FINITE DERIVATIVE CLOSURE OF PARTIAL FUNCTIONS AND A METHOD OF INFERENCE

Nishizawa, Teruyasu
Faculty of Economics, Niigata University

https://doi.org/10.5109/13397

出版情報：Bulletin of informatics and cybernetics. 23 (1/2), pp.89-93, 1988-03. Research Association of Statistical Sciences
バージョン：
権利関係：
FINITE DERIVATIVE CLOSURE OF PARTIAL FUNCTIONS
AND A METHOD OF INFERENCE

By

Teruyasu NISHIZAWA*

Abstract

We formulate a concept called a finite derivative closure of partial functions which is an abstraction of recursive structures in finite tree automata (transducers). Then we apply this concept to inference of functions from partial functions.

1. Preliminary Remarks and Notations

We can say that a partial function gives informations of a function partially. If a function \( \phi \) is decided by a recursive structure \( S \) and if a partial function \( \phi \) which is a restriction of \( \phi \) contains the same structure \( S \), then we can obtain \( \phi \) by naturally extending \( \phi \) using the structure \( S \). So we are interested in the problem to extract recursive structures from partial functions.

In this paper, we formulate a finite derivative closure as such a recursive structure. Under some restrictions, we discuss the methods for extracting such structures from given partial functions with finite domains.

We now give some notations needed in this paper. Let \( A \) and \( B \) be sets and \( \phi \) be a partial function from \( A \) to \( B \). \( D(\phi) \) denotes the domain of \( \phi \), that is, \( D(\phi)=\{x\in A; (\exists y\in B)[\phi(x)=y]\} \). For a subset \( X \) of \( A \), we set \( \phi(X)=\{y\in A; (\exists x\in X\cap D(\phi))[\phi(x)=y]\} \). For an element \( y \) of \( B \) and a subset \( Y \) of \( B \), we set \( \phi^{-1}(Y)=\{x\in A; \phi(x)=y\} \) and \( \phi^{-1}(Y)=\bigcup_{y\in Y}\phi^{-1}(y) \). \( \phi \) is said to be one-to-one (or 1-1) if \( \phi(x)=\phi(y) \) implies \( x=y \) for each \( x, y \) in \( D(\phi) \). \( \phi \) is said to be onto if \( \phi(A)=B \).

Let \( \phi \) and \( \psi \) be partial functions. If \( D(\phi)\subseteq D(\psi) \) and \( \psi(x) \) is equal to \( \phi(x) \) for each \( x \) in \( D(\phi) \), then we say that \( \phi \) is a restriction of \( \psi \) or \( \psi \) is an extension of \( \phi \) and we denote the situation by \( \phi\subseteq\psi \).

For a nonempty set \( W \) and a nonnegative integer \( m \), \( \Pi_m(W) \) denotes the family of all partial functions from \( W \) to \( W^m \). It should be noted that, in case of \( m=0 \), \( W^0 \) is a set containing only one element. Hence a partial function from \( W \) to \( W^0 \) is identified with a subset of \( W \). For an \( \vec{f} \) in \( \Pi_m(W) \), \( m \) is denoted by \( d(\vec{f}) \) and called the degree of \( \vec{f} \). We set \( \Pi(W)=\bigcup_{m=0}^{\infty}\Pi_m(W) \). For an \( \vec{f} \) in \( \Pi(W) \), the notation \( \vec{f}=(f_1, \ldots, f_m) \) means that

(1) \( d(\vec{f})=m \),

(2) for each \( i=1, \ldots, m \), \( f_i \) is in \( \Pi_i(W) \) and \( D(f_i)=D(\vec{f}) \), and

* Faculty of Economics, Niigata University, Igarashi, Niigata, Japan.
A partially ordered set $(W, \prec)$ is said to be a well founded set if there is no infinite decreasing sequence. Throughout this paper we assume the following (1)-(3):

1. "partial function" is abbreviated to "pf".
2. $W$ and $A$ are nonempty sets.
3. $i, j, k, l, m, n$ are nonnegative integers.

2. Derivatives of Partial Functions by Partial Functions

**DEFINITION 1.** Let $\omega$ be a pf from $W$ to $A$ and $f$ be in $\Pi^m(W)$. We say that $\omega$ is differentiable by $f$ iff $\omega(x) = \omega(y)$ implies $\omega(x) = \omega(y)$ for each $x, y$ in $D(f) \cap D(\omega)$.

When $\omega$ is differentiable by $f$, the derivative of $\omega$ by $f$, denoted by $\delta_f \omega$, is defined by

1. $\delta_f \omega$ is a pf from $W^m$ to $A$,
2. $D(\delta_f \omega) = f(D(\omega))$, and
3. for each $x$ in $W^m$ and each $z$ in $A$, $\delta_f \omega(x) = z$ if there exists $y$ in $D(f) \cap D(\omega)$ such that $f(y) = x$ and $\omega(y) = z$.

From the above definition, the following proposition obviously holds.

**PROPOSITION 1.** Let $f$ be in $\Pi^m(W)$ and $\omega$ be a pf from $W$ to $A$.

1. If $\omega$ is differentiable by $f$, then $\delta_f \omega(x)$ is the unique element of the set $\omega(f^{-1}(x))$ for each $x$ in $D(\delta_f \omega)$,
2. If $f$ is 1-1, then $\omega$ is differentiable by $f$.
3. In case that $f$ is constant on $D(f) \cap D(\omega)$, $\omega$ is differentiable by $f$ iff $\omega$ is also constant on $D(f) \cap D(\omega)$.
4. If $\omega$ is a function and differentiable by an onto pf $f$, then $\delta_f \omega$ is also a function.

It should be noted that if a pf $\omega$ from $W$ to $A$ is differentiable by $f$ in $\Pi(W)$ then $\delta_f \omega$ is an element of $A$ or undefined because $\omega(D(f))$ is a subset of $A$ containing at most one element. For instance, if $\omega(D(f)) = \{a\}$, then $\delta_f \omega = a$.

3. Case-Splitting Transformation

**DEFINITION 2.** A subfamily $F$ of $\Pi(W)$ is said to be a case-splitting transformation (abbreviated to cst) over $W$ iff the domains of pf's in $F$ constitutes a finite division of $W$, that is,

1. $F$ is a finite family and the union of all $D(f)$'s for $f$'s in $F$ covers $W$, and
2. all $D(f)$'s for $f$'s in $F$ are not empty and mutually disjoint.

**DEFINITION 3.** Let $W$ be a well founded set with a partial order $\prec$ and $F$ be a cst over $W$. Then $F$ is said to be descending iff for each $f = (f_1, \ldots, f_m)$ in $F$ and for each $x$ in $D(f)$, $f_i(x) \leq x$ for $i = 1, \ldots, m$.

**DEFINITION 4.** A pf $\omega$ is said to be differentiable by a cst $F$ iff $\omega$ is differentiable by every $f$ in $F$.

If a pf $\omega$ from $W$ to $A$ is differentiable by a cst $F$ over $W$, then the following logical formula is valid for each $x$ in $W$:

$$\bigvee_{f \in F} [x \in D(f) \cap D(\omega) = \delta_f \omega(f(x))]$$
4. Finite Derivative Closure of Partial Functions

We denote the family of all functions from $A^n$ to $A$ by $\mathcal{Q}(A)$ and set $\mathcal{Q}(A) = \bigcup_{m=0}^{\infty} \mathcal{Q}_m(A)$. It should be noted that $\mathcal{Q}(A)$ is identified with $A$, and hence $\mathcal{Q}(A)$ includes $A$. Let $h$ be in $\mathcal{Q}_m(A)$ and $\omega_1, \ldots, \omega_m$ be pf's from $W$ to $A$. Then $h^*(\omega_1, \ldots, \omega_m)$ denotes a pf from $W^m$ to $A$ such that the domain of the pf is $D(\omega_1) \times \cdots \times D(\omega_m)$ and the value of the pf is $h(\omega_1(x_1), \ldots, \omega_m(x_m))$ for $x = (x_1, \ldots, x_m)$ in the domain.

For an $f$ in $\mathcal{F}(W)$, a subset $H$ of $\mathcal{Q}(A)$ and pf's $\omega_0, \omega_1, \ldots, \omega_n$ from $W$ to $A$, let $E(f, \omega_0, H, \omega_1, \ldots, \omega_n)$ denote the set of all tuples $(h; j_1, \ldots, j_m)$, where $m = d(f)$, $h$ is in $H \cap \mathcal{Q}_m(A)$ and $j_1, \ldots, j_m$ are in $\{0, 1, \ldots, n\}$ such that $\partial_j h \subseteq h^*(\omega_{j_1}, \ldots, \omega_{j_m})$.

**Definition 5.** A finite derivative closure of pf's (abbreviated to fdcp) is a system $(H; \omega_0, \omega_1, \ldots, \omega_n; F_0, F_1, \ldots, F_n)$ where $H$ is a subset of $\mathcal{Q}(A)$, $\omega_i$'s are pf's from $W$ to $A$ and $F_i$'s are cst's over $W$ such that $\omega_i$ is differentiable by $F_i$ for each $i = 0, 1, \ldots, n$ and the set $E(f, \omega_0, H, \omega_1, \ldots, \omega_n)$ is not empty for all $f$ in $F_i$. It is also denoted by $E(f, \omega_0, H)$.

If $W$ is a well founded set and all $F_i$'s are descending, then $\Gamma$ is called a halting fdcp. Each $\omega_i$ is said to be computable by $\Gamma$.

A fdcp $\Gamma = (H; \omega_0, \omega_1, \ldots, \omega_n; F_0, F_1, \ldots, F_n)$ gives the following recursive program:

$$\begin{align*}
\omega_0 &= \{ x \in D(f) \mid x \leq y \} \\
\omega_i &= h(\omega_{j_1}(f_{j_1}(x)), \ldots, \omega_{j_m}(f_{j_m}(x)))
\end{align*}$$

where $f = (f_1, \ldots, f_m)$ and $(h; j_1, \ldots, j_m)$ is in $E(f, \omega_0, H)$. If $\Gamma$ is halting fdcp, then the above program defines functions $\omega_0, \omega_1, \ldots, \omega_n$ such that $\omega_i \subseteq \omega_i$ for $i = 0, 1, \ldots, n$.

Thus, in some case, the problem of inference of functions from partial functions is solvable by finding a halting fdcp. The above program is called a tree-automaton-like program generated by $\Gamma$. If every $f$ in each $F_i$ satisfies that $d(f) \leq 1$, then the program is called an automation-like program.

5. Simple Inference

In this section, we suppose that $W$ is a well founded set with a partial order $\prec$.

**Definition 6.** A subset $H$ of $\mathcal{Q}(A)$ is said to be simple iff, for each $a$ in $A$, $h^{-1}(a)$ is finite for all $h$ in $H$ and there exist only a finite number of functions $h$ in $H$ such that $h^{-1}(a)$ is nonempty.

For a subset $D$ of $W$, we set $C(D) = \{ x \in W ; (\exists y \in D)[x < y] \}$. Let $H$ be a simple subset of $\mathcal{Q}(A)$, $F$ is a descending cst over $W$ and $\omega$ is a pf from $W$ to $A$ with a finite domain. Then we can obtain all $\omega_i$'s such that $\partial_j \omega \subseteq h^*(\omega_1, \ldots, \omega_m)$ for some $f$ in $F$ owing to the simplicity of $H$ and the finiteness of $D(\omega)$, where $m = d(f)$. Such $\omega_i$'s are said to be obtainable by $(\omega, F, H)$.

Now suppose that a pf $\omega$ from $W$ to $A$ such that $C(D(\omega))$ is finite, a simple subset $H$ of $\mathcal{Q}(A)$ and a finite family $G$ of descending cst's over $W$ are given. Then, checking all possibility, we can easily construct all fdcp $\Gamma = (H; \omega_0, \omega_1, \ldots, \omega_n; F_0, F_1, \ldots, F_n)$ such that $\omega = \omega_0$, all $F_i$'s are in $G$ and for $i = 1, \ldots, n$, $\omega_i$ is obtainable by $(\omega_j, F_j, H)$ for some $j < i$. An fdcp which is constructed as above is said to be simply constructable.
A simply constructable fdc(p) is a representation of a method of inference of a function \( \omega \) such that \( \omega \subseteq \overline{\omega} \). This inference is called simple inference.

We can use the simple inference for many sorts of objects, for example, regular languages, linear context-free languages, recognizable tree sets, generalized sequential machine (abbreviated to gsm) mappings and so on.

**Example 1 (Linear context-free languages).**

Let \( W = \Sigma^* \) for some alphabet \( \Sigma \) and \( A = \{ \text{true}, \text{false} \} \). We introduce a partial order \( < \) on \( W \) such that \( x < y \) means that \( x \) is a subword of \( y \). We define pf's \( f_\sigma \) and \( g_\sigma \) in \( \Pi_\sigma(W) \) for each \( \sigma \) in \( \Sigma \) by setting \( f_\sigma(\sigma x) = x \) and \( g_\sigma(\sigma x) = x \) for all \( x \) in \( W \). Then \( D(f_\sigma) = \sigma \cdot W \) and \( D(g_\sigma) = W \cdot \sigma \). We also consider a pf \( \varepsilon \) in \( \Pi_\varepsilon(W) \) such that \( D(\varepsilon) = \{ \varepsilon \} \). We set \( F^{(1)} = \{ \varepsilon \} \cup \{ f_\sigma ; \sigma \in \Sigma \} \) and \( F^{(2)} = \{ \varepsilon \} \cup \{ g_\sigma ; \sigma \in \Sigma \} \), and set \( G = \{ F^{(1)}, F^{(2)} \} \).

Suppose that two disjoint finite subsets \( S_1, S_2 \) of \( W \) are given. The pair \( (S_1, S_2) \) is identified with a pf \( \omega_0 \) from \( W \) to \( A \) with \( D(\omega_0) = S_1 \cup S_2 \) defined by \( \omega_0(x) = \text{true} \) if \( x \in S_1 \), and \( \omega_0(x) = \text{false} \) if \( x \in S_2 \). We set \( H = \{ \text{id} \} \cup A \), where \( \text{id} \) stands for the identity mapping on \( A \). Then, clearly we can take an fdc(p) \( \Gamma = (H; \omega_0, \omega_1, \cdots, \omega_n; F_0, F_1, \cdots, F_n) \) simply constructable by \( (\omega_0, H, G) \) with minimum length \( n \).

The automaton-like program generated by \( \Gamma \) computes functions \( \omega_0, \omega_1, \cdots, \omega_n \) satisfying \( \omega_0 \subseteq \omega_i \) for \( i = 0, 1, \cdots, n \), and if we set \( L_1 = \{ x \in W ; \omega_0(x) = \text{true} \} \) and \( L_2 = \{ x \in W ; \omega_0(x) = \text{false} \} \), then \( L_1 \) and \( L_2 \) are linear context-free languages such that \( S_1 \subseteq L_1 \) and \( S_2 \subseteq L_2 \). The linear context-free grammars \( G_1 \) and \( G_2 \) generating \( L_1 \) and \( L_2 \) respectively are obtained from \( \Gamma \) as follows: We take nonterminals \( N_0, N_1, \cdots, N_n \). \( N_0 \) is the start symbol. If \( F_i = F^{(1)} \) and \( \partial f_{\sigma_0} \omega_i = \omega_j \) then both \( G_1 \) and \( G_2 \) contain a production \( N_i \rightarrow \sigma N_j \). If \( F_i = F^{(2)} \) and if \( \partial g_{\sigma_0} \omega_i = \omega_j \) then both \( G_1 \) and \( G_2 \) contain a production \( N_i \rightarrow N_j \sigma \). If \( \partial \omega_0 = \text{true} \), then \( G_1 \) contains \( N_0 \rightarrow \varepsilon \). If \( \partial \omega_0 = \text{false} \), then \( G_2 \) contains \( N_0 \rightarrow \varepsilon \).

**Example 2 (gsm Mappings).**

We take the same \( W, f_\sigma's \) and \( F^{(1)} \) as in Example 1. We set \( A = \Delta^* \) for some alphabet \( \Delta \) and set \( H = A \cup \{ h_x ; x \in A \} \), where \( h_x(y) = xy \) for all \( x, y \) in \( A \). Then clearly \( H \) is simple. We put \( F = F^{(1)} \) and \( G = \{ F \} \). Suppose that a pf \( \omega_0 \) from \( W \) to \( A \) with a finite domain is given satisfying the condition that \( \omega_0 \) can be a restriction of some gsm mapping from \( W \) to \( A \). Then clearly we can take an fdc(p) \( \Gamma = (H; \omega_0, \omega_1, \cdots, \omega_n; F_0, F_1, \cdots, F_n) \) simply constructable by \( (\omega_0, H, G) \) with minimum length \( n \).

The automaton-like program generated by \( \Gamma \) computes functions \( \omega_0, \omega_1, \cdots, \omega_n \) such that \( \omega_0 \subseteq \omega_i \) for \( i = 0, 1, \cdots, n \). The gsm computing \( \omega_0 \) is obtained from \( \Gamma \) as follows: We take states \( q_0, q_1, \cdots, q_n \), where \( q_0 \) is the initial state. Let \( \delta \) and \( \lambda \) be the state-transition function and the output function of the gsm, respectively. If \( \partial f_{\sigma_0} \omega_i \subseteq h_x \cdot \omega_j \), then \( \delta(q_i, \sigma) = q_j \) and \( \lambda(q_i, \sigma) = x \).

For other interesting examples, the inference of pure LISP functions can be treated by considering cst's constituted by variously restricted car and cdr.

If we give a suitable \( H \) which is not simple and a suitable \( G \) such that we can construct an fdc(p) using the \( H \) and \( G \), then we can make a nonsimple inference.

The inference using halting fdc(p)'s is very powerful and it is significant to supply many examples although we mentioned in this paper only two of them.
Finite derivative closure of partial functions and a method of inference

References


Received October 16, 1987