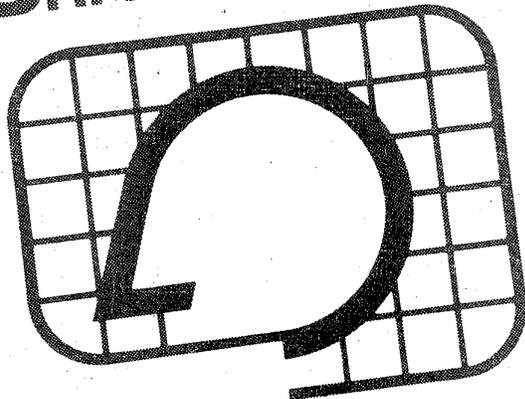


**2º SIMPÓSIO  
BRASILEIRO DE**



**INTELIGÊNCIA  
ARTIFICIAL**

006.306  
S612

---

**ANAIS**

**29 SIMPÓSIO BRASILEIRO DE INTELIGÊNCIA ARTIFICIAL**

**20-21-22 DE NOVEMBRO DE 1985**

**INPE - INSTITUTO DE PESQUISAS ESPACIAIS  
SÃO JOSÉ DOS CAMPOS - SP**

# RESOLUÇÃO DE PROBLEMAS POR ABSTRAÇÃO : O MÉTODO GULOSO

Christina Fraga E. Maciel Waga

Paulo Augusto Silva Veloso

Departamento de Informática, PUC/RJ

22453 Rio de Janeiro RJ

## I. Introdução

Encarando um problema como sendo uma pergunta geral a ser respondida podemos dizer que um algoritmo descreve uma seqüência ordenada de "ações", que dá a resposta desejada.

Assim, métodos para resolução de problemas, são métodos de construção de algoritmos.

Formalizar certo método, ou seja especificar um certo mecanismo, leva a um melhor entendimento de sua aplicabilidade e maior facilidade na obtenção de algoritmos.

## II. Visão Informal

O Método Guloso é uma técnica de fácil aplicação e entendimento empregada na solução de problemas. O algoritmo a ser projetado utilizando-se este método funcionará em estágios, tomando-se, um a um, os objetos da entrada. Em cada estágio, uma decisão será tomada: se o objeto em questão fará ou não parte da solução ótima do problema.

Para que esta decisões sejam tomadas sem erros, os objetos da entrada devem obedecer a alguma ordem de seleção ou política de seleção, baseada em uma medida de otimização associada ao problema.

O Método Guloso tenta resolver problemas combinatoriais de otimização. Desta forma, temos:

- para cada estágio  $i$ , temos um conjunto solução viável parcial  $S_i$ , onde  $i = 1, \dots, n$
- o algoritmo escolhe um objeto  $p$  da entrada
- Se  $S_i \cup \{p\}$  é admissível, então  $p$  é incluído, senão  $p$  é rejeitado e
- quando  $i=n$ ,  $S_n$  é a solução ótima para o problema.

### III. Visão Formal

#### III.1 Especificação Sintática

##### III.1.1 Tipo Abstrato de Dados

###### i. Sortes

Temos os seguintes sortes descritores dos domínios sintáticos do método: D de instâncias de problema, R de resultados, A de instâncias específicas e I de "elementos" de A.

###### ii. Funções

Existem duas funções de inicialização: init:  $D \rightarrow R$ , que a partir do domínio de entrada D inicializa o sorte de soluções R, e choice:  $D \rightarrow A$ , que faz uma redução do domínio D.

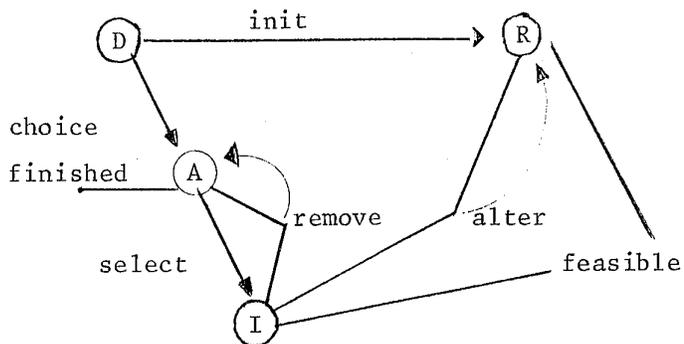
As funções remove:  $I \times A \rightarrow A$  e alter:  $I \times R \rightarrow R$  são de atualização sobre A e R, respectivamente.

A função select:  $A \rightarrow I$  seleciona um objeto a ser analisado, do domínio A.

###### iii. Predicados

Temos dois predicados bem intuitivos: finished em A, que detecta término de observação de elementos, e feasible em  $I \times R$ , que testa se é viável acrescentar um elemento a uma certa solução parcial.

##### III.1.2 Diagrama



### III.1.3 Programa Abstrato

```
PROCEDURE Guloso
  {d : entrada
   r : saída
   a, i: auxiliares}
BEGIN
  r := init (d);
  a := choice (d);
  WHILE ¬ finished (a) DO
    BEGIN
      i := select (a);
      a := remove (i,a);
      IF feasible (i,r)
      THEN
        r := alter (i,r)
    END
  END.
END.
```

### III.2 Especificação Semântica $\neg$ Axiomatização

Para a especificação semântica do tipo de dado é necessário definir:

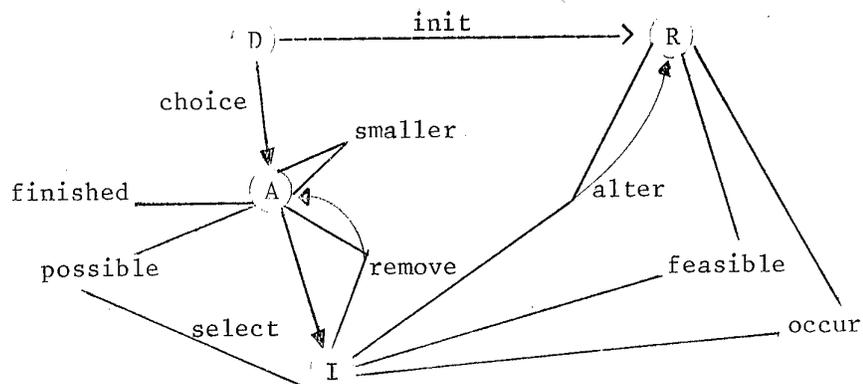
- um predicado possible em  $I \times A$ , que intuitivamente exprime a pertinência de um elemento em relação a  $A$  e
- um predicado occur em  $I \times R$ , que indica pertinência de um elemento em  $R$ ;
- um predicado smaller em  $A$ , que indica que o domínio  $A$  está sendo "esvaziado".

Assim temos os axiomas:

- Ax1.  $\forall j$  [occur (j,alter(select(a),r))  $\rightarrow$  j=select (a) v occur (j,r)]
- Ax2.  $\forall j$  [occur (j,init(d))  $\rightarrow$  possible (j,choice(d))]
- Ax3.  $\forall j$  [possible (j,remove(select(a),a))  $\rightarrow$  ¬ j=select(a) & possible (j,a)]
- Ax4. finished (a)  $\rightarrow$   $\forall j$  [¬ possible (j,a)]
- Ax5. ¬ finished (a) & feasible (select(a),r)  $\rightarrow$   $\forall j$  [feasible (j,r)  $\rightarrow$  feasible (j,alter(select(a),r))]
- Ax6. ¬ finished (a) & feasible (select(a),r)  $\rightarrow$   $\forall j$  [¬ feasible (j,r) & feasible (j,alter(select(a),r))  $\rightarrow$  possible (j,a)]
- Ax7.  $\forall j$  [¬ feasible (j,init(d))  $\rightarrow$  possible (j,choice(d)) & ¬ occur (j,init(d))]
- Ax8. ¬ smaller (a,a)
- Ax9. ¬ finished (a)  $\rightarrow$  smaller (remove(select(a),a),a)
- Ax10. smaller é bem fundada

Com esses axiomas pode-se provar a correção total do programa abstrato apresentado acima.

### III.2.1 Diagrama Expandido



### IV. Problemas

Os problemas típicos resolvidos pelo Método Guloso se enquadram nesta abordagem:

- . caminho de custo mínimo em grafos,
- . problema da mochila,
- . seqüência ótima de intercalação de listas,
- . árvores geradoras mínimas, etc.

### V. Conclusões:

Com a formalização de um método, procura-se, através de um entendimento melhor e mais preciso, alterar a conduta de trabalho face a um problema a ser resolvido. Pois, sem formalização, o entendimento do método é, de uma maneira geral, informal e decorrente da análise da solução de alguns problemas clássicos e a solução de um problema é, muitas vezes, obtida por meio de tentativa e erro. Outro aspecto é o da solução de um problema dada uma formalização para o método. Solucionar é sinônimo de interpretar um tipo abstrato de dados e provar que a solução encontrada está correta é traduzir axiomas para o escopo do problema.

### Referências Bibliográficas

- HOROWITZ, E. & SAHNI, S. Fundamentals of computer algorithms. Rockville, Md. Computer Science Press, 1978.
- VELOSO, P.A.S. Outlines of a mathematical theory of general Problems. Philosophia Naturalis, vol. 21, nº 2/3, 1984.
- WAGA, C.F.E.M. Métodos para resolução de problemas. Rio de Janeiro, PUC, D.I., 1984, Dissertação de Mestrado.