数理解析研究所講究録
189巻 1973年 7-12

Some Remarks in General Theory of Flow-Charts

By Ken Hirose and Makoto Oya

0. In this note, we state some remarks in general theory of flowcharts. In order to know the detail of definitions and proofs, see [1] and [2].

1.

We use following symbols to define "flowcharts".

variable symbols: $x_1, x_2, \cdots, y_1, y_2, \cdots$

function symbols: $f_1, f_2, \cdots$

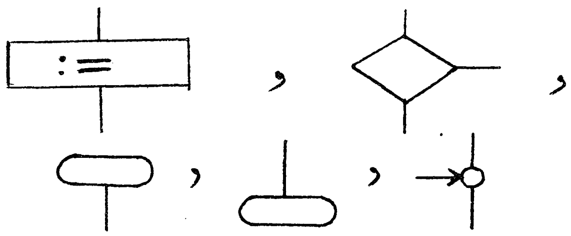
predicate symbols: $p_1, p_2, \cdots$

logical connectives: $\vee$, $\neg$

logical constants: T, F

auxiliary symbols: (,), ,

object symbols:



(Note: Individual constants are considered as 0-ary function symbols.)

From above symbols, terms, formulas, and thus flowcharts are defined. Simultaneously, we can give their interpretation. Interpreted flowcharts are called programs. Then, we can say programs are equivalent to relativized partial recursive procedures.

2.

We say a flowchart S is in normal form if S is as in Fig.1; where L, L_1 and L_2 are loop-free flowcharts (i.e. flowcharts without loops), and A is a formula.

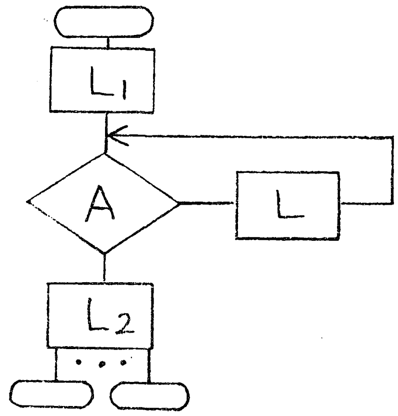


Fig.1 (Normal Form)

We proved following theorem.

Normal Form Theorem. For every flowchart S, there exists a flowchart in normal form which is equivalent to S.

In this theorem, it is important that there exists effective procedure to get the equivalent flowchart in normal form from a given flowchart.

3.

We make mention of some application of Normal Form Theorem.

The first application is that we can effectively get a predicate which is correct w.r.t. a given program. And we also obtain Davis' result about diophantine predicates.

The second application is about "termination", however we have not obtained any satisfiable result.

Let (S, I) be a program, where S is a flowchart in normal form and I is an interpretation.

Def.1 $\mathrm{Ter}(S, I, \xi) \Leftrightarrow [(S, I)$ terminates for input $\xi\,]$

$\mathrm{Ter}(S, I) \Leftrightarrow (\forall \xi)\mathrm{Ter}(S, I, \xi)$

$\mathrm{Ter}(S) \Leftrightarrow (\forall I)\mathrm{Ter}(S, I).$

On the other hand, from Normal Form Theorem, we can consider the condition that (S, I) terminates for ξ passing the loop (i.e. L in Fig.1) n times. This condition is denoted by $\mathrm{Ter}(S, I, \xi, n)$. Then,

(1) $\mathrm{Ter}(S, I, \xi) \Leftrightarrow (\exists n)\mathrm{Ter}(S, I, \xi, n).$

Hence,

(2) $\mathrm{Ter}(S, I) \Leftrightarrow (\forall \xi)(\exists n)\mathrm{Ter}(S, I, \xi, n).$

We shall define another kind of "termination". That is, (S, I) terminates boundedly.

Def.2 $b\text{-}\mathrm{Ter}(S, I) \Leftrightarrow (\exists N)(\forall \xi)(\exists n < N)\mathrm{Ter}(S, I, \xi, n).$

Then we have,

(3) $b\text{-}\mathrm{Ter}(S, I) \Rightarrow \mathrm{Ter}(S, I).$

Clearly the converse of (3) is not true.

Moreover,

Def.3 $b\text{-}\mathrm{Ter}(S) \Leftrightarrow (\forall I)[b\text{-}\mathrm{Ter}(S, I)]$

$b^*\text{-}\mathrm{Ter}(S) \Leftrightarrow (\exists N)(\forall I)(\forall \xi)(\exists n < N)\mathrm{Ter}(S, I, \xi, n).$

Then,

(4) $b^*\text{-}\mathrm{Ter}(S) \Rightarrow b\text{-}\mathrm{Ter}(S) \Rightarrow \mathrm{Ter}(S).$

(Note : Prof. Weyhrauch and Prof. Nozaki showed us a proof of $\Leftarrow$ in (4) at Kyoto Symposium.)

Following propositions are easily shown:

Prop.1 If we can answer to the problem of equivelence between loop-free programs, then

(1) Whether b - Ter(S, I) or not is semidecidable (i.e. if b - Ter(S, I), we can show b - Ter(S, I)).

(2) If (b-)Ter(S_1, I) and (b-)Ter(S_2, I), then equivalence between (S_1, I) and (S_2, I) is decidable.

Prop.2 (1) b^* - Ter is semidecidable.

(2) If b^* - Ter(S_1) and b^* - Ter(S_2), then equivalence between S_1 and S_2 is decidable.

(Note: In Props. 1 and 2, S, S_1 and S_2 denote flowcharts.)

Furthermore, in the case L (in Fig.1) has some property (e.g. "has only one path", etc.), b^* - Ter is decidable.

4.

We shall mention about the equivalence of loop-free flowcharts. $L_1, L_2, \cdots$ denote loop-free flowcharts in this section.

An interpretation with equality is said a structure in this paper.

We proved following two theorems:

Theorem I If I is a structure, for (L_1, I) and (L_2, I) it can be constructed a formula A satisfying

(5) $[(L_1, I)$ is equivalent to $(L_2, I)]$

$\Longleftrightarrow$ [A is valid in I].

Theorem II For a formula A, L_1 and L_2 can be constructed satisfying (5).

Above two say that the equivalence of loop-free programs is equivalent to the validity of an open formula.

From Theorem I, we get some results about decidability.

Prop.3 Whether $L_1 \sim L_2$ is decidable. (Where $L_1 \sim L_2$ means L_1 and L_2 are equivalent. That is (L_1, I) and (L_2, I) are equivalent for every interpretation I.)

Prop.4 If two loop-free programs P_1 and P_2 has only $+$ and $=$ as operations, then whether $P_1 \sim P_2$ is decidable.

Proof. Because it is well known that whether given formula A is valid or not is decidable if A has only $+$ and $=$ as operations.

From Theorem II, we also get a result about undecidability.

Prop.5 Let Π be the set of loop-free programs whose domain is integers and which has only $+$, $\cdot$ (product) and $=$ as operations. Then there is no algorithm that determines whether $P_1 \sim P_2$ or not for given $P_1, P_2 \in \Pi$.

Proof. By the negative solution of Hilbert's 10th problem [3], there is a Diophantine equation $D = 0$ having no algorithm that determines whether any solution of $D = 0$ exists or not for given coefficients of D.

Consider $D \neq 0$ as A in Theorem II. Then,

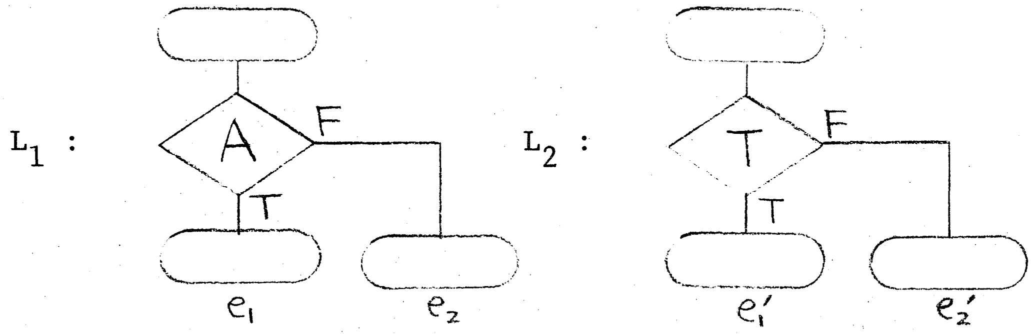
$$(D \neq 0 \text{ is identically true}) \Leftrightarrow P_1 \sim P_2$$

where $P_1 = (L_1, I)$ and $P_2 = (L_2, I)$ in Theorem II. Hence,

$$(D = 0 \text{ has some solution}) \Leftrightarrow \text{not } (P_1 \sim P_2).$$

So, if whether $P_1 \sim P_2$ or not is decidable, then whether $D = 0$ has any solution or not is decidable. That is a contradiction.

Note: In Theorem II, L_1 and L_2 are constructed as follows:



In the above proposition, we do not mind the domain. But the equivalence between loop-free flowcharts is also undecidable even if we consider "axioms" on it. (See [1].)

References

[1] K. Hirose and M. Oya, General Theory of Flow-Charts, Comm. Math. Univ. SANCTI PAULI, to appear.

[2] K. Hirose and M. Oya, Some results in General Theory of Flow-Charts, Proceedings of 1st USA-Japan Computer Conference, 1972.

[3] Ju. V. Matijasevič, Enumerable sets are Diophantine, Soviet Math. Dokl., Vol.11, No.2, 1970.

Other references are also in [1] and [2].

Note added in proof : Prof. Kasami and Prof. Tokura proved Prop. 5 in their paper " Equivalence Problem on Programs without Loops " Trans. IECE '71/7 vol. 54 - C , No. 7