

UNA TÉCNICA PARA ESTABILIZAR LA CANTIDAD DE INDIVIDUOS EN ALGORITMOS GENÉTICOS DE POBLACIÓN VARIABLE

Patricio A. Maller

*The University of Alabama
College of Engineering
Department of Computer Science
Box 870290, Tuscaloosa, Alabama 35487-0290
e-mail: pmaller@simplecom.net*

Ignacio Ponzoni* , Gustavo E. Vazquez

*Departamento de Ciencias de la Computación
Universidad Nacional del Sur
Av. Alem 1253 - Bahía Blanca, 8000, Argentina
e-mails: ponzoni@criba.edu.ar, gvazquez@criba.edu.ar*

**Planta Piloto de Ingeniería Química (UNS - CONICET)
Camino La Carrindanga, Km 7, CC 717
Bahía Blanca, 8000, Argentina*

Raúl Gallard

*Proyecto UNSL-338403¹
Departamento de Informática
Universidad Nacional de San Luis
Ejército de los Andes 950 - Local 106
5700 - San Luis - Argentina
E-mail: rgallard@unsl.edu.ar
Teléfono: +54 2652 420823 - Fax : +54 2652 430224*

RESUMEN

La elección del tamaño de población condiciona el desempeño de los algoritmos genéticos. Una población pequeña puede aumentar la presión selectiva, conduciendo a una convergencia prematura, mientras que para un número de individuos muy grande el alcance de la solución se torna lenta. Una alternativa es usar un algoritmo genético de población variable, el cual ajusta la cantidad de individuos dinámicamente mediante el concepto de edad. Esta técnica, aunque más flexible, presenta una gran sensibilidad en relación con sus parámetros: pequeñas variaciones en el máximo tiempo de vida de un individuo pueden llevar de la extinción a la superpoblación para un mismo problema.

En este trabajo se presenta la noción de inundación, la cual permite mantener estable la población. La idea es eliminar, con cierta probabilidad, a individuos por debajo del fitness medio. Asimismo, también se incluye un control para evitar la extinción. Los resultados revelan que, aunque el mejor individuo promedio no supera al obtenido por el método original, el nuevo algoritmo logra mejores tiempos de ejecución pues opera sobre poblaciones más chicas. Además, la media poblacional se incrementa debido a las inundaciones. De este modo, nuestra propuesta resulta apropiada cuando se requieren buenas poblaciones con bajos tiempos de ejecución.

Palabras clave: Algoritmos genéticos, tamaño de población, adaptabilidad.

¹ El grupo de investigación está apoyado por la Universidad Nacional de San Luis y la ANPCYT (Agencia Nacional de Promoción Científica y Tecnológica).

1. INTRODUCCIÓN

Una de las áreas de mayor aplicabilidad de los algoritmos genéticos es la optimización estocástica, especialmente en problemas donde la probabilidad de ser atrapado en un óptimo local es alta. Los algoritmos genéticos clásicos abordan el problema empleando una población de tamaño fijo y los operadores genéticos de selección, crossover y mutación.

La elección del tamaño de población condiciona el desempeño del algoritmo [1], [3], [4], [5], [7], [8]. Una población demasiado pequeña aumenta la presión selectiva, conduciendo a una convergencia prematura. Por otra parte, si la cantidad de individuos es excesivamente grande la convergencia se demora innecesariamente.

Una alternativa sería ajustar el tamaño de la población en forma dinámica. Este es el enfoque perseguido por los algoritmos genéticos con tamaño de población variable. Dichas técnicas no emplean la noción de selección, incorporando en cambio el concepto de edad de un individuo, es decir, el número de generaciones que éste permanece vivo. En este trabajo, se tratarán distintos aspectos de esta metodología.

2. ALGORITMOS GENÉTICOS PARA POBLACIÓN DE TAMAÑO VARIABLE

2.1. ALGORITMO TRADICIONAL

El algoritmo genético de población variables (GAVaPS) procesa en tiempo t una población $P(t)$ de individuos. Para obtener sus descendientes directos, se genera una población auxiliar ($AuxPopSize(t)$) de tamaño proporcional a $P(t)$. Esta proporción está determinada por la tasa de reproducción ρ . Es decir, $AuxPopSize(t) = \lfloor PopSize(t) * \rho \rfloor$. Cada individuo de la población puede ser elegido para apareamiento con igual probabilidad, independientemente de su valor de fitness. Los descendientes se obtienen aplicando los operadores genéticos usuales de crossover y mutación. Por otra parte, dado que la selección no depende del valor de fitness, se introducen los conceptos de *edad* del individuo y *máximo tiempo de vida*. Este último es un parámetro del algoritmo y se mantiene fijo. Asimismo, la edad de los individuos se asigna dinámicamente durante la evaluación, luego de la inicialización o recombinación de miembros de la población auxiliar. Esto quiere decir que el tiempo de vida no se vuelve a calcular para los viejos individuos. El deceso se produce cuando su edad, i.e. el número de generaciones que el individuo permanece vivo, excede su tiempo de vida. En otras palabras, el tiempo de vida establece el número de generaciones que el algoritmo lo mantendrá en la población. De esta manera, el tamaño de la población, después de una iteración, es:

$$PopSize(t+1) = PopSize(t) + AuxPopSize(t) - D(t)$$

donde $D(t)$ es el número de individuos que mueren en la t -ésima generación.

A continuación, se muestra el algoritmo genético de población variable:

```

procedure GAVaPS
begin
  t = 0;
  inicializar P(t);
  evaluar P(t);
  while (not condición_de_terminación) do
  begin
    t = t+1;
    incrementar la edad de cada individuo en 1;
    recombinar P(t);
    evaluar P(t);
    remover de P(t) los individuos con tiempo de vida
      mayor al tiempo máximo de vida;
  end;
end;

```

2.2. ESTRATEGIAS DE ASIGNACIÓN DE EDAD

Una de las cuestiones a resolver es la política a seguir para la asignación de edades. Es claro que fijar un valor constante (mayor que uno) causaría un crecimiento exponencial de la población. Además, al no existir un mecanismo de selección, no hay presión selectiva y por ende el algoritmo tendría un pobre desempeño. Luego, resulta fundamental determinar una estrategia apropiada para la asignación de edades de los individuos. Estos criterios deberían:

- Promover aquellos individuos con fitness por encima de la media.
- Acotar el crecimiento desmedido de la población.

El apoyo a los mejores individuos debería conducir a poblaciones auxiliares por encima de la media. Por tal motivo, se debería garantizar un mayor tiempo de vida a los individuos de mayor fitness. Michalewicz [6], propone las siguientes alternativas:

a) Asignación proporcional

$$\min(\minLT + \eta \frac{fitness(i)}{avgFit}, \maxLT)$$

b) Asignación lineal

$$\minLT + 2\eta \frac{fitness(i) - AbsMinFit}{AbsMaxFit - AbsMinFit}$$

c) Asignación bilineal

$$\begin{cases} \minLT + \eta \frac{fitness(i) - MinFit}{Avg - MinFit} & \text{si } AvgFit \geq fitness(i) \\ \frac{1}{2}(\minLT + \maxLT) + \eta \frac{fitness(i) - AvgFit}{MaxFit - AvgFit} & \text{si } AvgFit < fitness(i) \end{cases}$$

donde *avgFit*, *maxFit* y *minFit* corresponden al promedio, el máximo y el mínimo de los valores de fitness respectivamente, mientras que *minLT* y *maxLT* representan el mínimo y máximo tiempo de vida.

La primera alternativa tiene su origen en el método de selección por ruleta del algoritmo genético simple. El tiempo de vida de un individuo es proporcional, dentro de los límites $minLT$ y $maxLT$, a su fitness. Una desventaja de esta estrategia es que no considera la “bondad objetiva” del individuo, la cual puede ser estimada en relación con el fitness del mejor individuo encontrado. Esto motivó la política de asignación lineal, donde el tiempo de vida se calcula de acuerdo a esa relación. No obstante, la presencia de muchos individuos con fitness cercanos al mejor valor puede provocar la asignación de largos tiempos de vida ocasionando un aumento considerable en el tamaño de la población. Finalmente, la asignación bilineal provee una solución intermedia entre las anteriores, al considerar no sólo el valor medio del fitness sino el mínimo y máximo fitness encontrado.

2.3. NUEVA PROPUESTA

Los valores apropiados para los parámetros del GAVaPS (tasa de reproducción, mínimo y máximo tiempo de vida) resultan difíciles de determinar. Más aún, se observa una alta sensibilidad a los valores asignados a los mismos. En otras palabras, pequeños cambios dentro un rango acotado de valores pueden ocasionar efectos tan diferentes como la extinción de la población o su crecimiento desmedido. En el primer caso, se corre el riesgo de terminar la exploración sin alcanzar un óptimo. Por otra parte, el tiempo de ejecución crece en una mayor proporción con respecto al tamaño de la población. A modo de ejemplo, la Figura 1 muestra la relación run-time/popSize para una corrida. En dicha curva se aprecia que cuando el tamaño de población crece de 1000 a 1500 (50%), el run-time aumenta en un 200%. Luego, surge la necesidad de alcanzar un comportamiento más estable.

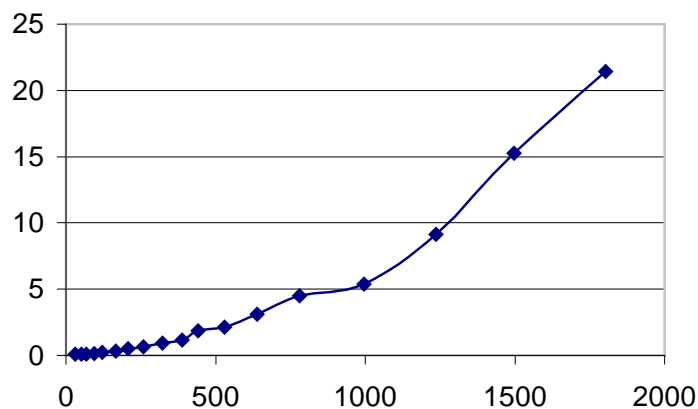


Figura 1. Relación run-time/popSize

En este trabajo se propone una variante del GAVaPS, la cual incorpora dos modificaciones. La primera, tendiente a evitar la extinción de la población, consiste en no decrementar la edad de los individuos cuando popSize es menor a un porcentaje del tamaño de la población inicial. La segunda, destinada a controlar el crecimiento exagerado de la población, introduce un nuevo concepto al cual denominamos “inundación”. Para comprender esta idea es necesario utilizar el concepto del “landscape” del problema, donde los individuos con mejor fitness se encuentran en los territorios más elevados. De este modo, los efectos de una inundación afectarían a aquellos con fitness por debajo del nivel de las aguas. En nuestra propuesta, fijamos ese nivel en el fitness medio dando a los

individuos inundados una probabilidad de supervivencia de $1 - pf$, donde pf es la probabilidad de perecer. Este último parámetro permite regular la presión selectiva y mejorar la sensibilidad del problema.

```

procedure GAVaPS con Inundación
begin
  t = 0;
  inicializar P(t);
  evaluar P(t);
  while (not condición_de_terminación) do
  begin
    t = t+1;
    incrementar la edad de cada individuo en 1;
    recombinar P(t);
    evaluar P(t);
    remover de P(t) los individuos con tiempo de vida
      mayor al tiempo máximo de vida;
    remover de P(t), con probabilidad pI, los individuos
      con fitness menor al fitness promedio de P(t).
  end;
end;

```

Ambos algoritmos se implementaron en MATLAB sobre la base del Algoritmo Genético Simple (GAOT) de Houck *et al.* (1996) [2]. En particular, ambos métodos siguen una política elitista. Se utilizó un crossover de un punto y mutación bit-flip.

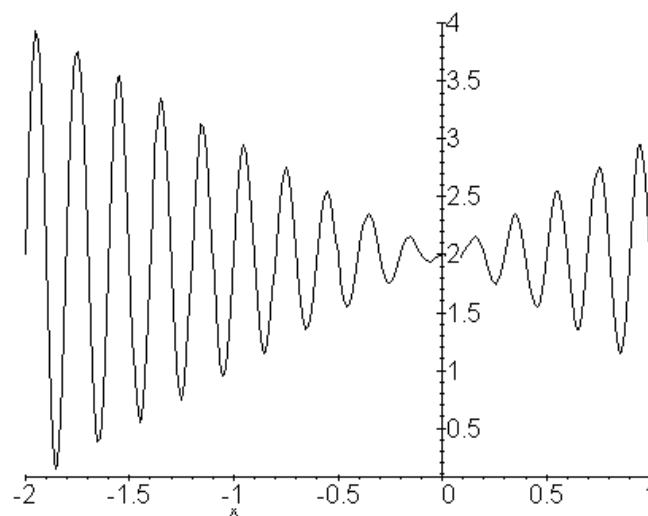
3. ANÁLISIS COMPARATIVO DEL DESEMPEÑO DE LAS TÉCNICAS

3.1. CASOS DE ESTUDIO

El algoritmo GAVaPS y su variante con inundación fueron testeados con las siguientes funciones:

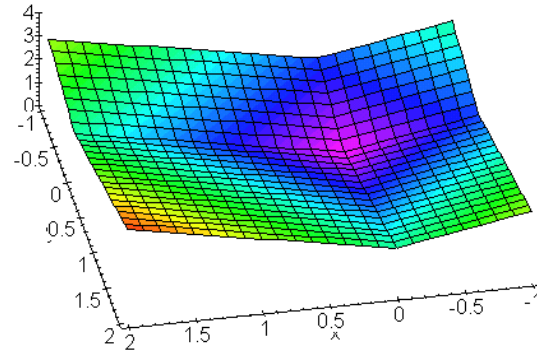
CE 1: $f1(x) = -x \cdot \text{sen}(10 \cdot \pi \cdot x) + 2 \quad -2 \leq x \leq 1$

Gráfico:



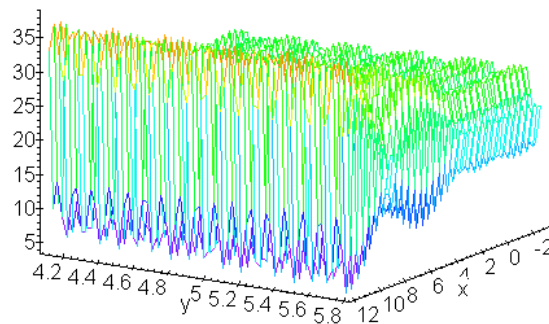
CE 2: $f_2(x, y) = |x| + |y| \quad -2 \leq x, y \leq 1$

Gráfico:



CE 3: $f_3(x, y) = 21.5 + x \cdot \sin(4 \cdot x \cdot \pi) + y \cdot \sin(20 \cdot y \cdot \pi) \quad -3 \leq x \leq 12.1 \quad 4.2 \leq y \leq 5.8$

Gráfico:



Los casos de estudio CE1 y CE3 han sido tomados de Michalewicz [6], los cuales tienen la particularidad de poseer numerosos máximos locales. El caso de estudio CE2 es un ejemplo de función deceptiva en el espacio bidimensional. En total se efectuaron 30 corridas para cada combinación de algoritmo, método de asignación de edad y función. La precisión se fijó en 1×10^{-4} .

El parámetro *Máximo Tiempo de Vida* se fijó en 6.4 para el método de asignación proporcional y 7.4 para el bilineal. Estos valores fueron determinados empíricamente, tratando de evitar la extinción o crecimiento desmedido de la población para los diferentes casos de estudio. Con respecto al método de asignación lineal, se observó que las poblaciones crecían excesivamente por lo que se decidió excluir esta alternativa del análisis. Por otra parte, se emplearon un *Mínimo Tiempo de Vida* de 1, una *Probabilidad de Mutación* de 0,15, una *Probabilidad de Crossover* de 0,65 y una *Tasa de Reproducción* de 0,4 para todas las corridas. Finalmente, para el método modificado se consideró una *Probabilidad de Perecer en la Inundación* de 0,015, 0,05 y 0,05 para CE 1, CE 2 y CE 3 respectivamente.

3.2. RESULTADOS

En esta sección se muestra el detalle estadístico de las corridas efectuadas. En la Tabla I se observa que salvo para CE 1, el mejor individuo promedio de ambos métodos es similar. Además, se aprecia un mejor desempeño de la estrategia de asignación de edad bilineal para ambos algoritmos. Por otro lado, la media poblacional promedio (mostrada en la Tabla II) es ligeramente superior en el método modificado para la mayoría de los casos.

En la Tabla III se muestra la media del promedio del tamaño de la población en cada corrida. Estos valores muestran que el crecimiento de la población es notablemente inferior para el método propuesto, lo que constituye una sensible mejora con respecto al algoritmo original.

Algoritmo	Estrategia de Asignación	CE1	CE2	CE3
GAVaPS	proporcional	3.7647	3.9569	37.3190
GAVaPS	bilineal	3.9150	3.9983	37.7809
GAVaPS/F	proporcional	3.7378	3.9293	37.2050
GAVaPS/F	bilineal	3.8105	3.9980	37.4095

Tabla I. Mejor Individuo Promedio

Algoritmo	Estrategia de Asignación	Función I	Función II	Función III
GAVaPS	proporcional	3.2374	3.4494	30.3625
GAVaPS	bilineal	3.9070	3.9845	37.6975
GAVaPS/F	proporcional	3.3688	3.5649	30.7269
GAVaPS/F	bilineal	3.7985	3.9852	37.3332

Tabla II. Media Poblacional Promedio

Algoritmo	Estrategia de Asignación	Función I	Función II	Función III
GAVaPS	proporcional	86.0175	144.7267	103.5308
GAVaPS	bilineal	352.8310	213.4400	45.5350
GAVaPS/F	proporcional	45.1567	64.0920	55.4721
GAVaPS/F	bilineal	186.3622	125.8417	34.5224

Tabla III. Tamaño Poblacional Promedio

4. CONCLUSIONES

De los resultados obtenidos y las experiencias desarrolladas se puede concluir que el algoritmo con tamaño de población variable (GAVaPS) presenta una gran sensibilidad en relación con sus parámetros. En particular, la elección del *Máximo Tiempo de Vida* constituye una de las tareas más difíciles, pudiendo pasar de la extinción de la población al crecimiento desmedido en intervalos muy pequeños.

En este trabajo se ha propuesto la variante GAVaPS/F que permite controlar el crecimiento de la población mediante el concepto de inundación. La idea es eliminar con cierta probabilidad a individuos por debajo del fitness medio. Otra característica es la inclusión de un control de extinción.

Con respecto a los resultados obtenidos, si bien el mejor individuo promedio no supera al obtenido por el método original, el nuevo algoritmo logra un mejor desempeño en cuanto al tiempo de ejecución debido a que opera sobre poblaciones de menor tamaño. Además, la media poblacional también se incrementa por efecto de las inundaciones. De este modo, el GAVaPS/F resulta apropiado para problemas que requieran obtener buenas poblaciones con bajos tiempos de ejecución.

5. REFERENCIAS

- [1] **De Jong K.**, “*An analysis of the behavior of a class of genetic adaptive systems*”. Dissertation Abstracts Int. 36 5140B (university microfilms No 76-9381) 1975.
- [2] **Houck C.R., Joines J., Kay M.G.**, “*A Genetic Algorithm for Function Optimization: A MATLAB implementation*”. ACM Transaction on Mathematical Software, submitted 1996.
- [3] **Goldberg D. E.**, “*Sizing populations for serial and parallel genetic algorithms*”, Proc. 3rd Int. Conf. on Genetic algorithms, San Mateo CA, Morgan Kaufmann, pp. 70-79, 1989.
- [4] **Goldberg D., Deb K., Clark J.**, “*Accounting for noise in sizing populations*”, Foundations of genetic algorithms 2, San Mateo CA, Morgan Kaufmann, pp. 127-140, 1992.
- [5] **Grefenstette J.J.**, “*Optimization of control parameters for genetic algorithms*”, IEEE Trans..Syst. Man cybernet. SMC 16, pp. 122-128, 1986.
- [6] **Michalewicz Z.**, “*Genetic Algorithms + Data Structures = Evolution Programs*”, Springer-Verlag, 3^o Ed., 1996.
- [7] **Smith R. E.**, “*Population size - handbook of evolutionary computation E1.1 – E1.5*”. Ed Back, Fogel, Michalewicz, Oxford University Press, 1977.
- [8] **Smith R.**, “*Adaptively resizing populations: an algorithm and analysis*”, TCGA report 93001, University of Alabama, 1993.