the set of strings with lengths 2^m such that $T(n) = m$ for some n.

Corollary: There is no effective procedure for deciding whether a given string of |s is generable by a given oss.

Outline of proof of theorem: Theorem IIa of Minsky 1961 states that any partial recursive function T can be represented by a program operating on an integer S using instructions I_j of the forms

(i) Multiply by k_j and go to I_j;

(ii) Divide by k_j and go to I_{j_1} if $k_j | S$, otherwise go to I_{j_2}.

where the system starts at I_0 with $S = 2^n$ and halts at I_h with $S = 2^{T(n)}$.

It is a simple matter to mimic the operation of such a program with an oss. We first choose an integer M greater than the number of any instruction in the program. We shall make the oss imitate the Minsky program by B-generating a string of length $MS+j$ whenever the program operates on the integer S with instruction I_j.

In order to reflect the fact that the Minsky program can operate on 2^n, for any n, with instruction I_0, the oss must B-generate all strings of length $M \cdot 2^n$. To achieve this, we introduce a predicate symbol A and axioms

$A||\ldots||$ M times

$Ax \to Axx$

$Ax \to Bx$

The oss then both A-generates and B-generates each string of the desired form (The purpose of introducing the extra symbol A, rather than B-generating the set directly, is to avoid writing the axiom $Bx \to Bxx$, which would B-generate unwanted strings at a later point.).

Then, for each instruction I_j of form (i), we add an axiom

$$\mathrm{B}\underbrace{\mathrm{xx}\ldots\mathrm{x}}_{M}\underbrace{|\cdot|\ldots|}_{j} \rightarrow \mathrm{B}\underbrace{\mathrm{xx}\ldots\mathrm{x}}_{M\cdot k_j}\underbrace{||\ldots|}_{j_1}$$

and for each instruction I_j of form (ii) we add the axiom

$$\mathrm{B}\underbrace{\mathrm{xx}\ldots\mathrm{x}}_{M\cdot k_j}\underbrace{||\ldots|}_{j} \rightarrow \mathrm{B}\underbrace{\mathrm{xx}\ldots\mathrm{x}}_{M}\underbrace{||\ldots|}_{j_1}$$

as well as an axiom

$$\mathrm{B}\underbrace{\mathrm{xx}\ldots\mathrm{x}}_{M\cdot k_j}\underbrace{||\cdot\ldots|}_{M\cdot l+j} \rightarrow \mathrm{B}\underbrace{\mathrm{xx}\ldots\mathrm{x}}_{M\cdot k_j}\underbrace{||\cdot\ldots|}_{M\cdot l+j_2}$$

for each l, $0 < l < k_j$.

Lemma: The oss described above B-generates a string of MS+j |s if and only if the Minsky program can apply the instruction I_j to the integer S at some stage of a computation which begins by applying I_0 to an integer of the form 2^n.

The lemma can be proved by induction in a straightforward fashion. By way of illustration, we shall go through a bit of the proof that if the program applies I_j to S, then a string of length MS+j is B-generated.

Let us assume, inductively, that the program applies I_q to the integer T and that MT+q is B-generated, where I_q is of form (ii). Then the next step of the program will be to apply I_{q_1} to T/k_q, if $k_q|T$, or I_{q_2} to T otherwise. We must thus show that a string of length $MT/k_q + q_1$ is B-generated if $k_q|T$ or that one of $MT+q_2$ is B-generated otherwise.

If $k_q|T$, the string of length MT+q which we assume to be B-generated can be decomposed as $M\cdot k_q$ blocks of |s of length T/k_q followed by a block of q |s. Thus the axiom

$$\mathrm{B}\underbrace{\mathrm{xx}\ldots\mathrm{x}}_{M\cdot k_q}\underbrace{||\ldots|}_{q} \rightarrow \mathrm{B}\underbrace{\mathrm{xx}\ldots\mathrm{x}}_{M}\underbrace{||\ldots|}_{q_1}$$

applies to the string, when we substitute a string of length T/k_q for the variable x, and insures that a string of length $MT/k_q + q_1$ is generated.

If $k_q \nmid T$, then $MT+q = M\cdot k_q\cdot x + Ml + q$ for some integer x and some l, $0 \leq l < k_q$. Therefore we can use the axiom

$$\mathrm{B}\underbrace{\mathrm{xx}\ldots\mathrm{x}}_{M\cdot k_q}\underbrace{||\cdot\ldots|}_{M\cdot l+q} \rightarrow \mathrm{B}\underbrace{\mathrm{xx}\ldots\mathrm{x}}_{M\cdot k_q}\underbrace{||\cdot\ldots|}_{M\cdot l+q_2}$$

with the desired result.

It follows from the lemma that if T is a partial recursive function, and m an integer, then T(n) = m, for some n, if and only if the oss corresponding to the Minsky machine which computes T B-generates $M \cdot 2^m + h$. Thus to prove the theorem we need only add the axiom

$$B\underbrace{xx\ldots x}_{M}\underbrace{||\ldots|}_{h} \rightarrow Cx$$

and the oss C-generates the desired set.

APPENDIX B.

We seek m, n, x_0, x_1, x_2, x_3 such that

(1)
$$\begin{aligned} x_0 &= m^2 \\ x_1 &= m^2 + 2n + 1 \\ x_2 &= m^2 + 4n + 4 \\ x_3 &= m^2 + 6n + 9 \end{aligned}$$

Equivalently, we seek m, n, k, h, l such that

(2)
$$\begin{aligned} m^2 + 2n + 1 &= (m + k)^2 \\ m^2 + 4n + 4 &= (m + k + h)^2 \\ m^2 + 6n + 9 &= (m + k + h + l)^2 \end{aligned}$$

From (2) it follows that $k > h > l$, so that we may set $k = h + r$, $l = h - s$, and search for positive integral solutions of

(3) $$m^2 + 2n + 1 = (m + h + r)^2$$

(4) $$m^2 + 4n + 4 = (m + 2h + r)^2$$

(5) $$m^2 + 6n + 9 = (m + 3h + r - s)^2$$

Equations (3) and (4) will be simultaneously satisfied if there are x, h, and r such that

(6) $$(m+h+r)^2 - m^2 + 2 = (m+2h+r)^2 - (m+h+r)^2$$

where n is defined by

(7) $$2n + 3 = (m+2h+r)^2 - (m+h+r)^2 .$$

Then from (6) it follows that

(8) $$m = 1/2r\,(2h^2 - r^2 - 2) = 1/r\,(h^2-1) - r/2 .$$

Similarly, (4) and (5) will be simultaneously satisfied if there are x, h, r, and s such that

(9) $$(m+2h+r)^2 - (m+h+r)^2 + 2 = (m+3h+r-s)^2 - (m+2h+r)^2$$

given the definition of n in (7). From (9) we derive

(10) $$m = 1/2s\ (2h^2-2+s^2-2sr-6sh) = 1/s\ (h^2-1) + s/2 - r - 3h$$

Thus, all of (3), (4), (5) will be satisfied if there are h, s, and r such that

(11) $$1/r\ (h^2-1) - r/2 = 1/s\ (h^2-1) + s/2 - r - 3h$$

with m defined as in (8) and n defined as in (7). Then, (11) is equivalent to the quadratic equation

(12) $$(r-s)h^2 - 3rsh - (1 + rs/2)(r-s) = 0$$

with the solution

(13) $$h = \frac{3rs \pm \sqrt{9r^2s^2 + 4(r-s)^2 + 2rs(r-s)^2}}{2(r-s)}$$

Now, the expression under the radical in (13) will be the square of $3rs+2(r-s)$ if and only if $r-s = 6$. Hence if we choose any even s and set

(14) $$r = s + 6$$

$$h = \frac{3rs \pm (3rs + 2(r-s))}{2(r-s)}$$

we will have a solution to (11). Since we want h to be a positive integer, we choose

(15) $$h = \frac{3rs + (3rs + 2(r-s))}{2(r-s)} = \frac{rs}{2} + 1$$

(which is an integer because s is even). Then, from (8),

(16) $$m = 1/r\ ((\frac{rs}{2} + 1)^2 - 1) - r/2 = 1/4\ rs^2 + rs - r/2$$

which is also a (positive) integer, as is n in (7). Thus, we have a solution of (3)-(5) with m, n, h, r, and s all positive integers; and there is one such solution for each choice of even s.

FOOTNOTES

*This work was supported in part by the 1965 and 1969 MSSB Advanced Research Seminars in Mathematical Linguistics (at the Massachusetts Institute of Technology and the University of Illinois, respectively), under a National Science Foundation grant to the Center for Advanced Study in the Behavioral Sciences, Stanford, California.

[1]Suppose that x and x+y are Fibonacci and <u>not</u> consecutive, and let z be the Fibonacci number immediately preceding x+y. Then $2x+y < x+y+z$. But x+y+z is the Fibonacci number immediately following x+y, so that 2x+y cannot be Fibonacci.

[2]To see this, note that $(x+y+1)^2 > K(x,y)$, so that $(x+y)^2$ is the greatest square less than or equal to K(x, y).

Then $(z+w)^2$ is the greatest square less than or equal to $K(z, w)$, so $K(x, y) = K(z, w)$ implies $(x+y)^2 = (z+w)^2$, which implies $x+y = z+w$. But then $x = K(x, y)-(x+y)^2-(x+y) = K(z, w)-(z+w)^2-(z+w) = z$, and hence $y = w$ also.

REFERENCES

Minsky, M. L. 1961. Recursive unsolvability of Post's problem of "Tag" and other topics in the theory of Turing Machines. Ann. Math. 74:437-55.

Smullyan, R. M. 1961. Theory of formal systems. Annals of Mathematics Studies, Number 47. Princeton.