A CHARACTERIZATION OF FLOYD-PROVABLE PROGRAMS

H.Andréka, I.Németi, and I.Sain
Mathematical Institute of the Hungarian Academy of Sciences
Budapest, Reáltanoda u. 13-15, H-1053 Hungary

This paper belongs to first order dynamic logic and within this field it belongs to the nonstandard-time school. A systematic exposition is [9], short ones are [3], [4], [8], [11]. An explicit characterization of the information content of the Floyd programverification method will be given here. Since the results might be of interest for researchers outside the scope of dynamic logic e.g. researchers in nonstandard recursion theory, here we use continuous traces instead of the more natural intensional semantics of [9], [3]. For the same reason here we assume that the theory T from which we prove partial correctness of our programs contains Peano's axioms. If we drop this condition then the results will be different, see [9], [8]. For more introductory and historical material, examples, and motivations see [9]. See also the end of the present paper.

We are interested in the behaviour of program schemes in first order axiomatizable classes of models (i.e. interpretations). We are unable to understand why some people investigate properties of programs either in a single fixed model or else in the similarity class of all possible models.

1. Syntax

t is a similarity type consisting of function and relation symbols.

ω denotes the set of natural numbers.

$Y \stackrel{d}{=} \{ y_i : i \in \omega \}$ is called the set of variable symbols and is disjoint from everything we use. Logical symbols: $\{ \wedge , \neg , \exists \}$. Other symbols: $\{ \leftarrow , \text{IF} , \text{GOTO} , (,) , : \}$. The set of "label symbols" is ω itself.

L_t denotes the set of all first order formulas of type t possibly with free variables (elements of Y of course), see e.g. [6]p.22 .

We shall refer to "terms of type t" as defined in e.g. [6] p.22 .

Now we define the set P_t of programs of type t .

The set U_t of commands of type t is defined by:

$(j: y \leftarrow \tau) \in U_t$ if $j \in \omega$, $y \in Y$ and τ is a term of type t .

$(j: \text{IF } \chi \text{ GOTO } v) \in U_t$ if $j, v \in \omega$, $\chi \in L_t$ is a formula without quantifiers.

These are the only elements of U_t .

If $(i:u) \in U_t$ then i is called the label of the command $(i:u)$.

By a program of type t we understand a finite sequence of commands (elements of U_t) in which no two members have the same label. Formally, the set of programs is:

$$P_t \stackrel{d}{=} \left\{ \langle (i_0:u_0) \ldots (i_n:u_n) \rangle : n \in \omega, (\forall e \leq n)(i_e:u_e) \in U_t, (\forall e < k \leq n) i_k \neq i_e \right\}.$$

For every $p \stackrel{d}{=} \langle (i_0:u_0), \ldots, (i_n:u_n) \rangle \in P_t$ we shall use the notation

$$i_{n+1} \stackrel{d}{=} \min(\omega \setminus \{ i_m : m \leq n \}) .$$

EXAMPLE: Let t contain the function symbols: "+,·,0,1" with arities "2,2,0,0" respectively. Now the sequence:

$\langle (0: y_1 \leftarrow 0), (1: \text{IF } y_1 = y_2 \text{ GOTO } 4), (2: y_1 \leftarrow y_1 + 1), (3: \text{IF } y_2 = y_2 \text{ GOTO } 1) \rangle$

is a program of type t .

2. Semantics

Let $p \in P_t$ be a program and $\underline{\underline{A}}$ be a structure or model of type t see [6] p.20 . The universe of a model denoted by $\underline{\underline{A}}$ will always be denoted by A .

V_p denotes the variable symbols occurring in p .
Note that V_p is a finite subset of Y .

By a valuation (of the variables of p) in $\underline{\underline{A}}$ we understand a function $q : V_p \longrightarrow A$ (cf. [5] p.55) .

Let τ be a term occurring in p. Now $\tau[q]_{\underline{\underline{A}}}$ denotes the value of the term τ in the model $\underline{\underline{A}}$ at the valuation q of the variable symbols, cf. [6] p.27 Def.1.3.13. We shall often write $\tau[q]$ i.e. we shall omit the subscript $\underline{\underline{A}}$.

From now on we work with the <u>similarity type</u> of arithmetic. I.e. t is fixed to consist of "+,·,0,1" with arities "2,2,0,0". We shall omit the index t since it is fixed anyway.

$\underline{\underline{N}}$ denotes the standard model, that is

$\underline{\underline{N}} \stackrel{d}{=} \langle \omega , + , \cdot , 0 , 1 \rangle$ where $+,\cdot,0,1$ are the usual.

<u>EXAMPLE</u>: Let $V_p = \{y_1,y_2\}$, $q(y_1)=2$, $q(y_2)=3$, $\tau=((y_1+y_2)+y_2)$. Then $\tau[q]_{\underline{\underline{N}}} = 8$.

We shall only be concerned with models of the Peano-axioms.

$PA \subseteq L$ denotes the (recursive) set of the <u>Peano-axioms</u> (together with the induction axioms), see [6] p.42 Ex.1.4.11.(axioms 1-7).

Next we define <u>continuous traces</u> of programs in models of PA.

Let $\underline{\underline{A}} \models PA$ be an arbitrary model (of Peano-arithmetic). Let $p \in P$ be a program with set V_p of variables.

A <u>trace</u> of p in $\underline{\underline{A}}$ is a sequence $s \stackrel{d}{=} \langle s_a \rangle_{a \in A}$ indexed by the elements of A such that (i) and (ii) below are satisfied:

(i) $s_a : V_p \cup \{\lambda\} \longrightarrow A$ is a <u>valuation</u> of the variables (of p) into $\underline{\underline{A}}$, where $\lambda \in Y \setminus V_p$ is a variable not occurring in p. λ can be conceived of as the "control variable of p". (We could call s_a a "<u>state</u>" of p in the model $\underline{\underline{A}}$.)

(ii) To formulate this condition, let $p = \langle (i_o:u_o),\ldots,(i_n:u_n) \rangle$ and recall the notation $i_{n+1} \stackrel{d}{=} \min(\omega \setminus \{ i_m : m \leq n \})$. Now we demand:

$s_o(\lambda) = i_o$ and for any $a \in A$,

if $s_a(\lambda) \notin \{ i_m : m \leq n \}$ then $s_{a+1} = s_a$ else,

for all $m \leq n$ such that $s_a(\lambda) = i_m$, conditions a) and b) below hold.

a) if u_m = " $y_w \leftarrow \tau$ " then:

$s_{a+1}(\lambda) = i_{m+1}$ and for any $x \in V_p$,

$$s_{a+1}(x) = \begin{cases} \tau[s_a]_{\underline{\underline{A}}} & \text{if } x = y_w \\ s_a(x) & \text{otherwise} \end{cases} .$$

b) if u_m = " IF χ GOTO v " then:

$$s_{a+1}(\lambda) = \begin{cases} v & \text{if } \underline{\underline{A}} \models \chi[s_a] \\ i_{m+1} & \text{otherwise} \end{cases} , \text{ and}$$

$s_{a+1}(x) = s_a(x)$, for every $x \in V_p$.

By this we have defined traces of a program in $\underline{\underline{A}}$ as sequences $\langle s_a \rangle_{a \in A}$ "respecting the structure" of the program.

It remains to define the continuous traces.

The sequence $\langle s_a \rangle_{a \in A}$ is continuous in $\underline{\underline{A}}$ if $\langle s_a \rangle_{a \in A}$ satisfies the induction axioms, that is if for any $\varphi \in L$ with free variables in $V_p \cup \{\lambda\}$,

$$\underline{\underline{A}} \models \big((\varphi[s_o] \wedge \bigwedge_{a \in A} (\varphi[s_a] \rightarrow \varphi[s_{a+1}])) \rightarrow \bigwedge_{a \in A} \varphi[s_a] \big) .$$

By a <u>continuous trace</u> of p in $\underline{\underline{A}}$ we understand a trace $\langle s_a \rangle_{a \in A}$ of p which is continuous.

Note that in the standard model $\underline{\underline{N}}$ every trace is continuous.

Intuitively, a trace $\langle s_a \rangle_{a \in A}$ is continuous if whenever a first order property $\varphi \in L$ changes during time (A), then there exists a <u>point of time</u> ($a \in A$) when this change is just happening:

$\underline{\underline{A}} \models \varphi[s_o]$ and $(\exists a \in A)\ \underline{\underline{A}} \not\models \varphi[s_a]$ together imply that

$(\exists a \in A)(\ \underline{\underline{A}} \models \varphi[s_a]$ and $\underline{\underline{A}} \not\models \varphi[s_{a+1}]\ )$.

Let $p = \langle (i_o : u_o), \ldots, (i_n : u_n) \rangle \in P$ and $\psi \in L$ be such that the free variables of ψ are in V_p . Let $\underline{\underline{A}} \models PA$.
The dynamic formula $\Box(p, \psi)$ is said to be valid in $\underline{\underline{A}}$ w.r.t. continuous traces if (*) below holds.

(*) For any continuous trace $\langle s_a \rangle_{a \in A}$ of p in $\underline{\underline{A}}$ and for any $a \in A$,

$$s_a(\lambda) \notin \{ i_m : m \leq n \} \quad \text{implies} \quad \underline{\underline{A}} \models \psi[s_a] .$$

$\underline{A} \overset{c}{\models} \Box(p,\psi)$ denotes that $\Box(p,\psi)$ is valid in $\underline{A}$ w.r.t. continuous traces.

Intuitively the formula $\Box(p,\psi)$ of dynamic logic means that the program p is partially correct w.r.t. the output condition ψ. See [10].

3. Derivation system (rules of inference)

In the following definition we shall recall the so called Floyd-Hoare derivation system. This system serves to derive statements $\Box(p,\psi)$ (where $p \in P$ and $\psi \in L$) from theories $T \subseteq L$.
We shall denote Floyd-Hoare derivability by $T \overset{F}{\vdash} \Box(p,\psi)$.

DEFINITION:
Let $p = \langle (i_0 : u_0), \ldots, (i_n : u_n) \rangle \in P$, let $\psi \in L$ and let $T \subseteq L$.
The set of labels of p is defined as follows:
$lab(p) \overset{d}{=} \{ i_m : m \leq n+1 \} \cup \{ v : (\exists m \leq n)\, u_m = \text{" IF } \chi \text{ GOTO } v \text{ "} \}$.
Note that $lab(p)$ is finite.

Now a Floyd-Hoare derivation of $\Box(p,\psi)$ from T consists of:
a mapping $\Phi : lab(p) \longrightarrow L$ together with
classical first-order derivations listed in (i)-(iv) below.
Notation: When $z \in lab(p)$ we write Φ_z instead of $\Phi(z)$.

(i) A derivation: $T \vdash \Phi_{i_0}$.

(ii) To each command $(i_m : y_j \leftarrow \tau)$ occurring in p a derivation:
$T \vdash (\Phi_{i_m} \rightarrow \exists x (x = \tau \wedge \exists y_j (y_j = x \wedge \Phi_{i_{m+1}})))$
where x does not occur neither in $\Phi_{i_{m+1}}$ nor in τ.

(iii) To each command $(i_m : \text{IF } \chi \text{ GOTO } v)$ occurring in p derivations:
$T \vdash ((\chi \wedge \Phi_{i_m}) \rightarrow \Phi_v)$ and $T \vdash ((\neg\chi \wedge \Phi_{i_m}) \rightarrow \Phi_{i_{m+1}})$.

(iv) To each $z \in (lab(p) \setminus \{ i_m : m \leq n \})$ a derivation:
$T \vdash (\Phi_z \rightarrow \psi)$.

Now the existence of a Floyd-Hoare derivation of $\Box(p,\psi)$ from T is denoted by $T \overset{F}{\vdash} \Box(p,\psi)$.

REMARKS: If T is decidable then the set of Floyd-Hoare derivations (of statements $\Box(p,\psi)$ where $p \in P$ and $\psi \in L$, from T) is also decidable. If T is recursively enumerable then the Floyd-Hoare deri-

vable formulas are also recursively enumerable that is
$\{ \Box(p,\psi) \ : \ T \overset{F}{\vdash} \Box(p,\psi) \}$ is recursively enumerable.

4. Completeness

Notation: For all $T \subseteq L$, $p \in P$, $\psi \in L$ we define
$$T \overset{c}{\models} \Box(p,\psi) \quad \overset{d}{\Longleftrightarrow} \quad (\forall \underline{\underline{A}}) \left[\underline{\underline{A}} \models T \implies \underline{\underline{A}} \overset{c}{\models} \Box(p,\psi) \right] .$$

THEOREM 1: Let T be such that $L \supseteq T \supseteq PA$. Let $p \in P$ and $\psi \in L$. Then
$T \overset{F}{\vdash} \Box(p,\psi)$ if and only if
$\Box(p,\psi)$ is valid in every model of T w.r.t. continuous traces.

In concise form:
$$T \overset{F}{\vdash} \Box(p,\psi) \quad \Longleftrightarrow \quad T \overset{c}{\models} \Box(p,\psi) \quad .$$

PROOF:
Let $T \supseteq PA$, $p \overset{d}{=} \langle (i_0 : u_0), \ldots, (i_n : u_n) \rangle \in P$ and $V_p = \{y_1, \ldots, y_k\}$.

(I) Assume $T \overset{c}{\models} \Box(p,\psi)$ for some $\psi \in L$. We show that $T \overset{F}{\vdash} \Box(p,\psi)$.
First we introduce an abbreviation: Let x,b,i,r be variables. Then
$$(\ \gamma(x,b,i)=r \ ) \overset{d}{\Longleftrightarrow} \exists z \left[\ (1+(i+1)b)z+r = x \ \wedge \ r < 1+(i+1)b \ \right] .$$
Intuitively: $\gamma(x,b,i)$ = remainder$(x, 1+(i+1)b)$. (Note that $<$ can be expressed in L , e.g. $x<y \overset{d}{\Longleftrightarrow} \exists z(\ x+z=y \wedge \neg x=y \ ) .$)

In models of PA this is a sound abbreviation because
$$PA \models (\forall x,b,i)(\forall r,r') \left[(\ \gamma(x,b,i)=r \ \wedge \ \gamma(x,b,i)=r' \ ) \rightarrow r=r' \right] .$$
Thus in the definition of the formulas Φ_z $(z \in lab(p))$ needed for a Floyd-Hoare derivation from T we can use the abbreviation $\gamma(x,b,i)$.

Let $\underline{\underline{A}} \models T$. Since $V_p = \{y_1, \ldots, y_k\}$, we let $\langle a_1, \ldots, a_k \rangle$ denote the valuation $q : V_p \longrightarrow A$ for which $q(y_j) = a_j$ if $1 \leq j \leq k$.

We want to prove the existence of a Floyd-Hoare derivation $T \overset{F}{\vdash} \Box(p,\psi)$. To this end first we define $\Phi_z \in L$ for every $z \in lab(p)$. Intuitively, Φ_z says that "there is a 'partial trace' (a 'codeable one') such that the state $(z, \langle y_1, \ldots, y_k \rangle)$ occurs in it". (Roughly, a partial trace $\langle s_a \rangle_{a \leq t}$ of lenght $t \in A$ is "codeable" if there is a "code" $\langle x,b \rangle$ such that $\langle s_a \rangle_{a \leq t} = \langle \gamma(x,b,i) \rangle_{i \leq t}$.)
Let $r \in Y$. $\Phi_{i_o} \overset{d}{=}$ " $(\forall r)$ $r=r$ " $\overset{d}{=}$ TRUE .

Let $x_0, \ldots, x_k, b_0, \ldots b_k, r, t \in (Y \setminus V_p)$ be different variable symbols.

Notations, abbreviations: $s(r) \overset{d}{=} \langle \gamma(x_j,b_j,r)\rangle_{j\le k}$ and $\ell(r) \overset{d}{=} \gamma(x_0,b_0,r)$.

Let $z \in (\mathrm{lab}(p) \setminus \{i_0\})$. Then

$$\Phi_z(y_1,\dots,y_k) \overset{d}{=} \text{"} (\exists t)(\exists x_0,\dots,x_k,b_0,\dots,b_k) \left[s(t)=\langle z,y_1,\dots,y_k\rangle \wedge \mathrm{PARTRACE}_p(\langle s(r)\rangle_{r\le t}) \right] \text{"}$$

where $s(t) = \langle z,y_1,\dots,y_k\rangle$ abbreviates the formula

$$(\gamma(x_0,b_0,t)=z \wedge \bigwedge_{j=1}^{k} \gamma(x_j,b_j,t)=y_j)$$

and $\mathrm{PARTRACE}_p(\langle s(r)\rangle_{r\le t})$ is defined as follows:

To formulate the formula " $\mathrm{PARTRACE}_p(\langle s(r)\rangle_{r\le t})$ " $\in L$, first we define a formula $\varphi_{pm}(r)$ for every $m \le n$:

Remark: $\varphi_{pm}(r)$ will have other free variables too, not only r. By writing $\varphi_{pm}(r)$ instead of $\varphi_{pm}(r,x_0,\dots,x_k,b_0,\dots,b_k)$ we deviate from the notational conventions of [6].

Let u_m = " $y_w \leftarrow \tau$ ". Then

$$\varphi_{pm}(r) \overset{d}{=} \text{"} (\forall y_1,\dots,y_k)((\bigwedge_{j=1}^{k} \gamma(x_j,b_j,r)=y_j) \rightarrow \left[\ell(r+1)=i_{m+1} \wedge \right.$$

$$\left. \wedge \gamma(x_w,b_w,r+1)=\tau \wedge \bigwedge_{\substack{j=1 \\ j\ne w}}^{k} \gamma(x_j,b_j,r+1)=y_j \right]) \text{"} .$$

Let u_m = " IF χ GOTO v ". Then

$$\varphi_{pm}(r) \overset{d}{=} \text{"} (\forall y_1,\dots,y_k)((\bigwedge_{j=1}^{k} \gamma(x_j,b_j,r)=y_j) \rightarrow$$

$$\rightarrow \left[(\chi \rightarrow \ell(r+1)=v) \wedge (\neg\chi \rightarrow \ell(r+1)=i_{m+1}) \wedge \bigwedge_{j=1}^{k} \gamma(x_j,b_j,r+1)=y_j \right]) \text{"} .$$

It can be checked that $\varphi_{pm}(r) \in L$ in both cases. Now

$$\mathrm{PARTRACE}_p(\langle s(r)\rangle_{r\le t}) \overset{d}{=} \text{"} (\ell(0)=i_0 \wedge (\forall r<t) \left[\bigwedge_{m\le n} (\ell(r)=i_m \rightarrow \varphi_{pm}(r)) \wedge \right.$$

$$\left. \wedge ((\bigwedge_{m\le n} \ell(r)\ne i_m) \rightarrow s(r+1)=s(r)) \right]) \text{"} .$$

By this the definition of Φ_z is completed for every $z \in \mathrm{lab}(p)$. It is easy to check that $\Phi_z \in L$.

Having defined $\Phi : \mathrm{lab}(p) \longrightarrow L$, we have to give some first-order derivations from T. By completeness of classical first-order logic, instead of giving a derivation $T \vdash \varphi$, it is enough to show $T \models \varphi$.

Notation: Let $\varphi \in L$ and τ be a term. Let x be a variable not occurring neither in φ nor in τ. Then we define $\varphi(y/\tau)$ to be the formula $\exists x(x=\tau \wedge \exists y(y=x \wedge \varphi))$.

(i) $T \models \Phi_{i_0}$ is trivial since Φ_{i_0} = "TRUE".

(ii) Let u_m = "$y_w \leftarrow \tau$" for some $m \le n$.

We have to show $T \models (\forall y_1,\dots,y_k)(\Phi_{i_m} \rightarrow \Phi_{i_{m+1}}(y_w/\tau))$.

Let $\underline{\underline{A}} \models T$ and $\underline{\underline{A}} \models \Phi_{i_m}[\langle a_1,\dots,a_k\rangle]$. This means the existence of $\bar{\tau},\bar{x}_o,\dots,\bar{x}_k,\bar{b}_o,\dots,\bar{b}_k \in A$ such that $\langle\langle\gamma(\bar{x}_j,\bar{b}_j,r)\rangle_{j\le k}\rangle_{r\le\bar{\tau}}$ is a "partial trace" of p in $\underline{\underline{A}}$ and $\langle\gamma(\bar{x}_j,\bar{b}_j,\bar{\tau})\rangle_{j\le k} = \langle i_m,a_1,\dots,a_k\rangle$.

Let $\langle n_o,\dots,n_k\rangle \overset{d}{=} \langle i_{m+1},c_1,\dots,c_k\rangle$ where

$$c_j \overset{d}{=} \begin{cases} \tau[\langle a_1,\dots,a_k\rangle]_{\underline{\underline{A}}} & \text{if } j=w \\ a_j & \text{if } j\neq w \end{cases} .$$

Consider the following formula $\pi \in L$ (we write it down by using the abbreviation $\gamma(x,b,i)$) :

π = "$(\forall x,b,t,n)(\exists x',b')[(\forall i\le t)\gamma(x,b,i)=\gamma(x',b',i)\wedge\gamma(x',b',t+1)=n]$" .

A detailed rigorous proof of $PA \models \pi$ can be found in [7]. Note that the related results which can be found in the literature prove only $\underline{\underline{N}} \models \pi$. But " $\underline{\underline{N}} \models \pi$ " is too weak to be of any use here. Hence we do need the result of Csirmaz [7].

Now by $\underline{\underline{A}} \models \pi$ there are $x'_o,\dots,x'_k,b'_o,\dots,b'_k \in A$ such that in $\underline{\underline{A}}$:

$$\langle\langle\gamma(x',b',r)\rangle_{j\le k}\rangle_{r\le\bar{\tau}+1} = \langle\langle\langle\gamma(\bar{x}_j,\bar{b}_j,r)\rangle_{j\le k}\rangle_{r\le\bar{\tau}}, \langle i_{m+1},c_1,\dots,c_k\rangle\rangle,$$

which can easily be seen to be a partial trace in $\underline{\underline{A}}$.

Thus by the definition of $\Phi_{i_{m+1}}$,

$\underline{\underline{A}} \models \Phi_{i_{m+1}}[\langle c_1,\dots,c_k\rangle]$ i.e. $\underline{\underline{A}} \models \Phi_{i_{m+1}}(y_w/\tau)[\langle a_1,\dots,a_k\rangle]$.

(iii) The case u_m = "IF χ GOTO v" can be treated similarly to the above (ii).

(iv) Let $z \notin \{i_m : m\le n\}$.

To show $T \models (\Phi_z \rightarrow \psi)$, we use our assumption $T \overset{c}{\models} \square(p,\psi)$.

Let $\underline{\underline{A}} \models T$ and $\underline{\underline{A}} \models \Phi_z[\langle a_1,\dots,a_k\rangle]$. This means the existence of a partial trace $\langle\langle\gamma(\bar{x}_j,\bar{b}_j,r)\rangle_{j\le k}\rangle_{r\le\bar{\tau}}$ of p in $\underline{\underline{A}}$ for which $\langle\gamma(\bar{x}_j,\bar{b}_j,\bar{\tau})\rangle_{j\le k} = \langle z,a_1,\dots,a_k\rangle$. For every $a \in A$ we define:

$$s_a \overset{d}{=} \langle\gamma(\bar{x}_j,\bar{b}_j,\min(a,\bar{\tau})\rangle_{j\le k} .$$

It is easy to see that $\langle s_a\rangle_{a\in A}$ is a trace of p in $\underline{\underline{A}}$. Now we show that $\langle s_a\rangle_{a\in A}$ is continuous too. Let $\varphi \in L$ be arbitrary for which $\underline{\underline{A}} \models \varphi[s_o] \wedge \bigwedge_{a\in A}(\varphi[s_a] \rightarrow \varphi[s_{a+1}])$. We have to show $\underline{\underline{A}} \models \bigwedge_{a\in A}\varphi[s_a]$.

Let $v \in Y$ be a new variable.

Now we define a formula $\bar{\varphi}(x_o,\dots,x_k,b_o,\dots,b_k,t,v)$ such that $(\forall a\in A)$ $(\underline{\underline{A}} \models \varphi[s_a]$ iff $\underline{\underline{A}} \models \bar{\varphi}(x_o,\dots,t,v)[\langle\bar{x}_o,\dots,\bar{\tau},a\rangle])$ holds.

(Recall that $\bar{x}_o,\dots,\bar{x}_k,\bar{b}_o,\dots,\bar{b}_k,\bar{\tau}$ are fixed elements of A .)

$\bar{\varphi}(x_0,\ldots,x_k,b_0,\ldots,b_k,t,v) \overset{d}{=}$

$$"\Big((v \le t \rightarrow (\forall r_0,\ldots,r_k)\Big[\bigwedge_{j=0}^{k} \gamma(x_j,b_j,v)=r_j \rightarrow \varphi(r_0,\ldots,r_k)\Big]) \wedge$$

$$\wedge (v \not\le t \rightarrow (\forall r_0,\ldots,r_k)\Big[\bigwedge_{j=0}^{k} \gamma(x_j,b_j,t)=r_j \rightarrow \varphi(r_0,\ldots,r_k)\Big]) \Big)".$$

It is easy to see that this $\bar{\varphi}$ will do. Thus $\bar{\varphi} \in L$ and $\underline{A} \models (\bar{\varphi}(x_0,\ldots,t,0) \wedge (\forall v)(\bar{\varphi}(x_0,\ldots,t,v) \rightarrow \bar{\varphi}(x_0,\ldots,t,v+1)))[\langle \bar{x}_0,\ldots,\bar{t}\rangle]$. Since $\underline{A} \models T \supseteq PA$, the induction axiom corresponding to $\bar{\varphi}$ is true in $\underline{A}$ and thus $\underline{A} \models (\forall v)\bar{\varphi}(x_0,\ldots,t,v)[\langle \bar{x}_0,\ldots,\bar{t}\rangle]$, i.e. $\underline{A} \models \bigwedge_{a\in A} \varphi[s_a]$. We have shown that $\langle s_a\rangle_{a\in A}$ is a continuous trace of p in $\underline{A}$. Since $z \notin \{i_m : m \le n\}$ and $s_{\bar{t}} = \langle z,a_1,\ldots,a_k\rangle$ by $\underline{A} \overset{c}{\models} \Box(p,\psi)$, we have $\underline{A} \models \psi[\langle a_1,\ldots,a_k\rangle]$. So far we have shown $T \overset{F}{\vdash} \Box(p,\psi)$.

(II) Let $T \overset{F}{\vdash} \Box(p,\psi)$. We want to show that $T \overset{c}{\models} \Box(p,\psi)$. Let $\underline{A} \models T$ and let $\langle s_a\rangle_{a\in A}$ be a continuous trace of p in $\underline{A}$. Let $\langle \Phi_z\rangle_{z\in lab(p)} : lab(p) \longrightarrow L$ belong to a Floyd-Hoare derivation of $\Box(p,\psi)$ from T. Recall that $y_1,\ldots,y_k$ are the variables occurring in $p = \langle (i_0:u_0),\ldots,(i_n:u_n)\rangle$. Therefore we may use y_0 as "control variable" (i.e. for λ). We define

$$\varphi(y_0,y_1,\ldots,y_k) \overset{d}{=} " \bigwedge_{m\le n} (y_0=i_m \rightarrow \Phi_{i_m}(y_1,\ldots,y_k)) \wedge$$

$$\wedge ((\bigwedge_{m\le n} y_0 \neq i_m) \rightarrow \psi(y_1,\ldots,y_k)) " .$$

Now $\varphi \in L$ and $\underline{A} \models \varphi[s_0] \wedge \bigwedge_{a\in A} (\varphi[s_a] \rightarrow \varphi[s_{a+1}])$. (This is true because $\Phi : lab(p) \longrightarrow L$ belongs to a Floyd-Hoare derivation of $\Box(p,\psi)$ and $\langle s_a\rangle_{a\in A}$ is a trace of p in $\underline{A}$.)

Now, since $\langle s_a\rangle_{a\in A}$ is, in addition, continuous, $\underline{A} \models \bigwedge_{a\in A} \varphi[s_a]$. Let $a \in A$ be such that $s_a(\lambda) \notin \{i_m : m \le n\}$. Then $\underline{A} \models \varphi[s_a]$ implies $\underline{A} \models \psi[s_a]$, by the definition of φ. This means $\underline{A} \overset{c}{\models} \Box(p,\psi)$ since $\langle s_a\rangle_{a\in A}$ was an arbitrary continuous trace of p in $\underline{A}$.

QED

In the definition of continuity of a trace, the induction did not have parameters. This was inessential, namely the above proof works for the case with parameters too [9], [8]. As a contrast, let d be an expan-

sion of t with arbitrary new relation and function symbols. Then $L_d \supsetneqq L_t$. Let $PA \subseteq L_t$ be the same as before. THEOREM 2: Let $L_d \supseteq T \supseteq PA$, $p \in P_d$ and $\psi \in L_d$. Then the conclusion of Thm.1 holds.

This result is due to Jeff B. Paris and L.Csirmaz. This Paris-Csirmaz theorem solves a problem which was open for a long while e.g.[1],[3]. The present proof of Thm.1 does not work for Thm.2 because

$$\left((R(0) \wedge \forall x(R(x) \rightarrow R(x+1))) \wedge \exists x \neg R(x) \right) \in T$$

is allowed for any new relation symbol R in d.

The present proof of Thm.1 first appeared in [1] in 1977. Later it was translated into English. The English translation is Preprint No.8/1978 of our institute. Its abstract is [2]. Since then a large number of papers (e.g.[3],[4],[7-9],[11], ones by Salwicki, Biró, Csirmaz, Gergely, Ury) quote it. Thus we decided to publish it in the form of the present paper.

For propositional dynamic logic see [10].

REFERENCES

1. Andréka,H. Németi,I., On the completeness problem of systems for program verification. (In Hungarian) Math.Inst.Hung.Acad.Sci. - SZKI Budapest, 1977.
2. Andréka,H. Németi,I., Completeness of Floyd Logic. Bull.Section of Logic (Wroclaw) Vol.7, No.3, 1978, pp.115-120.
3. Andréka,H. Németi,I. Sain,I., Henkin-type semantics for program schemes to turn negative results to positive. Fundamentals of Computation Theory FCT'79 Berlin. Ed.: L. Budach. Akademie Verlag 1979, pp.18-24.
4. Andréka,H. Németi,I. Sain,I., Completeness problems in verification of programs and program schemes. Mathematical Foundations of Computer Science MFCS'79 Olomouc. Lecture Notes in Computer Science 74, Springer Verlag 1979, pp.208-218.
5. Bell,J.L. Slomson,A.B., Models and Ultraproducts. North Holland, 1969.
6. Chang,C.C. Keisler,H.J., Model Theory. North Holland, 1973.
7. Csirmaz,L., On definability in Peano Arithmetic. Bull.Section of Logic (Wroclaw) Vol.8, No.3, 1979, pp.148-153.
8. Csirmaz,L., A survey of semantics of Floyd-Hoare derivability. Comput. Linguist. Comput. Lang. CL&CL (Budapest) Vol.14, 1981.
9. Németi,I., A complete first order dynamic logic. Preprint, Math. Inst.Hung.Acad.Sci. 1980, pp.1-120.
10. Pratt,V.R., Application of modal logic to programming. Studia Logica Vol.39, No.2/3, 1980, pp.257-274.
11. Sain,I., There are general rules for specifying semantics: Observations on abstract model theory. Comput. Linguist. Comput. Lang. CL&CL (Budapest) Vol.13, 1979, pp.251-282.