

INSTITUTO POLITÉCNICO NACIONAL
ESCUELA SUPERIOR DE CÓMPUTO

ALGORITMO GENÉTICO DE SELECCIÓN POR JERARQUÍA

Práctica 5

Dominguez de la Rosa Bryan

GRUPO 3CM5

Profesor: Morales Güitron Sandra Luz

25 de octubre de 2018

Introducción

La selección por jerarquía fue propuesta por Baker para evitar la convergencia prematura en las técnicas de selección proporcional. El objetivo de esta técnica es disminuir la presión de selección. En este caso, discutiremos el uso de jerarquías lineales, pero es posible también usar jerarquías no lineales, aunque la presión de selección sufre cambios más abruptos al usarse esta última. Los individuos se clasifican con base en su aptitud, y se les selecciona con base en su rango (o jerarquía) y no con base en su aptitud. El uso de jerarquías hace que no se requiera escalar la aptitud, puesto que las diferencias entre las aptitudes absolutas se diluyen. Asimismo, las jerarquías previenen la convergencia prematura (de hecho, lo que hacen, es alentar la velocidad convergencia del algoritmo genético).

Contenido

Para la implementación del algoritmo de selección por jerarquía, implementé 4 arreglos de bits para controlar las distintas etapas que se realizan en el algoritmo:

- Población inicial.
- Población de individuos seleccionados mediante el algoritmo de selección por torneo.
- Población después de cruza.
- Población después de mutación.

En la primer etapa, se llena aleatoriamente el arreglo de población inicial con series de 5 bits. Después se realiza una selección por ranking o jerarquía, el algoritmo inicia ordenando la población con base en su aptitud, de 1 a N (donde 1 representa al menos apto). Posteriormente el algoritmo es el siguiente:

- Elegir $Max(1 \leq Max \leq 2)$
- Calcular $Min = 2 - Max$
- El valor esperado de cada individuo será:
$$Valesp(i, t) = Min + (Max - Min) \frac{jerarquía(i, t) - 1}{N - 1}$$

Baker recomendó $Max = 1.1$
- Usar selección proporcional aplicando los valores esperados obtenidos de la expresión anterior.

Figura 1: Algoritmo de selección por jerarquía o ranking

Una vez teniendo la población de selección de padres, se realiza una cruza de individuos de la siguiente manera:

1. Se utilizan 2 individuos de la población de padres.
2. Se define un punto de cruza estático para todas las generaciones.
3. Se cruzan los individuos.
4. Se retorna el individuo resultante.

```
bitset<BITS_PER_INDIVIDUAL> crossAlgorithm(bitset<BITS_PER_INDIVIDUAL> &p1, bitset<BITS_PER_INDIVIDUAL> &p2, int cross_point) {  
  
    bitset<BITS_PER_INDIVIDUAL> aux = p1;  
  
    for (int i = 0; i <= cross_point; i++)  
    {  
        aux.set(cross_point - i, p2[cross_point - i]);  
    }  
  
    return aux;  
}
```

Figura 2: Algoritmo de cruce de individuos

Al obtener la población de individuos después de la cruce se necesita realizar una mutación. En este caso se generó una mutación del 10 % de la población. Nuestra población total es de 32 elementos, entonces la cantidad de individuos a redondear es 3.2, redondeado como 3.

La mutación se realiza de la siguiente forma:

1. El algoritmo se realizará 3 veces.
2. La mutación buscará mejorar al individuo, por lo tanto, se buscará cambiar un bit 0 por un bit 1.
3. Debido a que se requiere buscar un 0 en el individuo a mutar, y es posible que el individuo no tenga bits 0, se define un número máximo de iteraciones para evitar que el programa se cicle.
4. Cuando se encuentre un bit 0, se cambia por un bit 1.

```
bitset<BITS_PER_INDIVIDUAL> mutationAlgorithm(bitset<BITS_PER_INDIVIDUAL> individual){  
    bitset<BITS_PER_INDIVIDUAL> result = individual;  
  
    int cont = 0;  
  
    while(cont <= MAX_SEARCH_VALUE)  
    {  
        int mutation_point = rand() % BITS_PER_INDIVIDUAL;  
        if(result[mutation_point] == 0){  
            result.set(mutation_point, 1);  
            break;  
        }  
        cont++;  
    }  
  
    return result;  
}
```

Figura 3: Algoritmo de mutación de individuos

Una vez que se tenga la población mutada, se establece ésta como población inicial, para realizar el algoritmo de ruleta en la siguiente generación.

Al obtener la población final de una generación, se obtiene la aptitud del individuo de menor valor, la aptitud del individuo de mayor valor y el promedio de aptitud de cada generación.

A continuación se muestran 2 ejemplos con 10, 30, 50 y 100 generaciones, en los que la línea azul representa la aptitud del mejor individuo de cada generación, la línea roja representa la aptitud del peor individuo de cada generación y la línea blanca representa el promedio de aptitud de cada generación.

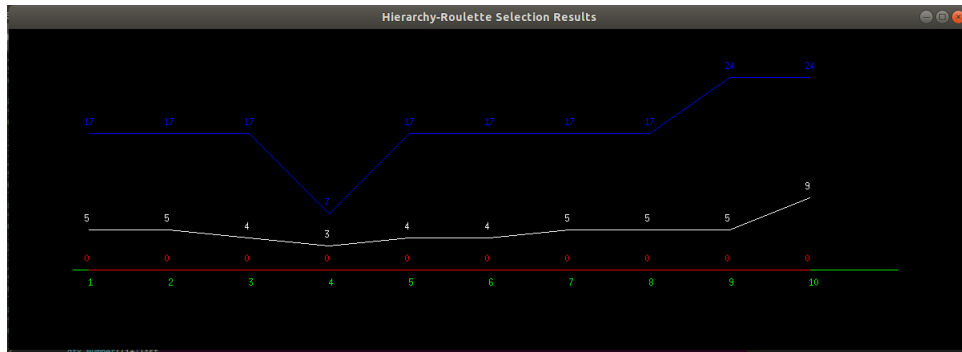


Figura 4: Resultado 1 con 10 generaciones

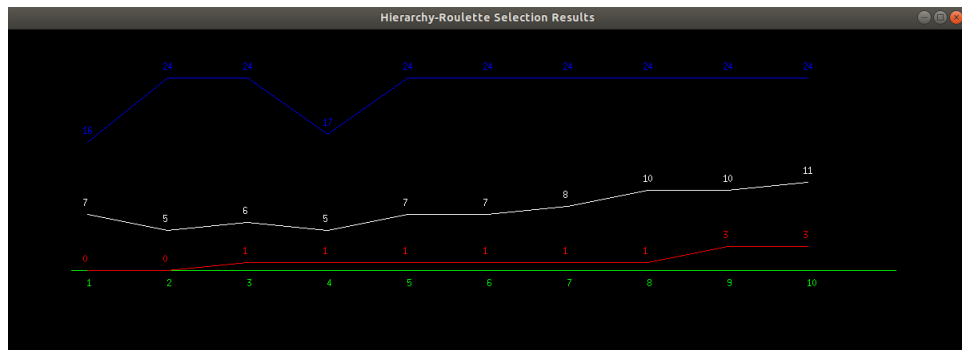


Figura 5: Resultado 2 con 10 generaciones

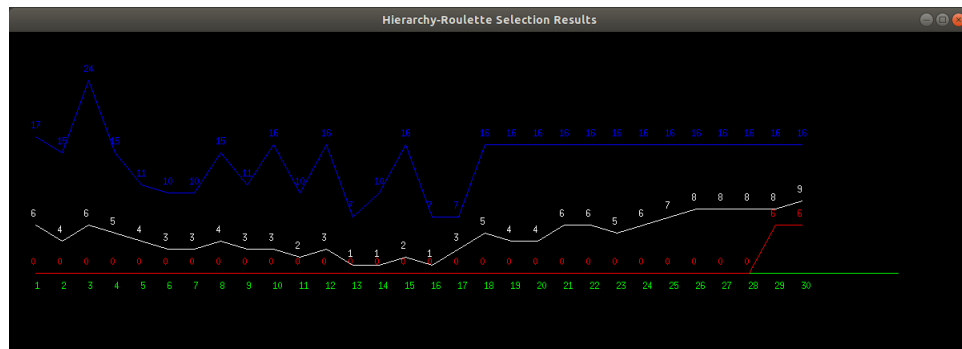


Figura 6: Resultado 1 con 30 generaciones

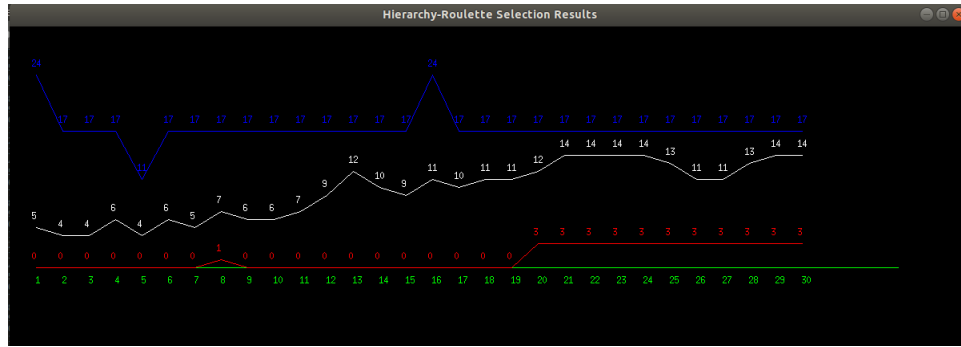


Figura 7: Resultado 2 con 30 generaciones

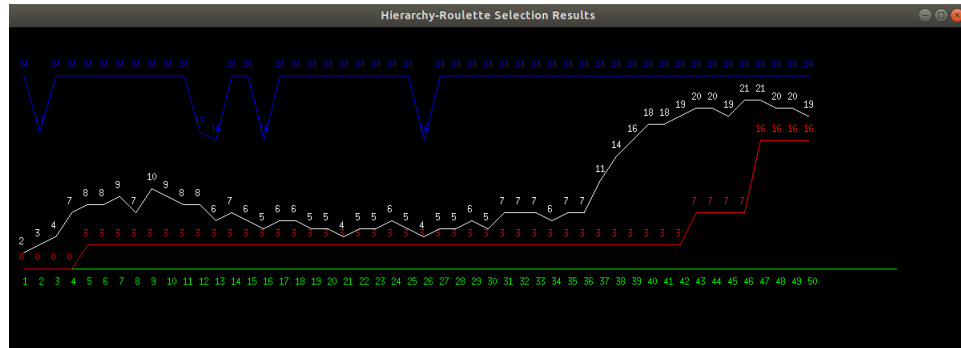


Figura 8: Resultado 1 con 50 generaciones

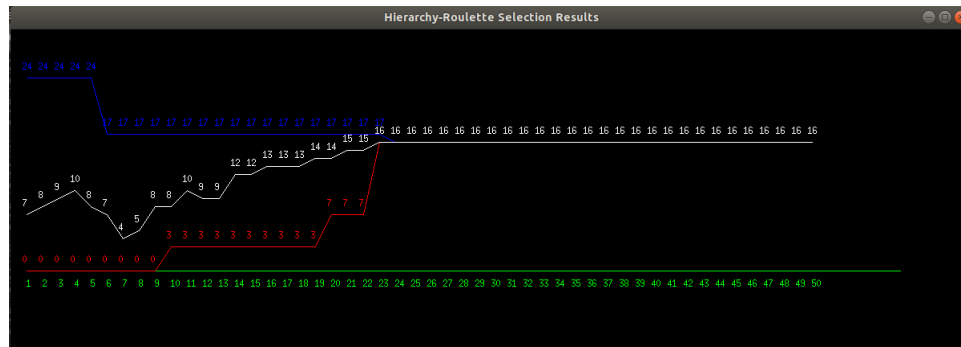


Figura 9: Resultado 2 con 50 generaciones

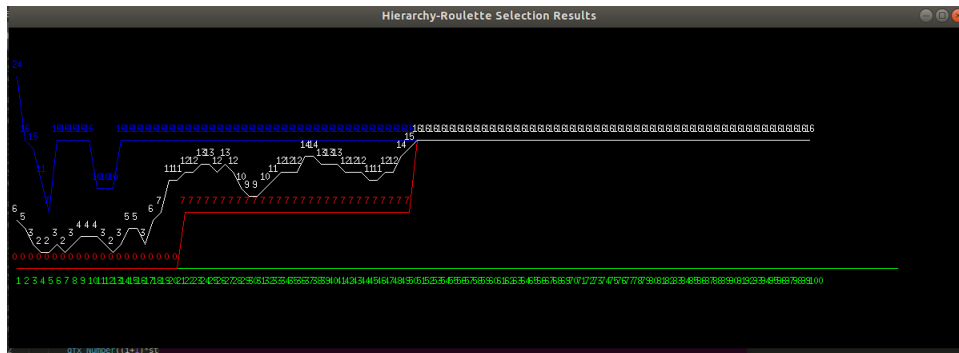


Figura 10: Resultado 1 con 100 generaciones

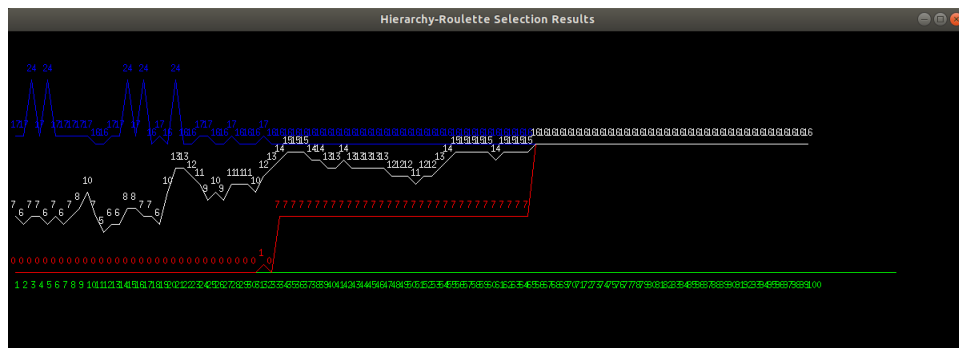


Figura 11: Resultado 2 con 100 generaciones

Conclusión

El fin de un algoritmo genético es simular la evolución de una población de individuos. En esta práctica, utilizamos distintos métodos aleatorios que se presentan en la naturaleza y generan una evolución, tales como la cruce y mutación de individuos, sin olvidar la selección de individuos para siguientes generaciones. Ésta selección se inicio con un ordenamiento de individuos con base en su ranking o jerarquía, lo que hace distinta ésta práctica a la práctica número 3, donde la selección por ruleta se hacia sin tomar en cuenta el ordenamiento de ranking.

El algoritmo genético de selección por ruleta es uno de los más utilizados, sin embargo considero que pueden existir casos en los que la selección de individuos se aplique sobre los individuos de menor aptitud, dado que el número aleatorio de referencia puede ocasionar que se seleccione un individuo con poca aptitud. Aún así, este algoritmo genético permite dar un vistazo de las distintas combinaciones que se presentan en la naturaleza, tanto de flora o fauna, y nos concientiza acerca de que no siempre los individuos más aptos son los que sobreviven, sin embargo, tienen una mayor probabilidad de conseguirlo.