

Implementación algoritmo genético.

Optimización del rendimiento del enzima 5-BR-triptófano halogenasa.



Rubén Rodríguez Fernández

[9 mayo 2014]

[rrodrf04@estudiantes.unileon.es]

Summary

1.- Programación algoritmo genético.....	3
1.1.- Operadores matemáticos de Perl.....	4
1.2.- Utilización del programa.....	4
2.- Crítica a la guía.....	5

1.- Programación algoritmo genético

Para llevar a cabo la programación del algoritmo genético hemos utilizado el lenguaje de programación Perl y la librería AI Genetics. En la guía se pide una serie de funciones que maximicen el valor de una función, es decir, que se obtenga el valor más alto posible. Los valores y sus rangos son los siguientes :

Catalasa : (2000 – 4000) μM

NADH : (5000 – 10 000) μM

L-Trp : (50.0 – 200.0) μM

NaBr : (30 000 – 60 000) μM

FAD : (10.0 – 50.0) μM

FLR : (120 – 200) μM

En el algoritmo genético cada componente es un gen, en el que sus valores varían entre los rangos establecidos arriba. La programación de este algoritmo genético es realmente sencilla y la única dificultad es encontrar los valores concretos de iteraciones, mutaciones, crossover y población inicial para obtener el mejor resultado. Para este problema se ha observado que es mejor establecer una probabilidad de mutación bastante alta (0.3) y un crossover de 0.9 . El número de iteraciones necesarias para estabilizarse son unas 5000 con una población inicial de 50 individuos. Los resultados de una ejecución con estos datos son los siguientes:

Catalasa = 4000 μM

NADH = 10000 μM

L-Trp = 200 μM

NaBr = 30000 μM

FAD = 50 μM

FLR = 131 μM

Con un valor de [5-Br-triptófano] = $8.09000051442427\text{e}+16 \mu\text{M}$.

(Nota, en caso de ejecutarlo y obtener un valor inferior a este, volver a ejecutar de nuevo)

A partir de los valores mostrados anteriormente hemos detectado que no se ha producido ninguna mejora, aunque un algoritmo genético no es determinista, diferentes ejecuciones pueden dar diferentes valores ya que los métodos para aplicar el crossover y mutaciones utilizan números aleatorios aunque la probabilidad de que de un número mayor a ese es bastante baja ya que el número de test realizado ha sido bastante elevado.

1.1.- Operadores matemáticos de Perl

Los operadores utilizados para portar la función de aptitud a perl son los siguientes :

- `*` = multiplicación. Ejemplo, $2 * 3$
- `**` = exponenciación , Ejemplo $2 ** 3 = 2^3$
- `sqrt` = raíz cuadrada (del inglés square root)

1.2.- Utilización del programa

Para ejecutar este programa simplemente tenemos que introducir lo siguiente en nuestra línea de comandos :

`perl programa.pl`

Y para modificar los datos referentes a la probabilidad de mutación, crossover, iteraciones o población inicial simplemente tenemos que modificar las variables que están al principio del programa :

`$population = 50;`

`$iterations = 5000;`

`$crossover = 0.9;`

`$mutation = 0.3;`

Asignandoles cualquier valor, teniendo en cuenta que el crossover y mutación son probabilidades y tienen que estar en un rango de 0 a 1 y todas las variables tienen que ser positivas.

(Nota, el nombre del programa y la forma de lanzar perl pueden cambiar, dependiendo del sistema operativo y del nombre del archivo)

2.- Crítica a la guía

La guía es clara y consisa, contiene todos los apartados requeridos y varios ejemplos desarrollados completamente.