Computación Evolutiva
Máster en Inteligencia Artificial Avanzada
Universidad Nacional de Educación a Distancia (UNED)

ACTIVIDAD 3

1. PRERREQUISITOS

La presente actividad deberá ser realizada tras el estudio de los temas 6, 8, 10 y 12 del texto base [Eiben&Smith-03] y el artículo [O'Neill&Ryan-01] (descargable desde el curso virtual). El tema 6 permitirá al alumno familiarizarse con la programación genética (PG) y, el mencionado artículo, con una de las variantes de la PG, denominada Evolución Gramatical (del inglés *Grammatical Evolution*, GE). El capítulo 8 describe distintos mecanismos para sintonizar de forma adaptativa cada uno de los diferentes parámetros de los que consta un algoritmo evolutivo (AE). El capítulo 10 describe la forma de hibridar un AE con otros métodos de búsqueda. Finalmente, en el capítulo 12, se muestran distintas estrategias para manejar la existencia de restricciones en problemas de optimización que son abordados mediante AEs.

2. DEFINICIÓN DEL PROBLEMA

Se pretende implementar un algoritmo evolutivo para calcular la derivada simbólica de una función $f: X \subseteq \Re \rightarrow \Re$. Para ello, haremos uso de las dos siguientes definiciones:

> *Definición de derivada de una función en un punto*: Sea $X \subseteq \Re$ un intervalo abierto, diremos que $f: X \rightarrow \Re$ es derivable en $x_0 \in X$, denotado por $f'(x_0)$, si existe y es finito el límite:

$$f'(x_0) = \lim_{\mathrm{h}\rightarrow 0} \frac{f(x_0+\mathrm{h}) - f(x_0)}{\mathrm{h}} \qquad (1)$$

> *Definición de derivada de una función en un intervalo:* Sea $X \subseteq \Re$ un intervalo abierto, diremos que $f: X \rightarrow \Re$ es derivable en el intervalo $[a, b] \subset X$ si f es derivable en cada uno de los puntos de dicho intervalo, es decir:

$$f'(x) = \lim_{\mathrm{h}\rightarrow 0} \frac{f(x+\mathrm{h}) - f(x)}{\mathrm{h}}, \forall \mathrm{x} \in [\mathrm{a}, \mathrm{b}] \qquad (2)$$

A partir de esta última definición, resulta inmediato establecer que si la función $f(x)$ tiene derivada en el intervalo $[a, b]$, entonces el problema de calcular dicha derivada se puede transformar en un nuevo problema de optimización consistente en encontrar una función $g(x)$ que minimice la siguiente expresión:

$$\underbrace{mín}_{g(x)} \quad \frac{1}{(b-a)} \int_a^b error[(f'(x), g(x))]dx \qquad (3)$$

donde $f'(x)$ se calcularía utilizando la expresión (2). No obstante, el problema anterior se puede resolver de forma aproximada discretizando el intervalo de definición, es decir, cambiando el operador integral por un sumatorio:

$$\underbrace{\text{mín}}_{g(x)} \quad \frac{1}{\mathrm{N}+1}\sum_{\mathrm{i}=0}^{\mathrm{N}} error_i(f'(a+i\cdot h), g(a+i\cdot h)) \qquad (4)$$

donde $h=(b-a)/N$ es la anchura del subintervalo de muestreo para conseguir muestrear $N+1$ puntos en el intervalo $[a,b]$, y $f'(a+i\cdot h)$ viene dado por:

$$f'(a+ih)=\frac{f(a+(i+1)\cdot h)-f(a+i\cdot h)}{h},\ \forall\ \mathrm{i}\in\{0,1,\dots,\mathrm{N}\} \qquad (5)$$

El valor de N lo elige el usuario. Obsérvese que cuanto mayor sea el valor de este parámetro, mayor será la precisión de la solución buscada, es decir, de la función derivada $g(x)$.

3. ACTIVIDADES A REALIZAR

Para la resolución del problema de optimización planteado en (4), se empleará una aproximación basada en GE. Así, una posible gramática Backus-Naur para resolver este problema podría ser la siguiente:

N = {expr, op, pre_op}
T = {“+”, “-“, “*”, “/”, “sen”, “cos”, “exp”, “ln”, “x”, “1.0”, “(”, “)”*}*
S = {expr}

Conjunto de reglas de producción (P):

<expr>	*::=*	*<expr><op><expr>*	(0)
	|	*(<expr><op><expr>)*	(1)
	|	*<pre_op>(<expr>)*	(2)
	|	*<var>*	(3)
<op>	*::=*	+ | - | * | /	(0), (1), (2), (3)
<pre_op>	*::=*	sen | cos | exp | ln	(0), (1), (2), (3)
<var>	*::=*	*x* | 1.0	(0), (1)

El alumno deberá tener en cuenta las siguientes recomendaciones:

- El paradigma de GE se centra principalmente en el proceso de codificación/decodificación de un cromosoma, siendo dicho proceso independiente del proceso de búsqueda de la solución. Por tanto, para realizar la búsqueda de la solución, es posible utilizar cualquier estrategia de búsqueda poblacional. En particular, el alumno puede reutilizar el AG implementado en la actividad 1 como motor de búsqueda. No obstante, dado que la codificación en GE se realiza a partir de cadenas lineales de enteros (codones), el alumno podría tener que adaptar su implementación AG para manejar este tipo de codificación basada en enteros.

- Otras variantes de la gramática indicada anteriormente son posibles. Por ejemplo, se podrían añadir nuevas funciones a la regla de producción <*pre_op*>. También son posibles otras reglas de producción para crear valores numéricos en la expresión resultante. Se recomienda que el alumno consulte la bibliografía relacionada.

- Se permite la reutilización de código dedicado al proceso de decodificación de un individuo. Para ello, se puede consultar, por ejemplo, el enlace de la página de Michael O'Neill dedicada a GE:

  http://www.grammatical-evolution.org/

- La función de evaluación $F(x)$ vendrá dada por la expresión (4). No obstante, existen diferentes formas de calcular el error. Por ejemplo, usando el error cuadrático, la expresión (4) se transforma en:

$$\mathrm{F(x)} = \frac{1}{\mathrm{N}+1}\sum_{\mathrm{i}=0}^{\mathrm{N}}[f'(a+i\cdot h) - g(a+i\cdot h)]^2 \qquad (6)$$

  Usando el error absoluto, se tiene:

$$\mathrm{F(x)} = \frac{1}{\mathrm{N}+1}\sum_{\mathrm{i}=0}^{\mathrm{N}}|f'(a+i\cdot h) - g(a+i\cdot h)| \qquad (7)$$

  Otra posibilidad es ponderar cada sumando de error. Por ejemplo, si se usa el error absoluto, la función de evaluación resultaría:

$$\mathrm{F(x)} = \frac{1}{\mathrm{N}+1}\sum_{\mathrm{i}=0}^{\mathrm{N}}\omega_{\mathrm{i}}\cdot|f'(a+i\cdot h) - g(a+i\cdot h)|$$

$$\omega_i = \begin{cases} K_0 & \text{si } |f'(a+i\cdot h) - g(a+i\cdot h)| \le U \\ K_1 & \text{si } |f'(a+i\cdot h) - g(a+i\cdot h)| > U \end{cases} \qquad (8)$$

  donde el parámetro U define el umbral mínimo por debajo del cual el error i-ésimo es admisible. En este caso, se dice que se ha alcanzado un *hit*. De otro lado, K_0 y K_1, con $K_0 < K_1$, definen, respectivamente, el valor de ponderación (penalización) cuando se alcanza o no un *hit*.

- Dado que la gramática no garantiza siempre la construcción de expresiones válidas (por ejemplo, podrían producirse divisiones por cero), resulta imprescindible implementar mecanismos para manejar restricciones (ver capítulo 12 del texto base).

- El alumno debe experimentar con los parámetros asociados a dicho algoritmo (tamaño de cromosoma siempre fijo, tamaño de cromosoma variable, valor de *wrapping*, operadores de variación y sus probabilidades, operadores de selección, tamaño de la población, etc.) con el objeto de intentar alcanzar la solución de todos los problemas propuestos y de la forma más óptima posible. En este sentido, también debería investigar la forma de incorporar algún mecanismo que incorpore una estrategia de control adaptativo o auto-adaptativo de alguno de dichos parámetros (ver capítulo 8 del texto base).

- Para mejorar o acelerar la convergencia, el alumno debería investigar la posibilidad de aplicar mecanismos de búsqueda local (ver capítulo 10 del texto base).

Asumiendo que las siguientes funciones son derivables en el intervalo indicado, se pide obtener su derivada:

1. $f(x) = 2x^3 + 5$ en $[0,5]$ (Sol: $g(x) = 6x^2$)
2. $f(x) = \frac{x-1}{x+1}$ en $[0,5]$ (Sol: $g(x) = \frac{2}{(x+1)^2}$)
3. $f(x) = \frac{1}{4}(x^2 + 1)(x - 1)$ en $[-2,2]$ (Sol: $g(x) = \frac{3}{4}x^2 - \frac{1}{2}x + 1$)
4. $f(x) = -e^{-x^2+3}$ en $[0,3]$ (Sol: $g(x) = 2xe^{-x^2+3}$)
5. $f(x) = \frac{e^{2x}+e^{-6x}}{6}$ en $[0,2]$ (Sol: $g(x) = \frac{1}{3}e^{2x} - e^{-6x}$)
6. $f(x) = xln(1 + x)$ en $[0,5]$ (Sol: $g(x) = \ln(1 + x) + \frac{x}{(1+x)}$)
7. $f(x) = e^x \sin(x)$ en $[-2,2]$ (Sol: $g(x) = e^x(\sin(x) + \cos(x))$)

El valor de N se tomará siempre como $N = 50$ (obsérvese que el intervalo en el que se busca la solución no es el mismo para todos los problemas planteados).

La función de evaluación a utilizar será la indicada por la expresión (8), con $U = 10^{-1}$, $K_0 = 1$ y $K_1 = 10$. Se dirá que se obtiene una solución para un problema dado si el algoritmo alcanza un *hit* en todos los puntos de muestreo. Un **aspecto muy importante** a tener en cuenta es que el valor de $f'(a + i \cdot h)$, mostrado en la expresión (8), nunca se evaluará a partir de la derivada exacta (se supone que ésta es desconocida), sino tal y como indica la expresión (5).

Hay que tener en cuenta que el algoritmo no tiene por qué encontrar la solución exacta, sino que puede encontrar una aproximación a dicha solución en el intervalo de estudio. Por ello, el conjunto de funciones usadas en la regla de producción <*pre_op*> siempre será como mínimo el expresado en la gramática descrita al principio de esta sección, es decir, no se puede adaptar este conjunto dependiendo de la derivada a calcular. Tal y como se decía antes, se supone que la solución no es conocida de antemano.

Finalmente, utilizando la mejor configuración del algoritmo evolutivo (aquella que produzca los mejores resultados en los siete problemas analizados), se deberá representar gráficamente la derivada exacta de cada problema junto a la obtenida evolutivamente.

4. CRITERIOS DE EVALUACIÓN

Sobre la presentación de la memoria (2 puntos sobre 10)
Debe incluir las siguientes secciones: (a) Descripción del problema a resolver, (b) Método para resolverlo (aquí se debe analizar la idoneidad o no del uso de GE para resolver el problema planteado). Además, se debe incluir la expresión matemática de la función de evaluación finalmente empleada, la descripción de los diferentes operadores de inicialización, variación y selección empleados, la forma de manejar las restricciones, los mecanismos de control de parámetros utilizados, así como los

mecanismos de búsqueda local implementados; (c) Los resultados de los distintos experimentos realizados; (d) Un análisis y comparación de resultados; (e) Una sección de conclusiones; y (f) Una descripción del código implementado. Se evaluará especialmente la claridad en la redacción de la memoria y la capacidad de síntesis.

Sobre el manejo de restricciones (1 puntos sobre 10)
Se valorará la originalidad del mecanismo o mecanismos usados para el manejo de restricciones.

Sobre la configuración del algoritmo (2 puntos sobre 10)
Aquí se valorará el procedimiento seguido por el alumno a la hora de elegir la mejor configuración de parámetros del algoritmo, incluyendo la implementación de mecanismos de control de parámetros adaptativos o auto-adaptativos

Hibridación del algoritmo con técnicas de búsqueda local (1 punto sobre 10)
Se valorará la originalidad del mecanismo de búsqueda local utilizado.

Sobre el análisis y comparación de resultados, y conclusiones (4 puntos sobre 10)
Se valorará la forma de interpretar y comparar los diferentes experimentos realizados. Es muy importante que dicha valoración se haga siempre en términos de los índices SR, MBF, AES y cualquier otra gráfica que considere oportuna como, por ejemplo, los plots de progreso de convergencia. Finalmente, se valorará la calidad de las conclusiones obtenidas a partir de la interpretación y comparación de resultados.

BIBLIOGRAFÍA

[Eiben&Smith-03] A.E. Eiben y J.E. Smith (2003). *Introduction to Evolutionary Computing*. Springer.

[O'Neill&Ryan-01] M. O'Neill and C. Ryan (2001). Grammatical Evolution. *IEEE Transactions on Evolutionary Computation*, 5 (4), pp 349-358.