



UNIVERSIDAD NACIONAL AUTÓNOMA – IZTAPALAPA  
DIVISIÓN DE CIENCIAS BÁSICAS E INGENIERÍA

**CONTROL DE CONGESTIÓN  
PARA REDES INALÁMBRICAS DE SENSORES,  
PARA APLICACIÓN EN MISIONES CRÍTICAS**

Idónea Comunicación de Resultados  
para obtener el grado de  
**MAESTRA EN CIENCIAS Y TECNOLOGÍAS DE LA INFORMACIÓN**

Presentada por:

**Daniela Aguirre Guerrero**

**Asesores:**

Dr. Ricardo Marcelín Jiménez

Dr. Enrique Rodríguez de la Colina

Dr. Miguel López Guerrero

Dr. Ricardo Marcelín Jiménez

M. en I. Martha Montes de Oca Cáliz

Defendida públicamente en la UAM Iztapalapa,  
el 31 de enero de 2014

Presidente: Dr. Miguel López Guerrero, UAM – Iztapalapa, Redes y Telecomunicaciones  
Secretario: Dr. Ricardo Marcelín Jiménez, UAM – Iztapalapa, Redes y Telecomunicaciones  
Vocal: M. en I. Martha Montes de Oca Cáliz, UNAM, Computo y Visualización

# Resumen

---

Las redes inalámbricas de sensores (WSN, por sus siglas en inglés) se componen de una gran cantidad de nodos con capacidades de monitorización, procesamiento y comunicación. Estos nodos monitorizan su ambiente y transportan paquetes de información, hasta uno o más centros de recolección o estaciones base, llamados nodos *sink*. Existen varios problemas abiertos que limitan la implementación práctica de WSN [1]. Estas limitaciones se deben principalmente a las reducidas capacidades de procesamiento y batería de los nodos.

El mayor reto que se enfrenta al diseñar WSN, consiste en programar cada nodo autónomo, de manera que sea capaz de tomar decisiones basadas sólo en sus condiciones locales y teniendo en cuenta que dichas decisiones impactarán en el comportamiento global de la WSN. Es por ello que la idea de construir soluciones globales a partir del comportamiento individual de los componentes de un sistema, es la perspectiva de “sistemas complejos” que se ha utilizado en este trabajo. En el diseño de WSN también se debe tomar en cuenta los requerimientos de una aplicación en particular, como son: el tiempo de vida de la WSN, tiempo de respuesta, tolerancia a fallas, control de congestión, etc.

Este trabajo aborda el problema de control de congestión para WSN orientadas a eventos (*event-driven*) [2], como las utilizadas en la detección temprana de incendios forestales. En estas aplicaciones cuando un evento de interés es detectado, el tráfico de la WSN se incrementa rápidamente, lo que puede conducir a su congestión.

Decimos que una red está congestionada cuando la densidad de tráfico es tal que la mayoría de los paquetes se pierden o experimentan retardos considerables, provocando así la degradación del desempeño global de la red y en casos extremos el colapso de la misma. Para evitar esta contingencia, se han diseñado técnicas de control de congestión [3 - 5], que buscan transportar la mayor cantidad posible de paquetes y minimizar su retardo. Estos mecanismos de control de congestión deben considerar los efectos adversos de transmisiones en el medio inalámbrico, tales como interferencia y atenuación. Es por ello que como parte de la propuesta que se presenta, fue diseñado un modelo de simulación de WSN que toma en cuenta los efectos inalámbricos y escenarios de congestión.

Esta investigación sostiene que las WSN pueden ser analizadas como un “sistema complejo”, por lo que se ha propuesto un modelo de WSN que considera a los paquetes de datos como agentes de un “sistema complejo”. El segundo aspecto de la investigación está basado en el modelo mencionado y consiste en la propuesta de un nuevo esquema de control de congestión, en el cual a partir de sus condiciones locales cada nodo es capaz de tomar decisiones referentes al encaminamiento, contención en el medio y eliminación

selectiva de paquetes. Así, a partir de controlar las interacciones locales entre los paquetes de datos es posible obtener un comportamiento del tráfico global de la WSN que sea emergente y autoorganizado. Como consecuencia la WSN es capaz de autoregular su densidad de tráfico.

El esquema de control de congestión propuesto cumple con los requerimientos de WSN orientadas a eventos, tales como: tiempo de respuesta, bajo consumo de energía en escenarios no congestionados, informe de todos los eventos detectados por la WSN y balance de carga.



# Abstract

---

A WSN consists of a large number of nodes, which have sensing, processing and communicating capacities. Not only do nodes in a WSN monitor their environment, but they also forward and route data packets to one or more appointed *sink* nodes. It is known that there are open issues that limit the practical adoption of WSN [1]. These limitations come from the fact that each individual wireless node in a WSN has reduced processing capabilities, as well as, a modest energy budget.

The major challenge faced when designing a WSN consists in programming each autonomous node to make decisions based only on its local conditions, having in mind that each decision impacts on the overall network behavior. Indeed, this is the Complex Systems perspective considered in this work, which achieves a global congestion control from the actions that each component takes, based on its immediate context.

Additionally, in event-based WSN applications, such as early forest fire detection, when the target event is detected, network traffic may suddenly increase and it could lead to the congestion of the underlying network.

Network congestion arises when the traffic density reaches a condition where most of the exchanged packets experience delays or are lost. Therefore, the overall system's performance is degraded and may collapse. To avoid this condition, congestion control techniques are designed [3][5] to carry as many packets as possible and to minimize the elapsed time from source to destination. Congestion mechanisms should also consider the side effects of wireless transmissions, such as interference and power loss.

In order to address congestion control, a simulation model has been proposed in this work. This model comprises wireless effects and MAC protocols, that can be switched on and off, as well as, traffic patterns that may induce network congestion. This thesis argues that a WSN can be understood as a Complex System. From this perspective, the WSN model considers data packets as agents that interact between each other at the nodes where they are temporarily stored on their way to the sink node(s). Based on these local interactions, each node takes decisions about routing, medium contention and packet dropping. As an emergent global behavior, it is shown that the network achieves a multi-path routing mechanism and a self-adaptive traffic level that matches the network capacity.



Como resultado de esta investigación, se realizó la publicación de un artículo de investigación titulado “*WSN Simulation Model with a Complex System Approach*”, el cual se presentó en *the 2013 Summer Computer Simulation Conference*, en la ciudad de Toronto.

As a result of this Project, it was performed a research paper publication titled “*WSN Simulation Model with a Complex System Approach*” that was presented in *the 2013 Summer Computer Simulation Conference*, in the city of Toronto.

# Agradecimientos

---

Agradezco la Universidad Autónoma Metropolitana y a la comisión del Posgrado de Ciencias y Tecnologías de la Información por abrirme las puertas de esta importante casa de estudios.

Agradezco a mis asesores el Dr. Ricardo Marcelín Jiménez y el Dr. Enrique Rodríguez de la Colina, por su invaluable apoyo y compromiso. Gracias por sus consejos y por su ejemplo, no sólo han enriquecido mi crecimiento profesional, sino también personal.

Agradezco a mis padres por su amor incondicional y por mantenerse siempre atentos de mis sueños, tristezas y alegrías. Gracias por estar siempre para mí.

Agradezco a mis hermanos Juan y Julio, por su compañía, sus abrazos y el amor que siempre me demuestran.

Agradezco a mi hermana Liz por ser mi compañera de vida, por su amor incondicional y porque siempre es parte de cada logro de mi vida. Siempre tengo mucho que agradecerle hermana.

Agradezco a Ismael por estos años compartidos, por su ternura y cariño. Gracias por todo lo hermoso que me haces sentir.





# Contenido

---

<b>Acrónimos .....</b>	<b>XII</b>
<b>Lista de Figuras .....</b>	<b>XIV</b>
<b>Lista de Tablas .....</b>	<b>XVI</b>
<b>1. Introducción .....</b>	<b>1</b>
1.1.Motivación.....	1
1.2.Objetivos.....	2
1.3.Metodología de la Investigación.....	2
1.4.Contribución .....	3
1.5.Estructura del Documento .....	4
<b>2. Estado del Arte .....</b>	<b>7</b>
2.1.Congestión en WSN .....	7
2.1.1. Mecanismos de Control de Congestión para WSN.....	7
2.1.2. Control de Congestión en la Capa de Enlace de Datos.....	8
2.1.3. Control de Congestión en la Capa de Transporte .....	8
2.1.4. Control de Congestión en la Capa de Red .....	9
2.1.5. Mecanismos de Detección de Congestión para WSN.....	11
2.1.6. Algoritmos de Control de Congestión para WSN.....	11
2.2.Análisis de WSN desde una perspectiva de Sistemas Complejos.....	15
<b>3. Análisis de Tráfico en WSN desde la Perspectiva de Sistemas Complejos .....</b>	<b>19</b>
3.1.Marco Conceptual .....	19
3.1.1. Satisfacción de un Paquete.....	20
3.1.2. Interferencia entre un Paquete y un Grupo de Paquetes .....	22
3.1.3. Satisfacción de la WSN .....	23
3.2.Modelo de Simulación de WSN, desde la Perspectiva de Sistemas Complejos .....	23
3.2.1. Configuración de la WSN.....	24
3.2.2. Configuración de los Nodos.....	25
3.2.3. Simulación de Efectos Inalámbricos.....	27
3.2.4. Evaluación de Desempeño .....	28
3.3.Diseño de Algoritmos Distribuidos para la Autoorganización de Tráfico en WSN	29

<b>4. Control de Congestión para WSN Orientadas a Eventos.....</b>	<b>31</b>
4.1.Fase de Inicialización.....	31
4.1.1. Parámetros de Encaminamiento.....	31
4.1.2. Tiempo Máximo de Espera en Buffer ( $\Delta t_{max}$ ) .....	33
4.1.3. Satisfacción de un Paquete ( $\sigma_{p,N}$ ).....	33
4.1.4. Interferencia de un Paquete ( $\phi_p$ ).....	34
4.2.Fase de Procesamiento .....	35
4.3.Fase de Encaminamiento.....	35
4.4.Fase de Transmisión.....	36
4.5.Discusión .....	37
<b>5. Evaluación de Desempeño .....</b>	<b>39</b>
5.1.Programa Simulador.....	39
5.2.Análisis del Desempeño .....	41
<b>Conclusiones y Trabajo a Futuro.....</b>	<b>47</b>
<b>Bibliografía.....</b>	<b>50</b>



# Acrónimos

---

<b>ABM</b>	Agent-Based Model
<b>AIMD</b>	Additive Increment / Multiplicative Decrement
<b>BER</b>	Bit Error Rate
<b>BPSK</b>	Binary Phase Shift Keying
<b>CAS</b>	Complex Adaptive Systems
<b>CSMA</b>	Carrier Sense Multiple Access
<b>DCCP</b>	Datagram Congestion Control Protocol
<b>DPCC</b>	Decentralized Predictive Congestion Control Protocol
<b>MAC</b>	Medium Access Control
<b>PDF</b>	Probability Distribution Function
<b>PI</b>	Propagation of Information
<b>RTT</b>	Round Trip Time
<b>TCP</b>	Transmission Control Protocol
<b>TDMA</b>	Time Division Multiple Access
<b>TTL</b>	Time to Live
<b>WSN</b>	Wireless Sensor Network



# Lista de Figuras

---

3.1. Ruta seguida por el paquete $p$ en su viaje del nodo fuente $S$ hacia el <i>sink</i> .....	22
3.2. Asignación de $\Delta t_{max}$ de acuerdo a la ubicación de los nodos con respecto al <i>sink</i> .	22
3.3. Árbol obtenido al ejecutar el algoritmo PI sobre un grafo .....	26
3.4. Comportamiento del tráfico generado y cursado por la red .....	29
5.1. Interfaz de usuario del simulador .....	40
5.2. Paquetes perdidos por congestión a nivel de nodo.....	42
5.3. Paquetes perdidos por congestión a nivel de enlace .....	43
5.4. Vista del grafo de la WSN.....	43
5.5. Porcentaje de paquetes recibidos de cada fuente .....	44
5.6. Paquetes perdidos variando el tamaño del <i>buffer</i> .....	44
5.7. Tráfico entregado .....	45





# Lista de Tablas

---

2.1.Mecanismos de Control de Congestión para WSN .....	8
2.2.Control de Congestión en la Capa de Red.....	10
2.3.Algoritmos de Control de Congestión para WSN .....	12
5.1.Configuración de la Simulación .....	42



# Capítulo 1

## Introducción

---

Las redes inalámbricas de sensores (WSN, por sus siglas en inglés) son una tecnología emergente, estas redes se componen de un gran número de pequeños nodos autónomos organizados en una red cooperativa *ad hoc*. Los nodos monitorizan y colectan información de su ambiente de operación, para después transportarla hasta uno o más nodos colectores, llamados nodos *sink*. Para cumplir con este propósito, los nodos deben cooperar, de manera que sea posible transportar la mayor cantidad de paquetes en el menor tiempo posible.

Al diseñar protocolos de comunicación para WSN se debe considerar que las capacidades de procesamiento de los nodos, así como su batería son reducidas. Otros aspectos que debe tomarse en cuenta son los requerimientos de cada aplicación en particular, tales como: tiempo de respuesta, tiempo de vida, tolerancia a fallas, control de congestión, etc. Dichas características y requerimientos hacen que existan varios problemas abiertos en el diseño e implementación de WSN, uno de estos retos es el control de congestión.

### 1.1 Motivación

La congestión en redes de comunicación, surge cuando la densidad de tráfico alcanza niveles en los que la mayoría de paquetes intercambiados se pierden o experimentan retardos considerables, esto en referencia al tiempo de respuesta que requiere la aplicación. Existen dos niveles de congestión en WSN, congestión a nivel de nodo y congestión a nivel de enlace. La congestión a nivel de nodo ocurre cuando el *buffer* de algún nodo se desborda; mientras que la congestión a nivel de enlace ocurre debido a colisiones de paquetes a causa de transmisiones simultáneas, como resultado de ello la utilización del canal decrece. En los dos tipos de congestión la cantidad de paquetes perdidos y el tiempo de respuesta de la WSN se reducen.

Los protocolos de control de congestión se pueden aplicar en la capa de enlace, en la capa de transporte o desde un enfoque entre capas (*crosslayer*) que incluya, además de las capas ya mencionadas, la capa de red. Dicho enfoque es aplicado en esta propuesta.

El mecanismo de control de congestión que se propone, es adecuado para WSN orientadas a eventos, como las utilizadas en la detección temprana de incendios forestales. Este tipo de aplicaciones presenta los escenarios de congestión más adversos, debido a que cuando se detecta un evento de interés, como la presencia de fuego, el tráfico de la WSN puede incrementarse súbitamente.

## 1.2 Objetivos

El objetivo principal de este trabajo es diseñar un algoritmo autoorganizado, autoconfigurable y distribuido de control de congestión para WSN orientadas a eventos.

Los objetivos específicos son:

- 1) Diseñar un marco conceptual, desde un enfoque de “sistemas complejos”, para analizar el tráfico en WSN.
- 2) Desarrollar un modelo de simulación de WSN que sirva para evaluar distintos algoritmos de control de congestión.

## 1.3 Metodología de Investigación

El proceso de investigación seguido durante el desarrollo de este trabajo, consistió en las siguientes etapas:

i. Formulación del problema de investigación

Durante esta etapa se definió el problema de investigación, estableciendo con ello los objetivos de la investigación. Además, se llevó a cabo la caracterización de los requerimientos del sistema.

ii. Investigación del estado del arte

Debido a la naturaleza de esta investigación, se realizó el estado del arte de en 2 sentidos: 1) control de congestión en WSN y 2) análisis de WSN desde la perspectiva de “sistemas complejos”.

iii. Desarrollo de un marco conceptual para el análisis de tráfico en WSN desde la perspectiva de “sistemas complejos”.

Dentro de este marco conceptual se definieron conceptos y métricas que permiten analizar el tráfico de WSN como un “sistema complejo”.

iv. Desarrollo de una herramienta de simulación

Se desarrolló un programa, en el simulador de “sistemas complejos” NetLogo [6], con ello se obtuvo un simulador que, de acuerdo a las métricas definidas en la etapa *iii*, permite simular y evaluar protocolos de control de congestión, encaminamiento y control de acceso al medio (MAC, por sus siglas en inglés).

v. Diseño del algoritmo de control de congestión para WSN

El algoritmo de control de congestión se diseñó bajo los principios establecidos en el marco conceptual diseñado en la etapa *iii*.

vi. Evaluación del algoritmo propuesto

Durante esta etapa se comparó el desempeño algoritmo propuesto con otros trabajos reportados en el estado del arte. El algoritmo propuesto se probó y evaluó mediante el simulador diseñado en la etapa *iv*.

vii. Reporte de resultados

Finalmente se documentó la investigación y se llevó a cabo el reporte de resultados.

## 1.4 Contribución

Con el objetivo de enfrentar el problema de control de congestión en WSN, se diseñó un modelo de simulación de WSN, mediante el cual es posible simular escenarios de tráfico típicos de WSN orientadas a eventos. El modelo propuesto, también considera los efectos de la transmisión por el medio inalámbrico, como la interferencia y la atenuación.

El enfoque aplicado en este trabajo, consiste en estudiar el tráfico en WSN desde la perspectiva de “sistemas complejos”. De acuerdo con [7] - *“Los sistemas complejos son un campo interdisciplinario de la investigación, que trata de explicar cómo un gran número de entidades relativamente simples se autoorganizan, sin ningún tipo de control central, en un ente colectivo que es capaz de crear patrones usando información y en algunas ocasiones, tiene lugar la evolución y el aprendizaje”*. El análisis de WSN, desde este nuevo enfoque se puede identificar en un pequeño conjunto de trabajos de investigación [8 - 12]. La mayoría de estas propuestas, consideran a los nodos de las WSN como agentes del “sistema complejo”, mientras que en nuestra propuesta los paquetes de datos son considerados como agentes del “sistema complejo”, esta perspectiva facilita el análisis del tráfico global de la WSN y las interacciones locales de los paquetes de datos, al controlar dichas interacciones es posible mantener a la WSN operando, de manera emergente y autoorganizada.

A partir del análisis de las interacciones locales de los paquetes de datos, se diseñó un algoritmo de control de congestión, en el cual cada nodo, basado únicamente en sus condiciones locales, es capaz de tomar decisiones acerca de encaminamiento, contención en el medio y eliminación de paquetes. Al tomar estas decisiones los nodos buscan mejorar el desempeño total de la WSN, como resultado, la WSN es capaz de autoregular el tráfico de paquetes. Al realizar esta adaptación de niveles de tráfico, la propuesta considera un esquema de balance de carga y de prioridad de paquetes, el cual garantiza que el nodo *sink* obtenga información de la mayoría de los eventos detectados. El esquema de control de congestión propuesto, es evaluado mediante un programa simulador, desarrollado en el simulador de “sistemas complejos” NetLogo [6]. La herramienta de simulación diseñada, toma en cuenta las métricas de análisis y de evaluación de desempeño propuestas en el modelo de WSN propuesto.

## 1.5 Estructura del Documento

Después de este capítulo introductorio, en el Capítulo 2 se presenta el estado del arte de la investigación, el cual se divide en 2 secciones: 1) algoritmos de control de congestión para WSN y 2) análisis de WSN, desde la perspectiva de “sistemas complejos”. La Sección 2.1 explica en qué consiste la congestión en redes, cuáles son sus causas y las estrategias más utilizadas en el control de congestión de WSN. Después se mencionan los principales algoritmos de control de congestión para WSN, finalmente se incluye una discusión acerca de sus ventajas y desventajas. En la Sección 2.2 se habla de la relación que existe entre los “sistemas complejos” y las WSN, y se presenta un conjunto de trabajos que utilizan este nuevo enfoque para el análisis de WSN.

En el Capítulo 3 se presentan un análisis de tráfico en WSN desde la perspectiva de “sistemas complejos”, para ello en la Sección 3.1 se propone un marco conceptual que permite llevar a cabo el análisis de tráfico en WSN desde la perspectiva de “sistemas complejos”. La principal idea de este marco conceptual es que una WSN puede ser estudiada como un “sistema complejo” que se compone de una gran cantidad de paquetes de datos, en el cual mediante un algoritmo de control de congestión, los nodos de la WSN regulan el comportamiento de los paquetes, con la finalidad de reducir los conflictos<sup>1</sup> entre ellos y así mejorar el desempeño de la WSN. Basado en el marco conceptual de la Sección 3.1, en la Sección 3.2 se describe el modelo de simulación de WSN que fue desarrollado para evaluar el esquema de control de congestión propuesto, dicho modelo de simulación fue presentado en un trabajo previo [13]. La Sección 3.3, aplica las ideas presentadas en la Secciones 3.1 y 3.2, mostrando una perspectiva de su aplicación en el diseño de algoritmos distribuidos de autoorganización de tráfico.

En el Capítulo 4 se detalla el esquema de control de congestión propuesto, el cual consiste en que cada nodo toma decisiones referentes al encaminamiento, contención en el medio y eliminación de paquetes. Estas decisiones se toman con base en las condiciones locales de tráfico de cada nodo. La propuesta también contempla mecanismos de prioridad de paquetes para garantizar que el nodo *sink* colecte información de la mayor cantidad de eventos detectados; además de un mecanismo de balance de carga, que mediante técnicas multitrayectorias permite evitar la congestión a nivel de nodo.

En el Capítulo 5 se muestra la evaluación de desempeño del esquema de control de congestión propuesto, la cual se llevó a cabo mediante una herramienta de simulación desarrollada en el simulador de “sistemas complejos” NetLogo [6]. La herramienta de simulación se diseñó tomando en cuenta las métricas de evaluación de desempeño propuestas en el modelo de WSN propuesto.

---

<sup>1</sup> Como son las colisiones, las pérdidas por desbordamiento de *buffer* y los retrasos por tiempo de espera en el *buffer*.

Finalmente, se mencionan las conclusiones y futuras líneas de investigación que pueden derivarse de este trabajo.





# Capítulo 2

## Estado del Arte

---

Este capítulo se divide en 2 secciones: en la Sección 2.1, se describe en qué consiste la congestión en WSN, así como las técnicas para detectarla y mitigarla, finalmente se presentan los principales algoritmos de control de congestión para WSN; en la Sección 2.2 se explica qué son los “sistemas complejos” y qué relación tienen con las WSN, después se presentan algunos trabajos representativos, en donde se analizan las WSN desde esta nueva perspectiva.

### 2.1 Congestión en WSN

Existen 2 tipos de congestión en WSN: congestión a nivel de nodo y congestión a nivel de enlace. La congestión a nivel de nodo ocurre cuando el *buffer* de algún nodo se desborda. Por otro lado, la congestión a nivel de enlace es causada por colisiones, debido a transmisiones simultáneas, lo que provoca que la utilización del canal se reduzca. Ambos niveles de congestión pueden presentarse en WSN orientadas a eventos, ya que tanto cerca de los nodos fuente<sup>1</sup>, como cerca del nodo *sink*; habrá una gran cantidad de nodos intentando transmitir, lo cual puede ocasionar colisiones, además de que el *buffer* de dichos nodos puede desbordarse debido a un incremento repentino de tráfico.

Con el fin de evitar que la WSN se colapse a causa de la congestión, los métodos de control de congestión regulan el tráfico de la WSN para mantenerlo en niveles óptimos de operación. Los esquemas de control de congestión en WSN pueden estar enfocados en: la capa de enlace de datos, la capa de transporte o pueden tener un enfoque *entre capas* que contemple, además de las capas mencionadas, la capa de red.

#### 2.1.1. Mecanismos de Control de Congestión para WSN

En la Tabla 2.1 se presentan los principales mecanismos de control de congestión para WSN. Estos mecanismos se combinan para construir algoritmos de control de congestión, los cuales pueden actuar *punto a punto* o *extremo a extremo*. En los algoritmos *punto a punto*, cada nodo de acuerdo a sus condiciones locales de tráfico, controla la congestión. En los algoritmos de *extremo a extremo*, mediante el intercambio de paquetes entre los nodos fuente y el *sink*, la congestión es controlada.

Otra clasificación que se puede dar a los algoritmos de control de congestión es de acuerdo a la forma en que aplican sus mecanismos de control de congestión; ya sea de manera reactiva, es decir una vez que se presenta la congestión; o de manera proactiva, aplicando mecanismos de prevención de congestión.

---

<sup>1</sup> Es decir los nodos que detectan algún evento de interés.

**Tabla 2.1. Mecanismos de Control de Congestión para WSN**

Mecanismo	Capa de Comunicación	Nivel de Congestión
Control de la tasa de generación de paquetes	Capa de enlace	Nodo
Ajuste de intervalos backoff <sup>2</sup>	Capa de enlace	Enlace
Contención de paquetes basada en CSMA	Capa de enlace	Enlace
Calendarización de paquetes basada en TDMA	Capa de enlace	Enlace
Control de tasa de transferencia	Capa de enlace y/o de transporte	Enlace y nodo
Ajuste de la ventana de congestión <sup>3</sup>	Capa de transporte	Enlace
Redistribución de tráfico con nodos multiradio	Capa de red	Enlace y nodo
Encaminamiento basado en colonias ( <i>clustering</i> )	Capa de red	Enlace y nodo
Encaminamiento multitrayectorias	Capa de red	Nodo
Encaminamiento adaptativo	Capa de red	Nodo

Como se puede observar en la Tabla 2.1, cada mecanismo actúa en una capa específica de comunicación y ataca alguno de los 2 niveles de congestión. A continuación se explica, la manera en que se aplican estos mecanismos en las capas de comunicación y la relación que tiene cada capa en el control de congestión.

### 2.1.2. Control de Congestión en la Capa de Enlace de Datos

La capa de enlace de datos juega un papel muy importante en el control de congestión, ya que un protocolo MAC eficiente, debe evitar, o en su defecto controlar, la congestión a nivel de enlace, para lograrlo se aplican técnicas basadas en acceso múltiple por división de tiempo (TDMA, por sus siglas en inglés) o basadas en CSMA. En el primer caso, con el fin de evitar transmisiones simultáneas, la transmisión de paquetes se calendariza, asignando a cada nodo de un vecindario un intervalo de tiempo para transmitir. En el segundo caso, antes de hacer uso del canal, los nodos deben monitorizarlo para saber si está libre o no, en caso de que esté libre se puede realizar la transmisión, en caso contrario se debe esperar un intervalo de tiempo (conocido en inglés como *backoff*) antes de volver a monitorizar el canal. Otras técnicas para evitar la congestión en la capa de enlace de datos, son: el control de la tasa de generación de paquetes en los nodos fuente y controlar *punto a punto* la tasa de transferencia, con lo que también se evita la congestión a nivel de nodo. Además se pueden definir criterios para optimizar los intervalos de tiempo de espera (*backoff*).

### 2.1.3. Control de Congestión en la Capa de Transporte

El control de congestión en la capa de transporte, se puede realizar controlando de *extremo a extremo* la tasa de transferencia; en ese caso cuando la congestión se presenta, un nodo coordinador, que normalmente es el nodo *sink*, envía avisos a los nodos cercanos a la zona congestionada para que reduzcan su tasa de transferencia. Cuando la congestión es superada, los nodos deben incrementar su tasa de transferencia, con la finalidad de

<sup>2</sup> Para protocolos de control de acceso al medio basados en CSMA.

<sup>3</sup> Para protocolos de transporte inspirados en el protocolo de control de transmisión (TCP, por sus siglas en inglés).

aumentar el caudal de envío<sup>4</sup>. Normalmente el control de la tasa de transferencia, para protocolos de transporte inspirados en protocolo de control de transmisión (TCP, por sus siglas en inglés), se realiza mediante el mecanismo de “incremento aditivo / decremento multiplicativo” (AIMD, por sus siglas en inglés), de la siguiente manera: sea  $w(t)$  la tasa de transferencia de un nodo  $i$ , durante un periodo de tiempo  $t$ ; entonces el ajuste de tasa de transferencia en el siguiente periodo de tiempo  $(t+1)$  está dado por la siguiente ecuación:

$$w(t + 1) = \begin{cases} w(t) + a & \text{Cuando la congestión no es detectada} \\ w(t) * b & \text{Cuando la congestión es detectada} \end{cases}$$

Dónde  $a$  y  $b$  son los parámetros de incremento y decremento, respectivamente, los cuales son definidos por el algoritmo de control de congestión.

Existen también algunas propuestas de control de congestión para WSN, como [14] y [15], que se basan en los principios de TCP. El control de congestión que realiza TCP consiste en asignar a cada nodo emisor una ventana de transmisión y una de congestión, la ventana de transmisión indica la tasa de bits que es capaz de recibir el receptor, es decir el *sink*, la ventana de congestión indica la tasa de bits que puede transportar la red sin tirar paquetes. El valor de la ventana de congestión se puede inicializar con el intercambio de paquetes de *extremo a extremo* y conforme las condiciones de tráfico van cambiando se ajusta, por lo regular mediante un algoritmo AIMD. De esta manera el emisor ajusta su tasa de transferencia a un valor menor a la ventana de transmisión y de congestión.

#### 2.1.4. Control de Congestión en la Capa de Red

En lo que respecta al control de congestión en la capa de red, en general se busca que los algoritmos de encaminamiento para WSN proporcionen balance de carga, para así evitar el congestionamiento de rutas; por ejemplo en [16] se presenta un protocolo de encaminamiento que funciona con nodos multiradio, es decir que pueden transmitir en distintas frecuencias, así cuando se presenta un escenario de congestión, los nodos activan una red secundaria con una frecuencia distinta a la que usaban originalmente, de manera que el tráfico excedente sea redistribuido hacia la red secundaria.

Otros protocolos de encaminamiento que tratan de evitar la congestión son los basados en la agrupación de nodos (*clustering*) en estos protocolos la WSN se divide en subredes llamadas *colonias*, cada *colonia* tiene un nodo líder que colecta los paquetes de los nodos de su *colonia*, así los nodos líderes forman una subred *ad hoc* para transportar los paquetes de sus *colonias* hasta el *sink*.

En la Tabla 2.2, se presentan algunas de las propuestas más recientes, relacionadas con el control de congestión, aplicadas a la capa de red. Aunque los protocolos de encaminamiento mencionados en la Tabla 2.2 hacen frente a la congestión, no son

---

<sup>4</sup> Cantidad de paquetes que son entregados al *sink*, por unidad de tiempo.

propriadamente protocolos de control de congestión, ya que un protocolo de control de congestión actúa principalmente en las capas de enlace y/o de transporte. Si el protocolo de control de congestión tiene un enfoque *entre capas*, puede incluir mecanismos de encaminamiento adaptativo y/o multitrayectorias, estas técnicas permiten redistribuir el tráfico, con lo que proveen balance de carga, y de energía. De esta manera es posible extender el tiempo de vida de las rutas y evitar así la pérdida de cobertura. Las estrategias para descubrir nuevas rutas, de manera reactiva, constituyen el reto principal de los mecanismos multitrayectorias, ya que durante este proceso el intercambio de paquetes de control puede generar más tráfico y contribuir a la congestión. Las rutas se pueden construir con estrategias *punto a punto*, o *extremo a extremo*; en el primer caso cada nodo, de acuerdo a sus condiciones locales de tráfico, toma la decisión de hacia que nodos enviar sus paquetes; en el segundo caso, mediante intercambio de paquetes entre los nodos fuente y el *sink*, se descubren nuevas rutas.

**Tabla 2.2. Control de Congestión en la Capa de Red**

Algoritmo	Mecanismos de control de congestión	Estrategia de construcción de rutas
<i>Priority based Traffic Load Reduction Algorithm (PBTR)</i> [17]	<ul style="list-style-type: none"> <li>▪ Establece rutas dedicadas para transportar información crítica</li> <li>▪ Encaminamiento multitrayectorias de acuerdo a criterios de prioridad de paquetes y a la presencia de congestión</li> </ul>	Extremo a extremo
<i>Clustering based Energy Efficient Congestion aware Protocol for WSN</i> [18]	<ul style="list-style-type: none"> <li>▪ Realiza el <i>clustering</i> tomando en cuenta la energía de los nodos</li> <li>▪ El <i>clusterhead</i> cambia de manera aleatoria después de cada transmisión</li> <li>▪ Cuando el <i>clusterhead</i> tiene información de todos sus hijos la comprime en un solo paquete que transmitirá</li> </ul>	Extremo a extremo
<i>Dynamic alternative path selection scheme (DAIPaS)</i> [19]	<ul style="list-style-type: none"> <li>▪ Encaminamiento multitrayectorias adaptativo</li> </ul>	Punto a punto
<i>Congestion avoidance and fairness in WSN</i> [20]	<ul style="list-style-type: none"> <li>▪ Asume una topología fija de malla</li> <li>▪ Encaminamiento multitrayectorias adaptativo</li> <li>▪ Control de la tasa de generación de paquetes</li> </ul>	Punto a punto
<i>Congestion avoidance control mechanism for multi-paths routing protocol (MR-CACM)</i> [21]	<ul style="list-style-type: none"> <li>▪ Encaminamiento multitrayectorias</li> </ul>	Extremo a extremo

### 2.1.5. Mecanismos de Detección de Congestión para WSN

Cómo ya se ha mencionado, en las WSN orientadas a eventos mientras que no sea detectado algún evento de interés, el tráfico de la WSN será mínimo; sin embargo una vez que se detecta un evento de interés, el tráfico de la WSN tiende a incrementarse repentinamente, por lo que se deben aplicar algoritmos de control de congestión reactivos, es decir que actúen de acuerdo a las condiciones de tráfico que se van presentando. Para que esto sea posible los algoritmos de control de congestión deben hacer uso de mecanismos que les permitan detectar la posibilidad de congestión.

Algunos de los mecanismos utilizados para detectar posibles congestiones consisten en que cada nodo monitoree su estado, y en algunos casos el estado de los nodos de su vecindario, de acuerdo a los siguientes parámetros:

- Porcentaje de ocupación de *buffer*
- Tasa de arribo de paquetes
- Energía remanente
- Condiciones del canal<sup>5</sup>
- Probabilidad de error por bit (BER, por sus siglas en inglés)
- Tiempo transcurrido entre el envío de paquete y la recepción del acuse de recibido (RTT, por sus siglas en inglés)

Si el algoritmo de control de congestión utiliza alguna técnica de encaminamiento, puede ser de gran utilidad que los nodos, además de lo anterior, conozcan su distancia en saltos, y la de sus vecinos, hacia el *sink*.

En la siguiente sección se presenta una descripción y discusión acerca de los principales algoritmos reactivos de control de congestión.

### 2.1.6. Algoritmos de Control de Congestión para WSN

La diferencia entre los distintos algoritmos de control de congestión para WSN, depende por un lado de la aplicación particular de WSN para la que están orientados y por otro de la combinación de los mecanismos de detección y control de congestión mencionados en la sección anterior.

El algoritmo propuesto en esta investigación es de tipo reactivo, debido a que ataca la congestión en WSN orientadas a eventos. La tarea fundamental de los algoritmos reactivos de control de congestión es detectar la posible congestión y aplicar mecanismos que ayuden a evitarla o mitigarla, en caso de que haya ocurrido. En la Tabla 2.3 se mencionan los principales algoritmos reactivos de control de congestión. Para cada algoritmo se menciona: la capa de comunicación en que trabaja, los parámetros que utiliza para la detección de congestión, los mecanismos de control de congestión que aplica y sus restricciones.

---

<sup>5</sup> Se determinan mediante la monitorización continua del canal (*overhearing*).

**Tabla 2.3. Algoritmos de Control de Congestión para WSN**

Algoritmo	Capa de Comunicación	Parámetros de Detección de Congestión	Mecanismos de Control de Congestión	Restricciones	Tipo de solución
<i>A New Control Congestion Mechanism for WSN</i> [14]	Capa de transporte	<ul style="list-style-type: none"> <li>▪ Tasa de arribo de paquetes</li> <li>▪ Porcentaje de ocupación de <i>buffer</i></li> </ul>	<ul style="list-style-type: none"> <li>▪ Ajuste de la ventana de transmisión mediante AIMD</li> <li>▪ Asignación de prioridad a cada paquete</li> </ul>	<ul style="list-style-type: none"> <li>▪ No considera la congestión a nivel de enlace</li> <li>▪ Asume que cada nodo tiene la misma tasa de transferencia</li> </ul>	Punto a punto
<i>Extended DCCP Congestion Control Algorithm</i> [15]	Capa de transporte	<ul style="list-style-type: none"> <li>▪ Tasa de arribo de paquetes</li> <li>▪ RTT</li> </ul>	<ul style="list-style-type: none"> <li>▪ Ajuste de la ventana de congestión</li> <li>▪ Ajuste de la tasa de transferencia de los nodos, de acuerdo su tasa de arribo de paquetes</li> <li>▪ Asignación de prioridad a cada paquete</li> </ul>	<ul style="list-style-type: none"> <li>▪ No considera la congestión a nivel de enlace</li> <li>▪ Utiliza retroalimentación de extremo a extremo para ajustar la tasa de transferencia</li> </ul>	Extremo a extremo
<i>Congestion avoidance based on lightweight buffer management in sensor networks</i> [22]	Capa de enlace	<ul style="list-style-type: none"> <li>▪ Porcentaje de ocupación de <i>buffer</i></li> </ul>	<ul style="list-style-type: none"> <li>▪ Contención en el medio, de acuerdo al porcentaje de ocupación de <i>buffer</i></li> </ul>	<ul style="list-style-type: none"> <li>▪ Los nodos nunca utilizan el <i>buffer</i> en su totalidad</li> </ul>	Punto a punto
<i>Decentralized Predictive Congestion Control Protocol (DPCC)</i> [23]	Capa de enlace	<ul style="list-style-type: none"> <li>▪ Porcentaje de ocupación de <i>buffer</i></li> <li>▪ Condiciones del canal</li> <li>▪ VER</li> </ul>	<ul style="list-style-type: none"> <li>▪ Ajuste de la tasa de transferencia</li> <li>▪ Ajuste de los intervalos <i>backoff</i>, de acuerdo a las condiciones del canal y al porcentaje de ocupación de <i>buffer</i></li> <li>▪ Asignación de prioridad a cada paquete</li> </ul>	<ul style="list-style-type: none"> <li>▪ Requiere procesar varias operaciones para realizar el cálculo de la tasa de transferencia y de los intervalos <i>backoff</i></li> </ul>	Punto a punto
<i>Cross-layer protocol for efficient communication in WSN (XLP)</i> [24]	Capas de enlace y de red	<ul style="list-style-type: none"> <li>▪ Energía remanente</li> <li>▪ Porcentaje de ocupación de <i>buffer</i></li> <li>▪ Distancia hacia el <i>sink</i></li> <li>▪ Condiciones del canal</li> </ul>	<ul style="list-style-type: none"> <li>▪ Integra encaminamiento y control de acceso al medio</li> <li>▪ Ajuste de la ventana de contención, de acuerdo a la distancia al <i>sink</i></li> <li>▪ Control de la tasa de generación de paquetes</li> <li>▪ Encaminamiento adaptativo de acuerdo a la ubicación de los nodos</li> </ul>	<ul style="list-style-type: none"> <li>▪ Asume que cada nodo conoce su ubicación (en coordenadas) dentro de la WSN</li> </ul>	Punto a punto

En la Tabla 2.3 se puede observar que debido a que los algoritmos [14] y [15] trabajan únicamente en la capa de transporte, no consideran la congestión a nivel de enlace. Estos algoritmos pueden presentar un desempeño pobre en escenarios de densidad extrema de tráfico, ya que uno de los parámetros de detección de congestión que utilizan es el tiempo de arribo entre paquetes, pero al calcular este tiempo el algoritmo no toma en cuenta los paquetes perdidos debido a las colisiones. Además, en el caso del algoritmo [15] los paquetes de control intercambiados para realizar ajustes en la tasa de transferencia, incrementan el tráfico en las zonas congestionadas. Una ventaja que podemos encontrar en [14] y [15] es que implementan un esquema de prioridad de paquetes, con lo que buscan que el *sink* recolecte información de la mayoría de los eventos detectados.

Como se muestra en la Tabla 2.3, al trabajar en la capa de enlace de datos, los algoritmos [22] y [23] enfrentan la congestión a nivel de enlace. Estos protocolos llevan a cabo un control de acceso al medio basado en CSMA que además toma en cuenta el estado de los *buffers* de los nodos para realizar la contención en el medio; con lo que, además, buscan evitar la congestión a nivel de nodo. Sin embargo, en el caso del algoritmo [22] lo anterior resulta en una reducción del caudal de envío ya que los nodos nunca utilizan sus *buffers* en su totalidad. En lo que respecta al algoritmo [23], al realizar el ajuste de los intervalos *backoff* considera las condiciones del canal y la BER, para ello los nodos se mantienen escuchando de manera continua el canal (*overhearing*), lo que implica un consumo constante de energía, es por ello que los algoritmos basados en *overhearing* no son recomendables para aplicaciones que requieren un tiempo de vida largo; pero pueden ser muy útiles en algunas aplicaciones de WSN orientadas a eventos, en las que una vez que el *sink* ha colectado suficiente información de los eventos de interés detectados, la WSN puede dejar de funcionar. Por ejemplo, en la detección temprana de incendios forestales, la WSN puede ser consumida por el fuego; en el caso de WSN utilizadas para detectar explosivos en campos de batalla una vez que es detectada la ubicación de los explosivos, puede ya no ser necesaria la WSN.

El algoritmo [24] aplica un enfoque *entre capas*, el cual busca evitar y controlar la congestión a nivel de nodo y de enlace. La congestión a nivel de nodo es enfrentada con un protocolo de encaminamiento multitrayectorias adaptativo, en donde se asume que cada nodo conoce su ubicación con respecto al *sink*, y además mantiene un registro del estado (energía remanente, porcentaje de ocupación del *buffer* y distancia al *sink*), de sus vecinos; de acuerdo a estos parámetros cada nodo decide a qué nodo enviar sus paquetes. La congestión a nivel de enlace es tratada con un esquema de acceso al medio, en el que cada nodo decide si participa en la retransmisión de un paquete, esta decisión se toma considerando el estado del nodo y su tasa de arribo. Si varios nodos deciden participar en la retransmisión de un paquete se elige uno mediante su ventana de contención, la cual se ajusta de acuerdo a la distancia que hay al *sink*.

En el diseño de algoritmos de control de congestión para WSN, se deben tomar en cuenta los requerimientos de cada aplicación en particular; por ejemplo en [25] se presenta un esquema de control de congestión para aplicaciones multimedia de WSN, en el cual se requiere un control de congestión para tasas de transferencia altas. El esquema de control de congestión propuesto en este trabajo, cumple con los requerimientos de WSN orientadas a eventos, tales como: tiempo de respuesta, bajo consumo de energía en escenarios no congestionados, informe de todos los eventos detectados por la WSN y balance de carga.

La propuesta de control de congestión fue diseñada a partir de un análisis de tráfico en WSN, desde la perspectiva de “sistemas complejos”, el cual se detalla en el Capítulo 3. Este novedoso enfoque permite analizar el impacto que tienen las interacciones locales de los paquetes, en el comportamiento global del tráfico de la WSN. Mediante este análisis se diseñó un esquema de control de congestión, en el cual a partir de las condiciones locales de tráfico, los nodos de la WSN son capaces de tomar decisiones de encaminamiento, contención en el medio y eliminación de paquetes; estas decisiones están encaminadas a mantener a la WSN operando, de manera emergente y autoorganizada. Con el fin de controlar los 2 niveles de congestión, el protocolo propuesto trabaja con un enfoque *entre capas* y proporciona una solución *punto a punto*, que funciona de manera reactiva, adaptándose a las condiciones de tráfico. La congestión a nivel de enlace es mitigada con un mecanismo de acceso al medio inspirado en el protocolo CSMA que además considera el porcentaje de ocupación de *buffer*. La congestión a nivel de nodo es tratada mediante un esquema de encaminamiento multitrayectorias y adaptativo, en el cual las rutas nuevas se construyen de manera reactiva y *punto a punto*, para ello cada nodo de acuerdo a sus condiciones locales de tráfico decide hacia qué nodo debe enviar sus paquetes, con lo cual también logra cierto de balance de carga. Para evitar que la densidad de tráfico exceda la capacidad de la WSN, el protocolo propuesto contempla un mecanismo de eliminación selectiva de paquetes; además de un mecanismo de equidad en la entrega de información, con lo que se garantiza que el *sink* obtenga información de todos los eventos de interés detectados.

El mecanismo de control de congestión propuesto también considera los efectos adversos de transmisiones en el medio inalámbrico, tales como la interferencia y la atenuación. Es por ello que como parte de la propuesta que se presenta, fue diseñado un modelo de simulación de WSN que toma en cuenta los efectos inalámbricos y escenarios de congestión.



## 2.2 Análisis de WSN desde la Perspectiva de Sistemas Complejos

-“El análisis de ‘sistemas complejos’ es una área interdisciplinaria de investigación que busca explicar cómo una gran cantidad de entidades, relativamente simples, se autoorganizan, sin ningún tipo de control central, en una entidad colectiva que es capaz de crear patrones, usar información y, en algunos casos, evolucionar y aprender” [7]. A partir de este enfoque es posible analizar el comportamiento de colonias de insectos, cerebros, redes de computadoras, etc.

De acuerdo con [7] los “sistemas complejos” consisten en una gran cantidad de entidades llamadas *agentes*, los cuales a partir de un conjunto de reglas simples y sin ningún tipo de control central, son capaces de exhibir un comportamiento elaborado, colectivo y autoorganizado. En lo que respecta al tráfico en WSN, existe una gran cantidad de paquetes de datos, cuyas interacciones locales están basadas en algoritmos que, al diseñarse de manera eficiente, pueden lograr que el tráfico de la WSN exhiba un comportamiento autoorganizado y sofisticado que se adapte a alteraciones en la WSN, tales como congestiones o cambios de topología ocasionados por el fallo de nodos.

Debido a que el estudio de “sistemas complejos” es un área de investigación joven, actualmente es reducido el número de trabajos que realizan un análisis de WSN desde este enfoque. En [8] se presenta un marco de referencia para adaptar los conceptos de “sistemas complejos” a WSN, los principios más importantes que se proponen son:

- Un “sistema complejo” se compone de agentes adaptativos, es decir que pueden interactuar con el ambiente y con otros agentes
- Una WSN puede ser estudiada como un “sistema complejo”, cuyos agentes adaptativos son los nodos de la WSN
- A partir de sus interacciones, los nodos obtienen “experiencia” y con ello “aprendizaje”, de manera que su comportamiento evoluciona, provocando a su vez que la WSN evolucione
- La capacidad de autoorganización de la WSN depende de su evolución
- Cada subsistema de la WSN (nodo o conjunto de nodos) contribuye a la autoorganización de toda la WSN
- De acuerdo a su contribución, cada subsistema de la WSN debe tener un peso de importancia en la autoorganización de la WSN

Mientras que la propuesta de [8] es sólo un marco conceptual para establecer una analogía entre “sistemas complejos” y WSN, en [9] se presenta una metodología para diseñar WSN autoorganizadas, los pasos de esta metodología son:

1. Definir el comportamiento autoorganizado y emergente, que se desea tenga la WSN
2. Diseñar el conjunto de reglas locales que seguirán los nodos de la WSN

3. Determinar los eventos que detonarán cambios en el comportamiento local de los nodos, es decir los eventos que detonarán el comportamiento emergente de la WSN
4. Definir los mecanismos de señalización para la coordinación de los nodos, tales como: campos de gradientes, feromonas digitales, etc.
5. Diseñar mecanismos de retroalimentación para refinar los eventos mencionados en el paso 3

Esta metodología, al igual que el marco conceptual presentado en [8], considera a los nodos de la WSN como agentes del “sistema complejo”, con ello se pretende que los nodos, mediante mecanismos de realimentación, acumulen experiencia, que les permita refinar la manera en que responden a los eventos de la WSN.

En [10] se presenta un marco de referencia para simular WSN y su ambiente de operación, la simulación es realizada mediante el enfoque de *modelado basado en agentes*, es decir que tanto la WSN, como su ambiente de operación son modelados como “sistemas complejos” cuyos agentes son los nodos de la WSN y las entidades que esperan sean detectadas por la WSN. El objetivo de esta propuesta es obtener una simulación de WSN en funcionamiento, es por ello que se hace hincapié en el modelado del ambiente de operación que en el modelado de la WSN.

En esta investigación se propone la idea de analizar el comportamiento del tráfico en WSN como un “sistema complejo” cuyos agentes son los paquetes de datos que transporta la WSN. El motivo de elegir como agentes del “sistema complejo” a los paquetes de datos, en lugar de los nodos de la WSN, es que a diferencia de [8 - 10], este trabajo no persigue un objetivo de modelar y analizar WSN en general (como es el caso de [8 - 10]), sino que el objetivo de la investigación es más específico, ya que se busca proporcionar control de congestión, basándose en el comportamiento autoorganizado del tráfico de la WSN. De esta manera el papel que juegan los nodos, como parte del “sistema complejo” es, por un lado, el de señalizadores que permiten la comunicación indirecta entre los nodos de la WSN y por otro lado, el de mediadores, es decir que regulan el comportamiento entre los paquetes de datos. El enfoque de análisis propuesto está basado en un trabajo previo [13] y será tratado a profundidad en el siguiente capítulo.

Existen otros trabajos que en su análisis de WSN, desde la perspectiva de “sistemas complejos”, consideran a los paquetes de datos como agentes del sistema, por ejemplo en [11] se propone una mejora al protocolo de encaminamiento *directed diffusion* para WSN basadas en peticiones (*query-based*) [2]. Esta propuesta considera a las WSN como “sistemas complejos” compuestos de agentes estacionarios y móviles. Se propone que los nodos estén dotados de un agente estacionario que tiene información del nivel de batería de su nodo y de los nodos de su vecindario; y que además, cuando lo demanda el *sink*, obtiene información de su ambiente. De esta manera para realizar una petición de información a la WSN, el *sink* despliega agentes móviles, que en esencia son paquetes de datos cuyo

objetivo es informar a los nodos fuente de la petición del *sink*, además de coleccionar, transportar y entregar al *sink* la información solicitada. Para transportarse los agentes móviles eligen rutas basándose en la información que obtienen de los agentes estacionarios.

Otro trabajo que considera a los paquetes de datos como agentes de la WSN, es el presentado en [12]. Se trata de un protocolo de encaminamiento inspirado en el comportamiento de parvadas de pájaros, en el que los paquetes de la WSN se transportan en grupos hacia el *sink*, esquivando zonas congestionadas. Para lograr este comportamiento, los nodos mantienen información de su tasa de arribo y la de sus vecinos, así cada nodo envía sus paquetes a otros nodos cuya tasa de arribo se encuentre dentro de un rango que permita mantener “parvadas de paquetes”. Al tratarse de un protocolo de encaminamiento, la propuesta no considera la congestión a nivel de enlace y en su lugar asume la aplicación de un protocolo acceso al medio de tipo CSMA, sin embargo en escenarios de congestión extrema es posible que ocurra congestión a nivel de enlace por la excesiva densidad de paquetes, en este caso el protocolo no ofrece algún esquema de eliminación de paquetes. Esta propuesta considera una topología de red estática y busca controlar la congestión, más no evitarla ya que no aplica mecanismos de prevención de congestión.

El algoritmo de control de congestión propuesto en este trabajo, considera, al igual que el trabajo en [12], a los paquetes de la WSN como agentes de un “sistema complejo”, sin embargo, a diferencia de lo realizado en [12], los nodos no sólo toman decisiones de encaminamiento para cada paquete, sino que también toman decisiones de contención en el medio y eliminación selectiva. Estas decisiones además de estar basadas en las condiciones locales de tráfico de cada nodo, también toman en cuenta la “experiencia” adquirida por el paquete en su viaje a través de la WSN. Para lo lograr esto, el algoritmo propuesto se basa en un análisis de tráfico en WSN desde la perspectiva de “sistemas complejos”, el cual permite obtener métricas del comportamiento “microscópico” del tráfico de la WSN, estas métricas son utilizadas por los nodos para regular el comportamiento entre los paquetes de datos y con ello lograr un comportamiento autoorganizado del tráfico de la WSN. Este análisis es explicado a detalle en el Capítulo 3.



# Capítulo 3

## Análisis de Tráfico en WSN desde la Perspectiva de Sistemas Complejos

---

Este capítulo<sup>1</sup> se divide en 3 secciones. La Sección 3.1 presenta un marco conceptual para analizar el tráfico en WSN desde la perspectiva de “sistemas complejos”. A partir de este marco conceptual, se diseñó un modelo de simulación de WSN que permite simular escenarios de tráfico, sobre los cuales es posible evaluar distintos protocolos de control de congestión para WSN, este modelo es presentado la Sección 3.2. Finalmente, en la Sección 3.3 se explica cómo utilizar el marco conceptual y el modelo de simulación presentados, en el diseño de algoritmo distribuidos de autoorganización de tráfico en WSN.

### 3.1 Marco Conceptual

Este marco conceptual es una herramienta de análisis que permite diseñar algoritmos distribuidos de autoorganización de tráfico en WSN, esto mediante el estudio del tráfico en WSN desde la perspectiva “sistemas complejos”.

Para el diseño de este marco conceptual se tomaron como punto de partida algunos de los conceptos e ideas referentes al estudio de “sistemas complejos” presentados en [26]. En donde se considera que un “sistema complejo” se compone de agentes, siendo los agentes entidades que actúan en su ambiente, y que tienen “metas” y un comportamiento dedicado a alcanzarlas.

Esta investigación propone que **el tráfico de WSN orientadas a eventos, puede ser estudiado como un “sistema complejo”, cuyos agentes son los paquetes de datos que transporta la WSN, los cuales tienen como “meta” llegar al *sink* con el menor retardo posible, dicho retardo está determinado por sus interacciones con otros paquetes y con el medio de comunicación.**

En el análisis de “sistemas complejos”, además de identificar los elementos del sistema es importante identificar de qué manera las interacciones entre los agentes; y las interacciones entre los agentes y su ambiente; afectan el comportamiento global del sistema.

En el caso de WSN, las interacciones entre los paquetes de datos pueden provocar congestión a nivel de enlace y a nivel de nodo. La congestión a nivel de enlace ocasiona pérdida de paquetes por colisiones, mientras que la congestión a nivel de

---

<sup>1</sup> Basado en un trabajo previo, presentado en [13].

nodo aumenta el tiempo de entrega de los paquetes y en casos extremos provoca pérdida de paquetes por desbordamiento de *buffer*.

En lo que respecta a la interacción entre los paquetes de datos y el canal de comunicación, tanto la congestión a nivel de nodo, como la presencia de ruido térmico ocasionan que la calidad del canal se reduzca, provocando con ello pérdidas de paquetes.

Con el propósito de estimar el impacto que tienen las interacciones mencionadas, en el desempeño global de la WSN, proponemos dos métricas que permiten realizar un análisis “microscópico” del tráfico de la WSN; una de ellas mide el grado de satisfacción de un paquete en el cumplimiento de su “meta”, la otra métrica mide la interferencia que ocasiona un paquete en su interacción con un grupo de paquetes. Además proponemos una tercer métrica para medir la satisfacción, es decir el desempeño global de la WSN.

### 3.1.1. Satisfacción de un Paquete

Sea  $p$  un paquete que se transporta a través de una WSN, desde un nodo fuente  $S$  hasta el nodo *sink* de la WSN. Si  $p$  actualmente se encuentra en el nodo  $N$ , siendo  $M$  el nodo inmediato anterior a  $N$ , en la ruta de  $p$  hacia el *sink* y además se sabe que  $N$  es el  $n$ -ésimo nodo en el viaje de  $p$  hacia el *sink*., ver Figura 3.1. Entonces la satisfacción del paquete  $p$  en el cumplimiento de su “meta” es representada con la variable  $\sigma_{p,N} \in [0,1]$  y es calculada por el nodo  $N$ , como se muestra a continuación:

$$\sigma_{p,N} = \begin{cases} \sigma_{p,M} + \frac{\left(1 - \frac{\Delta t}{\Delta t_{max}}\right) - \sigma_{p,M}}{n} & 0 < \Delta t \leq \Delta t_{max}, \\ 0 & \Delta t > \Delta t_{max} \end{cases} \quad (1)$$

Donde  $\Delta t$  es el tiempo que el paquete  $p$  ha permanecido en el *buffer* del nodo  $N$ ;  $\Delta t_{max}$  es el tiempo máximo que el paquete puede permanecer en dicho *buffer*, antes de ser eliminado.

En la Ecuación (1),  $\sigma_{p,N}$  indica el promedio del retardo normalizado que el paquete experimenta en su viaje hacia el *sink*. Por ello, el primer paso para que el nodo  $N$  calcule  $\sigma_{p,N}$  es calcular el retardo normalizado que el paquete  $p$ , experimenta mientras está en el *buffer* del nodo  $N$ . Dicho cálculo se realiza de acuerdo a la Ecuación (2):

$$\|\Delta t\| = \left(1 - \frac{\Delta t}{\Delta t_{max}}\right) \quad (2)$$

Una vez que el nodo  $N$  calcula  $\|\Delta t\|$ , debe calcular la contribución que  $\|\Delta t\|$  aporta a la satisfacción del paquete  $\sigma_{p,N}$ , es decir:

$$\Delta\sigma_{p,N} = \frac{\|\Delta t\| - \sigma_{p,M}}{n} \quad (3)$$

Entonces la Ecuación (1) se puede reescribir en términos de  $\Delta\sigma_{p,N}$ , de la siguiente manera:

$$\sigma_{p,N} = \begin{cases} \sigma_{p,M} + \Delta\sigma_{p,N} & 0 < \Delta t \leq \Delta t_{max}, \\ 0 & \Delta t > \Delta t_{max} \end{cases} \quad (4)$$

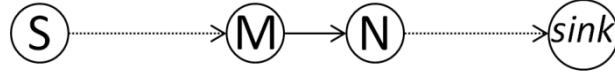
Se puede ver que  $\sigma_{p,N}$  es una función que actualiza el promedio de  $\|\Delta t\|$ , en cada nodo  $N$  por el que pasa  $p$ , en su recorrido del nodo fuente  $S$  al nodo *sink*. Por lo que cuando el nodo fuente  $S$ , envía por primera vez al paquete  $p$ ,  $\sigma_{p,S}$  corresponde a  $\|\Delta t\|$ , debido a que en ese caso  $n=1$  y  $\sigma_{p,M}=0$ , esto es:

$$\sigma_{p,S} = \begin{cases} \|\Delta t\| & 0 < \Delta t \leq \Delta t_{max}, \\ 0 & \Delta t > \Delta t_{max} \end{cases} \quad (5)$$

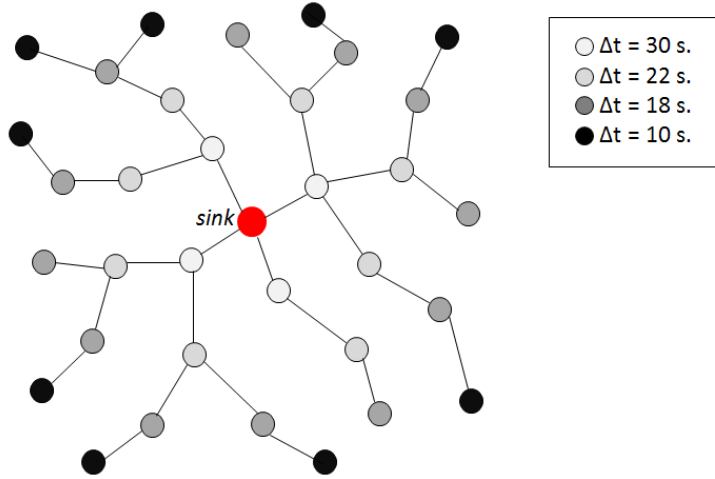
En lo que respecta al cálculo de  $\Delta t$ , el nodo  $N$  debe registrar el tiempo en que coloca al paquete  $p$  en su *buffer*. De manera que el nodo  $N$  pueda estimar  $\Delta t$  y así etiquete al paquete  $p$  con su  $\sigma_{p,N}$  correspondiente, antes de reenviarlo

El siguiente parámetro que el nodo  $N$  debe conocer para calcular  $\sigma_{p,N}$ , es  $\Delta t_{max}$ , este parámetro es propio del nodo  $N$ . Los criterios para establecer  $\Delta t_{max}$  pueden variar de acuerdo al tipo de comportamiento que se desee obtener en el tráfico de la WSN. Por ejemplo,  $\Delta t_{max}$  se puede asignar de acuerdo a la tasa promedio de arribo de paquetes y/o a la ubicación de los nodos de la WSN, en este caso puede resultar útil que el  $\Delta t_{max}$  sea mayor en los nodos cercanos al *sink* que en los nodos ubicados en zonas alejadas del *sink*, ver Figura 3.2. De esta manera se asegura que en caso de tener alta densidad de tráfico en la WSN: 1) los paquetes provenientes de zonas alejadas del *sink*, no acumulen grandes retardos y 2) los nodos en zonas cercanas al *sink* mantengan sus *buffers* a su máxima capacidad, con el fin de entregar la mayor cantidad posible de paquetes.

Al medir la satisfacción de los paquetes de datos, los nodos obtienen un estimado del comportamiento del tráfico local, como se verá en el Capítulo 4, esta métrica es de utilidad en el diseño de algoritmos que busquen obtener un comportamiento global y autoorganizado del tráfico de la WSN.



**Figura 3.1.** Ruta seguida por el paquete  $p$  en su viaje del nodo fuente  $S$  hacia el  $sink$



**Figura 3.2.** Asignación de  $\Delta t_{max}$  de acuerdo a la ubicación de los nodos con respecto al  $sink$

### 3.1.2. Interferencia entre un Paquete y un Grupo de Paquetes

Sea un grupo de  $m$  paquetes que interactúan en un nodo  $N$  y sea  $p$  un paquete que está contenido en el grupo de  $m$  paquetes. La interferencia entre el paquete  $p$  y el resto del grupo de paquetes con los que interactúa, indica el impacto que tiene el paquete  $p$  en la satisfacción del grupo de  $m-1$  paquetes y se calcula de la siguiente manera:

$$\phi_p = -\Delta\sigma_{p,N} - \sum_{i=1}^m \frac{\Delta\sigma_{i,N}}{m-1} \quad (6)$$

Para calcular  $\phi_p$  es necesario acotar el grupo de paquetes con los que  $p$  interactúa. En este marco conceptual se consideran sólo las interacciones que se dan entre los paquetes que pasan por un mismo nodo. Entonces, decimos que un paquete  $p$  interactúa con un grupo de paquetes, cuando el comportamiento de  $p$  “afecta” a la satisfacción de dicho grupo de paquetes. Dependiendo del tipo de algoritmo que se desea diseñar, se pueden utilizar distintos enfoques para definir cuando un paquete  $p$  “afecta” a la satisfacción de un grupo de paquetes. Por ejemplo, se puede decir existe interacción entre un grupo de paquetes de datos, cuando comparten un *buffer* o cuando, dentro de determinado rango de tiempo, pasan por un mismo nodo.

La Ecuación (7) mide la interferencia entre un paquete  $p$  y un grupo de paquetes que interactúan en un nodo  $N$ . Dicha interferencia puede ser positiva, negativa o neutral, de acuerdo a las siguientes premisas:



- Cuando el incremento en la satisfacción de un paquete  $p$  causa un decremento (que es mayor que el incremento) en la satisfacción promedio de un grupo de  $m-1$  paquetes, se dice que existe **fricción** entre dichos paquetes
- Cuando el incremento en la satisfacción del paquete  $p$  causa un incremento en la satisfacción promedio de un grupo de  $m-1$  paquetes, entonces existe **sinergia** (interferencia “positiva”) entre el paquete  $p$  y el grupo de  $m-1$  paquetes
- En cualquier otro caso, se dice que existe **interferencia neutral** entre el paquete  $p$  y el grupo de  $m-1$  paquetes

### 3.1.3. Satisfacción de la WSN

Sea una WSN que transporta  $n$  paquetes en un instante de tiempo  $t$ , se define a la satisfacción de la WSN en el tiempo  $t$ , como el promedio de la satisfacción de los  $n$  paquetes de datos que transporta en dicho instante, es decir que:

$$\sigma_{WSN,t} = \sum_{i=1}^n \frac{\sigma_{i,N}}{n} \quad (7)$$

Para calcular  $\sigma_{WSN,t}$  se requiere coleccionar información de la satisfacción de todos los paquetes  $i$  que están siendo transportados en el instante  $t$ .

La utilidad de  $\sigma_{WSN,t}$  se encuentra durante el proceso de diseño de algoritmos de autoorganización de tráfico, los cuales deben de buscar maximizar  $\sigma_{WSN,t}$ , a partir la reducción de la fricción entre paquetes.

## 3.2 Modelo de Simulación de WSN, desde la Perspectiva de Sistemas Complejos

Este modelo de simulación hace posible obtener representaciones, desde la perspectiva de “sistemas complejos”, de la operación de WSN orientadas a eventos. Se desarrolló a partir del marco conceptual presentado en la Sección 3.1 y forma parte de la propuesta de este trabajo, ya que fue utilizado en el diseño, prueba y evaluación del esquema de control de congestión propuesto.

Las funcionalidades y características de las WSN son agrupadas en 4 módulos:

1. Configuración de la WSN
2. Configuración de los nodos
3. Simulación de efectos inalámbricos
4. Evaluación de desempeño

El módulo 1 permite obtener la representación de distintas configuraciones de WSN. En el módulo 2 se especifica la configuración básica del funcionamiento de los nodos de la WSN, dicha configuración está basada en el marco conceptual presentado en la Sección 3.1. Este módulo puede ser manipulado por el usuario, con la finalidad

de diseñar algoritmos distribuidos y autoorganizados de encaminamiento y/o control de congestión. Mediante el módulo 3 se da mayor realismo a la representación de WSN, ya que los efectos del medio inalámbrico tienen gran impacto en la operación de WSN, esto aporta veracidad en la evaluación de desempeño de los algoritmos que se deseen evaluar. Finalmente el módulo 4 provee de funcionalidades para la evaluación de desempeño de algoritmos de encaminamiento y control de congestión, este módulo hace uso de las métricas propuestas en el marco conceptual de la Sección 3.1 y de métricas clásicas de evaluación de desempeño de redes de comunicaciones.

Una vez que se realiza la especificación de estos módulos puede elegirse algún *software* de simulación para su implementación. En las siguientes secciones se describe la manera en que deben especificarse los parámetros de cada módulo.

### 3.2.1. Configuración de la WSN

En este módulo se deben establecer 3 tipos de parámetros de configuración: parámetros de la WSN, de los nodos y de operación.

Los parámetros de la WSN que son configurados son:

- Escala espacial. Indica la longitud de área ocupada por la WSN, así como su forma (cuadrada, circular o irregular).
- Escala temporal. Determina el tiempo total de la simulación, así como los intervalos de tiempo en que se medirá  $\sigma_{WSN,t}$ .
- Tipos de nodos. En este modelo se consideran 3 tipos de nodos: 1) *sink*; 2) nodos fuente, que generan y reenvían paquetes; y 3) nodos intermedios, que sólo reenvían paquetes. El usuario debe definir la cantidad de nodos que serán desplegados, así como cuántos de ellos serán nodos fuente.
- Topología de la WSN. Las topologías se deben generar de manera aleatoria y determinística.

En lo que respecta a los nodos de la WSN se deben establecer los siguientes parámetros:

- Potencia de transmisión.
- Tasa de transferencia. Debido a que el modelo desprecia la velocidad de propagación en el medio inalámbrico, la tasa de transferencia permite estimar el tiempo en que un paquete es enviado de un nodo a otro.
- Tamaño del *buffer*. Con el propósito de dar mayor realismo a las simulaciones, este parámetro es asignado de acuerdo a una función de distribución de probabilidad (PDF, por sus siglas en inglés) Normal. Por otro lado, debido a que el tamaño del *buffer* tiene gran impacto en la congestión a nivel de nodo, los nodos puedan tomar distintos valores, entre los cuales se consideran *buffers* de tamaño cero e infinito.

- Cobertura. El radio de cobertura se establece mediante una PDF Normal y está en función de la potencia de transmisión, esto se detalla en la siguiente sección.
- Grado promedio de nodo. Este parámetro define el promedio de conexiones que puede tener el nodo. Se propone establecer el grado promedio de nodo mediante una PDF Normal.
- Esquema de consumo de energía. El consumo de energía en la recepción de datos y el procesamiento<sup>2</sup> es despreciado, de manera que sólo se toma en cuenta el consumo de energía por transmisión de datos. Este consumo de energía se puede simular permitiendo que el usuario elija el porcentaje de batería que gasta cada nodo en la transmisión de datos.

Los parámetros de operación que se toman en cuenta en este modelo son: potencia de transmisión, número de errores de bit tolerados por paquete, tamaño del paquete.

### 3.2.2. Configuración de los Nodos

En este módulo se consideran las siguientes funcionalidades:

- Generación de paquetes de datos
- Encaminamiento
- Control de acceso al medio
- Control de congestión

La generación de paquetes de datos se llevó a cabo de 3 formas distintas: de manera continua, mediante una PDF de Poisson y mediante una PDF Normal. En cada caso el usuario debe establecer los parámetros correspondientes, tales como la velocidad promedio de generación de paquete.

Los módulos de encaminamiento, control de acceso al medio y control de congestión deben tener una configuración básica, basada en el marco conceptual de la Sección 3.1. De manera que el usuario pueda manipular estos módulos, con la finalidad de diseñar algoritmos distribuidos que provean autoorganización de tráfico.

Los parámetros de encaminamiento se establecen mediante el conocido algoritmo de *Propagación de la Información* [26] (PI, por sus siglas en inglés). Este algoritmo permite formar un árbol con los nodos de un grafo; en el caso de WSN, la raíz del árbol es el *sink*. El algoritmo se ejecuta una sola vez antes de echar a andar la WSN. Cuando la ejecución del algoritmo termina cada nodo conoce su nivel dentro del árbol, el *sink* por ser la raíz tiene nivel 0.

---

<sup>2</sup> Debido a que se consideran aplicaciones en las cuales los nodos sólo realizan las tareas de procesamiento que demanda los protocolos de comunicación.

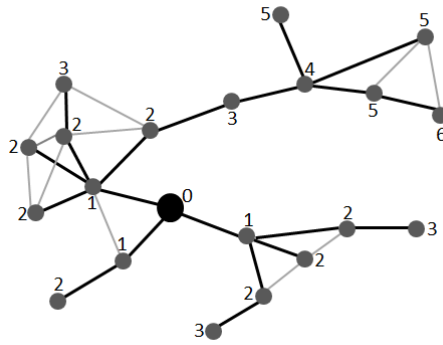
Como es sabido, una WSN actúa como una red *ad hoc* transportando todos los datos generados dentro de la WSN hacia el *sink*. Por lo tanto, tenemos que el nivel de cada nodo equivale a la distancia, en saltos, que tiene hacia el *sink*. En la Figura 3.3 se muestra un ejemplo del árbol que se forma al ejecutar el algoritmo PI sobre un grafo, la raíz del árbol está representada por el nodo más grande, las ramas del árbol obtenido están resaltadas, también se muestra el nivel de cada nodo.

En una configuración básica de encaminamiento, cada nodo envía sus paquetes de datos, a sus vecinos que tengan un nivel menor al suyo, sin embargo, el usuario puede manipular este módulo para diseñar esquemas de encaminamiento multitrayectorias que además del nivel de nodo, tomen en cuenta la interferencia de los paquetes, en la elección de rutas.

Para el módulo de control de acceso al medio se usa como configuración básica un esquema basado en CSMA. En este módulo, al igual que en el módulo de encaminamiento, el usuario puede modificar algunos parámetros, como la duración de los intervalos *backoff*, para así introducir la información de interferencia de los paquetes a las decisiones de contención en el medio.

El módulo de control de congestión será el encargado de calcular la interferencia y satisfacción de los paquetes que reenvía el nodo, por lo que en este módulo se deben calcular todas las métricas definidas en la sección 3.1 y 3.3. Como ya se mencionó estas métricas tendrán injerencia en las decisiones de encaminamiento y de acceso al medio. Adicionalmente, este módulo puede utilizarlas para aplicar esquemas de eliminación selectiva de paquetes, que permitan a la WSN autoregular su densidad de paquetes de datos.

Al tomar en cuenta la interferencia de los paquetes en la toma de decisiones de encaminamiento, acceso al medio y eliminación selectiva de paquetes, se debe considerar que la efectividad de estas decisiones depende en gran parte de la escala de tiempo elegida para medir los cambios de satisfacción de los paquetes. Es sabido que muchos mecanismos autoorganizados mantienen una fuerte dependencia con la escala temporal en la que se basan para tomar las decisiones adaptativas [26].



**Figura 3.3.** Árbol obtenido al ejecutar el algoritmo PI sobre un grafo

### 3.2.3. Simulación de Efectos Inalámbricos

Los efectos inalámbricos se simulan considerando un modelo de pérdidas en el espacio libre, el cual permite calcular la atenuación, con respecto a la distancia, en la señal recibida [27]. Esto se muestra en la siguiente ecuación:

$$P_r = \frac{P_t \lambda^2}{(4\pi d)^2} \quad (8)$$

Donde  $P_r$  es la potencia de la señal recibida, en *watts*;  $P_t$  es la potencia de la señal transmitida, en *watts*;  $\lambda$  es la longitud de onda, en metros; y  $d$  es la distancia entre el transmisor y receptor, en metros.

Este modelo también contempla los efectos del ruido térmico, para ello el usuario debe definir a que temperatura se encuentra operando el sistema. Con la siguiente ecuación se calcula la densidad espectral del ruido térmico ( $N_0$ ) [27].

$$N_0 = kT [W/Hz] \quad (9)$$

Donde  $k$  es la constante de Boltzmann ( $1.38 \times 10^{-23}$  J/K); y  $T$  es la temperatura, en grados Kelvin.

La capacidad máxima del canal [27], se calcula considerando un ancho de banda de  $8kHz$ , de siguiente manera:

$$C = B \log_2(1 + S/N) [b/s] \quad (10)$$

Donde  $B$  es el ancho de banda, en *hertz*;  $S$  es la potencia de la señal recibida, en *watts*; y  $N$  es la potencia del ruido de la señal recibida, que se calcula con la siguiente ecuación:

$$N = N_0 B [w] \quad (11)$$

A partir de la Ecuación (10) es posible calcular el valor de la relación señal a ruido normalizada  $(E_b/N_0)^3$ , de la siguiente manera [27]:

$$\frac{E_b}{N_0} = \frac{B}{c} (2^{C/B} - 1) \quad (12)$$

A partir de la  $E_b/N_0$  y considerando el esquema modulación por desplazamiento de fase de dos símbolos<sup>4</sup> (BPSK, por sus siglas en inglés), obtenemos la BER.

De acuerdo a los requerimientos de la aplicación, específicamente la tolerancia de errores de bits, obtenemos la probabilidad de error por paquete (PER). Con esta probabilidad podemos simular la pérdida de paquetes por efectos inalámbricos.

$$PER = 1 - \sum_{i=0}^n \binom{p}{i} [BER^i (1 - BER)^{p-i}] \quad (13)$$

Donde  $n$  es el número de errores de bit tolerados por la aplicación y  $p$  es el tamaño, en bits, del paquete.

La conectividad de los nodos es otro aspecto que se ve afectado por efectos inalámbricos, ya que aunque las características de los nodos sean las mismas, no todos tendrán la misma conectividad. Para simular este efecto el usuario elige un valor promedio de conectividad, así mediante una PDF Normal, cuya media será el valor que el usuario elige, se asignará a cada nodo un valor de cobertura.

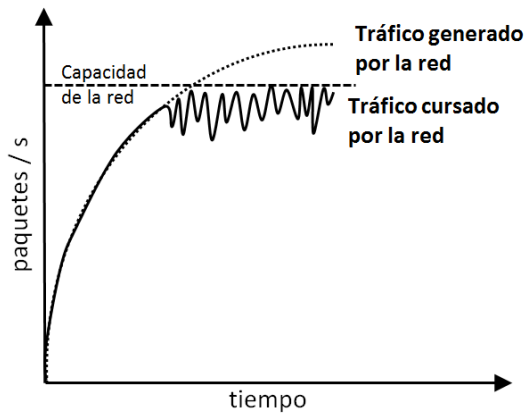
### 3.2.4. Evaluación de Desempeño

En este módulo se evalúa la efectividad de los algoritmos que se diseñen, para ello se utilizan métricas tradicionales de evaluación de WSN, como son: el caudal de envío, tiempo de almacenamiento en *buffer*, tiempo de vida de la red y porcentaje de paquetes perdidos.

Se pueden realizar ajustes en los parámetros de los algoritmos diseñados, esto con el propósito de refinarlos. Dichos ajustes deben estar encaminados a maximizar  $\sigma_{WSN,t}$ . Al medir los cambios en  $\sigma_{WSN,t}$  hay que considerar que puede haber instantes en que el valor de  $\sigma_{WSN,t}$  sea bajo (mínimos locales), pero al final el promedio de  $\sigma_{WSN,t}$  sea alto (máximo local), en otras palabras: “a corto plazo puede haber sacrificio, para a largo plazo cosechar recompensas” [26]. Esto se ilustra en la Figura 3.4.

<sup>3</sup> Es decir la relación de energía por bit ( $E_b$ ) respecto a la densidad espectral de ruido ( $N_0$ ).

<sup>4</sup> Se eligió el esquema de modulación BPSK por ser el esquema que requiere menor energía por bit ( $E_b$ ).



**Figura 3.4.** Comportamiento del tráfico generado y cursado por la red.

### 3.3 Diseño de Algoritmos Distribuidos para la Autoorganización de Tráfico en WSN

Para lograr su “meta” los paquetes de datos se transportan a través de la WSN, siguiendo una serie de reglas definidas principalmente en los protocolos de encaminamiento y MAC. Por lo cual, en el diseño de algoritmos que busquen obtener un comportamiento autoorganizado en el tráfico de la WSN, se deben definir reglas de encaminamiento y acceso al medio que tengan los siguientes objetivos:

- Reducir la fricción entre los paquetes de datos, para así incrementar la satisfacción de la WSN.
- Maximizar  $\sigma_{WSN,t}$ , lo cual implica limitar el comportamiento de los paquetes de datos que reducen otras  $\sigma_{p,N}$ 's mientras se preserva su funcionalidad, y además promover la sinergia entre los paquetes de datos
- Restringir o modificar el comportamiento de los paquetes de datos para que sus “metas” puedan ser alcanzadas, con la finalidad que bloqueen lo menos posible las “metas” de los paquetes con los que interactúan

Se tiene entonces que los nodos de la WSN, al aplicar un algoritmo autoorganizado, actúan como mediadores del sistema [26], regulando el comportamiento de los paquetes del sistema para minimizar la fricción, de modo que se maximice la cooperación y la sinergia.





# Capítulo 4

## Control de Congestión para WSN Orientadas a Eventos

---

En este capítulo se presenta un novedoso esquema de control de congestión para WSN orientadas a eventos, el cual se basa en el marco conceptual presentado en el Capítulo 3. Este esquema de control de congestión tiene 2 objetivos: 1) maximizar  $\sigma_{WSN,t}$ , reduciendo la fricción entre los paquetes de datos; y 2) mantener el tráfico de paquetes, dentro de niveles óptimos de operación. Para cumplir con estos objetivos se diseñó un esquema de encaminamiento basado en la interferencia entre paquetes; así como un esquema de acceso al medio, basado en CSMA que toma en cuenta el espacio remanente en *buffer*; y un mecanismo de eliminación selectiva de paquetes, basado en prioridad por ubicación.

El esquema de control de congestión propuesto, consta de 4 fases de operación: 1) inicialización, 2) procesamiento 3) encaminamiento y 4) transmisión. La primera fase se ejecuta una sola vez en toda la WSN y sirve para inicializar los parámetros del algoritmo. La segunda fase se ejecuta cuando un paquete es recibido por un nodo, esta fase tiene como objetivo registrar los parámetros del paquete y decidir si debe ser enviado o almacenado en *buffer*. En la tercera fase, los nodos toman decisiones de encaminamiento, contención en el medio y, en caso de ser necesario, eliminación selectiva; tomando en consideración la información de interferencia entre paquetes. En la cuarta fase, los nodos aplican un mecanismo de acceso al medio basado en CSMA, el cual toma en cuenta el espacio remanente en *buffer*.

En las siguientes secciones se describen las 4 fases de operación del esquema de control de congestión propuesto y finalmente se presenta una discusión acerca de esta propuesta.

### 4.1 Fase de Inicialización

Esta fase se ejecuta una sola vez, al inicializarse la WSN y sirve para establecer los parámetros del esquema de control de congestión. Estos parámetros son descritos en el Capítulo 3.

#### 4.1.1. Parámetros de Encaminamiento

Durante la inicialización, se utiliza el algoritmo PI [26] para crear un árbol a partir de los nodos de la WSN. En el árbol construido, el *sink* es la raíz y se le asigna el nivel 0. A cada uno de los demás nodos, también se les asigna un nivel, así como el

identificador de su nodo padre. Además, los nodos mantienen un registro del nivel que ocupan sus nodos vecinos.

Los parámetros del esquema de control de congestión; tales como: longitud de grupo, tiempo máximo de espera en el *buffer* y tiempo de vida; son asignados en cada nodo de acuerdo a su distancia, en saltos, al *sink*, por lo que la WSN es dividida en 3 zonas de operación<sup>1</sup>. Cada zona tiene distintos parámetros de operación, para asignarle a cada nodo una zona, una vez que se ha creado el árbol, se ejecutan los siguientes pasos:

1. Los nodo hoja<sup>2</sup>, calculan los límites de cada zona de acuerdo a las siguientes relaciones:

- **zona 1** =  $\{N: N \text{ es un nodo cuyo nivel es } n \text{ y además } 0 < n < \lfloor \frac{L}{3} \rfloor\}$
- **zona 2** =  $\{N: N \text{ es un nodo cuyo nivel es } n \text{ y además } \lfloor \frac{L}{3} \rfloor \leq n < \lfloor \frac{2L}{3} \rfloor\}$
- **zona 3** =  $\{N: N \text{ es un nodo cuyo nivel es } n \text{ y además } \lfloor \frac{2L}{3} \rfloor \leq n\}$

Donde  $L$  es el nivel del nodo hoja que está realizando el cálculo.

2. Los nodos hoja actualizan su atributo zona a 3 y comienzan el envío de paquetes de configuración de zona. Para ello, cada nodo hoja envía a su nodo padre un paquete que contiene la información del límite de las zonas, es decir  $\lfloor \frac{L}{3} \rfloor$  y  $\lfloor \frac{2L}{3} \rfloor$ .
3. Los nodos al recibir el paquete de actualización de zona, actualizan su atributo zona, de acuerdo a las relaciones definidas en el paso 1, y reenvían el paquete a su nodo padre, para continuar con la configuración de zonas. Los nodos que tengan más de un hijo deben esperar el paquete de configuración de zona de todos sus hijos. Para continuar con el proceso de actualización de zonas, elegirán a aquel paquete cuyos rangos de zona sean mayores.

Para su envío, los paquetes se organizan en grupos, cuyo tamaño varía en cada zona de operación, de acuerdo a la siguiente ecuación:

$$l_g = \begin{cases} \lfloor \frac{l_b}{3} \rfloor & \forall n \in \text{zona 1,} \\ \lfloor \frac{2l_b}{3} \rfloor & \forall n \in \text{zona 2,} \\ l_b & \forall n \in \text{zona 3,} \end{cases} \quad (13)$$

Donde  $l_g$  es la longitud del grupo de paquetes, en bits; y  $l_b$  es la longitud del *buffer* del nodo  $n$ , en bits.

---

<sup>1</sup> Se ha propuesto arbitrariamente la creación de 3 zonas de operación, sin embargo el número de zonas de operación puede variar, e incluso los parámetros pueden establecerse de acuerdo al nivel de los nodos

<sup>2</sup> Los nodos hoja de un árbol, son aquellos nodos que no tienen nodos hijos

Finalmente, se debe configurar el tiempo de vida (TTL, por sus siglas en inglés) en cada paquete. Este parámetro es configurado por el nodo fuente que crea al paquete, y se asigna de acuerdo a la zona a la que pertenece el nodo fuente, como se muestra en la siguiente ecuación:

$$TTL = \begin{cases} \left\lfloor \frac{l_b}{3} \right\rfloor & \forall n \in \text{zona 1}, \\ \left\lfloor \frac{2l_b}{3} \right\rfloor & \forall n \in \text{zona 2}, \\ l_b & \forall n \in \text{zona 3}, \end{cases} \quad (14)$$

#### 4.1.2. Tiempo Máximo de Espera en *Buffer* ( $\Delta t_{max}$ )

Cada nodo tiene definido un  $\Delta t_{max}$  que corresponde al tiempo máximo que un paquete puede mantenerse en el *buffer* de un nodo. En esta propuesta  $\Delta t_{max}$  se asigna de acuerdo de la zona de cada nodo, de la siguiente manera:

$$\Delta t_{max} = \begin{cases} \frac{l_g - l_p}{v_{tx}} & \forall n \in \text{zona 1}, \\ \frac{l_g}{v_{tx}} & \forall n \in \text{zona 2}, \\ \frac{2l_g - l_p}{v_{tx}} & \forall n \in \text{zona 3}, \end{cases} \quad (15)$$

Donde  $l_p$  es la longitud del paquete, en bits;  $l_p < l_g$ ; y  $v_{tx}$  es la velocidad de transmisión del nodo  $n$ , en b/s.

A partir de las Ecuaciones (13) y (15) vemos que tanto el valor de  $l_p$ , como el valor de  $\Delta t_{max}$  varían de acuerdo a la ubicación del nodo, siendo mayor en los nodos más cercanos al *sink*, es decir en los nodos que pertenecen a la zona 1, y menor en las zonas 2 y 3. De esta manera, se busca que en escenarios de alta densidad de tráfico: 1) los nodos que están en zonas cercanas al *sink* mantengan sus *buffers* a su máxima capacidad, con el fin de entregar la mayor cantidad posible de paquetes y 2) los paquetes provenientes de zonas alejadas del *sink*, no acumulen grandes retardos.

#### 4.1.3. Satisfacción de un Paquete ( $\sigma_{p,N}$ )

Cada paquete de datos debe mantener un control tanto de su satisfacción, ( $\sigma_{p,N}$ ), como de los cambios que se van experimentando ( $\Delta\sigma_{p,N}$ ) en ésta.

Inicialmente, cuando un nodo fuente crea un paquete  $p$ , etiqueta a dicho paquete con sus valores de  $\Delta\sigma_{p,N}$  y  $\sigma_{p,N}$ ; estos valores son calculados mediante las Ecuaciones (2) y (5), respectivamente. Cada nodo por el que pasa  $p$ , actualiza los valores de  $\Delta\sigma_{p,N}$  y  $\sigma_{p,N}$ . Para ello cada nodo de la WSN, mantiene un *arreglo de satisfacciones*, en donde registra el valor  $\Delta\sigma_{p,N}$  de los paquetes que tiene en su *buffer*. De esta manera

para que un nodo  $N$  actualice los valores de  $\Delta\sigma_{p,N}$  y  $\sigma_{p,N}$  de un paquete  $p$ , debe ejecutar los siguientes pasos:

1. Antes de almacenar a  $p$  en su buffer,  $N$  registra en su arreglo de satisfacciones, el valor de  $\sigma_{p,M}$  que está en la cabecera del paquete  $p$ .
2. Cuando  $N$  saca a  $p$  de su buffer, actualiza la cabecera de  $p$ , con los nuevos valores de  $\Delta\sigma_{p,N}$  y  $\sigma_{p,N}$ , los cuales calcula mediante las Ecuaciones (3) y (4), respectivamente.

#### 4.1.4. Interferencia de un Paquete ( $\phi_p$ )

La interferencia entre paquetes es un parámetro importante, ya que a partir de éste los nodos toman decisiones de encaminamiento, contención en el medio y eliminación selectiva. De acuerdo al contexto de este esquema de control de congestión, se dice que un grupo de paquetes interactúan, cuando pertenecen al mismo grupo. Entonces cuando un nodo reúne en su *buffer* la cantidad de paquetes necesarios para formar un grupo, Ecuación (13), dicho nodo debe calcular la interferencia de los paquetes del grupo.

La Ecuación (6) permite calcular la interferencia entre un paquete  $p$  y un grupo de paquetes que interactúan en un nodo  $N$ . Para calcular la interferencia que ocasiona un paquete  $p$  a un grupo de  $l_g - 1$  paquetes, se puede utilizar la Ecuación (6) reescrita de la siguiente forma:

$$\phi_p = -\Delta\sigma_{p,N} - \sum_{i=1}^{l_g-1} \frac{\Delta\sigma_{i,N}}{l_g-1} \quad (17)$$

Como se menciona en el Capítulo 3, la interferencia puede ser positiva, negativa o neutral. De acuerdo a los siguientes casos:

**Caso 1.** Sea un grupo de  $l_g$  paquetes que interactúan en un nodo  $N$ , se dice que existe **sinergia** entre un paquete  $p$  del grupo y el resto de  $l_g - 1$  paquetes de su grupo, si se cumplen las siguientes condiciones:

1.  $\Delta\sigma_{p,N}$  es positiva
2.  $\sum_{i=1}^{l_g} \frac{\Delta\sigma_{i,N}}{l_g-1}$  es positiva

**Caso 2.** Sea un grupo de  $l_g$  paquetes que interactúan en un nodo  $N$ , se dice que existe **fricción** entre un paquete  $p$  del grupo y el resto de  $l_g - 1$  paquetes de su grupo, si se cumplen las siguientes condiciones:

1.  $\Delta\sigma_{p,N}$  es positiva
2.  $\sum_{i=1}^{l_g} \frac{\Delta\sigma_{i,N}}{l_g-1}$  es negativa

$$3. \quad |\Delta\sigma_{p,N}| > \left| \sum_{i=1}^{l_g} \frac{\Delta\sigma_{i,N}}{l_g-1} \right|$$

**Caso 3.** Sea un grupo de  $l_g$  paquetes que interactúan en un nodo  $N$ , se dice que existe **interferencia neutral** entre un paquete  $p$  del grupo y el resto de  $l_g - 1$  paquetes de su grupo, cuando el paquete  $p$  no afecta ni positiva, ni negativamente el cumplimiento de las metas del resto de  $l_g - 1$  paquetes de su grupo. Es ocurre cuando se presentan condiciones distintas a las mencionadas en los casos 1 y 2.

## 4.2 Fase de Procesamiento

Esta es la fase previa al encaminamiento de paquetes, por lo que en esta fase se decide si un paquete debe ser transmitido, almacenado en *buffer* o eliminado.

Cuándo un paquete de datos  $p$  arriba a un nodo  $N$ , el nodo  $N$  ejecuta los siguientes pasos:

1. Reduce en una unidad, el valor de TTL del paquete  $p$ 
  - a. Si  $TTL = 0$ , elimina el paquete y continúa a la espera de más paquetes
2. Registra en su *arreglo de satisfacciones*, los valores de  $\Delta\sigma_{p,N}$  y  $\sigma_{p,N}$  del paquete  $p$ , así como el tiempo en que se cumplirá el  $\Delta t_{max}$  para dicho paquete
3. Revisa si tiene en su *buffer* la cantidad necesaria de paquetes, para completar un grupo de tamaño  $l_g$ 
  - a. Si ha reunido en su *buffer* los paquetes necesarios para completar un grupo de tamaño  $l_g$ , continúa con la Fase de Encaminamiento
  - b. Si aún no ha reunido los paquetes necesarios para completar un grupo de tamaño  $l_g$ , coloca a  $p$  en el *buffer* y continúa a la espera de más paquetes<sup>3</sup>

## 4.3 Fase de Encaminamiento

En esta fase se aplica un esquema de encaminamiento adaptativo, basado en la interferencia de los paquetes. Este esquema contempla dos tipos de encaminamiento, vertical y horizontal. En el encaminamiento vertical, los paquetes son enviados a nodos cuyo nivel es menor al nodo que los envía. Por el contrario, en el encaminamiento horizontal, los paquetes son enviados a nodos cuyo nivel es igual al nodo que los envía.

Cuando en la Fase de Procesamiento, el nodo  $N$  decide que un grupo de paquetes está listo para ser enviado, entonces el nodo  $N$  aplica los siguientes pasos:

---

<sup>3</sup> Mientras un nodo se encuentre a la espera de paquetes, debe revisar periódicamente si algunos de los paquetes que mantiene almacenados en su *buffer*, están por completar su  $\Delta t_{max}$ . Los paquetes que estén en esa situación deben pasar a la Fase de Encaminamiento.

1. Actualiza los valores de  $\Delta\sigma_{p,N}$  y  $\sigma_{p,N}$ , de todos los paquetes del grupo
2. Revisa si en el grupo hay paquetes con  $\Delta\sigma_{p,N} < 0$ 
  - a. Si no es así, todos los paquetes del grupo pasan a la Fase de Transmisión para ser encaminados verticalmente
  - b. Sí es así, por cada paquete con  $\Delta\sigma_{p,N} > 0$ , revisa si se cumple  $|\Delta\sigma_{p,N}| > \left| \sum_{i=1}^{l_g} \frac{\Delta\sigma_{i,N}}{l_g-1} \right|$ . Los paquetes que no cumplan con esta condición junto con los paquetes con  $\Delta\sigma_{p,N} < 0$ , pasan a la Fase de Transmisión para ser encaminados verticalmente. Los paquetes restantes pasan a la Fase de Transmisión para ser encaminados horizontalmente

#### 4.4 Fase de Transmisión

En esta fase se aplica un esquema de acceso al medio, basado en CSMA. Este esquema contempla un ajuste adaptativo de los intervalos *backoff*, en donde los intervalos *backoff* dependen de la zona de ubicación de cada nodo.

Cuando en la Fase de Encaminamiento, el nodo  $N$  decide que un grupo de paquetes está listo para ser transmitido, entonces el nodo  $N$  aplica los siguientes pasos:

1. Escucha el canal para saber si está ocupado
  - a. Si el canal está ocupado, espera la duración de un intervalo *backoff* y después regresa al Paso 1
2. Envía a sus vecinos, ya sea de igual o menor nivel (dependiendo si va a realizar encaminamiento horizontal o vertical), un paquete de petición de transmisión, indicando el tamaño de la ráfaga de paquetes que desea transmitir
  - a. Los nodos que reciben el mensaje, deben responder de acuerdo a su espacio en *buffer*, si desean participar en la comunicación y cuantos paquetes pueden recibir
3. De acuerdo a la respuesta de sus vecinos, realiza el envío de paquetes, distribuyendo la carga de paquetes entre la mayor cantidad de nodos posible. La transmisión comienza por los paquetes que tienen menor valor de  $\Delta\sigma_{p,N}$
4. Revisa si aún tiene paquetes por transmitir
  - a. Si es así, regresa al Paso 1 para transmitirlos mediante encaminamiento horizontal
  - b. Si no es así, revisa el estado de su batería, si tiene menos del 10% de su batería original, envía un mensaje de aviso a todos sus vecinos para que lo eliminen de la tabla de encaminamiento

## 4.5 Discusión

Este esquema de control de congestión provee una solución *punto a punto*, de manera que cada nodo, de acuerdo a sus condiciones locales de tráfico, toma decisiones de encaminamiento, contención y eliminación selectiva de paquetes. Los principales parámetros del esquema de control de congestión, se establecen de acuerdo con la zona de ubicación de los nodos, es decir, de acuerdo con la distancia en saltos que cada nodo tiene hacia el *sink*. Lo anterior debido a que las condiciones de tráfico de los nodos de una WSN, dependen en gran medida de su distancia al *sink*, ya que el tráfico siempre será mayor en zonas cercanas al éste, y por lo tanto la energía de estos nodos se agotará rápidamente.

La idea principal de esta propuesta es que al reducir la fricción entre los paquetes de datos y promover su sinergia, se incrementa el desempeño de la WSN, es decir  $\sigma_{WSN,t}$ .

El esquema de control de congestión ataca los dos tipos posibles de congestión. La congestión a nivel de enlace es atacada mediante un esquema de acceso al medio, basado en CSMA, en el cual los nodos deciden, de acuerdo con su espacio remanente en *buffer*, si participan en la transmisión y cuántos bits pueden recibir sin congestionar su *buffer*. Dentro de este esquema de acceso al medio los intervalos *backoff* son ajustados de acuerdo a la zona en que se encuentran los nodos.

La congestión a nivel de nodo es atacada mediante un esquema de encaminamiento multitrayectorias, en el cual para facilitar la administración del tráfico, los paquetes son enviados en grupos que se distribuyen en la mayor cantidad de rutas posibles. Cuando existen condiciones adversas de tráfico, los paquetes de datos son encaminados horizontalmente, es decir que se encaminan de manera que intenten rodear las zonas congestionadas.

Esta propuesta se basa en *overhearing*, lo cual representaría una desventaja para aplicaciones en las cuales se requiere que las WSN tengan un tiempo de vida largo, a pesar de mantener una transmisión continua de paquetes. Sin embargo en WSN orientadas a eventos, una vez que son detectados los eventos de interés, el principal objetivo de la WSN es informar acerca de los mismos, aunque esto implique que al hacerlo se agote la energía de los nodos que participan en su notificación.

En el siguiente capítulo se presenta la evaluación de desempeño del esquema de control de congestión propuesto, los mecanismos de encaminamiento y acceso al medio son evaluados en su operación tanto conjunta como individual.





# Capítulo 5

## Evaluación de Desempeño

---

Este capítulo muestra la evaluación de desempeño del esquema de control de congestión propuesto, el cual se probó y evaluó en una herramienta de simulación desarrollada usando el software de modelado de “sistemas complejos” NetLogo [6]. La Sección 5.1 muestra una descripción del simulador desarrollado y los módulos que lo componen. En la Sección 5.2 se presentan los resultados de la evaluación de desempeño.

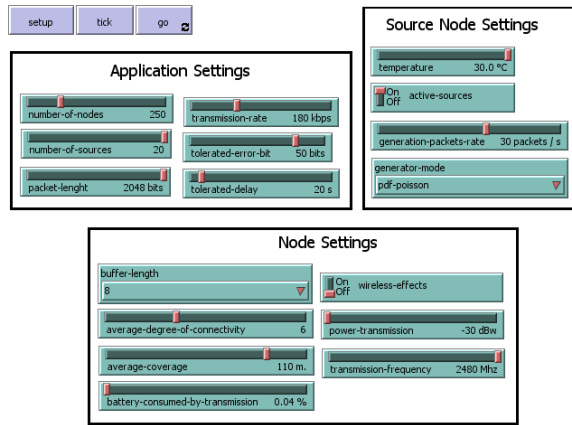
### 5.1. Programa Simulador

Este simulador se desarrolló bajo un enfoque de simulación orientada a agentes [28], en el cual los paquetes de datos se consideran entidades activas, cuyas interacciones modelan el comportamiento global de la WSN. El simulador implementa los módulos presentados en la Sección 3.2 y contempla la generación de topologías aleatorias dentro de una escala espacial de 10 ha.

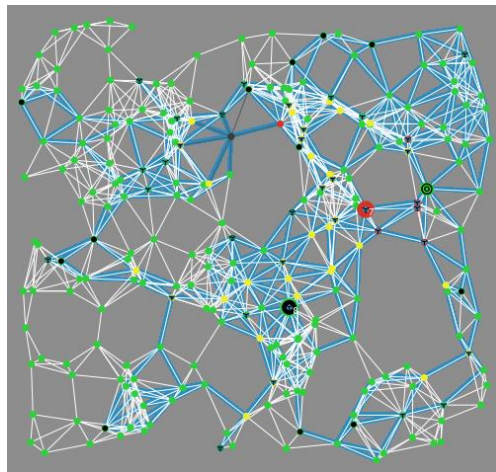
La Figura 5.1, muestra la interfaz de usuario del simulador, la cual divide en tres partes: interfaz de configuración, Figura 5.1a; vista de la simulación, Figura 5.1b; y gráficas y monitores de desempeño, Figura 5.1c. En la interfaz de configuración, el usuario puede establecer los siguientes parámetros:

- Parámetros de la aplicación: número de nodos desplegados, número de nodos fuente, velocidad de transmisión, tamaño de los paquetes de datos, retardo tolerado por paquete y número de errores de bits tolerados por paquete
- Parámetros de los nodos: tamaño del *buffer*, grado promedio de conectividad, cobertura promedio, potencia de transmisión, frecuencia de transmisión y porcentaje de batería consumida por transmisión
- Parámetros de los nodos fuente: velocidad de generación de paquetes y tipo de generación de paquetes, la cual puede ser de manera continua, mediante una PDF de *Poisson* o una PDF normal

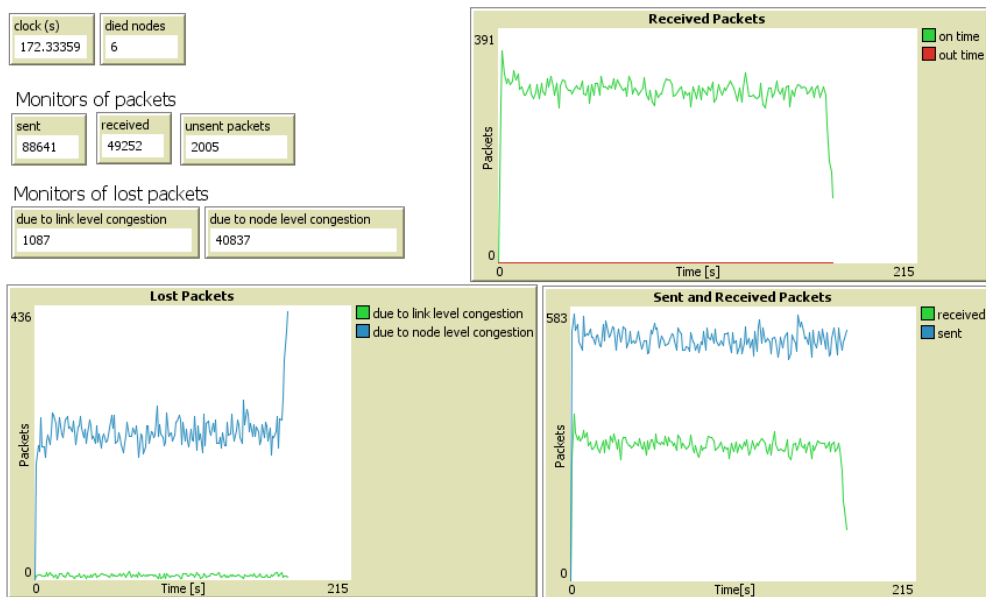
Adicionalmente, en la interfaz de configuración cuenta con botones de: ejecución por pasos, activación de efectos inalámbricos, lo que permite realizar simulaciones sin considerar los efectos inalámbricos; y de activación de generación de paquetes, lo que hace posible iniciar la creación de paquetes en cualquier momento de la simulación.



(a)



(b)



(c)

Figura 5.1. Interfaz de usuario del simulador.

La Figura 5.1b muestra la vista del grafo que representa a la WSN, esta vista se actualiza durante la simulación y tiene las siguientes características:

- El *sink* es representado con dos círculos concéntricos, los nodos fuente son representados con un círculo negro; y los nodos muertos, es decir los nodos que han agotado su batería, son representados con un círculo gris
- Los paquetes de datos son representados con una caja azul
- Los colores de los nodos cambian de acuerdo a su porcentaje de energía remanente, de la siguiente manera: verde si tienen entre el 75% y el 100% de batería, amarillo si tienen entre el 40% y 74% de batería, y rojo si tienen menos del 40% de batería
- El tamaño de los nodos se incrementa de acuerdo al número de paquetes que tienen en su *buffer*
- Las líneas azules representan los enlaces que han sido utilizados al menos una vez

En lo que respecta a la evaluación de desempeño, el programa permite obtener gráficas de:

- Satisfacción de la WSN
- Paquetes entregados
- Paquetes perdidos por congestión a nivel de nodo
- Paquetes perdidos por congestión a nivel de enlace
- Caudal de envío
- Retardo promedio de los paquetes entregados

## 5.2. Análisis del Desempeño

Con la finalidad de evaluar los beneficios de aplicar el esquema de control de congestión propuesto, se proponen 4 escenarios de operación:

1. Sin control de congestión
2. Con encaminamiento adaptativo, basado en la interferencia de los paquetes
3. Con control de acceso al medio basado en CSMA y tomando en cuenta el espacio remanente en *buffer*
4. Con control de congestión

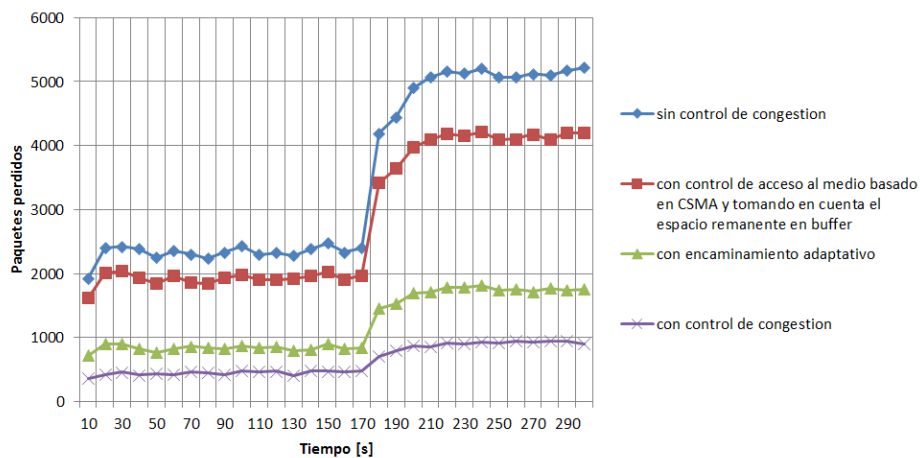
El escenario 4 se refiere al esquema de control de congestión propuesto, el cual contempla la aplicación de los mecanismos 2 y 3, tal como se explica en el Capítulo 4.

La configuración de la simulación para los escenarios mencionados, es mostrada en la Tabla 5.1.

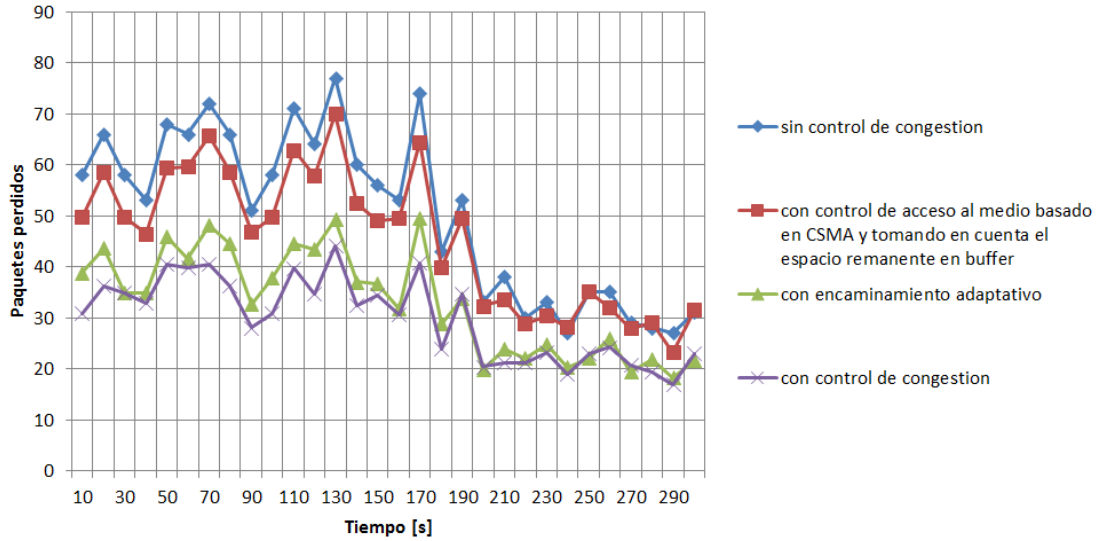
**Tabla 5.1. Configuración de la Simulación**

Parámetro	Valor
Número nodos	250
Número de nodos fuente	20
Tamaño del paquete	2048 bits
Velocidad de transmisión	180kb/s
Número de errores de bits tolerados por paquete	20
Tamaño de <i>buffer</i>	16384 bits
Grado promedio de conectividad de nodo	6
Cobertura promedio de nodo	110 m
Porcentaje de batería consumida por transmisión	.01
Potencia promedio de transmisión	-56 dBm
Frecuencia de transmisión	2048 MHz
Modelo de transmisión	Pérdidas en el Espacio Libre
Temperatura de operación	30°C
Esquema de Modulación	BPSK
Ancho de banda	4 kHz
PDF para la generación de paquetes	<i>Poisson</i>
Promedio de paquetes generados	30 paquetes / s

Como se mencionó en el Capítulo 3, el esquema de control de congestión propuesto aborda la congestión a nivel de enlace, mediante un esquema de control de acceso al medio basado en CSMA, que además toma en cuenta el espacio remanente en *buffer*. La congestión a nivel de nodo es tratada mediante un esquema multitrayectorias basado en la interferencia de los paquetes. Las Figuras 5.2 y 5.3, muestran la cantidad de paquetes perdidos debido a congestión a nivel de nodo y enlace, respectivamente. En estas figuras pueden observarse los beneficios de aplicar, tanto por separado, como juntos, los 2 mecanismos mencionados. Es importante subrayar que debido al enfoque entre capas que aplica el mecanismo de control de congestión existe sinergia en la aplicación de ambos mecanismos de control de congestión.

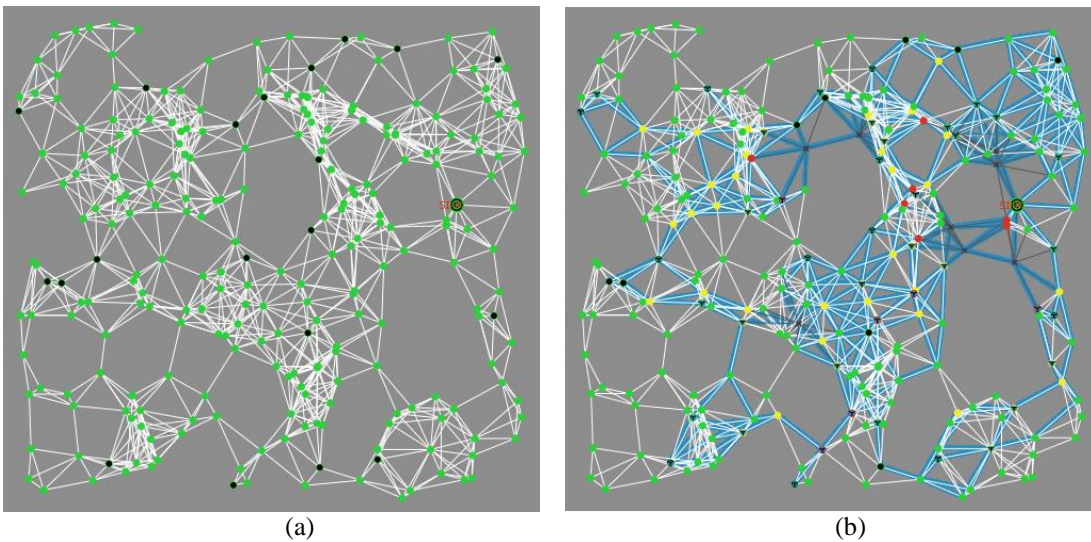


**Figura 5.2.** Paquetes perdidos por congestión a nivel de nodo.

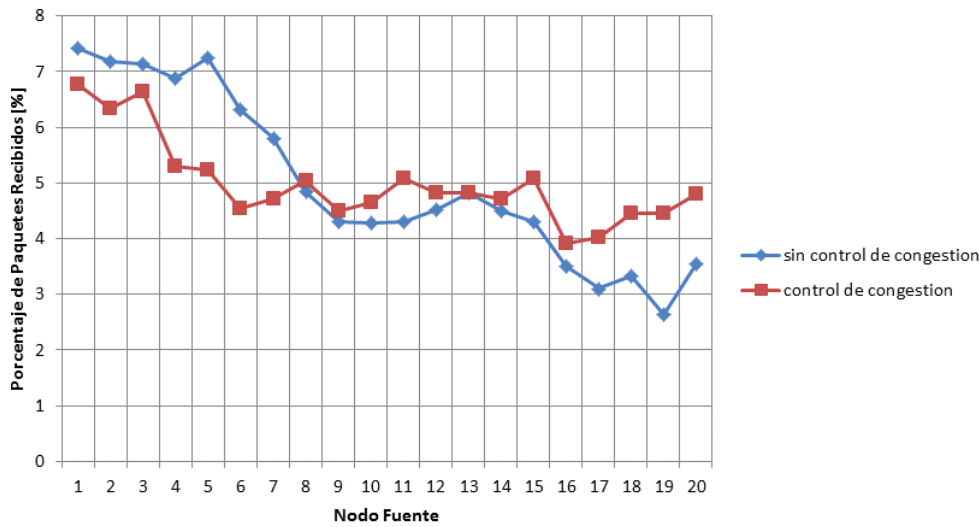


**Figura 5.3.** Paquetes perdidos por congestión a nivel de enlace.

En ambas figuras se puede observar que en los 4 escenarios propuestos, la cantidad de paquetes perdidos se incrementa a partir del segundo 170 de la simulación, esto se debe a que, como se muestra en la Figura 5.4, hasta el segundo 170 han muerto 9 nodos importantes por su ubicación. Sin embargo, al aplicar el esquema de control de congestión, el desempeño de la WSN mejora, ya que se pierden 5 veces menos paquetes que en el escenario que no aplica el control de congestión.



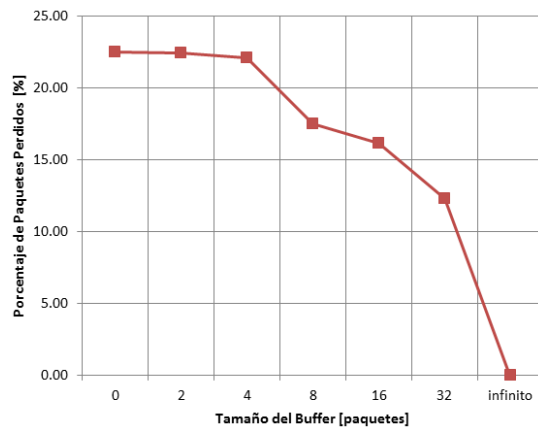
**Figura 5.4.** Vista del grafo de la WSN: (a) antes de iniciar su operación, (b) en el segundo 200.



**Figura 5.5.** Porcentaje de paquetes recibidos de cada fuente.

Otra de las ventajas que ofrece el esquema de control de congestión, es la equidad en la entrega de paquetes, ya que al asignar los parámetros de control de congestión mediante la ubicación de los nodos se busca que el *sink* reciba información de todos los nodos fuente. Con la finalidad de evaluar esta característica, la Figura 5.5 muestra el control de congestión propuesto para escenario de simulación de 1 hasta 20 nodos fuente, se puede ver que el esquema de control de congestión mejora la equidad en la entrega de paquetes.

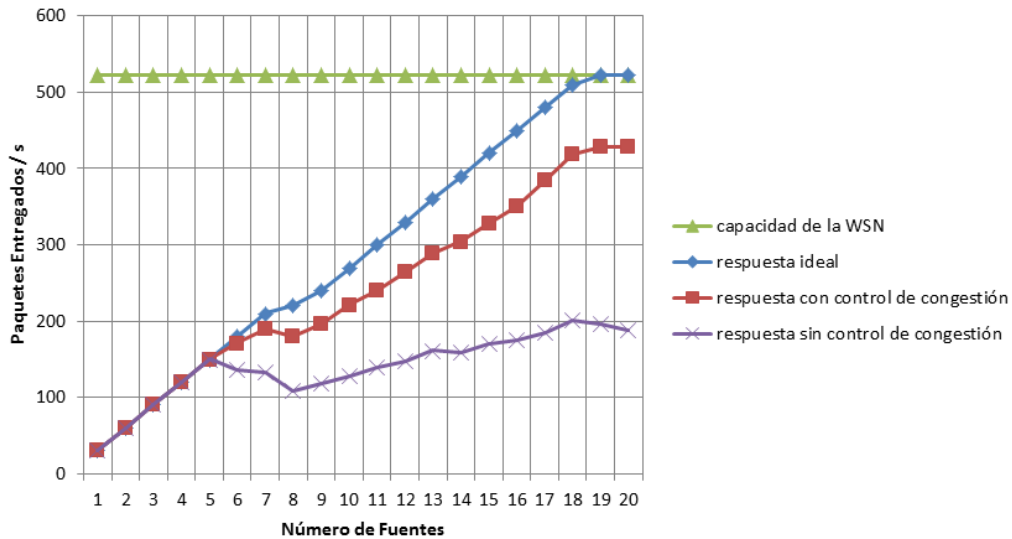
Un parámetro de incidencia en la congestión a nivel de nodo es el tamaño de *buffer*, para probar el impacto que tiene este parámetro en el esquema de control de congestión propuesto, se realizaron simulaciones variando el tamaño del *buffer*, Figura 5.6. Se observa que conforme se incrementa el tamaño de *buffer*, se reduce la cantidad de paquetes perdidos.



**Figura 5.6.** Paquetes perdidos variando el tamaño del *buffer*.

Finalmente, la Figura 5.7 muestra el desempeño del esquema de control de congestión propuesto, se observa que la capacidad de la WSN es de 522 paquetes por segundo, lo anterior se obtuvo a partir de la configuración mencionada en la Tabla 5.1, dicho escenario está representado por la Figura 5.4a, en donde se observa que el nodo *sink* tiene conexión con 9 nodos, de ahí obtenemos que la máxima cantidad de entrega de paquetes se obtiene cuando los 9 nodos están transmitiendo, de manera continua, al *sink*, a su máxima velocidad de transmisión.

Se puede ver en la Figura 5.7, que el desempeño de la WSN disminuye a partir de que hay 7 nodos fuente en operación, en este caso si no se aplica algún mecanismo de control de congestión, entonces el desempeño de la WSN decrece hasta un punto de colapso. Por el contrario al aplicar el esquema de control de congestión propuesto la cantidad de paquetes entregados se mantiene cercana a la capacidad de la WSN.



**Figura 5.7.** Tráfico entregado.





# Conclusiones y Trabajo Futuro

---

Se desarrolló un marco conceptual que permite analizar el tráfico de WSNs, desde una perspectiva de “sistemas complejos”. Mediante este marco conceptual es posible analizar el comportamiento global del tráfico en WSN, a partir de las interacciones locales de los paquetes de datos que transporta. Esta propuesta proporciona una manera novedosa y eficiente de analizar la operación de WSN, ya que aunque existen otras propuestas que analizan a las WSN como un “sistema complejo”, es importante mencionar que durante el desarrollo del presente trabajo no se encontró alguna propuesta que considerará a los paquetes de datos como los agentes activos de las WSN.

A partir del marco conceptual mencionado, se desarrolló un esquema de control de congestión que, en casos de extrema congestión, logra mantener a la WSN en niveles cercanos a su capacidad, lo cual es deseable en cualquier tipo de aplicación de WSN.

En el esquema desarrollado los paquetes son enviados en grupos, para ello cada nodo antes de enviar un grupo de paquetes evalúa si algún paquete está ocasionando fricción en el grupo, de acuerdo a dicha información se toman decisiones de encaminamiento que buscan mejorar el desempeño de la WSN, como resultado los paquetes, de manera emergente, son capaces de esquivar zonas congestionadas de la WSN.

En lo que respecta al control de acceso al medio, se aplica un mecanismo basado en el protocolo CSMA, que además toma en cuenta el espacio remanente en *buffer*, de manera que los nodos que son candidatos a recibir paquetes tienen la capacidad de decidir si participan en la comunicación y cuántos bits pueden recibir, con lo que se busca evitar la congestión no sólo a nivel de enlace, sino también a nivel de nodo.

Los parámetros del esquema de control de congestión, tales como: tiempo de vida de sus paquetes generados, tamaño de grupo y tiempo máximo de espera en *buffer*; son establecidos en cada nodo de acuerdo a su distancia en saltos al *sink*. De esta manera los paquetes generados en zonas alejadas al *sink* tienen un mayor tiempo de vida en comparación con aquellos que son creados en zonas cercanas al *sink*. Además, los nodos de zonas alejadas del *sink* mantienen a sus paquetes en *buffer*, menos tiempo que aquellos que se encuentran en zonas cercanas al *sink*; como resultado tenemos que el esquema de control de congestión, de manera emergente, mejora la equidad en la entrega de paquetes, lo cual es deseable pues interesa que el *sink* reciba información de todos los eventos que han sido detectados por la WSN.

Para realizar la evaluación de desempeño del esquema de control de congestión propuesto, se desarrolló una herramienta de simulación que incluye la perspectiva de “sistemas complejos” mediante un enfoque de simulación orientada a agentes, este tipo de simulación permite modelar el comportamiento global del tráfico de WSNs a partir del comportamiento individual de los paquetes de datos. El simulador desarrollado es útil para probar y evaluar algoritmos de encaminamiento, control de acceso al medio y control de congestión.

El esquema de control de congestión propuesto podría ser refinando mediante un mecanismo autoconfigurable que permita definir las áreas de operación de los nodos, tomando en cuenta las condiciones locales de tráfico. Otra mejora que puede incluirse es agregar un esquema de acceso al medio basado en TDMA, para los nodos que mantienen comunicación directa con el *sink*. También puede proponerse un esquema de agregación de datos para reducir la densidad de tráfico.

La propuesta de analizar el comportamiento de las WSN como un “sistema complejo” puede extenderse para diseñar protocolos de encaminamiento y acceso al medio.

La aplicación del esquema de control de congestión propuesto puede extenderse a otros sistemas de telecomunicaciones, tales como las redes cableadas.

# Bibliografía

---

- [1] G. H. Raghunandan and B. N. Lakshmi, "A Comparative Analysis of Routing Techniques for Wireless Sensor Networks," in *Proc. Nat. Conf. Innovations in Emerging Technology*, Erode, Tamilnadu, 2011, pp. 17-22.
- [2] T. Alsbouí et al., "An Overview and Classification of Approaches to Information Extraction in Wireless Sensor Networks" in *Proc. 5<sup>th</sup> Int. Conf. Sensor Technologies Applicat. (SENSORCOMM)*, Saint Laurent du Var, 2011, pp. 255-260.
- [3] J. Zhao et al., "A Survey of Congestion Control Mechanisms in Wireless Sensor Networks," in *Proc. 6th Int. Conf. Intelligent Inform. Hiding and Multimedia Signal Process. (IIH-MSP)*, Darmstadt, 2010, pp. 719-722.
- [4] S. E. Ploumis et al., "Congestion Avoidance in Wireless Sensor Networks: A Survey," in *Proc. 16th Panhellenic Conf. Informatics (PCI)*, Piraeus, 2012, pp.234-239.
- [5] D. F. Jenolin, et al., "A Survey on Congestion Control Techniques in Wireless Sensor Networks," in *Proc. Int. Conf. Emerging Trends in Elect. and Comput. Technology (ICETECT)*, Tamilnadu, 2011, pp.1146-1149.
- [6] U. Wilensky. (1999). *NetLogo* [Online], Available: <http://ccl.northwestern.edu/netlogo>
- [7] M. Mitchell, "What is complexity?," in *Complexity: a Guided Tour*, 1<sup>st</sup> ed. New York: Oxford University Press, 2009, pp. 3-14.
- [8] Ch. Yan and Q. Ji-Hong, "Application Analysis of Complex Adaptive Systems for WSN," in *Proc. Int. Conf. Comput. Applicat. and Syst. Modeling*, Taiyuan, 2010, vol. 7, pp. 328-331.
- [9] D. Orfanus, et al., "An Approach for Systematic Design of Emergent Self-Organization in Wireless Sensor Networks," in *Proc. Computation World: Future Computing, Service Computation, Cognitive, Adaptive, Content, Patterns (COMPUTATIONWORLD)*, 2009, pp. 92-98.
- [10] A. M. Niazi and A. Hussain, "A Novel Agent-Based Simulation Framework for Sensing in Complex Adaptive Environments," *IEEE Sensor J.*, vol. 11, no. 2, pp. 404-412, Feb 2011.

- [11] E. Shakshuki et al., "Software Agent-Based Directed Diffusion in Wireless Sensor Network," *Telecommun. Syst.: Model., Anal., Des. and Manage. J.*, vol. 38, no. 3-4, pp.161-174, Aug 2008.
- [12] P. Antoniou, et al., "Mimicking the Bird Flocking Behavior for Controlling Congestion in Sensor Networks," in *Proc. 3rd Int. Symp. Appl. Sci. in Biomedical and Commun. Technologies (ISABEL)*, 2010, pp. 1-7.
- [13] D. Aguirre, et al., "WSN Simulation Model with a Complex System Approach," in *Proc. Summer Comput. Simulation Conf. (SCSC)*, Toronto, 2013.
- [14] V. Michopoulos, et al., "A New Congestion Control Mechanism for WSN," in *Proc. 10<sup>th</sup> Int. Conf. Comput. Inform. Technology (CIT)*, Bradford, 2010, pp. 709-714.
- [15] Y. Liu and X. Jiang, "An Extended DCCP Congestion Control in Wireless Sensors Networks," in *Proc. Int. Workshop Intelligent Syst. Applicat. (ISA)*, Wuhan, 2009, pp. 4732-4737.
- [16] Ch. Wan et al., "Overload Traffic Management for Sensor Networks," *ACM Trans. Sensor Networks (TOSN)*, vol. 3, no. 4, Oct 2007.
- [17] S. K. Kaushik et al. "QoS Aware Routing Scheme Based on Traffic Priority for Wireless Sensor Network," in *Proc. Int. Conf. Computing Sci. (ICCS)*, Phagwara, 2012, pp. 147-152.
- [18] B. A. Sabarish et al. "Clustering based Energy Efficient Congestion Aware Protocol for Wireless Sensor Networks," in *Proc. Int. Conf. Emerging Trends Elect. Comput. Technology (ICETEECT)*, Tamil Nadu, 2011, pp. 1129-1135.
- [19] Ch. Sergiou, V. Vassiliou, "DAIPaS: A performance aware congestion control algorithm in Wireless Sensor Networks," in *Proc. 18th Int. Conf. Telecommun. (ICT)*, Nicosia, Cyprus, 2011.
- [20] M. Z. Ahmad and D. Turgut, "Congestion Avoidance and Fairness in Wireless Sensor Networks," in *Proc. IEEE Global Telecommun. Conf. (IEEE GLOBECOM)*, New Orleans, 2008, pp. 1-6.
- [21] J. Zhang, "Congestion Avoidance and Control Mechanism for Multi-paths Routing in WSN," in *Proc. Int. Conf. Comput. Sci. Software Eng.*, Wuhan, 2008, pp. 1318-1322.
- [22] Sh. Chen and N. Yang, "Congestion Avoidance Based on Lightweight Buffer Management in Sensor Networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 17, no. 9, pp. 934-946, Sept 2006.

- [23] M. Zawodniok and S. Jagannathan, "Predictive Congestion Control Protocol for Wireless Sensor Network," *IEEE Trans. Wireless Commun.*, vol. 6, no. 11, pp. 3955-3963, Nov. 2007.
- [24] C. Vuran and I. F. Akyildiz, "XLP: A cross layer protocol for efficient communication in wireless sensor networks," *IEEE Trans. Mobile Computing*, vol. 9, No. 11, pp. 1578-1591, Nov. 2010.
- [25] T. Jenn-Yue et al., "Interference-Minimized Multipath Routing with Congestion Control in Wireless Sensor Network for High-Rate Streaming," *IEEE Trans. Mobile Computing*, vol.7, no.9, pp.1124-1137, Sept. 2008.
- [26] C. Gershenson. (2007). Design and Control of Self-Organizing Systems (1st ed.) [Online]. Available: <http://arxiv.org/abs/nlin/0505009>
- [27] A. Segall, "Distributed Network Protocols," *IEEE Trans. Inf. Theory*, vol. 29, No. 1, pp. 23-35, Jan 1983.
- [28] W. Stallings, "Antennas y Propagation" in *Wireless Communications and Network*, Ed. Pearson Prentice Hall, 2005, pp. 95-126.
- [29] L. Yilmaz and T. I. Ören, "Agent-directed simulation system engineering," in *Proc. Summer Comput. Simulation Conf. (SCSC)*, San Diego, 2007, pp. 897-904.

