

Conocer técnicas como las de los algoritmos bioinspirados, que han mostrado simplicidad y eficiencia en la solución de problemas complejos, es útil para los Ingenieros de Sistemas. Bioescenarios IV crea modelos para el desarrollo de escenarios de aprendizaje sobre la tendencia inteligencia de enjambres: optimización por cúmulos de partículas.

Bioescenarios IV

Inteligencia de enjambres: optimización por cúmulos de partículas

Carlos Eduardo Casallas Fonseca
Steven Francisco Vargas Aristizábal

• introducción

Durante el siglo XX se observó un énfasis en el estudio y el avance pleno de la física; ahora, en el siglo XXI, la biología se consolida como la ciencia con mayor potencial de investigación y desarrollo. Son tan variados los campos de progreso de la biología y su influencia en la informática, que en la actualidad hay un número considerable de tendencias informáticas bioinspiradas, útiles tanto en la solución de problemas clásicos de la informática, como en las dificultades reales del complejo contexto actual [8] [11][13].

Bioescenarios IV tiene como objetivo desarrollar herramientas que apoyen el aprendizaje de la tendencia informática de inteligencia de enjambres, enfocada en optimización por cúmulos de partículas (*Particle Swarm Optimization, PSO*). En este artículo se presentan el contexto de Bioescenarios IV, la metodología aplicada, una síntesis del marco teórico, los modelos realizados,

tres escenarios de aprendizaje desarrollados usando los modelos y las conclusiones del proyecto.

Contexto

Bioescenarios es un proyecto de investigación cuyo propósito es construir escenarios de aprendizaje de conceptos fundamentales y estrategias de solución de problemas de tendencias informáticas bioinspiradas [8]. Los escenarios se construyen con base en un problema, cuya solución está guiada por los *modelos de conocimiento* para aprender los conceptos y estrategias de solución de una tendencia específica, apoyados sobre *modelos de experimentación* para explorar, validar y aplicar lo aprendido, solucionando así el problema base. Bioescenarios IV es un proyecto de grado que hace parte de Bioescenarios.

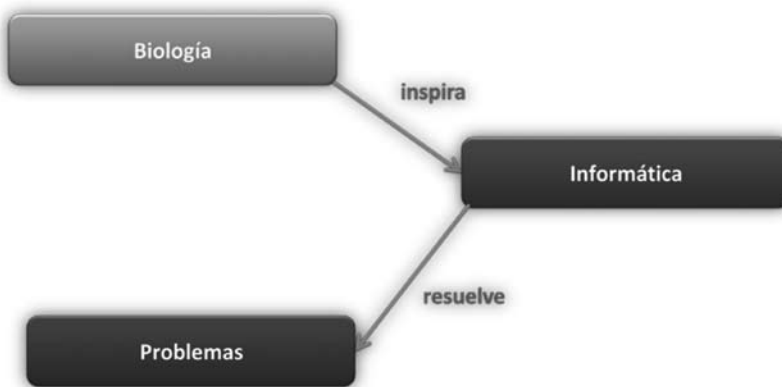


Figura 1. Mapa conceptual general bioescenarios.

El concepto de escenario se propuso en el proyecto de investigación Escenarios de aprendizaje [4], como “un conjunto de elementos armoniosamente acoplados para propiciar la comprensión y la interiorización de conceptos y principios, la formulación o respuesta de preguntas y la aproximación lúdica al estudio de problemas informáticos”.

Hasta el día de hoy se han realizado cuatro proyectos de grado de Bioescenarios:

- Bioescenarios I: Algoritmos genéticos (2005-2 a 2006-1) [13]
- Bioescenarios II: Colonia de hormigas (2006-2 a 2007-1) [8]
- Bioescenarios III: Sistema inmunológico artificial (2007-2 a 2008-1) [11]

- Bioescenarios IV: Optimización por cúmulos de partículas (2008-2 a 2009-1)



Figura 2. Mapa conceptual de Bioescenarios IV.

Cada uno de estos proyectos ha contribuido a cumplir el propósito de Bioescenarios, extendiendo el marco teórico de tendencias bioinspiradas, enriqueciendo la propuesta pedagógica de Escenarios [3] y construyendo los modelos de conocimiento y experimentación para el desarrollo de escenarios de aprendizaje propios de cada tendencia.

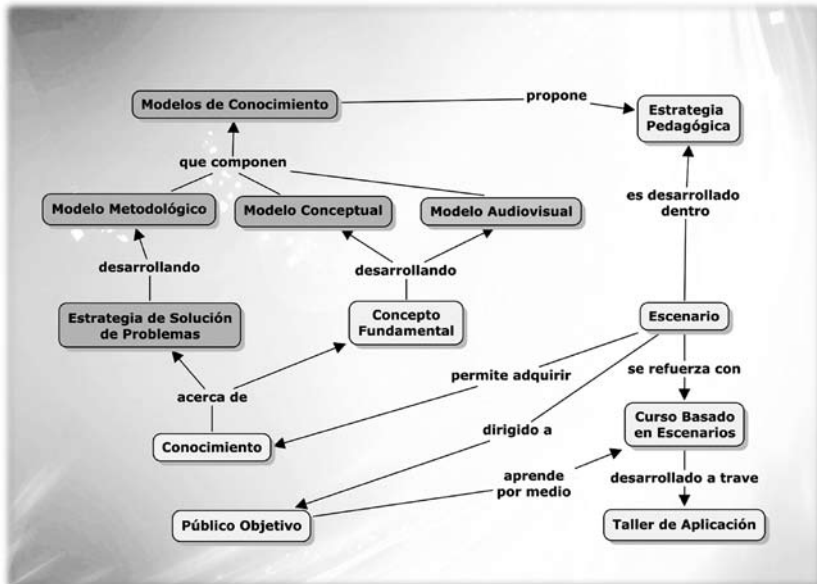


Figura 3. Mapa conceptual parcial de Escenarios de aprendizaje enriquecido (se resalta en color amarillo un nuevo modelo desarrollado).

Metodología

El proyecto se realizó en cinco etapas no secuenciales: **i)** Apropiación de los proyectos Escenarios y Bioescenarios; **ii)** Selección e investigación de la tendencia bioinspirada; **iii)** Desarrollo de modelos de conocimiento; **iv)** Desarrollo de modelos de experimentación; y, **v)** Validación de modelos construyendo escenarios.

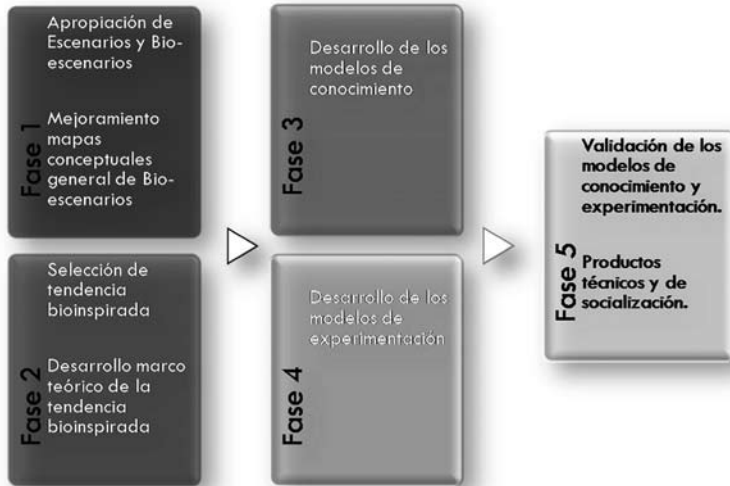


Figura 4. Metodología.

Marco teórico

En el marco teórico de Bioescenarios IV se presentan los aspectos biológicos e informáticos generales de la tendencia, estudiados por la teoría de enjambres y la inteligencia de enjambres, respectivamente. Contempla una aproximación teórica a los problemas de optimización y se explica en detalle el algoritmo de optimización por cúmulo de partículas.

Teoría de enjambres

Un enjambre es una agrupación considerable de animales de una misma especie que persiguen un objetivo común; no obstante, se pueden encontrar enjambres conformados por especies de animales diferentes, pero con semejanza morfológica o de tamaño entre los individuos [2]. Los enjambres más conocidos son los cardúmenes de peces, las bandadas de pájaros, las colonias de bacterias y los enjambres de insectos como hormigas, termitas y abejas.

Los objetivos más frecuentes para la conformación de enjambres son la búsqueda de comida o pareja, la protección ante actividad predatoria y el seguimiento de rutas migratorias [16] [17]. Aun cuando los individuos del enjambre no son inteligentes, las interacciones locales de estos permiten al enjambre cumplir con su objetivo de manera inteligente, beneficiando a todos los individuos en una actividad que no pueden hacer por sí solos o cuya realización demandaría un gran esfuerzo individual [9].

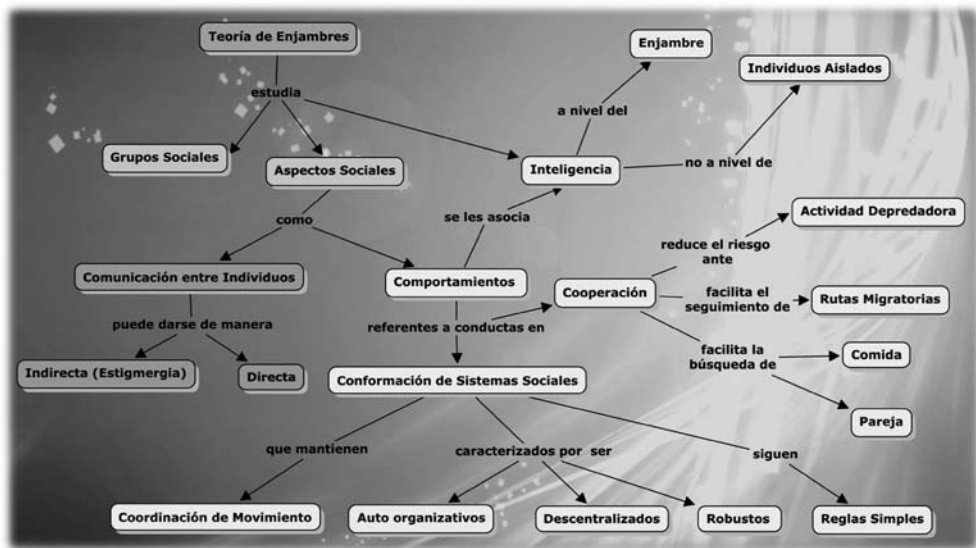


Figura 5. Mapa conceptual parcial de teoría de enjambres.

Una de las particularidades encontradas en los estudios biológicos de los enjambres, es que su comportamiento se basa en el cumplimiento de reglas simples [9]. Según algunos modelos de adaptación cultural, el comportamiento de un individuo en un enjambre sigue tres principios: evaluar, comparar e imitar [8]. Así, los individuos son capaces de modificar su opinión, considerando la de los de mayor éxito, formando así un conjunto de individuos con opiniones muy relacionadas, que guía el enjambre a su objetivo. Estos tres principios son la base de las reglas simples generales que siguen la mayoría de los enjambres.

Inteligencia de enjambres

La inteligencia de enjambres es una tendencia informática inspirada en la teoría de enjambres [8], que forma parte de la rama de sistemas multiagentes de la inteligencia artificial. Esta tendencia surgió porque los informáticos encontraron

aspectos interesantes en los enjambres, como los comportamientos sociales descentralizados y autoorganizativos [11], que podrían modelarse para solucionar problemas eficazmente, sin recurrir a implementaciones complejas.



Figura 6. Mapa conceptual parcial de inteligencia de enjambres.

Los algoritmos más conocidos que se basan en inteligencia de enjambres son [11] [13]: optimización por colonia de hormigas (ACO), el cual se desarrolló en la segunda fase de Bioescenarios [7]; enrutamiento basado en el comportamiento de termitas [13]; colmena de abejas artificiales (ABHA) [14]; y, el PSO, el cual se trabaja en Bioescenarios IV.

Todos estos algoritmos tienen un grupo de agentes computacionales autónomos que cumplen unas reglas simples, determinadas por el algoritmo particular. Las reglas están formuladas para que dichos agentes funcionen sin la necesidad de una estructura de control centralizada, de manera que interacciones locales de cada agente y sus vecinos permitan la emergencia de comportamientos globales inteligentes.

Problemas y optimización

Un problema tiene diferentes connotaciones, dependiendo de la perspectiva con la que se le mire; a pesar de ello, todos los problemas poseen una misma estructura, gracias a la cual se pueden estudiar. Básicamente, están compuestos por tres factores: una situación inicial, un estado u objetivo que se desea alcanzar y

unas limitaciones o restricciones [1]. Cuando se debe trabajar en un problema, lo que se busca es obtener una solución que satisfaga los objetivos, cumpliendo las restricciones impuestas.

Los problemas de optimización son una clase específica de problemas con los que nos enfrentamos a diario: mejorar la eficiencia de algún recurso, hallar el camino más corto entre dos puntos, distribuir uniformemente un recurso entre varias personas [3]. Un problema de optimización, en general, se formula como la minimización o maximización de una función objetivo que relaciona las variables relevantes del dominio del problema y durante el proceso de optimización se busca el conjunto de valores que maximicen o minimicen dicha función. Las variables representan el dominio del problema, y la función objetivo caracteriza el objetivo propio de este. Al igual que en cualquier problema, las variables se someten a restricciones que garantizan soluciones aceptables dentro del contexto del problema.

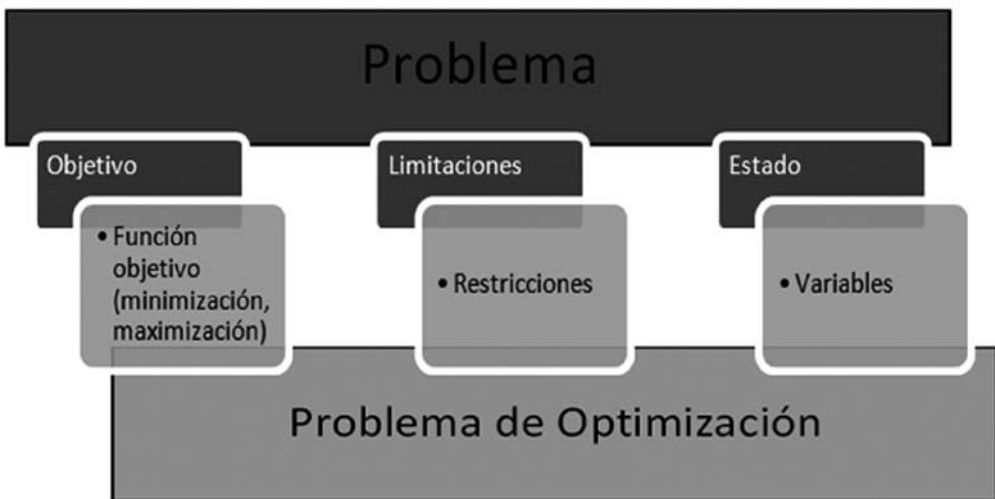


Figura 7. Un problema de optimización.

Optimización por cúmulos de partículas (PSO)

Un cúmulo de partículas es un sistema multiagente conformado por un grupo de agentes conocidos como partículas. Un cúmulo alude a un cardumen de peces o una bandada de pájaros y, por tanto, una partícula se refiere a un pez o a un pájaro [8][11]. Los pájaros y aves se desplazan en su entorno; en analogía,

una partícula explora un *espacio de búsqueda* que no necesita modelarse explícitamente.

En Optimización por cúmulo de partículas (PSO, por su sigla en inglés) el objetivo del cúmulo es explorar el espacio de búsqueda hasta llegar a su objetivo, representado por la *mejor posición en el espacio*; por tanto, cada partícula, o de manera más estricta su posición, se considera una solución candidata [5][8]. Para evaluar la bondad de la posición de una partícula se define una función de adaptación (*fitness*), la cual debe tener la suficiente información para cuantificar qué tan buena es con respecto al objetivo y las restricciones propias del problema. Cada partícula es una solución cooperativa, en el sentido en que su posición es el resultado de intercambios de información con las partículas vecinas del cúmulo durante la ejecución del algoritmo.

Una partícula generalmente almacena la información de su posición actual, su mejor posición hasta el momento y su velocidad [5][8]. El dominio de la posición define el espacio de búsqueda en el cual se desplaza el cúmulo y debe tener un significado de acuerdo con el contexto del problema que se intenta resolver. La velocidad corresponde al movimiento de la partícula y, por ende, debe asegurar que cada nueva posición continúe perteneciendo al dominio del problema. Las restricciones del problema tratado pueden truncar el dominio de la posición de la partícula o servir como mecanismo de penalización en la función de adaptación de la partícula.

El cúmulo realiza su movimiento considerando un modelo social, en el cual cada partícula ignora su experiencia y se ajusta a la experiencia exitosa de sus vecinas [9]; un modelo cognitivo, en el cual cada partícula es un agente aislado y dependiente exclusivamente de su expe-

riencia [8], o una mezcla entre los dos modelos anteriores. De acuerdo con el modelo definido, el cúmulo explora el espacio de búsqueda haciendo un número determinado de movimientos o hasta que se cumple una condición de parada, siendo la mejor partícula la solución al problema de optimización.

Algoritmo PSO General

mientras no se alcance la condición de parada

mientras existan partículas por evaluar en el cúmulo

si valor de adaptación de la posición actual es mayor que el valor de adaptación de la mejor posición de la partícula

 Actualice mejor posición de la partícula

fin si

si valor de adaptación de la mejor posición es mayor que el valor adaptación de la posición del mejor vecino de la partícula

 Actualice mejor posición del vecino

fin si

fin mientras

mientras existan partículas por evaluar en el cúmulo

 Actualice la nueva velocidad de la partícula considerando su velocidad y posición actual, su mejor posición y la posición del mejor vecino

 Actualizar la nueva posición de la partícula considerando la posición actual y la nueva velocidad.

fin mientras

fin mientras

retorne mejor solución encontrada

Tabla 1. Macro algoritmo PSO.

Resultados

Mapas conceptuales

Para enriquecer el marco general de Bioescenarios se perfeccionaron los mapas conceptuales sobre tendencias informáticas bioinspiradas y problemas relacionados con estas tendencias, incluyendo una tipología de problemas complejos. Adicionalmente, se desarrollaron mapas detallados, correspondientes a las tres jerarquías básicas de Bioescenarios: en biología, teoría de enjambres; en informática, inteligencia de enjambres, y en problemas, optimización. Estos mapas brindan un acceso efectivo a los conceptos más relevantes de Bioescenarios y son de gran utilidad para la construcción de los modelos.



Figura 8. Mapa conceptual parcial de optimización por cúmulos de partículas.

Modelos de conocimiento

a) Modelo conceptual

El modelo conceptual que se elaboró tiene dos vistas: una biológica y otra informática. La primera, referente a la teoría de enjambres, presenta a los individuos y sus relaciones estructurales y de comportamiento. La segunda vista, referente a cúmulos de partículas, muestra los conceptos y relaciones que inspiran el algoritmo PSO, y establece una relación directa de estos con el mapa conceptual biológico.

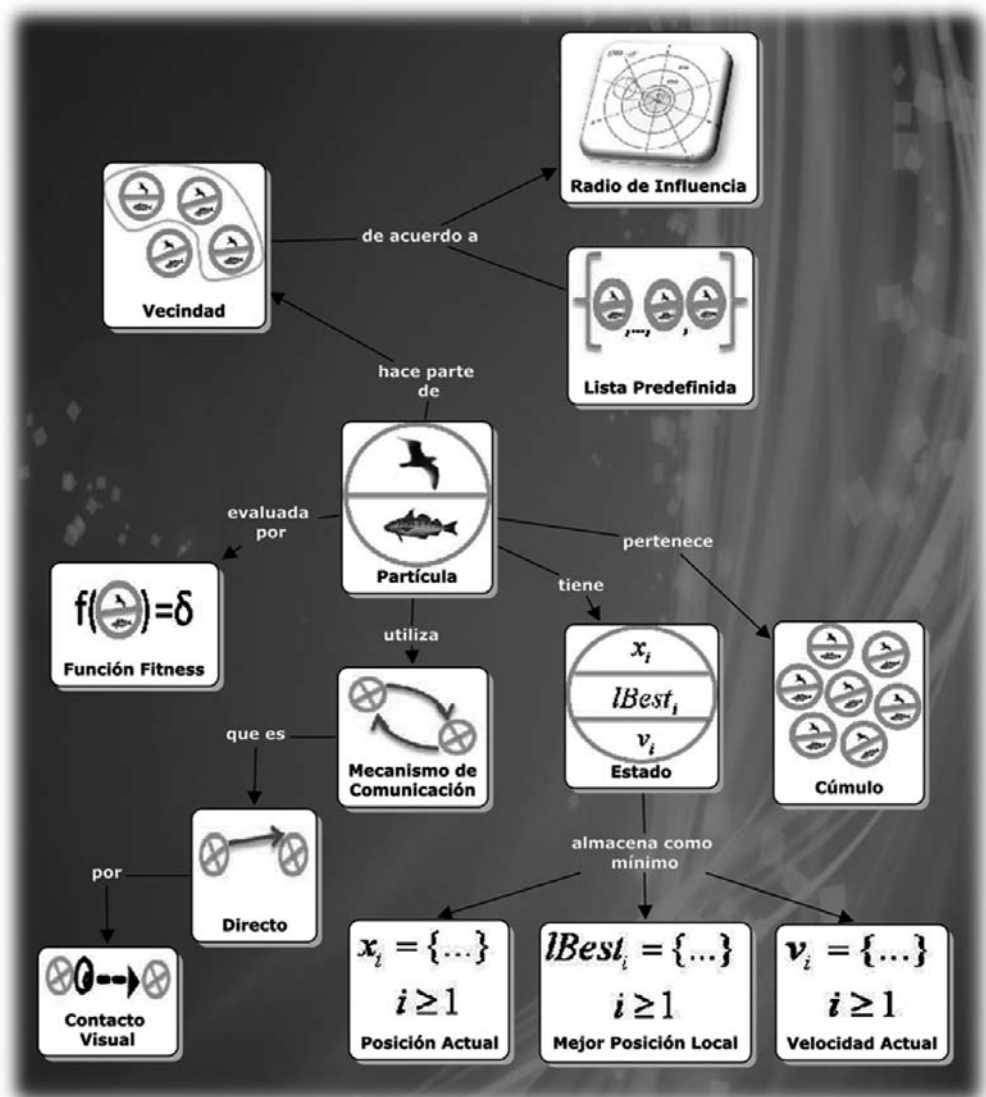


Figura 9. Modelo conceptual de cúmulos de partículas.

b) Modelo metodológico

Este modelo le permite al usuario, dado un problema y siguiendo una secuencia de pasos, abstraer la información necesaria para formularlo, el caso, como un problema de optimización y adaptar su formulación a la técnica de solución del PSO.



Figura 10. Modelo metodológico.

c) Modelo audiovisual

El modelo audiovisual ilustra en detalle el proceso de toma de decisión de una partícula sobre el cambio en el movimiento, considerando cúmulos con modelos de comportamiento diferentes. Este es un nuevo modelo de conocimiento propuesto en Bioescenarios IV, con el objetivo de reforzar comportamientos críticos de los algoritmos bioinspirados, que requieren presentarse de una manera más gráfica y dinámica para su aprendizaje.





Figura 11. Modelo audiovisual (vistas de pantalla mientras se ejecuta la animación.)

Modelos de experimentación

a) Modelo del programador

Este modelo es un *framework* en lenguaje Java que le permite a un programador construir soluciones a problemas de optimización, usando los demás modelos, especialmente el metodológico. El *framework* tiene dos partes: aplicación y presentación.

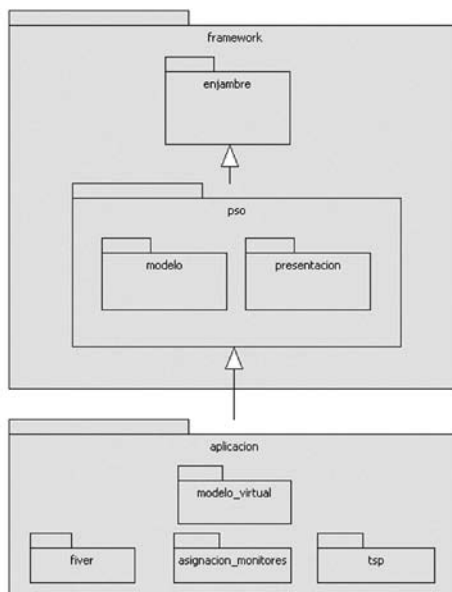


Figura 12. Diagrama de paquetes framework.

El *framework* de aplicación contiene la estructura general de inteligencia de enjambres y la específica de cúmulos de partículas, donde se incluyen las implementaciones estándar del algoritmo PSO. En el *framework* de presentación hay un control para configurar los parámetros del algoritmo PSO y algunas vistas gráficas para conocer el comportamiento en ejecución del algoritmo.

El *framework* de presentación permite al programador sintonizar fácilmente los parámetros de comportamiento del algoritmo PSO, para obtener mejores resultados, usando el control y recibiendo retroalimentación de las vistas gráficas. El control de este *framework* está enlazado directamente con algunos puntos críticos del modelo metodológico.

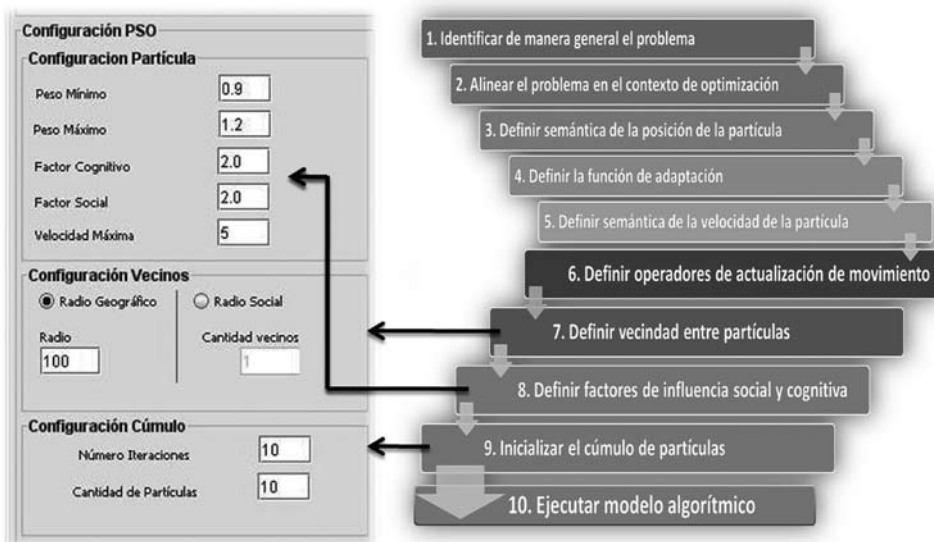


Figura 13. Relación entre framework de presentación y modelo metodológico.

b) Modelo virtual

Este modelo simula el comportamiento de una bandada de pájaros real buscando alimento (**objetivo**), en un espacio con helicópteros (restricciones), usando el algoritmo PSO. El usuario logra apreciar cómo al modificar la posición del alimento, los pájaros convergen hacia este evadiendo los helicópteros; es decir, que al modificar los parámetros propios de PSO se modifica el comportamiento de la bandada. Este modelo se construyó usando el modelo del programador.

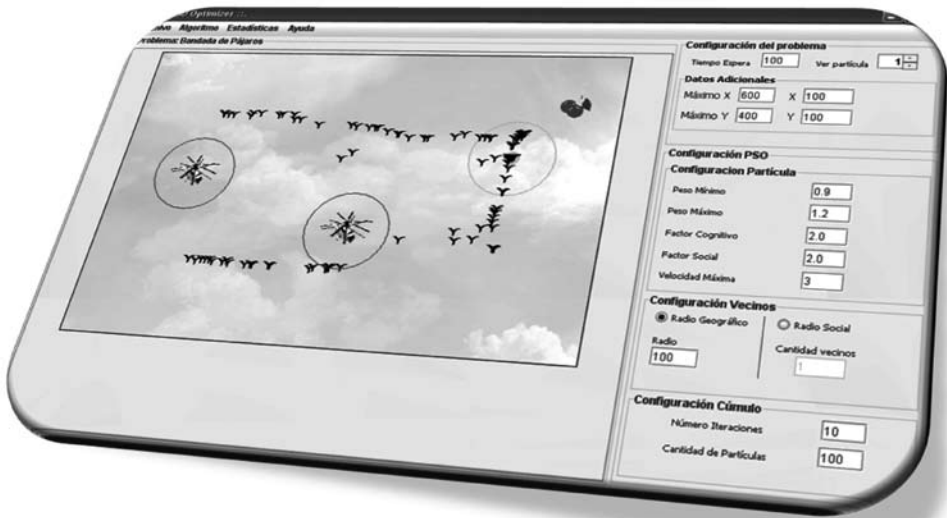


Figura 14. Modelo virtual.

Escenarios

Para validar los modelos se desarrollaron tres escenarios con problemas de tres niveles de dificultad: bajo, medio y alto. Para cada uno de los problemas se aplicó paso a paso el modelo metodológico y, posteriormente, se construyó una aplicación de solución PSO abierta, usando el modelo del programador, lo que permitía visualizar el comportamiento ante distintas configuraciones de parámetros.

Como primer problema, se seleccionó el juego fiver [6], en el cual, a partir de un tablero inicial vacío, se busca llegar a un tablero completo en la menor cantidad de pasos. El segundo problema fue la asignación de proyectos de grado a estudiantes, en el que conociendo las preferencias de estudiantes por proyectos y de directores de proyecto por perfiles de estudiantes, se quiere maximizar la satisfacción tanto de estudiantes como de profesores. El último problema abordado fue un problema TSP, en el cual se desea obtener la trayectoria que describa la ruta óptima para un circuito entre ciudades de una región [15]. Cada uno de estos escenarios permite afianzar de manera gradual los conceptos relacionados con el algoritmo PSO para la solución de problemas.

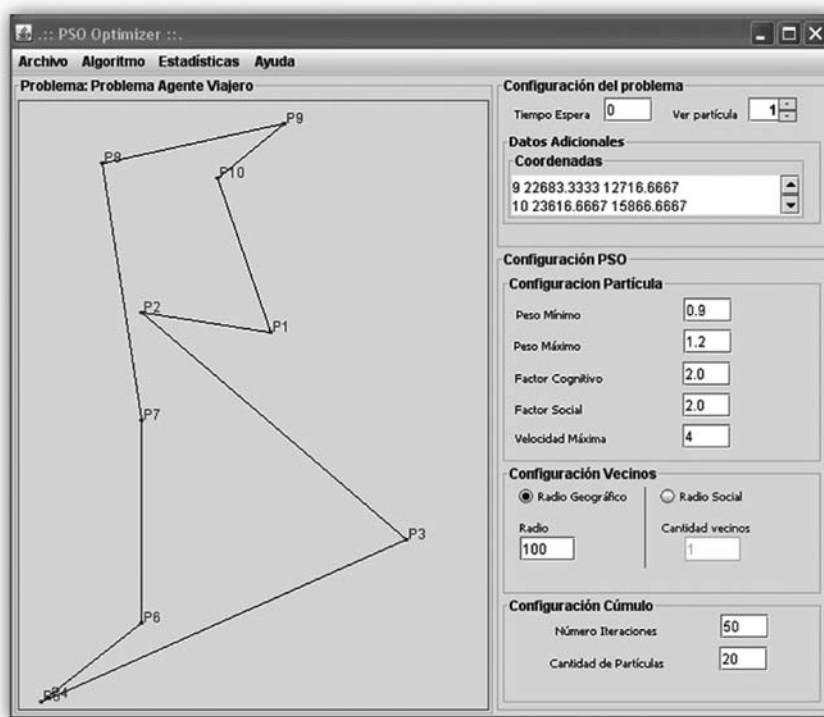


Figura 15. Escenario: problema TSP (uso del modelo del programador para el desarrollo del escenario).

Conclusiones

Una de las grandes ventajas del algoritmo PSO radica en la simplicidad de su implementación, debida en gran medida a los pocos elementos que utiliza en comparación con otros métodos de optimización clásicos. Esto se pudo comprobar con la construcción de los escenarios de aprendizaje.

El uso de mapas conceptuales facilitó el aprendizaje sobre teoría de enjambres, inteligencia de enjambres, optimización general y optimización basada en cúmulos de partículas, confirmando de esta manera el valor del aprendizaje significativo en los proyectos de Bioescenarios.

El uso conjunto de los modelos conceptual, metodológico y del programador (*framework*), enfocados en la solución de problemas, permitió recrear escenarios de aprendizaje. La construcción de estos escenarios fue una de las principales medidas para validar los modelos desarrollados.

Las fuentes de documentación sobre inteligencia de enjambres en Colombia, y en general en los países de habla hispana, son muy limitadas, por lo que se requiere llevar a cabo proyectos de aprendizaje como Bioescenarios, en particular sobre esta tendencia informática.

Referencias

- [1] Azinián, Herminia, Resolución de problemas matemáticos, Visualización y manipulación con computadora, 1ª ed, Buenos Aires: Novedades Educativas, 1997.
- [2] Caimary Loza, M. A. (2008). Sincronización en Cardúmenes Estacionarios de Pomadasys Incisus [En línea].
<http://www.webs.ulpgc.es/etologia/revista2008/AngelesCaimariLoza.pdf>
- [3] Chaparro, A. & Albornoz, J. (2002). Escenarios 2002. Bogotá, Colombia: s.n.
- [4] Definicion.de, Definición de problema, ONLINE, visitado el 3 de febrero de 2009, <http://definicion.de/problema/>
- [5] García Nieto, J. M., Algoritmos basados en cúmulos de partículas para la resolución de problemas complejos. [En Línea] http://neo.lcc.uma.es/staff/jmgn/doc/Memoria_PFC_JMGN.pdf
- [6] Herzog, D. Kirkland, B., “Fiver”, Mazeworks. [En línea], <http://www.mazeworks.com/fiver/>, 2002.
- [7] Jiménez, D. & Peñuela, D., (2007). Bioescenarios: Colonia de Hormigas. Bogotá, D.C.: Escuela Colombiana de Ingeniería Julio Garavito.
- [8] Kennedy, J., Eberhart, R., Shi, Y. (2001). Swarm Intelligence. San Francisco, CA.
- [9] Miller, P. (2007). Swarm Theory. [En Línea] <http://ngm.nationalgeographic.com/2007/07/swarms/miller-text>.
- [10] Mora, J. & Rodríguez, S. (2008). Bioescenarios III. Sistemas Inmunológicos Artificiales. Bogotá, D.C.: Escuela Colombiana de Ingeniería Julio Garavito.
- [11] Muñoz, A. M., López, J. A. & Caicedo, E. F., Inteligencia de enjambres: sociedades para la solución de problemas (una revisión). [En línea] http://www.revistaingenieria.unal.edu.co/Resumenes/28_2/15_779.pdf
- [12] Rojas, D., (2006). Bioescenarios: Algoritmos Genéticos. Bogotá, D.C.: Escuela Colombiana de Ingeniería Julio Garavito.
- [13] Roth, M. & Wicker, S., (2003). Termite: Emergent Ad-Hoc Networking. [En línea] <http://www.deutsche-telekom-laboratories.de/~rothmart/docs/SIDM2005.pdf>

- [14] Vries, H., Biesmeijer, J., (1998) Modelling collective foraging by means of individual behaviour rules in honey-bees.
- [15] WI29 - Western Sahara, The traveling salesman problem,[En línea] <http://www.tsp.gatech.edu/world/wipoints.html>
- [16] Wikipedia. (2008). Sociobiología. Wikipedia, la enciclopedia libre. [En línea], <http://es.wikipedia.org/wiki/Sociobiología>.
- [17] Wikipedia. (2008). Swarm. Wikipedia, la enciclopedia libre. [En línea], <http://en.wikipedia.org/wiki/Swarm>.

Carlos Eduardo Casallas Fonseca. *Estudiante de último semestre de Ingeniería de Sistemas de la Escuela Colombiana de Ingeniería Julio Garavito. Habilidades e interés en ingeniería de software, bases de datos y administración de servicios en las organizaciones. Excelente en trabajo en equipo y desarrollo de relaciones interpersonales.*

Steven Vargas Aristizábal. *Estudiante de de último semestre de Ingeniería de Sistemas de la Escuela Colombiana de Ingeniería Julio Garavito. Experiencia en desarrollo de software y seguridad informática. Sociable, íntegro, activo, independiente, eficiente, y siempre abierto a aprender nuevas herramientas y tecnologías.*

María Irma Díaz Rozo. *Directora del proyecto. Ingeniera de Sistemas y Computación de la Universidad de los Andes y máster en Ingeniería de Software de la Universidad Politécnica de Madrid. Profesora de la Escuela Colombiana de Ingeniería Julio Garavito, adscrita al Centro de Estudios de Ingeniería de Software de la Decanatura de Ingeniería de Sistemas. Experiencia en las áreas de ingeniería de software e ingeniería de conocimiento.*