

und.1 O Problema da Decisão

tur:und:dec:
sec

Dizemos que a lógica de primeira ordem é *decidível* se, e somente se, há um método efetivo para determinar se uma dada **sentença** é válida ou não. Acontece que não existe tal método: o problema de decidir a validade de sentenças de primeira ordem é insolúvel.

A fim de estabelecer esse importante resultado negativo, provamos que o problema da decisão não pode ser resolvido por uma máquina de Turing. Isto é, mostramos que não existe máquina de Turing que, sempre que iniciada em uma fita que contém uma **sentença** de primeira ordem, eventualmente para e produz 1 ou 0 conforme a **sentença** seja válida ou não. Pela tese de Church-Turing, toda função que é computável é Turing-computável. Logo, se essa “função de validade” fosse de algum modo efetivamente computável, ela seria Turing-computável. Se ela não é Turing-computável, então também não pode ser efetivamente computável.

Nossa estratégia para provar que o problema da decisão é insolúvel é reduzir o problema da parada a ele. Isso significa o seguinte: provamos que a função $h(e, w)$ que para com saída 1 se a máquina de Turing descrita por e para na entrada w e produz 0 caso contrário não é Turing-computável. Mostraremos que se houvesse uma máquina de Turing que decide a validade de sentenças de primeira ordem, então haveria também uma máquina de Turing que computa h . Como h não pode ser computada por uma máquina de Turing, também não pode haver uma máquina de Turing que decide a validade.

O primeiro passo nessa estratégia é mostrar que, para toda entrada w e máquina de Turing M , podemos descrever efetivamente a **sentença** $\tau(M, w)$ representando o conjunto de instruções de M e a entrada w , e a **sentença** $\alpha(M, w)$ exprimindo “ M eventualmente para”, tal que:

$$\models \tau(M, w) \rightarrow \alpha(M, w) \text{ se, e somente se, } M \text{ para na entrada } w.$$

A maior parte de nossa prova consistirá em descrever essas sentenças $\tau(M, w)$ e $\alpha(M, w)$ e em verificar que $\tau(M, w) \rightarrow \alpha(M, w)$ é válida se, e somente se, M para na entrada w .

Créditos das Imagens

Referências Bibliográficas