

RESUMO

Os transformadores de predicados são um instrumento cada vez mais utilizado em diversos domínios da programação citam-se diversos exemplos de aplicação e referem-se os principais trabalhos teóricos sobre o assunto.

Desenvolve-se depois um modelo relacional para programas não deterministas sobre o qual se definem alguns transformadores de predicados úteis, entre os quais um que corresponde a função wp de Dijkstra. Por fim aproveita-se o modelo desenvolvido para formalizar o conceito de invariante. Mostra-se que os invariantes são pontos fixos de certos transformadores de predicados e refere-se o problema do cálculo de invariantes.

1. INTRODUÇÃO

Nos últimos anos tem-se assistido, em diversos domínios da Programação, a um aumento de interesse pelos transformadores de predicados tp's). Na origem deste interesse esta sem dúvida os notáveis trabalhos de Dijkstra, (Dij75) e em especial [Dij76] em que os tp's são utilizados de maneira marcante. Neles, Dijkstra define uma pequena linguagem não determinista <a "linguagem dos comandos guardados"> cuja caracterização semântica é feita em termos de um tp, desde então conhecido por "a função wp de Dijkstra". Ele utiliza depois esta função para a construção sistemática de programas correctos.

Este ultimo objectivo é também o de outros autores. por exemplo, Sintzoff e van Lamsweerde [LaS79]. que usam os tp's também no caso de programas paralelos, em particular nos aspectos de construção (Sin79), e melhoramento (i.e., a modificação de programas para que eles passem a ter algumas propriedades suplementares), (Lam78).

Na Área dos programas paralelos situam-se ainda os trabalhos de Flon e Suzuki [F1S78a] [F1S78b], que se dedicam aos problemas de verificação. o método que estes autores usam, consiste em transformar os programas paralelos em programas sequenciais não deterministas, sobre os quais a então feita a verificação, utilizando tp's. Para cada propriedade estudada, o tp adequado permite obter as precondições que garantem a validade da propriedade. Supõe-se que as precondições assim encontradas são aplicáveis ao programa inicial.

A verificação de programas paralelos a igualmente um dos objectivos de Sifakis [Sif79a] [sif79b]. Neste caso, os programas são modelizados por meio de sistemas de transições [Kel76] No modelo, é definido um conjunto coerente de tp's, que permite classificar e estudar as propriedades que, em geral, se quer poder estabelecer para os programas paralelos.

Noutro domínio se situa o trabalho recente de Guttag [Gut79], o qual desenvolve uma técnica de especificação que combina tp's e axiomas algébricos.

Do ponto de vista puramente teórico, vários trabalhos sobre tp's têm sido publicados. Além dos de Dijkstra já citados, refiramos os de Hoare, de Roever, Milne, de Bakker, Wand e Best.

Hoare [Hoa78a] propõe um modelo para a linguagem dos comandos guardados, baseado em "traces", isto é, sequências descrevendo a execução de um programa. O seu objectivo principal é- demonstrar a consistência dos axiomas de Dijkstra para a linguagem (Dij76). Para isso é necessário caracterizar rigorosamente a função wp no modelo dos "traces

De Roever [Roe76] estuda a função wp, formalizada a partir de relações binárias. Atenção especial é dedicada a influência do não-determinismo sobre as propriedades de continuidade, [Dij76, cap 9].

Milne [Mil78] e de Bakker [Bak7B] consideram sobretudo as relações que existem entre os tp's e outras técnicas de definição de linguagens de programação.

Wand [Wan77] e Best [Bes79] estudam em particular o problema de descobrir, a partir de um tp com

certas propriedades, qual o "mecanismo" que lhe está associado.

Verifica-se, pois, que os tp's constituem um instrumento cada vez mais utilizado, nos vários domínios, e ao mesmo tempo um objecto matemático cujas propriedades se procura conhecer. Estes dois aspectos não são, evidentemente independentes. As aplicações indicam que tipo de tp's e que tipo de propriedades a preciso estudar os resultados teóricos, que traduzem um melhor domínio do conceito, permitem utiliza-lo em mais situações e com maior segurança.

Em principio, o conceito de tp é simples. Como o nome indica, trata-se de uma aplicação de um conjunto de predicados num conjunto de predicados. Esta noção elementar nem sempre é posta em evidencia, talvez porque suposta trivial. Contudo, as notações utilizadas nem sempre são claras quanto ao aspecto funcional dos tp's. Dijkstra, por exemplo, afirma [Dij76, pag 17]: "For a fixed mechanism S the rule which is fed with the predicate R , ..., and delivers a predicate $wp(S,R)$, is called a "predicate transformer". Portanto, o tp em causa é a função $wp(S)$ e não a função wp .

O objectivo deste artigo é ilustrar o conceito de tp e através da análise de alguns tp's Úteis e das suas propriedades, e mostrar como a partir de ideias simples é possível chegar a uma caracterização formal da função wp de Dijkstra. A nossa aproximação baseia-se num modelo relacional para programas não-deterministas, isto é, admitimos que os programas em que estamos interessados podem ser modelizados por meio de relações binárias e utilizamos essas relações como suporte de raciocínio, em vez dos programas.

O modelo relacional escolhido servir-nos-a ainda para formalizar o conceito de invariante, também instrumento indispensável quando se raciocina sobre programas, mas cuja caracterização fica geralmente ao nível da intuição informal.

O tratamento apresentado neste artigo foi abreviado e as demonstrações omitidas. O leitor interessado poderá fazê-las ele próprio ou consultar as referências, em particular [Gue79].

2 - O MODELO DE BASE

O comportamento de um programa determinista é geralmente caracterizado por uma função de um conjunto de dados num conjunto de resultados, ou, mais abstractamente, de um conjunto conveniente de estados nele mesmo. Por analogia, esperamos que o comportamento de um programa não-determinista, entendido como um programa que pode produzir vários resultados a partir de um mesmo estado inicial, possa ser descrito por uma relação binária num tal conjunto de estados.