ALGORITMOS DE ESTIMACION DE DISTRIBUCIONES

1 Introducción

El comportamiento de los dos heurísticos introducidos hasta el momento –enfriamiento estadístico y algoritmos genéticos– depende de un buen número de parámetros asociados a los mismos. En el caso del enfriamiento estadístico tenemos entre otros: el valor inicial de la temperatura, la longitud de la cadena para cada valor de temperatura, la política de reducción de la temperatura, el criterio de terminación, etc. Por lo que respecta a los algoritmos genéticos algunos de los parámetros a determinar son: los operadores de cruce y mutación, las probabilidades de cruce y mutación, el tamaño de la población, la tasa de reproducción generacional, el número de generaciones, etc. De hecho la determinación de valores adecuados para dichos parámetros constituye por si mismo un verdadearo problema de optimización. Por otra parte una mala elección de los valores de los parámetros puede llevar a que el algoritmo obtenga soluciones alejadas del óptimo. Este es uno de los motivos por los que desde hace varios años se han venido estudiando alternativas a los metodos heurísticos estocásticos existentes que no necesitasen el ajustar un número alto de parámetros.

Otro motivo básico por el que se ha desarrollado la búsqueda de nuevos heurísticos de optimización es por la necesidad de identificar las interrelaciones entre las variables utilizadas para representar a los individuos con la codificación utilizada.

Todo ello ha motivado el nacimiento de un nuevo método de búsqueda conocido como Algoritmo de Estimación de Distribuciones, que denotaremos por EDAs (*Estimation of Distribution Algorithms*).

Los EDAs son algoritmos heurísticos de optimización que basan su búsqueda –al igual que los algoritmos genéticos– en el caracter estocástico de la misma. Tambien al igual que los algoritmos genéticos los EDAs están basados en poblaciones que evolucionan. Sin embargo a diferencia de los algoritmos genéticos en los EDAs la evolución de las poblaciones no se lleva a cabo por medio de los operadores de cruce y mutación. En lugar de ello la nueva población de individuos se muestrea de una distribución de probabilidad, la cual es estimada de la base de datos conteniendo al conjunto de individuos seleccionados de entre los que constituyen la generación anterior.

Mientras que en los algoritmos genéticos las interrelaciones entre las variables representando a los individuos se tienen en cuenta de manera implícita, en los EDAs dichas interrelaciones se expresan de manera explícita a través de la distribución de probabilidad asociada con los individuos seleccionados en cada generación. De hecho la estimación de dicha distribución de probabilidad conjunta asociada a los individuos seleccionados en cada generación es la principal dificultad de esta aproximación.

En este capítulo vamos a presentar una aproximación general a los EDAs, es decir una aproximación en la cual la distribución de probabilidad con la cual se modeliza el comportamiento de los individuos seleccionados en cada generación es genérica, si bien posteriormente particularizaremos para el caso más simple en el cual la distribución de probabilidad conjunta es factorizada como producto de distribuciones marginales univariantes.

2 Ilustrando EDAs por medio de un ejemplo

Con el objetivo de entender el funcionamiento de los diferentes componentes y pasos de los EDAs, aplicaremos la versión más simple de esta aproximación a un ejemplo muy sencillo de optimización.

Tabla 7.1: La población inicial, D_0.

	X_1	X_2	X_3	X_4	X_5	X_6	$h(\boldsymbol{x})$
1	1	0	1	0	1	0	3
2	0	1	0	0	1	0	2
3	0	0	0	1	0	0	1
4	1	1	1	0	0	1	4
5	0	0	0	0	0	1	1
6	1	1	0	0	1	1	4
7	0	1	1	1	1	1	5
8	0	0	0	1	0	0	1
9	1	1	0	1	0	0	3
10	1	0	1	0	0	0	2
11	1	0	0	1	1	1	4
12	1	1	0	0	0	1	3
13	1	0	1	0	0	0	2
14	0	0	0	0	1	1	2
15	0	1	1	1	1	1	5
16	0	0	0	1	0	0	1
17	1	1	1	1	1	0	5
18	0	1	0	1	1	0	3
19	1	0	1	1	1	1	5
20	1	0	1	1	0	0	3

Supongamos que tratamos de maximizar la función *OneMax* definida en un espacio de dimensión 6. Es decir, tratamos de obtener el máximo de la función $h(\boldsymbol{x}) = \sum_{i=1}^{6} x_i$ con $x_i = 0, 1$.

La población inicial se obtiene al azar por medio del muestreo de la siguiente distribución de probabilidad: $p_0(\boldsymbol{x}) = \prod_{i=1}^{6} p_0(x_i)$, donde $p_0(X_i = 1) = 0.5$ para $i = 1, \ldots, 6$. Esto significa que la distribución de probabilidad conjunta de la cual se muestrea, se encuentra factorizada como un producto de seis distribuciones marginales univariantes, cada una de las cuales sigue un modelo de Bernouilli con parámetro igual a 0.5. Denotamos por D_0 el fichero conteniendo 20 casos –ver tabla 7.1– obtenida por medio de esta simulación.

En un segundo paso, seleccionamos algunos de los individuos de D_0. Esto puede hacerse usando uno de los métodos de selección estandard en computación evolutiva. Supongamos que nuestro método de selección es truncación, y que seleccionamos a la mitad de la población. Denotamos por D_0^{Se} el fichero de casos conteniendo los indiviuos seleccionados. En caso de que existan empates en la función de evaluación de algunos individuos (situación que se verifica en los individuos numerados como 1, 9, 12, 18 y 20) la selección se lleva a cabo de manera probabilística entre los mismos. Por ejemplo, en este caso se necesitan seleccionar 3 individuos de entre el conjunto de individuos cuya función de evaluación vale 3.

Tabla 7.2: Individuos seleccionados, D_0^{Se}, a partir de la población inicial.

	X_1	X_2	X_3	X_4	X_5	X_6
1	1	0	1	0	1	0
4	1	1	1	0	0	1
6	1	1	0	0	1	1
7	0	1	1	1	1	1
11	1	0	0	1	1	1
12	1	1	0	0	0	1
15	0	1	1	1	1	1
17	1	1	1	1	1	0
18	0	1	0	1	1	0
19	1	0	1	1	1	1

Una vez que tenemos seleccionados 10 individuos –véase la tabla 7.2– nos interesa expresar explícitamente –por medio de la distribución de probabilidad conjunta– las características de los

individuos seleccionados. Aunque somos conscientes de que sería interesante que dicha distribución de probabilidad conjunta tuviera en cuenta las interdependencias entre las variables, en este caso utilizaremos el modelo más simple posible para expresar dicha distribución de probabilidad conjunta. En dicho modelo, cada variable se va a considerar independiente del resto. Esto se expresa matemáticamente por medio de:

$$p_1(\boldsymbol{x}) = p_1(x_1, \ldots, x_6) = \prod_{i=1}^{6} p(x_i|D_0^{Se}). \tag{1}$$

Es decir que tan sólo necesitamos 6 parámetros para especificar el modelo. Cada uno de dichos parámetros, $p(x_i|D_0^{Se})$ con $i = 1, ..., 6$, será estimado a partir del fichero de casos D_0^{Se} por medio de la frecuencia relativa correspondiente, $\hat{p}(X_i = 1|D_0^{Se})$.

En este ejemplo los valores de los parámetros resultan ser:

$$\begin{array}{lll} \hat{p}(X_1 = 1|D_0^{Se}) = 0.7 & \hat{p}(X_2 = 1|D_0^{Se}) = 0.7 & \hat{p}(X_3 = 1|D_0^{Se}) = 0.6 \\ \hat{p}(X_4 = 1|D_0^{Se}) = 0.6 & \hat{p}(X_5 = 1|D_0^{Se}) = 0.8 & \hat{p}(X_6 = 1|D_0^{Se}) = 0.7. \end{array} \tag{2}$$

Muestreando la distribución de probabilidad, $p_1(\boldsymbol{x})$, obtenemos una nueva población de individuos, D_1. La tabla 7.3 representa el fichero de casos consistente en los 20 individuos obtenidos por medio de la simulación de $p_1(\boldsymbol{x})$. En dicha tabla se indica asimismo para cada individuo su valor de $h(\boldsymbol{x})$ asociado.

Table 7.3: La primera generación de individuos: D_1.

	X_1	X_2	X_3	X_4	X_5	X_6	$h(\boldsymbol{x})$
1	1	1	1	1	1	1	6
2	1	0	1	0	1	1	4
3	1	1	1	1	1	0	5
4	0	1	0	1	1	1	4
5	1	1	1	1	0	1	5
6	1	0	0	1	1	1	4
7	0	1	0	1	1	0	3
8	1	1	1	0	1	0	4
9	1	1	1	0	0	1	4
10	1	0	0	1	1	1	4
11	1	1	0	0	1	1	4
12	1	0	1	1	1	0	4
13	0	1	1	0	1	1	4
14	0	1	1	1	1	0	4
15	0	1	1	1	1	1	5
16	0	1	1	0	1	1	4
17	1	1	1	1	1	0	5
18	0	1	0	0	1	0	2
19	0	0	1	1	0	1	3
20	1	1	0	1	1	1	5

De nuevo seleccionamos 10 individuos de D_1, obteniendo –como puede verse en la tabla 7.4– el fichero D_1^{Se}.

Tabla 7.4: D_1^{Se}: Individuos seleccionados de la primera generación de inidviduos.

	X_1	X_2	X_3	X_4	X_5	X_6
1	1	1	1	1	1	1
2	1	0	1	0	1	1
3	1	1	1	1	1	0
5	1	1	1	1	0	1
6	1	0	0	1	1	1
8	1	1	1	0	1	0
9	1	1	1	0	0	1
15	0	1	1	1	1	1
17	1	1	1	1	1	0
20	1	1	0	1	1	1

A partir de este fichero de casos obtenemos:

$$p_2(\boldsymbol{x}) = p_2(x_1, \ldots, x_6) = \prod_{i=1}^{6} p(x_i|D_1^{Se}) \tag{3}$$

donde $p(x_i|D_1^{Se})$ con $i = 1, \ldots, 6$ se estima de D_1^{Se} por medio de su correspondiente frecuencia relativa, $\hat{p}(X_i = 1|D_1^{Se})$.

Como puede comprobarse, en este caso los valores de los parámetros son:

$$\begin{array}{lll} \hat{p}(X_1 = 1|D_1^{Se}) = 0.9 & \hat{p}(X_2 = 1|D_1^{Se}) = 0.8 & \hat{p}(X_3 = 1|D_1^{Se}) = 0.8 \\ \hat{p}(X_4 = 1|D_1^{Se}) = 0.7 & \hat{p}(X_5 = 1|D_1^{Se}) = 0.8 & \hat{p}(X_6 = 1|D_1^{Se}) = 0.7. \end{array} \tag{4}$$

Los pasos anteriores se repiten hasta que se verifique una determinada condición de parada.

3 Aproximación general

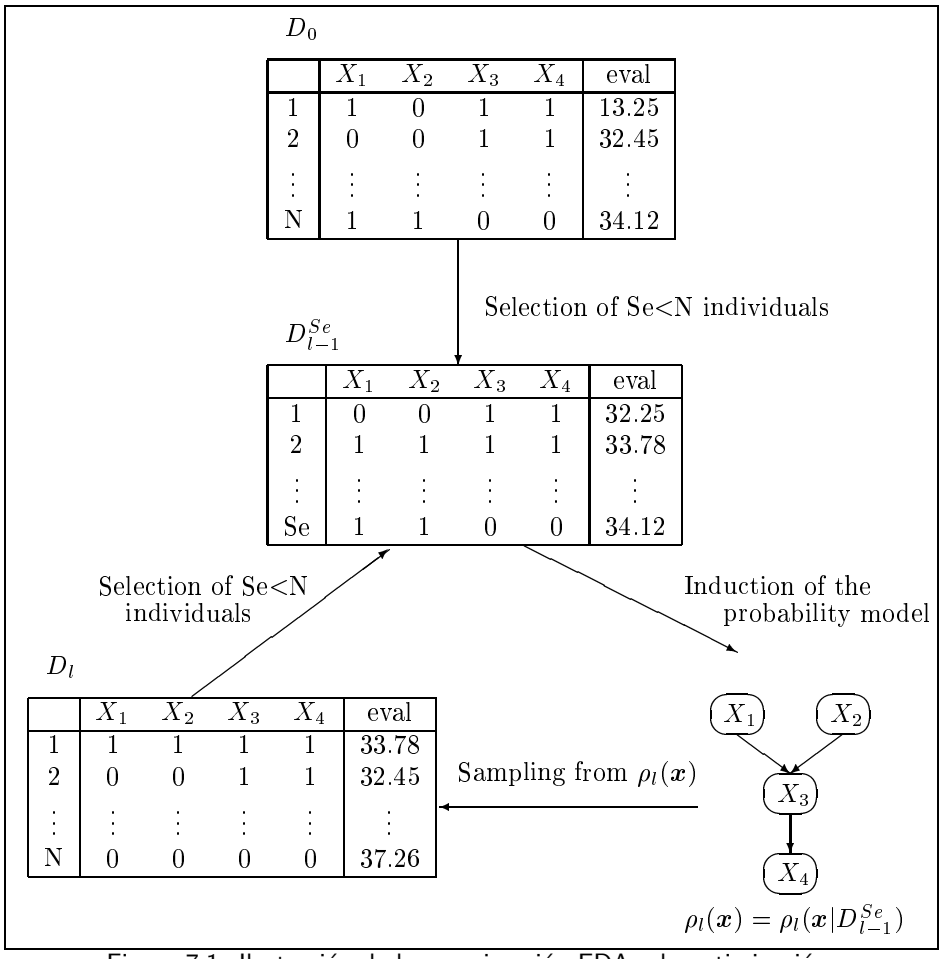
En este apartado vamos a generalizar lo presentado en el apartado anterior. Tal y como se ha visto, la aproximación EDA consiste en un heurístico probabilístico de búsqueda, basado en poblaciones que evolucionan, y fundamentado en tres pasos básicos a iterar.

Estos tres pasos consisten en:

1. seleccionar algunos individuos de la población
2. estimar el modelo probabilístico subyacente a dichos individuos seleccionados
3. muestrear de la distribución de probabilidad aprendida, con objeto de obtener una nueva población de individuos

y son repetidos hasta que se verifique un criterio de parada, previamente establecido.

En la figura 7.1, al igual que en el pseudocódigo de la figura 7.2, podemos ver un esquema de la aproximación EDA. Partimos de M individuos generados al azar, en nuestro ejemplo anterior por medio de una distribución uniforme para cada variable. Estos M individuos constituyen la población inicial, D_0. A continuación, cada uno de dichos individuos es evaluado. En un primer paso, un número $N (N \leq M)$ de individuos es seleccionado (habitualmente aquellos con mejor función objetivo). En un segundo paso se lleva a cabo la inducción del modelo probabilístico n-dimensional que mejor refleja las interdependencias entre las n variables. En un tercer paso, M nuevos individuos (la nueva población) se obtienen por medio de la simulación de la distribución de probabilidad aprendida en el paso anterior. Estos tres pasos se repiten hasta que se verifique una condición de parada. Ejemplos de condiciones de parada son: determinación de un número máximo de poblaciones, o de un número máximo de individuos evaluados, uniformidad en la población generada, no mejora con respecto al mejor de los individuos obtenidos en las generaciones previas, etc.



	X_1	X_2	X_3	X_4	eval
1	1	0	1	1	13.25
2	0	0	1	1	32.45
⋮	⋮	⋮	⋮	⋮	⋮
N	1	1	0	0	34.12

	X_1	X_2	X_3	X_4	eval
1	0	0	1	1	32.25
2	1	1	1	1	33.78
⋮	⋮	⋮	⋮	⋮	⋮
Se	1	1	0	0	34.12

	X_1	X_2	X_3	X_4	eval
1	1	1	1	1	33.78
2	0	0	1	1	32.45
⋮	⋮	⋮	⋮	⋮	⋮
N	0	0	0	0	37.26

Figura 7.1: Ilustración de la aproximación EDA a la optimización.

EDA
$D_0 \leftarrow$ Generar M individuos (la poblacion inicial) al azar

Repeat for $l = 1, 2, \ldots$ hasta que se verifique el criterio de parada

$D_{l-1}^{Se} \leftarrow$ Seleccionar $N \leq M$ individuos de D_{l-1} de acorde con el metodo de seleccion

$p_l(\boldsymbol{x}) = p(\boldsymbol{x}|D_{l-1}^{Se}) \leftarrow$ Estimar la distribucion de probabilidad de que un individuo se encuentre en los individuos seleccionados

$D_l \leftarrow$ Muestrear M individuos (la nueva poblacion) de $p_l(\boldsymbol{x})$

Figura 7.2: Pseudocódigo de la aproximación EDA.

El mayor problema con los EDAs es como estimar la distribución de probabilidad $p_l(\boldsymbol{x})$. Obviamente la computación de todos los parámetros necesarios para especificar la distribución de probabilidad conjunta no es práctica. Este problema trae como consecuencia la aproximación de la distribución de probabilidad conjunta por medio de distintas factorizaciones –más o menos complejas– de la misma.

4 La aproximación UMDA

Tal y como puede verse en el pseudocódigo de la figura 7.3, el modelo probabilístico utilizado por el algoritmo UMDA (*Univariate Marginal Distribution Algorithm*) es el más simple posible, y coincide

con el que nos ha servido para ilustrar el ejemplo del apartado anterior. En concreto, en cada generación la distribución de probabilidad conjunta, $p_l(\boldsymbol{x})$, que sirve para estimar el comportamiento de los individuos seleccionados, se factoriza como un producto de distribuciones marginales univariantes e independientes. Es decir:

$$p_l(\boldsymbol{x}) = p(\boldsymbol{x}|D_{l-1}^{Se}) = \prod_{i=1}^{n} p_l(x_i). \tag{5}$$

UMDA

$D_0 \leftarrow$ Generar M individuos (la poblacion inicial) al azar

Repeat for $l = 1, 2, \ldots$ hasta que se verifique el criterio de parada

$D_{l-1}^{Se} \leftarrow$ Seleccionar $N \leq M$ individuos de D_{l-1} de acorde al metodo de seleccion

$p_l(\boldsymbol{x}) = p(\boldsymbol{x}|D_{l-1}^{Se}) = \prod_{i=1}^{n} p_l(x_i) = \prod_{i=1}^{n} \frac{\sum_{j=1}^{N} \delta_j(X_i = x_i|D_{l-1}^{Se})}{N} \leftarrow$ Estimar la distribucion de probabilidad conjunta

$D_l \leftarrow$ Muestrar M individuos (la nueva poblacion) de $p_l(\boldsymbol{x})$

Figura 7.3: Pseudocódigo del algoritmo UMDA.

Cada distribución marginal se estima a partir de las frecuencias marginales:

$$p_l(x_i) = \frac{\sum_{j=1}^{N} \delta_j(X_i = x_i|D_{l-1}^{Se})}{N} \tag{6}$$

donde

$$\delta_j(X_i = x_i|D_{l-1}^{Se}) = \begin{cases} 1 & \text{si en el } j\text{-ésimo caso de } D_{l-1}^{Se},\ X_i = x_i \\ 0 & \text{en otro caso.} \end{cases} \tag{7}$$

5 Otras aproximaciones más complejas

En problemas de optimización que surgen en el mundo real la aproximación anterior (UMDA) no es muy adecuada, ya que supone que las variables no interactúan entre si. Esta falta de interdependencia entre las variables hace que el modelo UMDA esté muy alejado de lo que en realidad ocurre. Es por ello por lo que se han desarrollado algoritmos, que perteneciendo a la familia de EDAs, incorporan la posibilidad de tener en cuenta dichas interrelaciones entre las variables. Para ello se hace necesario utilizar una estimación de la distribución de probabilidad conjunta que no consista simplemente en el producto de las distribuciones de probabilidad marginales univariantes, sino que utilice modelos más complejos.

Aumentando en complejidad, se pueden considerar modelos que tengan en cuenta estadísticos de orden dos o incluso de orden superior. En realidad la aproximación más genérica se obtiene cuando en cada generación la distribución de probabilidad se factoriza por medio de redes Bayesianas (optimización combinatorial) o de redes Gaussianas (optimización en dominios continuos). Queda sin embargo, lejos del objetivo de estos apuntes el presentar ejemplos de tales aproximaciones.

6 Notas bibliográficas

Los EDAs se introdujeron por vez primera en el campo de la computación evolutiva por mediación del trabajo de Mühlenbein y Paaß(1996). El algoritmo UMDA desarrollado en este tema fué introducido por Mühlenbein (1998). Planteamientos más generales que utilizan las redes Bayesianas para factorizar la distribución de probabilidad conjunta en cada generación se han propuesto por

Etxeberria y Larrañaga (1999), Pelikan, Goldberg y Cantú-Paz (1999), y Larrañaga, Etxeberria, Lozano y Peña (2000a) en el mundo de optimización combinatorial. En problemas de optimización en dominios continuos, en Larrañaga, Etxeberria, Lozano y Peña (2000b) se propone la utilización de las redes Gaussianas para llevar a cabo, en cada generación, la factorización de la función de densidad conjunta. Larrañaga y Lozano (2001) recoge una revisión de trabajos relacionados con EDAs.

7 Recursos en internet

- http://www.sc.ehu.es/isg
  Página web del grupo *Intelligent Systems Group* del departamento de Ciencias de la Computación e Inteligencia Artificial de la UPV–EHU
- http://ais.gmd.de
  Página web del *National Research Center for Information Technology*
- http://www-illigal.ge.uiuc.edu/index.html
  Página web del grupo *Illinois Genetic Algorithms Laboratory* dirigido por el Prof. Goldberg.

Ejercicios

Diseñar Algoritmos de Estimación de Distribuciones (función de coste, representación de individuos) para los siguientes problemas:

1. *El problema del agente viajero*
   Dada una colección finita de ciudades, determinar la gira de mínimo coste, visitando cada ciudad exactamente una vez y volviendo al punto de partida.
   Sea $n \geq 3$, y $C = (c_{ij})$ una matriz $M(n,n)$ de números reales positivos, encontrar la permutación cíclica π de los enteros de 1 a n que minimice $\sum_{i=1}^{n-1} c_{\pi(i)\pi(i+1)} + c_{\pi(n)\pi(1)}$

2. *El problema de la mochila* $0-1$
   Dado un conjunto finito de items, cada uno de los cuales tiene asocido un peso y una ganancia, seleccionar el subconjunto de items a incluir en una mochila –capaz de soportar un peso máximo finito– cuya inclusión proporcione una ganancia máxima.
   Sean n items y una mochila capaz de soportar un peso máximo C. Denotamos por b_j el beneficio obtenido al introducir el item j en la mochila, mientras que w_j denota el peso asociado a dicho item j. Se trata de seleccionar un subconjuto de items que maximicen $Z = \sum_{j=1}^{n} b_j x_j$, sujeto a la restricción $\sum_{j=1}^{n} w_j x_j \leq C$, siendo

   $$x_j = \begin{cases} 1 & \text{si el item } j \text{ seleccionado} \\ 0 & \text{si el item } j \text{ no seleccionado} \end{cases}$$

3. *El problema de las n reinas*
   En un hipótetico tablero de ajedrez $n \times n$, posicionar n reinas, de tal manera que ninguna ataque al resto.

4. *Problema de agrupamiento I*
   Agrupar N números en k grupos disjuntos minimizando la suma de las diferencias entre los grupos.

5. *Problema de agrupamiento II*
   Disponer los N^2 primeros números naturales en una matriz cuadrada $M(N,N)$ de tal manera que las sumas tanto por filas como por columnas coincidan.

6. *Problema de emplazamiento de bloques rectangulares*
   Dado un conjunto de n bloques rectangulares de distintas anchuras y alturas, se trata de encontrar un emplazamiento –asignación de los centros de los bloques rectangulares a puntos de un espacio cartesiano bidimensional– de tal forma que no haya solapamientos entre los

bloques rectangulares, y se minimice la función de coste $f = A + \lambda C$, donde A es el área del rectángulo que engloba todos los bloques rectangulares, λ es una constante positiva y C un término de conectividad $C = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} w_{ij} d_{ij}$, siendo d_{ij} la distancia entre los centros de los bloques rectangulares y w_{ij} el coste relacionado al unir el bloque rectangular i-ésimo con el bloque rectangular j-ésimo.

7. *Particionamientos de un grafo. Problema "max-cut". Problema "min-cut".*
   Dado un grafo $G = (V, E)$ sin ciclos, donde cada arista lleva asociada un peso entero positivo, se trata de encontrar una partición de V en dos conjuntos disjuntos V_0 y V_1 de tal manera que la suma de los pesos de las aristas que tienen un extremo en V_0 y el otro extremo en V_1, sea máximo (*problema "max-cut"*) o mínimo (*problema "min-cut"*).

8. *Problema del conjunto de vértices independientes.*
   Dado un grafo $G = (V, E)$, se trata de encontrar el denominado conjunto maximal de vértices independientes, es decir el conjunto $VIM \subset V$ de mayor cardinalidad para el cual ninguna de sus aristas se encuentre en E ($\forall u, v \in VIM \Longrightarrow \{u, v\} \notin E$).

Referencias

1. R. Etxeberria, P. Larrañaga (1999). Global optimization with Bayesian networks. *II Symposium on Artificial Intelligence. CIMAF99. Special Session on Distributions and Evolutionary Optimization*, 332–339.

2. P. Larrañaga, R. Etxeberria, J. A. Lozano, J. M. Peña (2000a). Combinatorial optimization by learning and simulation of Bayesian networks. *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence*, 343–352.

3. P. Larrañaga, R. Etxeberria, J. A. Lozano, J. M. Peña (2000b). Optimization in continuous domains by learning and simulation of Gaussian networks. *Proceedings of the 2000 Genetic and Evolutionary Computation Conference Workshop Program*, 201–204.

4. P. Larrañaga, J. A. Lozano (2001). *Estimation of Distribution Algorithms. A New Tool for Evolutionary Computation.* Kluwer Academic Publishers.

5. H. Mühlenbein (1998). The Equation for Response to Selection and its Use for Prediction. *Evolutionary Computation*, **5**, 303–346.

6. H. Mühlenbein, G. Paaß(1996). From Recombination of Genes to the Estimation of Distributions I. Binary Parameters. *Lecture Notes in Computer Science 1411: Parallel Problem Solving from Nature - PPSN IV*, 178–187.

7. M. Pelikan, D. E. Goldberg, E. Cantú-Paz (1999). BOA: The Bayesian optimization algorithm. *Proceedings of the Genetic and Evolutionary Computation Conference GECCO-99*, 525–532.