

Enrutamiento de Paquetes en una Red Simulada por Medio de Algoritmos Inspirados en Hormigas

Sandra Milena Suárez Calderón

Universidad Distrital, Bogotá, Colombia, sandramilena@ieee.org

Marco Aurelio Alzate Monroy, M.Sc.

Universidad Distrital, Bogotá, Colombia, malzate@udistrital.edu.co

Resumen

Para modelar redes modernas de telecomunicaciones se han propuesto diversos caminos, entre ellos la adaptación de sistemas propios de la naturaleza al análisis de estas nuevas tecnologías. Este trabajo se basa en que las principales características de estos sistemas biológicos, como son la inteligencia emergente, adaptabilidad al medio, auto-organización, trabajo de sociedad, entre otras; pueden aplicarse al control de dichas redes, para obtener cualidades deseables a la hora de su implementación y mantenimiento. Partir del estudio de la topología de dos enrutadores unidos por dos enlaces deja las bases fuertes para extenderlo e implementarlo en las dimensiones de las redes reales actuales y dando paso a nuevos desarrollos dentro del campo de la inteligencia computacional.

Palabras Clave

Comportamiento Auto-emergente, Adaptabilidad, Ant Routing System.

1. Introducción

Se plantea encontrar un algoritmo inspirado en el comportamiento de las hormigas que mejore la asignación de rutas en una red simulada conmutada punto a punto, estudiando el uso de dicho algoritmo bajo un medio dinámico (varía sus características en el tiempo), el cual es un modelo que se aproxima a la realidad de las redes modernas. Dicho algoritmo debe desarrollar por sí mismo un proceso que comienza con el envío de un paquete y concluye sólo hasta llegar a su destino final, a través de una red compuesta por diferentes enrutadores unidos por enlaces; los primeros decidirán cuál es el paso siguiente de acuerdo a unos parámetros establecidos según las propiedades de la red. Dichos parámetros son principalmente heurísticos locales y los cambios dinámicos en las cualidades de los enlaces. Para encontrar un algoritmo que se ajuste a estos parámetros, se van a estudiar diferentes propuestas en una configuración específica, caracterizada por tener dos enrutadores comunicados por dos enlaces diferentes, por medio de simulaciones en software, usando la herramienta Network Simulator 2 (NS-2). Se hará un análisis comparativo de cada experimento y se espera encontrar el mejor algoritmo según las variables y estimadores usados.

2. Antecedentes

En la historia, la tarea de usar algoritmos inspirados en hormigas para el enrutamiento de paquetes comenzó desarrollándose alrededor de 1990 con ACO (Ant Colony Optimization), que es un algoritmo metaheurístico que construye una solución, por medio de la suma de soluciones parciales estocásticamente; así se puede clasificar como constructivo (va construyendo soluciones parciales de acuerdo a una viabilidad local) y repetitivo (la experiencia es base para nuevas y mejores soluciones). AS (Ant System) fue el primer algoritmo basado en ACO, y aunque se lograron grandes avances en su aplicación a diferentes problemas no resultó tan bueno como se esperaba. Así que luego se postuló el ACS (Ant Colony System), que basado en AS logró mejoras importantes: La hormiga se mueve por la línea más probable, sólo las hormigas con resultados óptimos pueden actualizar el recorrido, existe la realimentación negativa y se plantea una lista de información. El algoritmo que se espera encontrar se basará en ARS (Ant Routing System), éste también es un algoritmo de ACO para enrutamiento de redes, en él se utilizan las trayectorias más cortas hasta su máxima capacidad, las cuales se dejan de usar según el largo de sus colas, y converge en buenas soluciones mezclando la exploración con la explotación.

Utiliza los siguientes mecanismos: Regla del valor heurístico, que estima la probabilidad del siguiente enrutador; Regla de Probabilidad de Transición; Paso Aleatorio, que contrarresta la estagnación; Realimentación Positiva, que establece que el camino más usado es el más probable; y Realimentación Negativa que permite la posibilidad de exploración evitando pérdida de paquetes y largas colas (Dorigo M. et al, 2000a).

“La experiencia muestra a ARS como un algoritmo óptimo para asignar diferentes rutas en una red simulada conmutada punto a punto. Actualmente se investiga si ARS puede obtener un aumento en eficiencia mientras evita la pérdida de paquetes en una red de carga pesada” (Bundgaard M. et al ,2002a). Con este fin, se investiga cómo los diferentes efectos heurísticos caracterizan la calidad de la ruta ejecutada. El comportamiento de ARS dependerá de la prioridad dada a la realimentación positiva, la realimentación negativa y los heurísticos locales, con la posibilidad de ajustar sus parámetros característicos para lograr distribución de tráfico por varios caminos cuando la red está muy cargada y así producir una alta eficiencia y perder la menor cantidad de paquetes.

3. Comportamiento Colectivo de los Insectos Sociales y su Aplicación en el Enrutamiento de Paquetes

El comportamiento de un insecto social como son las abejas o las hormigas, es muy interesante abordándolo desde el punto de vista colectivo, ya que un solo insecto no puede sobrevivir solo, sino que se apoya en una estructura social conformada por un grupo gracias a su propia naturaleza biológica (Bonabeau E. et al, 2000a). Las hormigas artificiales son agentes que se crean con base en las características de las hormigas de la naturaleza, copiando sus características pero con ciertos ajustes cuya interpretación se desarrolle en una red. Una diferencia es que estos agentes tienen memoria de dónde han estado y la aleatoriedad de su trayectoria se expresa por el “Estado de Transición Estocástico”, que es el proceso de decisión de pasar de un estado a otro (Bundgaard M. et al ,2002b). Los rasgos más interesantes se exponen a continuación detallando la similitud que se plantea entre hormiga real y artificial.

3.1. Auto-organización

El trabajo de las hormigas funciona sin un jefe que coordine las labores. Esto describe patrones macroscópicos que se explican por interacciones y otros procesos microscópicos. Entonces, el agente es autosuficiente en la toma de decisiones (Bonabeau E. et al, 2000b).

3.2. Estigmergia

Forma indirecta y asíncrona de esquema de comunicación, los insectos manipulan el medio para informar a otros insectos, los cuales toman decisiones según los cambios que hayan sido reportados. Entonces un insecto no debe estar en el mismo lugar ni al mismo tiempo que otro para comunicarse con él, depositando feromona en su recorrido. Para el agente virtual existen dos formas de depositar la feromona: la primera es un trazo, en el cual una sola hormiga descubre una solución total y se devuelve por ella misma depositando feromona; se caracteriza por ser una forma demorada en actualizar. Y la segunda es la actualización del trazo paso por paso, que es adicionar un vértice a la trayectoria mientras transcurre el tiempo. En ciertos algoritmos se presenta una combinación de ambas formas (Dorigo M. et al, 2000b).

3.3. Realimentación Positiva

Los insectos convergen en dirección de las mejores soluciones en el problema de la búsqueda de alimento. Por ejemplo, las hormigas dejan rastros de feromona, y la realimentación positiva se produce porque las hormigas de atrás rastrean esa feromona según su tendencia natural. Entre más hormigas sigan el rastro, la feromona se reafirma más y más, aumentando la probabilidad de encontrar esa ruta en las hormigas que busquen. Para la hormiga virtual las buenas soluciones que se refuerzan (por medio de feromona virtual) son de mayor calidad. Inclusive puede conducir a un “efecto autocatalítico” o popularmente conocido como bola de nieve ya que conduce a una única solución, evitando la búsqueda de nuevas soluciones (Bundgaard M. et al ,2002c).

3.4. Realimentación Negativa

Contrarresta la realimentación positiva. Un claro ejemplo es la evaporación de la feromona, el cual es un fenómeno propiamente ambiental. Este efecto puede ser bueno para que las hormigas no tomen malas soluciones, por ejemplo, si la fuente de alimento se agota, las hormigas, sin necesidad de regresar al agujero, buscan nuevas fuentes permitiendo que la feromona se vaya evaporando. Virtualmente, se representa como una disminución en el porcentaje inicial con respecto a la suma de feromona.

En resumen, una buena trayectoria deberá reforzarse constantemente de acuerdo a la cantidad de feromona que hay en ella, teniendo en cuenta los efectos de la evaporación.

3.5. Aleatoriedad en Insectos Sociales

Es un factor crucial en el descubrimiento de nuevas soluciones. Una hormiga perdida puede “tropezarse” con una buena fuente de alimento, que quizás esté más cerca de otra conocida. Los agentes crean estructuras espacio-temporales en un medio inicialmente homogéneo (área alrededor del hormiguero). Cuando los parámetros en un momento cambian, la aleatoriedad produce bifurcaciones, presentando diferentes rutas, algunas que aún están vigentes y otras olvidadas en cierta medida, cuyo resultado es la posible coexistencia de diferentes estados de estabilidad (Kramer K. et al., 1999).

4. Descripción del Problema

Para describir el problema, se detallan las condiciones según la red que se va a estudiar, los parámetros que debe usar el algoritmo, y regla de Probabilidad que validará los efectos del algoritmo.

4.1.Requerimientos de la Red

La red planteada se caracteriza por contener sólo dos enrutadores, unidos por medio de dos enlaces. La transferencia de paquetes se hará desde el primer enrutador (enrutador 1) hasta el otro (enrutador 2) y no en dirección contraria. Los enrutadores 3 y 4 son transparentes a la transmisión ya que no se tiene en cuenta su retardo y se utilizaron para poder simular los dos enlaces en NS-2. De acuerdo a la figura 1, tomar el enlace 1, significará seguir la trayectoria 1-3-2; y seguir la trayectoria 1-4-2, significará tomar el enlace 2.

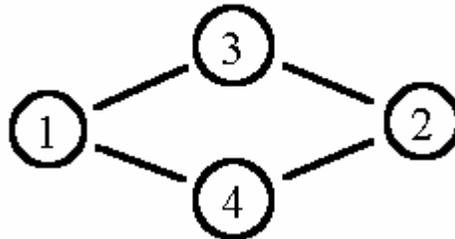


Figura 1: Topología Dos Enlaces

A continuación se muestran los parámetros propios de la red.

4.1.1.Tablas de Enrutamiento

Deben guardar toda la información, de dónde ha estado un paquete. Una red interconectada completamente tiene una tabla con número de entradas está dado por (R) , donde R es el número de enrutadores de la red, en este caso $R=2$, así que la tabla de enrutamiento de un paquete contiene sólo un elemento; esta información debe complementarse con el enlace que fue elegido para llegar allá para obtener las próximas rutas (Bundgaard M. et al ,2002d).

$$((R|-1) \times (R|-1)) \quad (1)$$

4.1.2. Las Hormigas son los Paquetes.

La memoria se almacena según el recorrido de cada uno de los paquetes en un archivo general.

4.1.3. Instante de Actualización del Enrutador

Esta información recuerda a las hormigas las mejores rutas. Es un intervalo de tiempo cuyo valor es crítico ya que debe dar la información a tiempo para el enrutamiento de cada futuro paquete.

4.2. Parámetros Usados en el Algoritmo

Los parámetros α , β y δ se usan para establecer la importancia del trazo, la realimentación negativa y el retardo en la transmisión, respectivamente. Para omitir alguna de estas características, el valor del parámetro correspondiente se hace cero. Tanto el trazo, la realimentación negativa, el retardo y sus parámetros hacen parte de la ecuación que determinará la probabilidad de envío por uno de los dos enlaces disponibles.

4.3. Método de Evaluación del Algoritmo

El algoritmo desarrollado deberá superar tres pruebas.

4.3.1. Adaptabilidad

Esto significa tener la capacidad de adaptarse rápidamente a los cambios que se puedan dar en cantidad de tráfico que deba manejar y por supuesto, los cambios en las tablas de enrutamiento que se asignen deben ser usadas en el instante siguiente para tener validez y efectividad.

4.3.2. Paquetes Recibidos

Debe ser la mayor cantidad posible.

4.3.3. Paquetes Perdidos

Debe ser la menor cantidad posible.

5. Resultados Obtenidos

Después de hacer diferentes pruebas con respecto al algoritmo propuesto se encontró que el que cumple con los requisitos de mejor adaptabilidad y manejo de información se describe como sigue.

5.1. Descripción del Algoritmo

El algoritmo se basa en la realimentación de la información. Comienza con un entorno homogéneo al que se le van dando valores de acuerdo a su comportamiento en el tiempo, cada paquete que llega al primer enrutador mira cuál es su mejor opción de acuerdo a lo que han decidido los que lo han precedido. Un punto importante para tener en cuenta es que esa información se obtiene del último intervalo de tiempo, desechando los anteriores, lo cual asegura que la información procesada pertenezca al estado inmediatamente anterior del sistema, construyendo una solución óptima cada vez que se presenta un paquete, ya que está usando información actualizada y suficiente para decidir cuáles son las mejores rutas.

Cada intervalo evalúa cuántos bytes se han recibido exitosamente, cuántos bytes se han perdido y cuántos bytes hay en la cola de espera instantánea para cada enlace. Después toma estos parámetros, los escaliza entre 0 y 1 y aplica la fórmula de probabilidad. Se genera un número aleatorio entre 0 y 1 con distribución uniforme y si este número es menor a la probabilidad toma el primer enlace, si es mayor, toma el segundo.

$$P = \frac{\frac{(recibidosen1)^\alpha}{(perdidosen1)^\beta * (encola1)^\delta}}{\frac{(recibidosen1)^\alpha}{(perdidosen1)^\beta * (encola1)^\delta} * \frac{(recibidosen2)^\alpha}{(perdidosen2)^\beta * (encola2)^\delta}} \quad (2)$$

La ecuación (2) determina la probabilidad de tomar el enlace 1, la probabilidad de tomar el enlace 2 será: 1-P. Se caracteriza por garantizar que la probabilidad va a ser alta si se reciben muchos paquetes, se pierden pocos y hay pocos en la cola de espera; que va a ser baja si los parámetros evaluados son pésimos para el enrutamiento que se desea; y la probabilidad será intermedia si ambos enlaces están en condiciones aceptables según los parámetros ya mencionados.

5.2. Ambiente de Prueba

En todas las pruebas realizadas se dejaron los siguientes parámetros constantes.

5.2.1. Capacidad de los Enlaces

Los enlaces están caracterizados por tener una capacidad máxima de 0.5MB y su distancia se interpreta a través de un retardo en tiempo establecido de 10mseg.

5.2.2. Tiempo de Prueba

El tiempo de prueba está limitado a diez segundos de simulación; esta es una característica muy útil, ya que los resultados se pueden medir comparando su eficiencia dentro del mismo intervalo de tiempo.

5.2.3. Tráfico

El tráfico se distribuyó en dos partes: un tráfico de tipo Exponencial en paquetes de 1000 bytes, a una tasa de 1000kB/seg; y el segundo, es un tráfico que simula la transferencia de una película cuya tramas están distribuidas en paquetes de tipo UDP de 200 bytes a una tasa de 10 KB/seg.

5.2.4. Intervalos de Tiempo para cada Tráfico

Con el fin de simular una red real, entre 0.5 y 2 seg., entre 4 y 6 seg, y entre 8 y 9.5 seg, se transmitió el Tráfico Exponencial, y entre 2 y 4 seg, 6 y 8 seg se transmitió Tráfico UDP.

5.3. Prueba de Adaptabilidad

El algoritmo se adapta fácilmente si los intervalos de tiempo de actualización son lo suficientemente rápidos como para cubrir al menos la llegada de un paquete, este tiempo se logra en 0.1 segundos.

5.4. Prueba de Paquetes que Llegaron y los que se Perdieron

Para probar el funcionamiento de los parámetros evaluados, se hicieron 4 pruebas. Y para analizar los resultados se hizo una inferencia estadística, realizando un muestreo al azar en este caso la población es infinita, así que el tamaño de la muestra es de 97 para un intervalo de confianza del 95%, un porcentaje de error de 10% y una variabilidad de 0.5 (Spiegel M.,1970). Se analiza la media y la varianza como parámetros de calidad. La tabla 1 resume los resultados obtenidos para cada una de las pruebas. Se debe tener en cuenta que α representa la realimentación positiva, β representa la realimentación negativa y δ representa los heurísticos locales (tamaño de colas en un instante).

Tabla 1: Resultados Según Los Parámetros Considerados

α	β	δ	Media/paquetes que llegaron	Varianza/paquetes que llegaron	Media/paquetes perdidos	Varianza/paquetes perdidos
1	1	1	2611,30159	277,412317	27,72222222	379,690222
1	1	0	2495,12121	891,250464	130,61	875,553434
1	0	1	2602,71134	490,915808	33,742268	530,297466
0	1	1	2489,25	188,25	157	942

Como se puede observar, si se tienen en cuenta los tres parámetros se obtienen mejores resultados mejorando el uso del algoritmo.

6. Conclusiones

Se prueba que el mejor algoritmo es el que tiene en cuenta para cada enlace, la cantidad de bytes que han llegado exitosamente a su destino, la cantidad de bytes que se han perdido y la cantidad de bytes que están en las colas de espera.

ARS se utilizó en este estudio ya que es un problema aplicado a dominio dinámico. Al combinar realimentaciones negativa y positiva con heurísticos locales, se asegura alta calidad (velocidad) y baja pérdida de paquetes. Para dar prioridades entre los heurísticos locales y las realimentaciones sobre un trazo a seguir, ARS se debe ajustar siguiendo la distribución del tráfico de paquetes, el primer criterio de evaluación es su costo en tiempo de enrutamiento traducido en los paquetes que han llegado, y el segundo criterio es la pérdida de paquetes. ARS encuentra buenas soluciones usando conjuntamente ambos criterios.

Una ventaja importante es que estos algoritmos se pueden implementar gráficamente y en un ambiente dinámico; se pueden combinar los diferentes mecanismos; el comportamiento del algoritmo se puede adaptar según las propiedades que se quieran reforzar entre: alta o baja tendencia a la distribución, exploración, y reducir pérdida de paquetes.

Referencias

Bonabeau E, Dorigo M. y Theraulaz G.(2000). “Inspiration for Optimization From Social Insect Behaviour”. *Nature Magazine*, Vol 406, pp 39-42.

Bundgaard M. et al (2002). “Ant Routing System”. Internet Technology Department. University of Copenhagen, Dinamarca

Dorigo M., Maniezzo V y Colonni A. (1996). “The Ant System: Optimization by a Colony of Cooperating Agents”. *IEEE Transactions on Systems, Man and Cybernetics*, Vol 26, No 1, pp.1-26

Spiegel M. (1970). *Estadística*. 1ra edición, Mcgraw-Hill, U.S.A.

Kramer K., Minar N. y Maes P. (1999). “Tutorial: Mobile Software Agents for Dynamic Routing”. www.media.mit.edu/~nelson/research/routes/

Información Biográfica

Sandra SUAREZ, Sandra Milena Suárez es estudiante de Ingeniería Electrónica de la Universidad Distrital Francisco José de Caldas. Ha sido miembro IEEE durante 3 años consecutivos. Fue presidenta del Capítulo Estudiantil de Procesamiento de Señales IEEE y actualmente es presidenta de la Rama Estudiantil IEEE de la Universidad Distrital, Bogotá, Colombia.

Marco ALZATE, Marco Aurelio Alzate es ingeniero electrónico de la Universidad Distrital, Bogotá, Colombia (1989) y Máster en Ingeniería Eléctrica de la Universidad de los Andes en Bogotá (1991). Obtuvo la calificación doctoral de la Universidad de Maryland en College Park, Maryland, en 2001. Actualmente es profesor titular de la Facultad de Ingeniería de la Universidad Distrital, donde imparte cursos de pregrado y posgrado sobre procesamiento digital de señales y análisis de desempeño de redes de comunicaciones. Participa en los grupos de investigación en procesamiento digital de señales (GI-DSP-UD) y en telecomunicaciones (GITUD) de la Universidad Distrital.