

Asignación automática de aulas en la Universidad Nacional del Chaco Austral mediante la aplicación de algoritmos genéticos con métodos de nichos

Ruben Ernesto Andreu, Licenciado en Matemáticas; Pedro Daniel Leguiza, Licenciado en Matemáticas; Patricia Paola Zachman, Ingeniera en Computación; Marina Beatríz Bloeck, Licenciada en Matemáticas; Ricardo Umbert Nasif, Ingeniero en Química

Todos los autores pertenecen a la Universidad Nacional del Chaco Austral, Presidencia Roque Sáenz Peña, Chaco, Argentina - Correos electrónicos: rubenandreu@uncaus.edu.ar, pdleguiza@gmail.com, ppz@uncaus.edu.ar, marina@uncaus.edu.ar, ricardo@uncaus.edu.ar

Abstract—En el cuerpo del trabajo se aborda en primer lugar los motivos que dan lugar a la necesidad en la Universidad Nacional del Chaco Austral, ubicada en Presidencia Roque Sáenz Peña, provincia del Chaco, República Argentina, para solucionar un problema de optimización de recursos como lo es la asignación de aulas para cursado de las asignaturas de carreras presenciales. En segundo lugar, se introduce la idea del llamado “problema de cursada universitaria”, mencionando además sus características. Luego se detalla el funcionamiento de la técnica de inteligencia artificial denominada algoritmos genéticos y los métodos de nichos vinculados a los mismos, a partir del estado de arte sobre el tema. Posteriormente se proporciona una hipótesis de solución al problema de la asignación automática de aulas a través de algoritmos genéticos con aplicación de métodos de nichos. Finalmente se sintetizan las actividades a desarrollar para un plan de investigación de la solución propuesta.

Index Terms—cursado, asignación, automática, algoritmos genéticos, nichos

I. INTRODUCCIÓN

La Universidad Nacional del Chaco Austral (UNCAUS), ubicada en la ciudad de Presidencia Roque Sáenz Peña, provincia del Chaco - Argentina, es una institución creada en el año 2008 con un constante crecimiento tanto en su oferta académica como en su matrícula. En la actualidad cuenta con 15 ofertas académicas de grado, 17 ofertas de posgrado (Especializaciones, Diplomaturas y Maestrías) y diferentes cursos de actualización y perfeccionamiento, todos ellos presenciales. Estas propuestas académicas aglutinan a aproximadamente 7000 estudiantes. Teniendo en cuenta que cada una de estas ofertas tienen características y necesidades distintivas, tales como espacialidad, horarios de desarrollo de clases, herramientas de trabajo, correlatividades, asignaturas que comparten profesor, entre otras, se visualiza como una tarea compleja la de asignar aulas para el cursado de materias. Hasta el momento esta tarea se realiza manualmente con el apoyo de herramientas digitales de ofimática, y es llevada a cabo por personas idóneas del área de alumnado, que van arreglando o mejorando una solución inicial, adecuándola a los nuevos requerimientos, tales como: asignación de aulas para nuevas materias, docentes que ingresan y se incorporan para las mismas, aulas nuevas que se agregan, entre

otras exigencias; considerando además transversalmente los horarios disponibles de los actores y recursos. Esto conduce al siempre persistente inconveniente de utilizar de la mejor manera posible los mismos, conjugando horarios de alumnos y docentes, tarea que es muy difícil de lograr considerando que se va arreglando una única solución inicial. Lo expuesto motivó una posible respuesta a la cuestión a través de la asignación automática de aulas mediante algoritmos genéticos con métodos de nichos.

En lo que sigue del presente artículo se va a denominar “problema de optimización” (maximización o minimización) a aquel en que dados una región R , una función f y un conjunto de restricciones r , consiste en hallar un punto de R que optimice el valor de f (función objetivo) cumpliendo con las restricciones impuestas por el conjunto r .

En la actualidad, la optimización brinda como alternativas herramientas algebraicas o numéricas sumamente confiables, pero que solo pueden aplicarse a funciones objetivo con determinadas condiciones tales como linealidad, unimodalidad (presencia de un único óptimo), diferenciabilidad y continuidad, entre otras. Sin embargo, la mayoría de los problemas con los que se enfrenta la ciencia no cumplen con las condiciones de aplicación para los métodos tradicionales [1] [2]. En consecuencia, durante la década de 1960, algunos investigadores recurrieron a algoritmos de búsqueda aleatorios, pero fracasaron en virtud a que un desbocado azar no permitía el control sobre los resultados. En la siguiente década, John Holland [3] recurre a la imitación de métodos biológicos como forma de controlar la aleatoriedad en la búsqueda de puntos para optimizar una función. Nacen así, los Algoritmos Genéticos (en adelante AG).

Por otra parte, dentro de los problemas de optimización existe uno particular con especial importancia para las instituciones educativas universitarias, conocido en idioma español como: problema de horario de cursado universitario (PHCU), o en inglés como: class teacher timetabling problem (CTTP) o university course timetabling problem (UCTP). El mismo se define como un problema de asignación de recursos con el fin de determinar aulas y horarios para clases, de tal forma

que no se dicten dos simultáneamente en el mismo aula o que sean dictadas por el mismo profesor en cualquier momento, proponiéndose restricciones de dos tipos: blandas y duras. Las primeras pueden no ser cumplidas con cierto grado de flexibilidad, mientras que las restricciones duras deben ser cumplidas en su totalidad de manera inflexible [4] [5].

II. MÉTODO

A. Conceptos básicos de algoritmos genéticos simples

Una población en un AG es un conjunto de estructuras de datos donde cada una de ellas representa una posible solución. Una estructura de datos o individuo consiste en uno o más cromosomas, en tanto uno de éstos es generalmente una cadena de bits o un vector de números reales [6] [7]. Cada cromosoma (cadena) es una concatenación de un número de subcomponentes llamados genes. Los genes se encuentran en varias posiciones o locus del cromosoma, y toman valores llamados alelos. En una representación de cadena de bits, un gen es un bit, un locus es su posición en la cadena, y un alelo es su valor (0 o 1). El término biológico genotipo se refiere a la composición genética total de un individuo, y corresponde a una estructura en el AG. Mientras que el término fenotipo se refiere a las características externas de un individuo y corresponde a la estructura decodificada del AG.

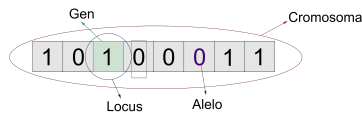


Fig. 1. Esquema de conceptos básicos de un individuo en un AG.

Además, para optimizar una función usando un AG, debe ser posible asignar una medida de calidad a cada individuo en la población -o estructura en el espacio de búsqueda-. La responsable de ésta tarea es la función de adaptación. Los AGs maximizan por defecto, para problemas de minimización, se niegan los valores de la función objetivo.

B. Funcionamiento

Una vez creada la población inicial, el funcionamiento de un AG simple consta esencialmente de tres fases: selección, cruzamiento y mutación. Completándose el ciclo referido anteriormente, se dice que ha pasado una generación. En primer lugar, los AG simples generan aleatoriamente una población inicial de n individuos de longitud fija l . Para luego ejecutarse por un número fijo de generaciones o hasta que se satisfaga algún criterio de parada. Como se mencionó previamente, durante cada generación los AG realizan una selección de individuos proporcional a la adaptación, luego el cruzamiento o reproducción, y por último la mutación [8]. La función de adaptación proporcional (1) asigna a cada estructura individual en la población una probabilidad de selección $p_s(i)$, de acuerdo a la cantidad de adaptación del i -ésimo individuo $f(i)$, sobre la adaptación total de la población $\sum_{j=1}^n f(j)$:

$$p_s(i) = f(i) / \sum_{j=1}^n f(j) \quad (1)$$

Entonces el AG selecciona con reemplazo un total de n individuos para el posterior procesamiento genético, de acuerdo a la distribución definida por $p_s(i)$ [9]. Después de la selección, los n individuos escogidos se someten al cruzamiento (también llamado recombinación o apareamiento) con una probabilidad fija p_c por cadena [8] [10]. En esta fase, las n cadenas seleccionadas son pareadas al azar, produciendo $\frac{n}{2}$ parejas. Para cada una, el cruzamiento puede o no ocurrir. Con probabilidad $1 - p_c$ el cruzamiento no ocurre y ambos individuos pasan a la fase de mutación. De lo contrario, la pareja produce dos descendientes (hijos) vía cruzamiento y solo éstos últimos continúan a la fase de mutación.

Después que finaliza la fase de cruzamiento, comienza la de mutación. Para cada cadena que ha avanzado a la fase de mutación, cada uno de sus genes es cambiado con una probabilidad p_m . Luego, la población resultante de la fase de mutación reemplaza a la antigua población (la primera previa a la selección), completando una generación. Las generaciones sucesivas siguen el mismo ciclo de selección, cruzamiento y mutación.

Existen numerosas alternativas a los operadores de selección, mutación y recombinación presentados anteriormente [6] [7] [8] [9] [10], los que modifican sustancialmente el funcionamiento de AG diseñado, y deben ser utilizados de acuerdo a los objetivos propuestos para el mismo.

En resumen, se puede esquematizar el funcionamiento de un AG simple a través del diagrama de flujo representado en la Fig. 2.

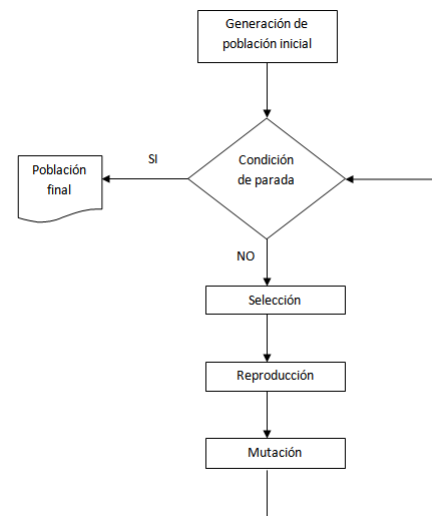


Fig. 2. Esquema de funcionamiento de un AG simple.

C. Diversidad

Los mecanismos de selección tradicional de los AGs simples replican soluciones muy adaptadas y descarta las demás,

conduciendo a una convergencia de la población a un único punto -unimodalidad-. Sin embargo, dadas varias soluciones de idéntica adaptación, la población seguirá convergiendo pero el AG de manera aleatoria visitará cada uno de los óptimos. Los AGs simples pierden soluciones y subsoluciones debido a tres causas:

- 1) presión de selección
- 2) ruido de selección
- 3) disrupción por operadores

La presión de selección es el resultado del valor esperado del proceso de selección: es esperable que soluciones menos adaptadas tiendan a desaparecer de una población finita.

El ruido de selección resulta de la variación del proceso de selección: en una población finita, elecciones aleatorias entre subsoluciones idénticamente adaptadas que compiten entre sí, agregan ruido al recuento esperado para cada individuo, forzando eventualmente buenas soluciones de la población.

La disrupción por operadores resulta de la aplicación del cruzamiento y la mutación, los cuales son capaces de destruir directamente buenas soluciones.

Existen diversas técnicas que se aplican a AGs simples para diversificar una población, y que comúnmente reducen la presión, el ruido de selección, la disrupción por operadores, o alguna combinación de ellos [11] [12] [13] [14]. Sin embargo, un AG simple no puede conseguir ni mantener múltiples puntos óptimos.

D. Algoritmos genéticos con métodos de nichos

Los métodos de nichos son técnicas para promocionar la formación y el mantenimiento de subpoblaciones estables en un AG. Además, pueden ser aplicados a la formación y mantenimiento de subsoluciones intermedias en la búsqueda de una solución final única. Sin embargo, se los utiliza generalmente para formar y mantener múltiples soluciones finales [15].

Un método de nicho debe ser capaz de formar y mantener varias soluciones finales y diversas, aunque sean de igual adaptación o variable. También, debe tener la capacidad de mantener estas soluciones por un período de tiempo exponencial o infinito, con respecto al tamaño de la población. Lo que en su momento se requirió de un AG, no fue solo menor presión de selección, ruido reducido de selección, o poca disrupción por operadores, sino un nuevo tipo de algoritmo que promocionara la diversidad a través de dimensiones útiles de la misma, mientras permitía converger a otras dimensiones [15].

Con respecto al espacio de búsqueda, la convergencia puede ocurrir en algún grado dentro de regiones locales, pero la diversidad debe prevalecer a través de las regiones más prominentes o adaptadas. Los métodos de nicho alteran el algoritmo de selección para proveer presión de selección dentro, pero no a través de las regiones del espacio de búsqueda. La presión de selección dentro de regiones individuales puede ser substancial y preservar propiedades de los nichos.

El mejor método de nicho será capaz de localizar el pico más alto en presencia de un gran número de picos menores, como también en presencia de picos falsos o engañosos. En

conclusión, el mejor método de nicho no será solamente selectivo, también tendrá la habilidad de formar y mantener tanto óptimos globales, como no globales. Según Mahfoud [15] se asume que los AGs simples pueden formar buenas soluciones, y que ésta capacidad se transfiere a los AGs con nichos. Si esta capacidad no se transfiere a un algoritmo en particular, no puede ser considerado un método de nicho.

Los métodos de nicho generalmente pueden ser clasificados de dos formas:

- a) con respecto al espacio o tiempo
- b) a la cantidad de entornos en los que se desarrollan.

La primera clasificación se puede dar en los métodos de nicho tomados como algoritmos espaciales, que forman y mantienen subpoblaciones dentro del espacio de una sola población. Los métodos de nicho espaciales son llamados en ocasiones métodos de nicho en paralelo, porque conceptualmente desarrollan nichos en paralelo dentro de una sola población. En el extremo opuesto a la clasificación comentada previamente, se tiene a los métodos temporales, que forman y mantienen subpoblaciones en el tiempo. Los métodos de nicho temporales son llamados también en ocasiones métodos de nicho secuenciales, porque desarrollan nichos secuencialmente en el tiempo. Esta designación de paralelo y secuencial es independiente del número de procesadores físicos empleados.

La segunda clasificación de los métodos de nichos se da entre aquellos que se desarrollan en un único ambiente o en múltiples ambientes. Podemos ver un entorno completo como simple/múltiple, o espacial/temporal. Algunos métodos de nicho son capaces de formar y mantener nichos estables dentro de un único entorno (el entorno o ambiente en funciones de optimización multimodal consiste en una sola función de adaptación).

La Tabla 1 ubica cada categoría de métodos de nichos [15], con respecto a las dos clasificaciones abordadas en los párrafos anteriores. Los métodos de nicho en cada cuadrante de la tabla son similares en cuanto a los tipos de problemas para los cuales son aplicables, por ende entre paréntesis se expondrán algunos de ellos.

TABLE I
CLASIFICACIÓN DE AG CON APLICACIÓN DE MÉTODOS DE NICHOS SEGÚN CANTIDAD DE AMBIENTES Y DESARROLLO DE NICHOS.

Desarrollo de nichos/Ambiente	Único	Múltiple
Temporal	Localización secuencial (clasificación y optimización de funciones multimodales y multiobjetivos)	Sobre especificación - AGs ecológicos (simulación adaptativa)
Espacial	Ventaja heterocigota - Hacinamiento - Competición restringida - Adaptación compartida (optimización de funciones multimodales)	Ags ecológicos - Sistemas inmunes (optimización de funciones multiobjetivos)

III. TRABAJOS PREVIOS

Considerando que existe una amplia gama de programas para diseñar AG, pero los mismos son pagos y además funcionan como cajas negras, anteriormente se realizó un

trabajo [16] para profundizar sobre la temática y poner en funcionamiento tanto un AG simple, como otro con método de nicho. Inicialmente se realizó una investigación bibliográfica sobre el tema y posteriormente se diseñó el AG simple y sobre la base del mismo, otro con aplicación de métodos de nichos, denominado: “nichos secuenciales”. Posteriormente se efectuó pruebas de los algoritmos codificados sobre funciones de prueba con múltiples picos y diferentes tipos de complejidades anexas, que modificaban cantidad, altura y distancias entre los puntos críticos. Al finalizar la investigación se probó el correcto funcionamiento de los algoritmos diseñados y codificados, como también se verificó la supremacía de los AG con métodos de nichos sobre los simples. El trabajo reseñado previamente fue elaborado por un egresado de la UNCAUS para obtener el título de grado de Licenciado en Matemáticas.

IV. PROPUESTA

Tomando como punto de partida lo expuesto en la sección anterior y considerando el PHCU, surgió la idea de proyectar una solución respecto de la optimización en la asignación de aulas en la universidad en forma automática mediante la aplicación de algoritmos genéticos con métodos de nicho. Además, el proyecto persigue el objetivo de generar una batería propia de AG de distintos tipos, cada uno de éstos aplicables a diferentes clases de problemas y con amplia posibilidad de manipulación de parámetros internos de los AG, como así también se propone formar personal para trabajar en forma interdisciplinaria con otros grupos de investigación a fin de adecuar los AG a requerimientos específicos de otras áreas de investigación. Durante el desarrollo de la investigación se codificarán diversos AG evaluándose en un primer momento su trabajo aplicándolos a funciones de prueba complejas. Paralelamente se abstraerán elementos esenciales del problema de asignación de aulas para su tratamiento con los mismos. Posteriormente se analizarán diversas combinaciones de rutinas de creación de poblaciones, selección de individuos, cruzamiento y mutación, en la búsqueda de mejorar la performance del algoritmo general. Simultáneamente, se analizarán modificaciones de parámetros en cada AG.

Para el desarrollo del trabajo se utilizará el lenguaje de programación de código libre Octave. Además se planea conformar la población de individuos codificados como matrices de tres dimensiones (horarios diarios, aulas y días de la semana). La adaptación dentro de la población se medirá con una función de penalización que se deberá minimizar, en la misma cuando un individuo presente en sus genes aspectos contrarios a las restricciones sumará puntos como castigo o pena.

Considerando la naturaleza del problema y que en la búsqueda de la solución al mismo se involucra a muchas personas, la idea de conseguir una única solución que cumpla con todas las restricciones es utópica. Por ello, con la aplicación de los métodos reseñados en la sección anterior se busca la formación de diferentes nichos que permitan al personal del área de alumnado tener buenas referencias de soluciones efi-

cientes al problema, para llevar adelante la aplicación práctica de las mismas gestionando acciones entre actores y recursos.

V. ESTADO DEL ARTE

La mayor parte de los trabajos consultados utilizaron AG simples para el proceso de optimización [17] [18] [19] [20] [21] [22] [23] [24]. Sin embargo también se realizaron investigaciones con algoritmos meméticos que proponen un método de integración de búsqueda global con búsqueda local [25], y con una mezcla entre AG simple y un algoritmo de creación de un padre aleatorio como forma de contrarrestar el estancamiento de la búsqueda [26].

Considerando que el PHCU involucra cantidades discretas [24], en los trabajos analizados para la codificación de los individuos de las poblaciones se estableció como regla, debido a su mejor funcionamiento, la utilización de vectores de números reales que se convierten a vectores de números enteros de diversas formas preestablecidas [25].

Por otra parte, se analizó en los artículos la forma de representar los individuos, determinándose que se han utilizado matrices tridimensionales (con aulas, horarios y días de la semana como dimensiones) [26]; matrices bidimensionales donde se conjugan solamente horarios y aulas por día [21] [24], como también vectores con segmentos de genes de la forma clase-docente-aula-hora-grupo que se replican hasta cubrir las opciones semanales.

En cuanto a los operadores propios de la ejecución de los AG se han puesto en funcionamiento estrategias tradicionales de selección, reproducción y mutación. Sin embargo se encontró la utilización de una técnica de recombinación con dos puntos de cruce [27], como también el uso de una mutación estática y otra progresiva, buscándose con el diseño dual de dos mecanismos de mutación mantener una diversidad de individuos que eviten que el algoritmo quede atrapado en un óptimo local. En otro caso [22] se agregó una mutación por intercambio.

En cuanto a la función de fitness o adaptación se observó que se recurre a la asignación de una medida de acuerdo a la penalización de individuos por errores [20] [22] [24] [26] o por desperdicio de espacio y minimización de cambios en función a una buena solución actual –obtenida manualmente o por experiencia práctica– [17]. Presentándose también la existencia de funciones de adaptación global y locales [20] [22], ya sea trabajando forma simultánea o alternada.

En todos los casos analizados, de acuerdo al problema bajo examen, se trabajan con restricciones duras y blandas, cambiando solamente si el tratamiento de las mismas es simultáneo o alternado [24] [25].

Asimismo, se realizó un estudio [23] sobre cómo afecta al AG las variaciones en tres de los parámetros con mayor ocurrencia en las diversas instancias, como son: a) el tamaño de la instancia, entendiéndose tal como cantidad de aulas, docentes, horarios y asignaturas a combinar, b) la afinidad de los docentes con las diferentes asignaturas y c) la disponibilidad de los docentes o afinidad de estos con los horarios de clase. Lo anterior manteniéndose especial interés en dos factores:

el tiempo que transcurre para encontrar la mejor solución alcanzable y la optimalidad de dicha solución con respecto al máximo posible. Los investigadores concluyeron de acuerdo a los hallazgos realizados que el tamaño de la instancia es el único parámetro significativo en cuando al tiempo de solución; no así en cuanto a la calidad de la solución obtenida, ya que en ese caso tanto el tamaño de la instancia como la afinidad de los docentes con las asignaturas se distinguieron como los más. Además acotaron los investigadores que el método de solución mediante algoritmos genéticos representa una alternativa conveniente para la resolución de problemas de asignación de recursos académicos.

En todos los artículos bajo examen se reportaron buenos resultados con las diferentes metodologías implementadas, más allá de las heterogéneas formas de medir la adaptación de las soluciones. Dentro del contexto de estudio se han analizado resultados de investigaciones vinculados con problemas similares (Job Shop Scheduling) al que aborda el presente artículo [5] [27] [28], como algunos que consideraron los ritmos cognitivos (cercanía de determinadas materias o tipos de conocimiento para facilitar el proceso de enseñanza aprendizaje) para determinar las soluciones óptimas [18] [26], concluyendo la investigación que el AG que considera los ritmos cognitivos es un quince por ciento más eficiente en las soluciones que aquellos que consideran solamente parámetros tradicionales.

VI. CONCLUSIONES

Lo expuesto previamente en el artículo permite afirmar la factibilidad de desarrollar el proyecto de investigación con el objetivo de poner en funcionamiento un sistema que permita asignar automáticamente aulas en la UNCAUS, tomando como base el trabajo de elaboración de códigos abiertos realizado por un egresado de la misma para obtener el título de Licenciado en Matemática. Además, analizado el estado del arte plasmado se pudo determinar que si bien se buscó soluciones al problema planteado mediante AG o variantes de los mismo complementados con otros tipos de heurísticas, no se abordó el problema con la implementación de técnicas de nichos, por lo que tal trabajo a futuro aportará alguna novedad al cuerpo de conocimientos establecidos en la materia.

VII. BIBLIOGRAFÍA

REFERENCES

- [1] G. Gonzalez Oliva, "Métodos clásicos de optimización para problemas no lineales sin restricciones", Departamento de Ingeniería Matemática, Universidad de Chile, 2006.
- [2] J.A. Caballero Suarez, J.R. Ruiz Femenia e I. Aracil Sáez, "Simulación de procesos químicos (curso 2011-2012)", Departamento de Ingeniería Química, Universidad de Alicante, 2011.
- [3] J. H. Holland, "Adaptation in natural and artificial systems", University of Michigan Press, Ann Arbor, 1975.
- [4] T. Muller, H. Rudova y Z. Mullerova, In PATAT 2018 - Proceedings of the 12th International Conference on the Practice and Theory of Automated Timetabling, 2018. Recuperado de: "https://www.unitime.org/papers/itc2019-patat2018.pdf".
- [5] N. Aziz, A. Liyana y N.A.H. Aizam, "A brief review on the features of university course timetabling problem", AIP Conference Proceedings. Vol. 2016, Doi:10.1063/1.5055403, 2018. Recuperado de: "https://aip.scitation.org/doi/abs/10.1063/1.5055403".
- [6] L. Davis, "Handbook of genetic algorithms", Van Nostrand Reinhold, 1991.
- [7] D.E. Goldberg, "Real-coded genetic algorithms, virtual alphabets and blocking", Complex Systems, 1991.
- [8] D.E. Goldberg, "Genetic algorithms in search, optimization and machine learning, Addison-Wesley, Reading, MA, 1989.
- [9] D.E. Goldberg y K. Deb, "A comparative analysis of selection schemes used in genetic algorithms", Foundation of Genetic Algorithms, 1991.
- [10] G. Syswerda, "Uniform crossover in genetic algorithms", Proceedings of the third International Conference on Genetic Algorithms, 1989.
- [11] A. Brindle, "Genetics algorithms for function optimization", Tesis Doctoral, University of Alberta, 1981.
- [12] D. Thierens y D.E. Goldberg, "Mixing in genetic algorithms", en Proceedings of the fifth International Conference on Genetic Algorithms, 1993.
- [13] M.L. Mauldin, "Maintaining diversity in genetic search", Proceedings of the National Conference on Artificial Intelligence, 1984.
- [14] R. Tanese, "Distributed genetic algorithm", Proceedings of the third International Conference on Genetic Algorithms, 1989.
- [15] S.W. Mahfoud, "Nicheing methods for genetic algorithms", Illinois Genetic Algorithms Laboratory (IlliGAL) - Department of General Engineering - University of Illinois, 1995. Recuperado de: "http://www.cse.unr.edu/~sushil/class/gas/papers/mahfoud95nicheing.pdf".
- [16] R.E. Andreu, "Una aproximación a los algoritmos genéticos simples y con métodos de nichos", Trabajo Final para obtener el título de grado de Licenciado en Matemáticas, Universidad Nacional del Chaco Austral, Argentina, 2014.
- [17] M.A. Antunez, "Sistema para optimizar asignación de aulas UNICEN". Trabajo final de la carrera de grado de Ingeniería en Sistemas, Universidad Nacional del Centro de la Provincia de Buenos Aires, Argentina, 2015.
- [18] O. Castrillón, "Ritmos cognitivos y algoritmos evolutivos en la programación de horarios universitarios", Revista de Matemática Teoría y Aplicaciones, 22(1), 135-152, 2015.
- [19] H.A. Castañeda, "Un algoritmo genético para la asignación de aulas, docentes y asignaturas de la Escuela Profesional de Ingeniería en Sistemas de la Universidad Nacional de Callao", Informe final de proyecto de investigación, Universidad Nacional de Callao, Perú, 2016.
- [20] J.M. Mejía Caballero, "Asignación de horarios de clases universitarias mediante algoritmos evolutivos", Trabajo de grado para obtener el título de magíster en Ingeniería Industrial, Universidad de Barranquilla, Colombia, 2008.
- [21] Y. Solano Sabatier, M. Calvo Marin y L. Trejos Picado, "Implementación de un algoritmo genético para la asignación de aulas en un centro de estudio", Uniciencia 22, págs. 115-121, Costa Rica, 2008.
- [22] M.J. Broca, "Algoritmos genéticos en la generación de horarios escolares", Tesis para obtener el grado de Maestro en Ciencias de la Computación, Universidad Autónoma del estado de México, México, 2016.
- [23] A.F. Salazar, J.F. López, A. Tavizón y M.J. Araiza-Vázquez, "Estudio de un Algoritmo Genético para la Administración Académica", Formación universitaria, 12(4), 63-72, 2019. Recuperado de: "https://dx.doi.org/10.4067/S0718-50062019000400063".
- [24] Rodriguez Maya N., Flores J.J. y Rodriguez Rangel H., "Performance Comparison of Evolutionary Algorithms for University Course Timetabling Problem", Computación y Sistemas, 20(4), 623-634, 2016. Recuperado de: "https://dx.doi.org/10.13053/cys-20-4-2504".
- [25] F.J. Pitol Reyes, "Uso de algoritmos evolutivos para resolver el problema de asignación de horarios escolares en la Facultad de Psicología de la Universidad Veracruzana", Tesis, Universidad Veracruzana, México, 2007.
- [26] O.D. Castrillón, "Combinación entre Algoritmos Genéticos y Aleatorios para la Programación de Horarios de Clases basado en Ritmos Cognitivos", Información tecnológica, 25(4), 51-62, 2014. Recuperado de: "https://dx.doi.org/10.4067/S0718-07642014000400008".
- [27] M. Jiménez-Carrión, "Algoritmo Genético Simple para Resolver el Problema de Programación de la Tienda de Trabajo (Job Shop Scheduling)", Información tecnológica, 29(5), 299-314, 2018. Recuperado de: "https://dx.doi.org/10.4067/S0718-07642018000500299".
- [28] T. Muller, In PATAT 2016 - Proceedings of the 11th international conference on the Practice And Theory of Automated Timetabling, 2016.