

PRACTICA VI: MÉTODOS DE CRUZA O RECOMBINACIÓN

ALGORITMOS GENÉTICOS

3CM5

Colín Varela, Alejandro

Contents

1	Introducción	2
2	Desarrollo	2
2.1	Cruza de un punto	3
2.2	Cruza de dos puntos	3
2.3	Cruza Uniforme	4
2.4	Cruza Acentuada	5
3	Pruebas	6
3.1	Cruza de un punto	8
3.2	Cruza de dos puntos	9
3.3	Cruza Uniforme	10
3.4	Cruza Acentuada	11
4	Conclusiones	11

1 Introducción

Con la experiencia que contamos ahora vamos a probar distintos métodos de cruce para poder optimizar ese paso en la construcción de un algoritmo genético, convenientemente tenemos cada uno de ellos en un menú.

2 Desarrollo

Primero se muestra la parte del código que construirá el menú una vez declaradas las variables a usar.

```
printf("\n\nPractica 6: Métodos de cruce o recombinación\n");
printf("Alumno: Alejandro Colin Varela\n");
printf("Menu:\n");
printf("1) s Cruce de un punto\n");
printf("2) Cruce de dos puntos\n");
printf("3) Cruce Uniforme\n");
printf("4) Cruce Acentuada\n\n");
printf("0) Salir\n");
printf(": ");
scanf("%d",&op);
//Generamos los padres para la cruce
int A1[8],A2[8],B1[8],B2[8],i,H1[8],H2[8];
for(i=0;i<8;i++)
{
    A1[i] = rand()%2;
    A2[i] = rand()%2;
    B1[i] = rand()%2;
    B2[i] = rand()%2;
}
```

Veamos los códigos de las cuatro partes del menú.

2.1 Cruza de un punto

```
// Cruza de un punto
if(op==1)
{
    int num = (rand()%7) + 1;
    printf("\nCruza de un punto en %i\n", num);
    for(i=0;i<num;i++)
    {
        H1[i] = A1[i];
        H2[i] = B1[i];
    }
    for(i=num;i<8;i++)
    {
        H1[i] = A2[i];
        H2[i] = B2[i];
    }
}
```

Tan sencillo como calcular un punto de cruce y decirle al programa que copie del primer padre hasta el punto de cruce y del segundo padre herede del punto de cruce al final del padre.

2.2 Cruza de dos puntos

```
//Cruza de dos puntos
else if(op==2)
{
    int num = (rand()%7) + 1,num2;
    do
    {
        num2 = (rand()%7) + 1;
    }while(num==num2);
    printf("\nCruza de dos puntos en %i y %i\n", num,num2);
    if(num2<num)
    {
        int hp = num2;
        num2 = num;
        num = hp;
    }
    for(i=0;i<num;i++)
    {
        H1[i] = A1[i];
        H2[i] = B1[i];
    }
    for(i=num;i<num2;i++)
    {
        H1[i] = A2[i];
        H2[i] = B2[i];
    }
    for(i=num2;i<8;i++)
    {
        H1[i] = A1[i];
        H2[i] = B1[i];
    }
}
```

Exactamente lo mismo que con el punto anterior pero con dos puntos, de inicio al punto 1, del punto 1 al punto 2 y luego del punto 2 al final.

2.3 Cruza Uniforme

```
//Cruza Uniforme
else if(op==3)
{
    int m[8];
    for(i=0;i<8;i++)
    {
        m[i] = rand()%2;
    }
    for(i=0;i<8;i++)
    {
        if(m[i]==1)
        {
            H1[i] = A1[i];
            H2[i] = B1[i];
        }
        else
        {
            H1[i] = A2[i];
            H2[i] = B2[i];
        }
    }
    printf("\nCruza Uniforme con la\n");
    printf("Mascara : \t");
    for (i = 0; i < 8; ++i)
    {
        printf("%i | ", m[i]);
    }
}
```

Para la cruce uniforme vamos a generar primero una mascara aleatoria que se utilizara para todos los cruces, después hacemos un for y dependiendo de los 1's y 0's de la mascara decidirá tomar del padre 1 o del padre 2.

2.4 Cruza Acentuada

```
//Cruza Acentuada
else if(op==4)
{
    int m1[8],m2[8];
    for(i=0;i<8;i++)
    {
        m1[i] = rand()%2;
        m2[i] = rand()%2;
    }
    printf("\n\nPadre 1 : \t");
    for (i = 0; i < 8; ++i)
    {
        if(m1[i]==1)
            printf("%i! | ", A1[i]);
        else
            printf("%i | ", A1[i]);
    }
    printf("\nPadre 2 : \t");
    for (i = 0; i < 8; ++i)
    {
        if(m2[i]==1)
            printf("%i! | ", A2[i]);
        else
            printf("%i | ", A2[i]);
    }

    int cnt=0;
    for(i=0;i<8;i++)
    {
        if(m2[i]==1)
            m1[i] = 1;
        if(m1[i]==1)
            cnt = abs(cnt-1);
        if(cnt==0)
        {
            H1[i] = A1[i];
            H2[i] = A2[i];
        }
        else
        {
            H1[i] = A2[i];
            H2[i] = A1[i];
        }
    }
}
```

Para esta cruce llenamos a los padres con símbolos '!' (Que en realidad son ceros y unos en un arreglo) y le decimos al algoritmo que cambie de padre cada que encuentra una '!', EL hijo 2 suele heredar estos '!'.

Finalmente terminamos imprimiendo los padres y los hijos, anexando una confirmación para regresar al menú.

```
if(op==1||op==2||op==3)
{
    printf("\n\nPadre 1 : \t");
    for (i = 0; i < 8; ++i)
    {
        printf("%i | ", A1[i]);
    }
    printf("\n\nPadre 2 : \t");
    for (i = 0; i < 8; ++i)
    {
        printf("%i | ", A2[i]);
    }
    printf("\n\nHijo : \t\t");
    for (i = 0; i < 8; ++i)
    {
        printf("%i | ", H1[i]);
    }
    printf("\n\nPadre 3 : \t");
    for (i = 0; i < 8; ++i)
    {
        printf("%i | ", B1[i]);
    }
    printf("\n\nPadre 4 : \t");
    for (i = 0; i < 8; ++i)
    {
        printf("%i | ", B2[i]);
    }
    printf("\n\nHijo : \t\t");
    for (i = 0; i < 8; ++i)
    {
        printf("%i | ", H2[i]);
    }
    char c = getchar();
    printf("\n\nENTER para volver al menu\n");
    c = getchar();
}
```

3 Pruebas

Nota: Favor de leer el README.txt para mas información sobre como compilar y correr la practica

Una vez compilado usando 'make' vamos a correr usando './correr' y nos debe aparecer el siguiente menú:

```
Terminal - colin@Manjaro-de-Colin-42:~/Documents/Escuela/Algoritmos Geneticos/Practica 6 AG Colin_Varela
File Edit View Terminal Tabs Help
[colin@Manjaro-de-Colin-42 Practica 6 AG Colin_Varela]$ ./correr

Practica 6: Métodos de cruza o recombinación
Alumno: Alejandro Colin Varela
Menu:
1) Cruza de un punto
2) Cruza de dos puntos
3) Cruza Uniforme
4) Cruza Acentuada
0) Salir
: 
```

En este menú podemos escoger cualquiera de las cuatro formas de cruza.

3.1 Cruza de un punto

```
Terminal - colin@Manjaro-de-Colin-42:~/Documents/Escuela/Algoritmos Geneticos/Practica 6 AG Colin_Vare
File Edit View Terminal Tabs Help
Menu:
1) Cruza de un punto
2) Cruza de dos puntos
3) Cruza Uniforme
4) Cruza Acentuada
0) Salir
: 1
Cruza de un punto en 7
Padre 1 :      0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 |
Padre 2 :      0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 |
Hijo :        0 | 1 | 1 | 0 | 1 | 0 | 1 | 1 |
Padre 3 :      1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |
Padre 4 :      0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |
Hijo :        1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 |
ENTER para volver al menu
```

Lo primero que hará el programa es avisarnos en que bit será el cruce, posteriormente solo se imprimirán a los padres y a su respectivo hijo.

3.2 Cruza de dos puntos

```
Terminal - colin@Manjaro-de-Colin-42:~/Documents/Escuela/Algoritmos Geneticos/Practica 6 AG Colin_Vare
File Edit View Terminal Tabs Help
Menu:
1) Cruza de un punto
2) Cruza de dos puntos
3) Cruza Uniforme
4) Cruza Acentuada
0) Salir
: 2
Cruza de dos puntos en 3 y 2

Padre 1 :      0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |
Padre 2 :      0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 |
Hijo :        0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 |
Padre 3 :      0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 |
Padre 4 :      0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
Hijo :        0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 |
ENTER para volver al menu
```

Exactamente como la versión de un punto, pero en esta ocasión desde el programa vemos los dos puntos que se utilizaran para esa cruce.

3.3 Cruza Uniforme

```
Terminal - colin@Manjaro-de-Colin-42:~/Documents/Escuela/Algoritmos Geneticos/Practica 6 AG Colin_Vare
File Edit View Terminal Tabs Help
Menu:
1) Cruza de un punto
2) Cruza de dos puntos
3) Cruza Uniforme
4) Cruza Acentuada
0) Salir
: 3
Cruza Uniforme con la
Mascara : 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 |
Padre 1 : 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 |
Padre 2 : 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 |
Hijo : 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 |
Padre 3 : 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 |
Padre 4 : 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
Hijo : 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 |
ENTER para volver al menu
```

Aquí se nos muestra la mascara que se utilizó para hacer la cruza, los padres y sus respectivos hijos.

3.4 Cruza Acentuada

```
Terminal - colin@Manjaro-de-Colin-42:~/Documents/Escuela/Algoritmos Geneticos/Practica 6 AG Colin_Varela
File Edit View Terminal Tabs Help

Practica 6: Métodos de cruza o recombinación
Alumno: Alejandro Colin Varela
Menu:
1) Cruza de un punto
2) Cruza de dos puntos
3) Cruza Uniforme
4) Cruza Acentuada
0) Salir
: 4

Padre 1 :      0! | 0 | 1! | 1 | 0! | 0 | 0 | 0 |
Padre 2 :      1! | 0! | 1 | 0! | 1 | 0 | 1! | 0! |
Hijo 1 :       1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 |
Hijo 2 :       0! | 0! | 1! | 0! | 0! | 0 | 1! | 0! |

ENTER para volver al menu
```

Finalmente en esta cruza , se nos mostraran los bits incluyendo las '!' que se han anexado, se ve en el segundo hijo los '!' heredados.

4 Conclusiones

Una vez que hemos entendido la manera en la que se seleccionan los padres en un algoritmo genético, también es muy bueno saber como se cruzan para tener los mejores hijos que pueden tener. Con esta practica los aprendimos a programar y a representar estos métodos de cruza.