

MODELO DE UN SISTEMA MANEJADOR DE COMUNIDADES AUTORGANIZATIVO PARA UN SISTEMA OPERATIVO WEB MULTIAGENTE

*NIRIASKA PEROZO, DOCARLY ROMERO

Resumen

Este artículo describe la simulación y evaluación funcional de un Sistema Manejador de Comunidades (SMC) para un Sistema Operativo WEB Multiagente (SOWM), siguiendo un diseño de referencia y los conceptos asociados al área de Inteligencia de Enjambre (“Swarm Intelligence”) evaluando así, la autorganización y emergencia en la gestión de comunidades que requiere este tipo de sistemas. Para ello, se ha empleado un algoritmo de agrupamiento basado en el comportamiento emergente de las hormigas, para recoger y depositar cadáveres y formar pilas de ellos. Siguiendo un esquema de gestión de comunidades dinámica, autorganizativa y emergente, se realizó la búsqueda de servicios y recursos a nivel Intra e Inter en el SMC de acuerdo con los requerimientos y objetivos del SOWM y sus subsistemas.

Palabras Clave: Sistemas Multiagente, Comunidades Virtuales, Autorganización, Inteligencia Colectiva, Sistema Operativo Web.

Abstract

A model of the simulation and functional evaluation for a Community Manager System (CMS), on a Multiagent Web Operating System (MWOS) is presented. Following a reference design and concepts associated with swarm intelligence, the evaluation focused on auto-organization and emergent community management requirements using a gathering algorithm based on the behavioral emergent patterns of ant colonies. Both, Intra and Inter-wise CMS resources were searched using a dynamic auto-organizational management approach according to CMS requirements and the SWOM’s objectives.

Keywords: Multiagent systems, virtual communities, auto-organization, collective intelligence, web operating system

1. Introducción

El Sistema Operativo Web Multiagente (SOWM) surge como una solución para integrar los numerosos recursos disponibles sobre la INTERNET, suministrando así una plataforma computacional que permita a los usuarios compartir recursos y resolver problemas de heterogeneidad y adaptabilidad dinámica. Para satisfacer estas necesidades se propone en [1] una arquitectura conformada por cuatro (4) entidades o subsistemas que se interrelacionan entre sí y que tienen una identidad propia. Una de esas entidades o subsistemas del SOWM es el Sistema Manejador de Comunidades (SMC).

El SMC propuesto plantea una solución adaptativa que pretende mejorar la eficiencia y el desempeño del SOWM. El SMC establece un mecanismo de agrupamiento dinámico de nodos, es decir, crea comunidades virtualmente organizadas que exhiben afinidades funcionales que favorece la búsqueda de recursos y servicios en la Web, en otras palabras, se buscaría por comunidades y no, de nodo en nodo. El SMC es autónomo en cuanto a la toma de decisiones, actualización y agrupamiento de los nodos involucrados

* *Unidad de Investigación en Inteligencia Artificial, Universidad Centroccidental “Lisandro Alvarado, Barquisimeto, Venezuela”, Barquisimeto, Lara 3001*

en las comunidades que forman parte de él, por ello cada comunidad es capaz de autorganizarse y adaptarse a su entorno, emergiendo nuevas comunidades. Así, se simula el comportamiento de este subsistema realizando, verificando y evaluando el diseño propuesto en [14] desde el punto de vista de implementación.

2. Marco Teórico

2.1. Antecedentes

Por ser el modelo propuesto autorganizativo, es interesante mencionar algunas aplicaciones descentralizadas y autorganizativas que operan en base a sus interacciones locales. Estos sistemas son particularmente robustos porque se adaptan a los cambios del ambiente. En algunos casos los sistemas autorganizativos están complementados con un comportamiento emergente en el sentido que los componentes individuales del sistema ejecutan tareas simples que favorecen la emergencia de tareas complejas mediante sus interacciones. Las propiedades de autorganización y emergencia han sido aplicadas en sistemas de control de manufactura [9], en sistemas de redes sensoriales para el control de tráfico [7], ha inspirado e influenciado desarrollos algorítmicos en optimización combinatoria, enrutamiento en redes de comunicaciones, análisis de datos exploratorios, partición de grafos, entre otros [3], [6], [2], [13].

En el caso de los sistemas naturales autorganizados tenemos como ejemplo las sociedades de insectos tales como las hormigas, las termitas o las abejas, la comunicación entre ellos ocurre mediante un mecanismo llamado “Stigmergy” (comunicación indirecta vía el entorno, Grassé 1959) [8], [9]. El comportamiento social de los humanos también es autorganizativo, dado que el hombre comienza a realizar una tarea con información propia y mediante interacciones locales directas o indirectas producen sociedades complejas que manifiestan un comportamiento global emergente; además en el ramo de la biología esta propiedad se pone de manifiesto en el sistema inmune y en la regeneración de células del cerebro garantizando la emergencia y autorganización de los mismos [7], [11].

Las técnicas de agrupación o clustering también ofrecen un aporte importante para el modelo propuesto, en este sentido tenemos el trabajo: “Towards Improving Clustering Ants: An Adaptive Ant Clustering Algorithm” [18] que consiste en incorporar propiedades adaptativas al algoritmo estándar de agrupación “Standard Ant Clustering Algorithm” (SACA) generalizado por Lummer y Faieta en base a un esquema de visión progresiva y heurística de feromona, este algoritmo propuesto denominado “Adaptive Ant Clustering Algorithm (A2CA)” se propuso evaluar en un conjunto de datos bioinformáticos las propiedades de ejecución y convergencia garantizando una estabilización luego de un número de pasos, quedando demostrado que con estas modificaciones, el A2CA es robusto en cuanto a que se encuentra el número correcto de grupos (“clusters”), existen bajas variaciones de los resultados y se estabiliza después de un número fijo de iteraciones definidas automáticamente por el algoritmo. Otro trabajo en esta área es: “Ant-Based Clustering and Topographic Mapping” [10] donde el agrupamiento y el ordenamiento basado en colonias de hormigas y aplicado en el contexto de minería de datos (“data-mining”), este trabajo incorporó una mejora a la versión básica del algoritmo de agrupación de hormigas (“Ant Clustering”, AC) llamada “Adaptive Time-dependent Transporter Ants” (ATTA), incorporando adaptatividad y heterogeneidad, a las actividades de transporte dependiente del tiempo, y un método de agrupación que transforma el espacio, es por esto que ATTA fue objeto de rigurosas evaluaciones de funciones y colecciones de datos para verificar que el agrupamiento y ordenamiento basado en colonias de hormigas identifica automáticamente el número de grupos en una colección de datos y produce soluciones de buena calidad. Además, es importante señalar según [17] que el clustering o agrupamiento tiene múltiples aplicaciones dentro de la ciencia de la computación, como compresión de imágenes y voz digitalizadas; en la recuperación de información; en minería de datos; en la segmentación de imágenes médicas; en la clasificación de componentes de software y en el ambiente de la Web, entre otros.

2.2. Aspectos Generales

Un sistema multiagente (SMA) necesita operar en ambientes heterogéneos y dinámicos, capaces de manejar frecuentemente los cambios requeridos, para ello deben ser flexibles, robustos y adaptables a las circunstancias [7]. Actualmente muchas aplicaciones que requieren cierto grado de autorganización y un comportamiento emergente, son diseñadas bajo el enfoque de un sistema multiagente que utiliza mecanismos de comunicación y coordinación. En [3] se define la inteligencia de enjambre en términos de dos conceptos: Autorganización y Stigmergy. Así tenemos según [4] que Autorganización, es un proceso donde el patrón (arreglo organizado particular de objetos en espacio o tiempo) en el nivel global de un sistema emerge solamente de las interacciones numerosas entre los componentes de nivel inferior del sistema. Por otra parte, las reglas que especifican las interacciones entre los componentes del sistema se ejecutan usando solamente información local, sin referencia al patrón global.

La esencia de la autorganización es que un sistema adquiere una estructura espacial, temporal o funcional sin interferencia específica del exterior. La característica principal de estos sistemas autorganizativos es su capacidad de lograr tareas colectivas complejas con comportamientos individuales simples, sin un control central o estructura jerárquica. Este comportamiento usualmente emerge de todas las interacciones que se producen en el sistema [15], [16].

En el caso de “Stigmergy”, es un mecanismo de retroalimentación usado para reflejar las interacciones indirectas entre los agentes y el entorno. Permite alcanzar formas emergentes de comportamiento coordinado a nivel de la sociedad [8]. Además, fue introducido por el estudio de las sociedades biológicas de insectos y sus conceptos de interacción e intercambio de información. “Stigmergy” describe una forma de interacción e intercambio de información asíncrona entre agentes mediado por un ambiente activo es decir, un ambiente que es modificado y a su vez, capaz de modificar el comportamiento de los agentes involucrados.

Los beneficios de la inteligencia de enjambre son ilustrados por una variedad de aplicaciones descritas en [3], [4], [12]. Estas aplicaciones incluyen el enrutamiento del tráfico de la red en sistemas de telecomunicaciones, problemas de transporte multi-robot, análisis y clasificación de datos y búsquedas de soluciones aproximadas a problemas complejos computacionalmente, entre otras. Por otro lado, el agrupamiento o clustering consiste en dividir un conjunto de objetos en un número de clusters o grupos [5]. La motivación de agrupar un conjunto de datos está en encontrar una estructura inherente de datos que exhiban un alto grado de similitud para poder agruparse; estos grupos deben tener un alto grado de cohesión (similitud entre los elementos de los nodos del mismo grupo), y bajo acoplamiento (los grupos deben ser lo más diferentes entre sí) con respecto a otros grupos [17].

3. Descripción de la Propuesta

De acuerdo al diseño del SMC propuesto en [14], se tienen dos (2) componentes: el manejador de comunidades (responsable de la gestión de comunidades) y el coordinador de búsqueda (encargado de la búsqueda de recursos y servicios a nivel de comunidades). A continuación se describe el algoritmo utilizado y las políticas establecidas para la implementación del SMC con respecto a los componentes mencionados.

3.1. Algoritmo Utilizado

Recientemente se han propuesto algunos algoritmos de agrupamiento basados en el comportamiento de algunas sociedades de insectos, tal es el caso del Algoritmo de Agrupamiento basado en el Comportamiento emergente de las hormigas al recoger y depositar cadáveres para formar pilas de ellos (“Ant Clustering (AC)”); este modelo estándar del algoritmo de agrupamiento AC fue generalizado por Lummer y Faieta para aplicarlo al análisis exploratorio de datos [3]. El Algoritmo de Agrupamiento utilizado es presentado en la tabla . En el algoritmo de agrupamiento basado en el comportamiento de las hormigas (“Ant

Clustering”) [3], la probabilidad de recoger un dato i viene dada por:

$$P_p(i) = \left(\frac{k_p}{k_p + f(i)} \right)^2 \quad (1)$$

y la probabilidad de depositar por:

$$P_d(i) = \begin{cases} 2f(i), & \text{si } f(i) < k_d \\ 0, & \text{Caso contrario} \end{cases}$$

Donde k_p y k_d son constantes iguales a 0.1 y 0.15 respectivamente y $f(i)$ es la medida de densidad de similitud del dato i en una localidad en particular r y se define así:

$$f(i) = \begin{cases} \frac{1}{S^2} \sum_{\text{vecindad}} (1 - \frac{d(i,j)}{\alpha}), & \text{Si } f > 0 \\ 0 & \text{Si } f \leq 0 \end{cases}$$

Donde S^2 es el área de la región de percepción de un agente con centro en su localidad, α es un factor de escalamiento de la medida de similitud, y $d(i, j)$ es la medida de similitud usada en este caso (distancia euclidiana) dada por:

$$d(i, j) = \sqrt{\sum_{h=1}^D (x_i - y_j)^2} \quad (2)$$

La distancia Euclidiana en los elementos de vectores binarios, representa la presencia del atributo como 1 y la ausencia como 0.

3.2. Políticas Establecidas en el Manejador de Comunidades del SMC

- La incorporación de un nodo a una comunidad del SMC, a través del Manejador de Comunidades, se realiza siguiendo el diagrama presentado en la figura .

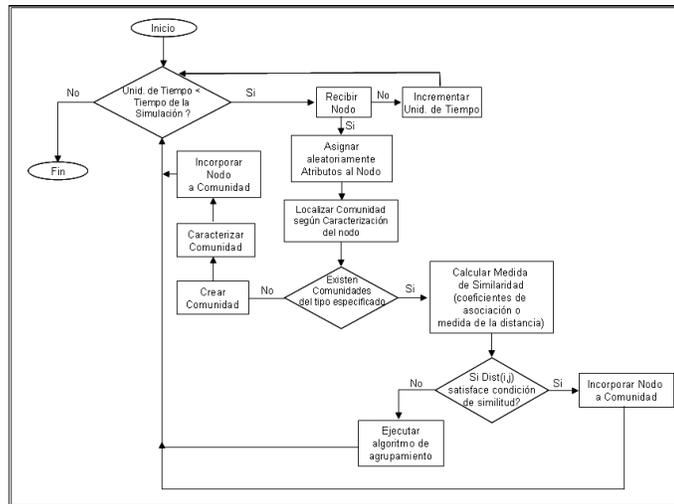


Figura 1: Diagrama de tareas involucradas en la incorporación de un nodo a través del Manejador de Comunidades

- La eliminación de un nodo del SMC, a través del Manejador de Comunidades se realiza siguiendo el diagrama presentado en la figura 2.

	Inicio del Algoritmo
inicialización	<pre> For Cada Elemento Oi Do Colocar Oi aleatoriamente sobre el Grid End For For Todos los Agentes Do Colocar el agente en un lugar seleccionado aleatoriamente en el grid bidimensional End For </pre>
Ciclo Principal	<pre> For t=1 to tmaxpasos Do For Todos los agentes Do If (Agente_Desocupado) and (Lugar_Ocupado_oi) Entonces Calcular_F(oi) and Pp(oi) Obtener aleatoriamente un valor R (Entre 0 y 1) If R<=Pp(oi) then Recoger_oi Else If (Agente_Cargando_Oi) and (Lugar_Vacio) Entonces Calcular_F(oi) and Pd(oi) Obtener aleatoriamente un valor R (Entre 0 y 1) If R<=Pd(Oi) then Depositar_oi End if Moverse aleatoriamente en el grid a un lugar no ocupado por otro agente End For /*Agentes*/ End For /*Tiempo*/ Imprimir_Localizacin_Elementos </pre>
	Fin del Algoritmo

Cuadro 1: Algoritmo Utilizado para el Agrupamiento

- La transferencia de un nodo es otra de las tareas a cargo del Manejador de Comunidades, la transferencia de un nodo es virtual entre comunidades y se apoya en las tareas de *eliminar nodo* (para quitar el registro del nodo de la comunidad origen) y de *localizar comunidad* (para incorporar el nodo a otra comunidad), siguiendo las políticas de agrupamiento de nodos establecidas

3.3. Políticas establecidas en el Coordinador de Búsqueda del SMC

El coordinador de búsquedas del SMC se encarga de recibir peticiones de búsqueda de servicios de un SMR o de otro SMC y procesarlas realizando búsquedas internas (Nivel Intra, dentro de las comunidades a las que pertenece el nodo) y/o externas (Nivel Inter, en las comunidades que no pertenece el nodo), dependiendo del éxito o no de la búsqueda y siguiendo el diagrama presentado en la figura .

4. Ejecución y Evaluación de la Propuesta

La implementación de esta propuesta se realizó usando el simulador Starlogo, que es una herramienta libre y con un ambiente de modelado programable para explorar el procesamiento de sistemas descentralizado. Starlogo provee un ambiente de ejecución para correr miles de agentes (o tortugas como son llamadas en el entorno) en paralelo y facilita un comando observador central para configurar e influenciar

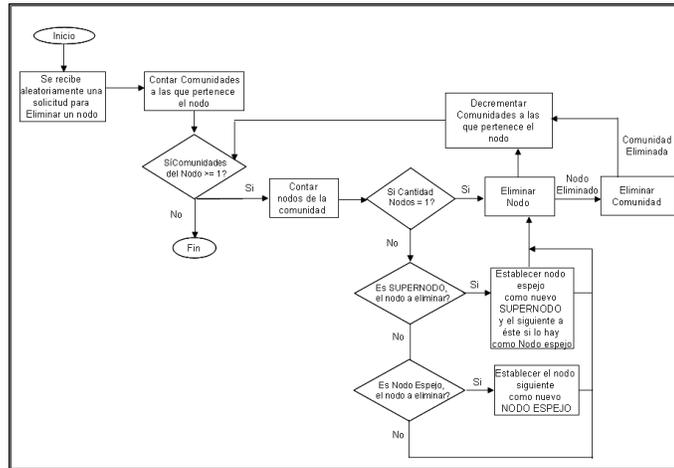


Figura 2: Diagrama de la tarea Eliminar Nodo a través del Manejador de Comunidades

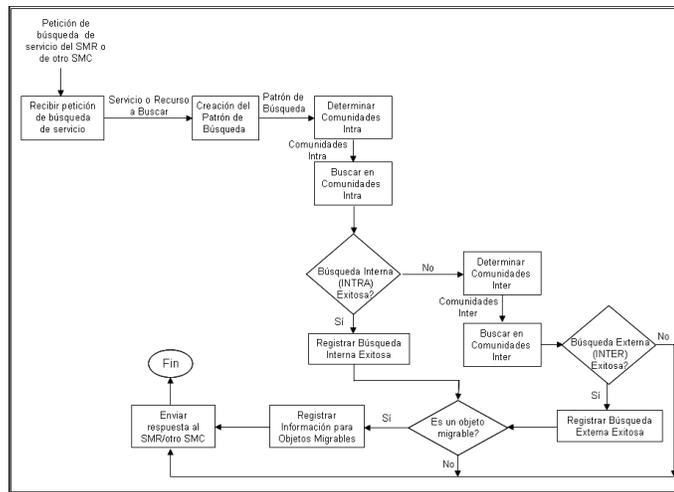


Figura 3: Diagrama de las Tareas de Localización del CB

el sistema. En la figura 4 se muestra la interfaz gráfica elaborada para la simulación del SMC y sobre la que se utilizó una configuración inicial (ver tabla 2) que servirá de escenario inicial para realizar las evaluaciones del modelo propuesto.

Para efectos de la simulación [20], se crean 3 tipos de comunidades: Software, Hardware y Ubicación Geográfica para las que se ejecutó por separado la misma configuración inicial con la finalidad de obtener resultados más detallados.

4.1. Resultados Obtenidos

4.1.1. Comunidades Tipo Hardware

En la tabla 3 se muestra el comportamiento de la gestión de comunidades de tipo hardware con los datos de la configuración inicial.

El número de comunidades de hardware que se crearon luego de la ejecución de la simulación es

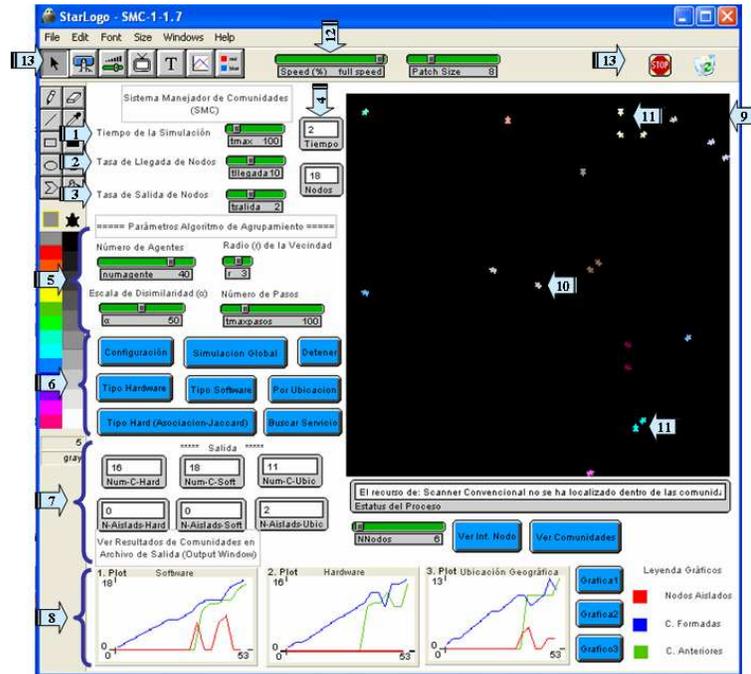


Figura 4: Interfaz Gráfica de la Simulación del SMCT

Parámetro	Valor
Tiempo de la Simulación (tmax)	100 unidades de tiempo
Tasa de Llegada (tllegada)	10 nodos/unidad de tiempo
Tasa de Salida (tsalida)	2nodos/unidad de tiempo
Número de Agentes (numagente)	40
Radio de la Vecindad de cada agente (r)	3
Escala de Disimilaridad (α)	50
Número de Pasos o Iteraciones (tmaxpasos)	100

Cuadro 2: Configuración Inicial utilizada en la Simulación del SMC

Elemento		Resultados	
Unidades de Tiempo Transscurridas	40	70	100
Nodos Recibidos	399	699	999
Comunidades de Tipo Hardware creadas	80	80	80

Cuadro 3: Resultados Obtenidos con la Configuración Inicial en Comunidades de Tipo: Hardware

bastante elevado y el mismo, es producto de la cantidad de nodos aislados que quedaban luego de cada iteración del algoritmo de agrupamiento, ya que los mismos se convierten en comunidades con un sólo

nodo, esto se debe a que el nodo es recogido por un agente para encontrarle ubicación y no logra depositarlo en ninguna comunidad existente, o el nodo, no es seleccionado durante la simulación antes de finalizar la ejecución de la misma.

4.1.2. Comunidades Tipo Software

Los resultados obtenidos de la gestión de comunidades de tipo software con los datos de la configuración inicial son mostrados en la tabla 4.

Elemento		Resultados	
Unidades de Tiempo Transcurridas	40	60	100
Nodos Recibidos	399	591	999
Comunidades de Tipo Software creadas	80	85	85

Cuadro 4: Resultados Obtenidos con la Configuración Inicial en Comunidades de Tipo: Software

El comportamiento de los resultados de las comunidades de tipo software es similar al obtenido con las comunidades de tipo hardware, puesto que ambos expresan la presencia o ausencia de atributos en vectores binarios.

4.1.3. Comunidades tipo Ubicación Geográfica

La administración de comunidades de tipo ubicación geográfica es más dinámica en la ejecución de la simulación puesto que se nota el incremento y decremento de comunidades a medida que llegan los nodos, esto no se nota en los otros tipos de comunidades, ya que el cálculo de la medida de similitud basada en la distancia favorece la creación de este tipo de comunidad debido a que se calcula la distancia entre las coordenadas que dan ubicación a los nodos y no en base a los atributos de un vector binario. En la tabla 5 se muestra los resultados del comportamiento de las comunidades en diferentes unidades de tiempo.

Elemento		Resultados	
Unidades de Tiempo Transcurridas	40	70	100
Nodos Recibidos	399	699	999
Comunidades de Tipo Ubicación Geográfica Creadas	35	40	47

Cuadro 5: Resultados Obtenidos con la Configuración Inicial en Comunidades de Tipo: Ubicación Geográfica

Una vez analizados los resultados obtenidos de la simulación bajo una configuración de parámetros establecida como inicial, se pudo observar Comunidades de Tipo Ubicación Geográfica Creadas que se obtuvieron resultados satisfactorios de acuerdo con los objetivos planteados, pero es necesario comparar estos resultados con los que resulten luego de evaluar algunos parámetros claves en el algoritmo de agrupamiento, y así poder detectar su sensibilidad ante diversos cambios intencionados. A continuación se muestra los resultados obtenidos al ajustar los parámetros: número de agentes (“numagente”), radio de la vecindad (“r”), número de iteraciones (“tmaxpasos”) y cambio de la medida de similitud.

4.2. Evaluación y Calibración de Parámetros

4.2.1. Ajuste del Parámetro “numagente” (Número de Agentes)

El número de agentes es uno de los principales parámetros que intervienen en el algoritmo de agrupamiento implementado, debido a que un agente es el encargado de recoger o depositar un determinado nodo en una posición específica del gris, siempre y cuando satisfaga ciertas condiciones establecidas, ver resultados en la tabla 6.

Numagente	10	40	70
Unidad de Tiempo	30	100	100
Nodos Recibidos	296	999	999
Nro. Comunidades de Tipo Hardware creadas	+100	91	80

Cuadro 6: Comportamiento de Comunidades de Tipo Hardware al Ajustar el Parámetro “numagente”

4.2.2. Ajuste del Parámetro “r” (radio de la vecindad)

Para la evaluación de este parámetro se tomaron las comunidades por ubicación geográfica para notar su comportamiento cuando el radio de la vecindad de los nodos se incrementa o decrementa, ver resultados en la tabla 7.

“r”	1	3	5
Unidad de Tiempo	100	100	100
Nodos Recibidos	999	999	999
Nro. Comunidades de Tipo Ubicación Geográfica creadas	42	47	35

Cuadro 7: Comportamiento de Comunidades de Tipo Hardware al Ajustar el Parámetro “numagente”

4.2.3. Cambiando la Medida de Similitud

Una de las evaluaciones clave es el cambio de la medida de similitud, que se emplea para asociar dinámicamente un nodo a una comunidad de acuerdo a grado de similaridad y en el cálculo de la función de densidad, para determinar la probabilidad de recoger o depositar un nodo en el algoritmo de agrupamiento. En el desarrollo de la simulación se tomó la distancia euclidiana como medida de similitud, pero los coeficientes de asociación son también utilizados como medida de similitud [19] en el caso de utilizar vectores binarios, están basados en la medición del número de atributos coincidentes de un vector en relación al otro, debido a que se resalta la ausencia o presencia de un atributo, ver resultados obtenidos en la tabla 8.

Medida de Similitud	Unidad de Tiempo	Cantidad de Nodos	Comunidades de Hardware formadas
Distancia Euclidiana	100	999	66
Coefficiente de Asociación de Jaccard	100	999	10

Cuadro 8: Comportamiento de las Comunidades de Tipo Hardware al cambiar la medida de similitud

4.2.4. Ajuste del Parámetro “tmaxpasos” (número de iteraciones del algoritmo de agrupamiento)

La evaluación de este parámetro se realizó en base a las comunidades por ubicación geográfica tomando cuatro valores (100, 200, 300 y 400), ver los resultados obtenidos en la tabla 9.

5. Conclusiones

Al modelar el SMC para el SOWM se determinó después de un número de evaluaciones que el algoritmo de análisis exploratorio de datos propuesto por Lummer y Faieta [10], favorece la gestión de

t-maxpasos	100	200	300	400
Unidad de Tiempo	100	100	100	100
Nodos Recibidos	999	999	999	999
Comunidades por Ubicación Geográfica Creadas	47	26	14	10

Cuadro 9: Comportamiento de las Comunidades por Ubicación Geográfica al Ajustar el Parámetro “tmax-pasos”

comunidades de manera dinámica, autoorganizada y emergente de acuerdo con el diseño propuesto. Además, se estableció los valores ideales (calibración) para los parámetros claves del algoritmo utilizado.

Con respecto a las medidas de similaridad en el agrupamiento de ítem u objetos, representan un aspecto clave, por tal razón se evaluaron dos (2) tipos de medidas; los coeficientes de asociación y la distancia euclidiana, determinando que los coeficientes de asociación específicamente el de Jaccard, reduce la creación de comunidades o agrupamiento de los nodos en comunidades, cuando se plasman los atributos en vectores binarios, tal es el caso de las comunidades de Hardware y Software. Por otro lado, la medida de similaridad basada en la distancia euclidiana ofrece mejores resultados en las comunidades por ubicación geográfica, debido a que se mide la similaridad en base a la distancia real que existe entre nodos.

En las evaluaciones realizadas a los diversos parámetros que participan en el algoritmo se determinó, que a mayor número de agentes interactuando en el proceso de ejecución del algoritmo de agrupamiento utilizado menor es el número de comunidades o grupos que se forman; que a mayor número de iteraciones (en el intervalo [300, 400]) se logra una mejor convergencia y estabilización del sistema.

En el ajuste del parámetro radio de la vecindad, no varió significativamente el número de comunidades que se forman, pero si fue interesante los resultados obtenidos en relación al nivel de cohesión y acoplamiento de las comunidades, mientras menor sea el radio, mayor es el grado de similitud que existe entre los nodos de una comunidad y mayores las diferencias entre las distintas comunidades. Debido a esta situación se recomienda establecer el valor de este parámetro en $r = 1$.

En cuanto a las tareas del coordinador de búsqueda del SMC, se realizó el proceso en tres (3) escenarios distintos de acuerdo a los requerimientos del SMC y de los otros subsistemas. Como trabajo futuro se podría ensayar con otros tipos de medidas de similaridad, distintas a las medidas implementadas para comparar los resultados en cuanto al número de comunidades formada. Se podría implementar el algoritmo de agrupamiento con agentes heterogéneos que tenga la capacidad de memoria a corto plazo, de tal manera, que ellos puedan recordar cuales fueron los últimos nodos visitados para no volver a visitarlos en un tiempo determinado, ya que este comportamiento podría reducir el número de comunidades formadas y acelerar el tiempo de convergencia del algoritmo. Finalmente, se podría variar dinámica e incrementalmente el parámetro radio de vecindad en el intervalo [1, 5], para acelerar la disolución de pequeñas comunidades preliminares.

Referencias

- [1] Aguilar, J., Perozo, N., Ferrer, E., Vizcarrondo, J. 2005, “*Architecture of a Web Operating System based on Multiagent Systems*”, Publicado en Lecture Notes in Artificial Intelligence, Springer-Verlag, Vol. 3681, pp. 700-706
- [2] Aguilar, J. 2004. “*La Inteligencia Colectiva en la Resolución de Problemas de Optimización Combinatoria Dinámicos*”, Proceeding of the XIV Simposio Internacional de Métodos Matemáticos Aplicados a las Ciencias, pp. 21-22, San José, Costa Rica, Febrero 2004.
- [3] Bonabeau E., Dorigo M., Theraulaz G. “*Swarm Intelligence: from Natural to Artificial Systems*”, Oxford University Press, 1999.

- [4] Camazine, S., Deneubourg, J. Franks, N., Sneyd, J., Theraulaz, G. and Bonabeau, E. “*Self-Organization in Biological Systems*”, Princeton University Press, Princeton, NJ, USA, 2001.
- [5] Cui, X., Gao, J., Potok, T. 2006. “*A flocking based algorithm for document clustering analysis*”, Journal of Systems Architecture. USA.
- [6] Di Caro, G., Dorigo, M. 1998. “*AntNet: Distributed Stigmergic Control for Communication Networks. Journal of Artificial*”, Intelligence Research 9.
- [7] Di Marzo, G., Pierre, M., Karageorgos, A. 2006. “*Self-Organization and Emergence in MAS: An Overview*”, Informática 30 (2006): 45-54.
- [8] Grassé P.P. 1959. “*La reconstruction du nid et les coordinations inter-individuelles chez Bellicositermes natalensis et Cubitermes sp. La thorie de la Stigmergie : Essai d’interprétation du comportement des termites constructeurs*”, Insectes Sociaux, 6, p.p. 41-80. 3.
- [9] Hadeli, P. et al. 2003. “*Self-Organising in Multi-Agent Coordination and Control Using Stigmergy*”, K.U. Leuven Clestijnenlaan.
- [10] Handl, J., Knowles, J., Dorigo, M. 2006. “*Ant-Based Clustering and Topographic Mapping*”, Artificial Life Volume 12. pp 35-61. Massachusetts.
- [11] Holland, O., Chris, M. 1999. “*Stigmergy, Self-Organization, and Sorting in Collective Robotic*”, Instituto de tecnología de Massachussets (Artificial life r: 173-202).
- [12] Izquierdo Torres, Eduardo. 2004. “*Collective Intelligence in Multi-Agent Robotics: Stigmergy, Self-Organization and Evolution*”, University of Sussex.
- [13] Montes de Oca M. et al. 2005. “*Efectos de la comunicación directa entre agentes en los algoritmos de agrupación de clases basados en el comportamiento de insectos sociales*”, Revista Iberoamericana de Inteligencia Artificial Nro 25.
- [14] Perozo N., Aguilar J. “*Design Of A Community Manager System For A Multiagent Web Operating System*”, Publicado en Ciencia y Tecnología Vol 2 N 1 Pg 14-22, Julio (2008).
- [15] Rennard, J.P. 2003. “*Social-Insect and self-organization*”, URL: <http://www.rennard.org/alife>. (Consulta Sep-tiembre 27, 2006).
- [16] Rodriguez, M.A.. 2005. “*A Self-Organizing Collective-Intelligence approach to the Peer-Reviewed Publication Process*”, University of California, Santa Cruz (USA).
- [17] Vicente, E., Tupia, M., Rivera, L. 2006. “*GraspKM en la Recuperación de la Estructura de Software*”.
- [18] Vizine, A., De Castro, L., Hrusehka, R., Gudwin, R. 2004. “*Towards Improving Clustering Ant: A Adaptive Ant Clustering Algorithm*”. Informática 29 (2005) pp 143-154.
- [19] Wiggerts, T. A. 1997. “*Using clustering algorithms in legacy systems modularization*”, In WCRE’97 Proceedings of the Fourth Working Conference On Reverse Engineering. Washinton, USA. Pag 33.
- [20] Romero D. “*Modelo de un Sistema Manejador de Comunidades Autorganizado para un Sistema Operativo Web Multiagente*”, Tesis de Maestría, Universidad Centrocidental “Lisandro Alvarado” 2008.