Programação Genética

Na programação genética, os indivíduos da população não são seqüências de bits, mas sim programas de computador, armazenados na forma de árvores sintáticas. Tais programas é que são os candidatos à solução do problema proposto. Programação genética não usa o operador mutação e a recombinação se dá pela troca de sub-árvores entre dois indivíduos candidatos a solução.

Implementação

Para implementar este engenho evolutivo, tomaram-se as seguintes definições operacionais.

1. As operações envolvidas são AND (com valência 2), OR (2), NOT (1) e IF (3). Os sensores são 8: n, ne, e, se, s, sw, w e nw. As ordens são $north$, $east$, $south$ e $west$. As constantes são 0 e 1.
2. a definição de AND(x,y) é
   se x=0 **então**
   AND(x,y) vale 0
   senão
   AND(x,y) deve ser substituída por y
   fim se
3. a definição de OR(x,y) é
   se x=1 **então**
   OR(x,y) vale 1
   senão
   OR(x,y) deve ser substituída por y
   fim se
4. a definição de NOT(x) é
   se x=1 **então**
   NOT(x) vale 0
   senão
   NOT(x) vale 1
   fim se
5. a definição de IF(x,y,z) é
   se x=1 **então**
   IF(x,y,z) deve ser substituída por y
   senão
   IF(x,y,z) deve ser substituída por z
   fim se
6. A avaliação de um programa indivíduo começa com a localização aleatória do robot em (x_i, y_i). O programa deve ser repetido enquanto houver movimento do robot. Quando a aplicação do programa não resultar em movimento, (isto é, $x_{n+1} = x_n$ e $y_{n+1} = y_n$, todo o ciclo deve ser encerrado.
7. Quando uma ordem é usada como entrada, por exemplo em `(OR (e) (west))`, a ordem (neste caso $west$) é executada, e não sobe nenhum resultado na árvore, pelo que o programa pára.
8. Quando um sensor é usado em uma saída (por exemplo em `IF(0 (west) (e))`) todo o IF vale o sensor (neste caso `e`) que é levado para cima na árvore e, segue o baile!
9. o programa é avaliado pela ordem (da raiz para baixo), até que o robot se movimente ou o programa termine. Se o robot se movimenta, esta encarnação do programa é encerrada e volta a repetição, agora com uma nova localização do robot. Se o programa termina sem movimento, acabou a avaliação deste indivíduo.
10. O laberinto é sempre 8×8 e é fornecido em uma matriz contendo 0s e 1s, onde 1 significa caminho livre e 0 significa obstáculo. Sempre há pelo menos dois espaços contendo 1 entre paredes paralelas (um para ir e outro para voltar). Por exemplo

1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1	1
3	1	1	1	1	1	1	1	1
4	1	1	1	0	1	1	1	1
5	1	1	1	0	1	1	1	1
6	1	1	1	0	1	1	1	1
7	1	1	1	0	1	1	1	1
8	1	1	1	0	1	1	1	1

Na análise do valor do sensor, deve-se imaginar a resposta à seguinte pergunta: "existe parede nessa direção?". Se SIM (e o robot não pode ir lá), a resposta é 1. Se NÃO a resposta é 0.

As direções são:

nw	n	ne
w	robot	e
sw	s	se

11. a análise sempre vem da direita para a esquerda.

Exemplo feito

1. Seja o indivíduo, com o robot colocado em (6,1) e sendo o laberinto o mesmo que consta anteriormente nesta página.

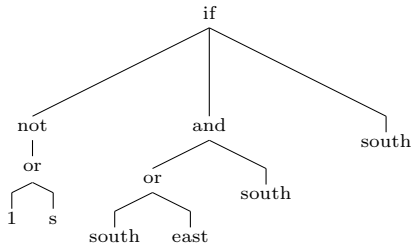
   `(if north,(not (if east,0,(not 1))),south)`

   Devemos descobrir qual é o tamanho do caminho percorrido pelo robot. Ao olhar a expressão, devemos lembrar que a análise **sempre** vem da direita para a esquerda. O primeiro parênteses, deve ser ignorado, pois ele é o fecho total da expressão. Assim, a primeira coisa a verificar é $south$ que é obedecida e o robot anda 1 posição indo para (7,1). O ciclo recomeça e por idênticas razões ele vai para a posição 8,1. Daí em diante ele não se move mais, o ciclo termina e a resposta é 2. Para nossa sorte, neste exemplo não foi necessário desenvolver os sinais n, ne ..., mas isso foi apenas sorte.
2. Sejam os indivíduos:

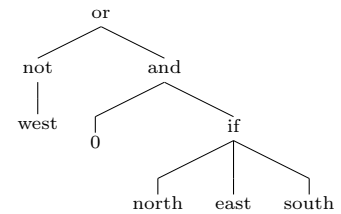
```
(if (not (or 1,s)),(and (or south,east),south),south)
12345678901234567890123456789012345678901234567890123
         1         2         3         4         5
```

   e

```
(or (not west),(and 0,(if north,east,south)))
1234567890123456789012345678901234567890123 45
         1         2         3         4
```



1: 10, 8 - v2
2: 5,14 - v1
3: 25,15 - v2
4: 20,27 - v2
5: 1,53 - v3



a: 16,29 - v2
b: 1,45 - v2
c: 5,10 - v1
d: 23,21 - v3

E finalmente, pareando as valências iguais, temos os seguintes pares possíveis: (1,a), (1, b), (2, c), (3, a), (3, b), (4, a), (4, b) e (5, d). Total= 8 pares.

☞ Para você fazer

1. Seja o laberinto acima, o robot colocado na posição: linha 3 e coluna 3 . Seja o indivíduo em análise

   `(AND east,(IF west,south,(AND (IF s,nw,south),0)))`

   Simule a avaliação deste cromossomo e informe o fitness alcançado:

2. Seja agora uma operação de *crossover*. Seja o ascendente 1

   `(AND e,(AND (IF sw,north,(IF north,nw,(NOT west))),ne))`

   e seja o ascendente 2

   `(OR (NOT n),(AND east,(AND east,south)))`

   A pergunta é: quantas são as possibilidades de operação de crossover para estes dois casos ? (sugestão: desenhe as árvores e considere as valências 1, 2 ou 3, conforme o caso.