

**UNIVERSIDAD MICHOACANA DE
SAN NICOLÁS DE HIDALGO
FACULTAD DE INGENIERÍA ELÉCTRICA**

TESIS:

**ALINEACIÓN DE REGISTROS DE
RELÁMPAGOS DETECTADOS DESDE
MÚLTIPLES NODOS**

Para obtener el grado de
Ingeniero en Computación

Presenta:

Javier González Gómez

Asesor de Tesis:

Dr. Jaime Cerda Jacobo

Morelia, Michoacán, Noviembre del 2018

A mis padres.
A flor.

Índice de contenido

Índice de contenido	II
Lista de figuras	V
Lista de tablas	VI
Lista de algoritmos	VII
Listados	VIII
Lista de acrónimos	IX
1. Introducción	1
1.1. Descripción del problema	1
1.2. Motivación	2
1.3. Antecedentes	3
1.4. Objetivos	4
1.5. Organización del documento	5
2. Las Tormentas y los Relámpagos	7
2.1. Introducción	7
2.2. El balance de energía de la tierra	8
2.3. Formación de tormentas en latitudes medias	10
2.4. Formación de nubes de tormentas y relámpagos	14
2.5. Principios para la detección de relámpagos	20
2.6. Importancia de la detección de relámpagos	21
2.7. Comentarios finales	24

ÍNDICE DE CONTENIDO

3. Detección de Relámpagos: Antenas Boltek	25
3.1. Introducción	25
3.2. Hardware	28
3.3. Software	29
3.4. Opciones de Acceso a los Datos	31
3.5. Formato de los datos en FlashGate	34
3.6. Comentarios finales	36
4. El Tiempo Global en Sistemas Distribuidos	38
4.1. Introducción	38
4.2. El orden y el tiempo	40
4.2.1. El orden lógico	40
4.2.2. El orden temporal	41
4.2.3. Los relojes físicos	42
4.3. La medición del tiempo	44
4.3.1. Precisión y exactitud	45
4.3.2. Tiempo global	46
4.4. Sincronización de tiempo	48
4.4.1. Sincronización interna	49
4.4.2. Sincronización externa	49
4.4.3. La Condición de resincronización	51
4.4.4. Corrección de estado y corrección de velocidad	52
4.5. Formatos de tiempo externo	53
4.6. Comentarios finales	55
5. Sistema Propuesto	57
5.1. Introducción	57
5.2. La recolección de datos	60
5.2.1. Acceso a los registros de detección	61
5.2.2. Instalando y configurando NTP	62
5.2.3. Algoritmo de recolección	64
5.3. Procesamiento y almacenamiento	66
5.3.1. Concentración y procesamiento: RESTful y Node.js	66
5.3.2. Criterios de alineación	67
5.3.3. Algoritmo de alineación	70
5.3.4. Almacenamiento y consulta: MongoDB y RESTful	73
5.4. Comentarios finales	76

ÍNDICE DE CONTENIDO

6. Pruebas y Resultados	77
6.1. Precisión de relojes locales	77
6.2. Recolección de registros	79
6.3. Algoritmo de Alineación	80
6.4. Selección de parámetros de alineación	82
6.5. Consulta de datos	87
6.6. Comentarios finales	88
7. Conclusiones y Trabajos Futuros	90
7.1. Conclusiones generales	90
7.2. Trabajos futuros	92
Apéndice A: Detección de Relámpagos desde el Espacio	94
Apéndice B: Drivers de Antenas Boltek	97
Apéndice C: Tecnologías Utilizadas	99
Apéndice D: Archivo de Configuración de NTP	105
Apéndice E: Algoritmo de Alineación en JavaScript	107
Bibliografía	111

Lista de figuras

2.1.	Absorción de radiación solar dependiente de la latitud	9
2.2.	Promedio anual de radiación neta sobre la tierra	9
2.3.	Representación de un sistema de baja presión	12
2.4.	Formación de una tormenta frontal	12
2.5.	Sección transversal del desarrollo de una tormenta	12
2.6.	Diferentes estados de una nube de tormenta	16
2.7.	Tipos de relámpagos y tipos de pulsos emitidos	17
3.1.	Diagrama de un nodo	27
3.2.	Dispositivos de pre-procesamiento para antenas Boltek	29
4.1.	Modos de falla de un reloj físico	44
4.2.	Estampas de tiempo generadas para un evento	47
4.3.	Orden temporal de eventos con un tick de diferencia	48
4.4.	La condición de resincronización	51
5.1.	Nodos con áreas de detección solapadas	59
5.2.	Diagrama del sistema propuesto	59
5.3.	Nuevos campos agregados a un registro	64
5.4.	Multi-detecciones de diferentes eventos	69
6.1.	Comportamiento del reloj después de NTP	78
6.2.	Comportamiento del algoritmo de alineación	81
6.3.	Comportamiento de relojes locales con picos de desviación	85
6.4.	Resultados de diferentes tipos de consultas	88

Lista de tablas

2.1. Características mínimas de un relámpago.	23
3.1. Comparación de antenas Boltek	26
3.2. Ventajas y desventajas en opciones de acceso a detecciones . .	33
3.3. Campos de un registro	35
3.4. Ejemplos de los tipos de registros	36
7.1. Comparación de conceptos en SGBDR y MongoDB.	104

Lista de algoritmos

5.1. Algoritmo de recolección.	65
5.2. Algoritmo de alineación.	72

Listados

5.1. Estructura de un documento que representa un registro. . . .	74
5.2. Estructura de un documento que representa un grupo.	74
6.1. Consulta de grupos utilizando AJAX y JQuery.	87
7.1. Configuración utilizada para NTP.	105
7.2. Algoritmo de alineación escrito en JavaScript, con Node.js y MongoDB.	107

Lista de acrónimos

AJAX	Asynchronous JavaScript And XML
API	Application Programming Interface
BSON	Binary JSON
CCD	Charge-Couple Device
CG	Cloud-to-Ground
DEP	División de Estudios de Postgrado
EUA	Estados Unidos de América
FIE	Facultad de Ingeniería Eléctrica
FORTE	Fast On-Orbit Recording of Transient Events
FTP	File Transfer Protocol
GPS	Global Positioning System
H^+	Catión de hidrógeno
HATEOAS	Hypermedia as the engine of application state
HTML	HyperText Markup Language
HTTP	Hypertext Transfer Protocol
IC	Intra-cloud
IEEE	Institute of Electrical and Electronics Engineers
IPC	Inter-Process Communication
JNA	Java Native Access

LISTA DE ACRÓNIMOS

JSON	JavaScript Object Notation
LAN	Local Area Network
LANNL	Los Alamos National Laboratory
LF	Low Frecuency
LIS	Lightning Imaging Sensor
MCC	Mesoscale Convective Complexes
MCS	Mesoscale Convective System
NASA	National Aeronautics and Space Administration
NPM	Node Package Manager
NTP	Network Time Protocol
<i>OH⁻</i>	Anión de oxígeno-hidrógeno
OTD	Optical Transient Detector
PC	Personal Computer
PCI	Peripheral Component Interconnect
PPS	Pulse-per-second Signal
PTP	Presicion Time Protocol
RAE	Real Academia Española
RDF	Radio Direction Finding
REST	REpresentational State Transfer
RPC	Remote Procedure Call
SD	Sistema Distribuido
SGBDR	Sistema de Gestión de Bases de Datos Relacionales
SNTP	Simple Network Time Protocol
SOAP	Simple Object Access Protocol
TAI	Temps Atomique Internationale

LISTA DE ACRÓNIMOS

TLE	Transient Luminus Event
TRAC	Thunderstorm Rangin and Acquisition subprocess
TRMM	Tropical Rainfall Measuring Mission
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
USB	Universal Serial Bus
UTC	Tiempo Universal Coordinado
VHF	Very High Frecuency
VLF	Very Low Frecuency
WSDL	Web Services Description Language
XML	eXtensible Markup Language

Alineación de relámpagos detectados desde múltiples nodos

TESIS

Javier González Gómez

dk_jgg_@hotmail.com

Resumen

En este trabajo se presenta un algoritmo que permite realizar la alineación de registros de detecciones de relámpagos utilizando las estampas de tiempo de cuando fueron detectados. Los relámpagos representan un peligro para las diversas actividades humanas. Por esto es importante poder detectarlos y así reaccionar de manera oportuna. Cuando se tienen varias antenas de detección distribuidas, también se tienen diferentes conjuntos de detecciones para cada antena. Para poder concentrar las diferentes fuentes de datos sin ocasionar inconsistencias, es necesario realizar una alineación de los registros de las detecciones. El principal criterio de alineación utilizado es la estampa de tiempo, sobre el tiempo de detección, que contiene cada registro. Por esto, la alineación está atada a la precisión que puedan ofrecer las estampas de tiempo. Para mejorar la calidad de la alineación, es necesario asegurar que los relojes utilizados como fuente de tiempo sean precisos. Adicionalmente son utilizados otros criterios secundarios, que permiten mejorar la calidad de alineación cuando el criterio del tiempo no es suficiente. El algoritmo de alineación resultante realiza su mejor esfuerzo para alinear los registros, debido a que los resultados dependen completamente de la precisión de los datos de entrada.

Palabras clave: Boltek, tiempo, global, timestamps, NTP

Alignment of lightning detected from multiple nodes

THESIS

Javier González Gómez

dk_jgg_@hotmail.com

Abstract

This paper presents an algorithm that allows the alignment of lightning detection records using the timestamps when they were detected. Lightning represents a danger to various human activities. This is why it is important to be able to detect them and thus react in a timely manner. When you have several detection antennas, you also have different sets of detections for each antenna. In order to concentrate the different data sources without causing inconsistencies, it is necessary to perform an alignment of the detection registers. The main alignment criterion used is the timestamp on the detection time, that each record contains. Because of this, the alignment is tied to the precision that the timestamps can offer. To improve the quality of the alignment, it is necessary to ensure that the clocks used as the source of time are accurate. Additionally, other secondary criteria are used, which allow improving the quality of alignment when the time criterion is not sufficient. The resulting alignment algorithm does its best effort to align the records, because the results depend entirely on the accuracy of the input data.

Keywords: Boltek, tiempo, global, timestamps, NTP

Capítulo 1

Introducción

Durante este capítulo se describe el origen y el contexto del problema que se pretende afrontar durante este trabajo. Se presenta el origen de la motivación para desarrollar esta línea de investigación. Después se describen los trabajos previos que se relacionan a esta línea de trabajo. Posteriormente se establecen los objetivos que se pretenden lograr durante el presente. Por último se da una descripción sobre la organización del resto del documento.

1.1. Descripción del problema

Los relámpagos son fenómenos naturales imprevisibles que se originan durante tormentas. Es de conocimiento general que el impacto de un relámpago tiene suficiente poder para atentar contra la vida de una persona. También pueden causar daño al entorno del impacto, al equipo eléctrico y equipo electrónico que se encuentre en las inmediaciones. En resumen los relámpagos pueden afectar en gran medida el desarrollo de las actividades humanas. Aunque no es posible predecir el impacto de los relámpagos, actualmente existen sistemas de detección de tormentas y de relámpagos. Mediante la detección de tormentas es posible tomar mayores precauciones debido a la alta probabilidad de que pueda ocurrir un impacto de relámpago. Por otro lado, con un sistema detector de relámpagos es posible tener monitorizada una zona estratégica y así poder reaccionar de manera oportuna ante un impacto de relámpago.

En la División de Estudios de Postgrado (DEP) de la Facultad de Ingeniería Eléctrica (FIE) se cuenta con tres antenas de detección de relámpa-

gos, las cuales están instaladas en Morelia, Yurécuaro y Lázaro Cárdenas. Las antenas almacenan solo la información relevante de la detección de un relámpago, al conjunto de datos producto de una detección le llamamos *registro* de una detección de relámpago. Los datos producto de las detecciones, los registros, están disponibles para ser compartidos con aplicaciones externas. Cada antena tiene un alcance limitado de detección, por lo cual se encuentran instaladas en diferentes puntos de Michoacán para aumentar el rango de detección. A cada punto donde se encuentra instalada una antena de detección le llámanos *nodo*. Cada nodo es independiente en su capacidad de detección, así, cada nodo produce su propio conjunto de registros de detecciones. Debido a que el área que cubre cada antena es diferente y a que puede haber un error inherente de la detección, cada nodo tendrá diferente cantidad de registros de detecciones. Los nodos no comparten información de sus detecciones entre ellos, siendo imposible saber si dos nodos detectaron los mismos relámpagos o saber cuáles registros de detección pertenecen al mismo relámpago. Si un nodo a detecta relámpagos y produce el conjunto de registros A , y una antena b produce el conjunto de registros B . ¿Ambas antenas detectaron los mismos relámpagos, $A \equiv B$? ¿Cuáles registros de detección de A también están en B , $d_A \equiv d_B : d_A \in A, d_B \in B$?

Por lo anterior, es deseable poder establecer un mecanismo que permita determinar cuáles registros de detección de relámpagos de una antena a también se encuentran en otra antena diferente b . Llamamos a tal mecanismo *alineación de registros relámpagos*. Para lograr la alineación de registros de relámpagos primero es necesario acceder a los registros de detecciones de los nodos. Después será necesario tomar en cuenta la información que contiene los registros y establecer reglas que permitan alinear los registros de diferentes antenas. Finalmente, deberá ser posible realizar la consulta de los resultados de la alineación.

1.2. Motivación

Actualmente no existen trabajos que revisen el problema de alineación de registros de detecciones de relámpagos. En [1] se propone una forma para mejorar la precisión de la posición de las detecciones. En [2] se propone un método para el seguimiento de tormentas. Sin embargo, en ambos casos, se asume que en la entrada de datos se encuentran registros de detecciones alineados. Aquellos trabajos que dependen de una entrada de registros ali-

neados no tendrán un uso práctico a menos que se resuelva el problema de la alineación, de otra forma solo se quedarán en la teoría. Por esto durante el presente trabajo se propone una solución al problema de alineación, con el fin de rellenar este espacio vacío de conocimiento.

Mediante la alineación de registros de detecciones es posible concentrar los diferentes registros producidos por los nodos. Cuando la concentración de registros de diferentes nodos no involucra una alineación, aquellos registros de detección que pertenecen al mismo relámpago serán contados como relámpagos separados. De esta forma, sin una correcta alineación el conteo del número de relámpagos ocurridos en una zona podría no ser correcto. Si los registros de los diferentes nodos no son concentrados, y por el contrario solo se procesan por separado es posible evitar estos inconvenientes. Sin embargo al tratar cada conjunto de registros por separado, se pierde la ventaja de ampliar el rango de detección mediante múltiples nodos. Así pues, es necesario realizar una alineación de registros de detecciones de relámpagos si se quiere aprovechar al máximo la utilización de múltiples nodos de detección.

1.3. Antecedentes

En [1] se afrontó el problema de cómo mejorar la precisión con que las antenas determinaban la posición donde ocurrió el relámpago detectado. El resultado del trabajo fue una aplicación web que procesaba los diferentes registros y estimaba la mejor aproximación de los relámpagos. Esta aplicación necesita como entrada un conjunto de registros ya alineados, y son procesados utilizando bilateración, trilateración y las fórmulas de Vincenty. Esta aplicación fue desarrollada en Javascript utilizando Node.js y MongoDB, los resultados gráficos se desplegaban sobre un mapa de Google Maps.

Los registros usados como entrada eran generados por otra aplicación web, también desarrollada durante ese trabajo, que simulaba el desarrollo de tormentas y la detección de los relámpagos por diferentes nodos. Como la aplicación tenía acceso a todos los datos generados por las tormentas, y al contexto de detección de las antenas (también simulado), podía fácilmente alinear los registros de salida. Estos registros generados por simulación se utilizaron para validar el desarrollo principal: la trilateración y bilateración de los relámpagos.

En [2] se afrontó el problema de determinar las trayectorias de las tormentas, a partir del tiempo y posición de los relámpagos. El resultado fue

una aplicación web que recibía como entrada un conjunto de registros de detecciones, y los procesaba utilizando k-means para realizar una agrupación temporal y espacial. Esta aplicación fue desarrollada en javascript utilizando Node.js y MongoDB, las trayectorias resultantes son visualizadas sobre un mapa de Google Maps.

El conjunto de registros usados como entrada para esta aplicación podría ser el de cualquier nodo, permitiendo generar las rutas de las tormentas con los datos disponibles. Sin embargo no es posible utilizar los registros de detecciones de más de una antena para ampliar el rango de las trayectorias. Esto debido a que habría registros de detecciones pertenecientes a un mismo relámpago, detectado desde varias antenas, que serían tomadas como relámpagos diferentes. Por lo cual se producirían trayectorias adicionales asignadas a diferentes tormentas, cuando en realidad son la misma tormenta.

Adicionalmente a los trabajos descritos previamente, también existen aplicaciones que permiten la recolección de los registros de detecciones. Sin embargo estas aplicaciones solo realizan la concentración de los registros de los nodos y no proveen una forma para lograr la alineación de los registros de detección. En [3] se encuentran dos aplicaciones para obtener los registros desde los nodos. La aplicación *readerNSms* extrae los registros desde cada nodo y los envía hacia una base de datos en PostgreSQL. Los registros son extraídos y enviados al servidor inmediatamente que están disponibles. Es una aplicación de recopilación y concentración de registros desde los diferentes nodos. La aplicación *localNSms* extrae los registros disponibles de cada nodo, los guarda en archivos de texto plano y, almacena los archivos de forma local en el nodo de origen. Es una aplicación que respalda localmente los datos capturados.

1.4. Objetivos

Conocer la teoría detrás del origen de los relámpagos para entender su naturaleza y comportamiento, y así poder establecer aquellas características que son relevantes durante su detección. Revisar la documentación existente para poder acceder a los registros de detecciones que proporciona cada nodo, con el fin de concentrar todos los registros de los diferentes nodos. Crear un método que permita realizar la correcta alineación de registros de detecciones de relámpagos detectados desde múltiples nodos. Finalmente,

permitir consultar los resultados de la alineación.

1.5. Organización del documento

Durante el segundo capítulo se repasan las bases teóricas sobre el origen de los relámpagos y sobre la detección de descargas originadas por los relámpagos. Primero se revisa la naturaleza de las descargas eléctricas; desde el origen del clima, pasando por el comportamiento de las tormentas, hasta la descripción de cómo se genera un relámpago dentro de una tormenta. Después se revisa brevemente los métodos utilizados para la detección de los relámpagos y, finalmente, se describe la importancia de la detección de los relámpagos.

En el tercer capítulo se presentan las antenas Boltek. Este tipo de antenas son con las que cuenta la DEP de la FIE. Primero se describen brevemente las características de estas antenas. Posteriormente se hace una revisión sobre sus componentes hardware, drivers y software. Por último se presentan las opciones disponibles para poder acceder a los datos detectados y el formato de los registros adquiridos.

En el capítulo 4 se revisa la relación entre el tiempo y el orden, las implicaciones de un tiempo global, y los tipos de sincronización entre diferentes dispositivos. Se describen brevemente las características de la sincronización interna y la sincronización externa. Finalmente, se describen brevemente los principales protocolos de sincronización externa, NTP y IEEE 1588.

En el capítulo 5 se describen las características y algoritmos de nuestro sistema propuesto. El proceso de alineación propuesto se divide principalmente en dos etapas: recolección y procesamiento. Durante la etapa de recolección se pone especial atención a los registros desde la perspectiva de los nodos. Por otro lado, en la etapa de procesamiento se presentan los procesos realizados por servidor conforme los datos son recibidos. Finalmente, se presentan las diferentes formas para realizar consultas de los datos almacenados.

Durante el capítulo 6, se muestran los resultados de la implementación del sistema. Se verifica la precisión alcanzada por los relojes de los nodos. Se describe el método utilizado para realizar pruebas con datos reales. Así mismo, se presenta el comportamiento del algoritmo propuesto con diferentes parámetros. Después, se describe una metodología para seleccionar los valores de los parámetros, y las implicaciones a tomar en cuenta. Por último, se presenta la interfaz de consulta de datos y las formas para realizar dichas

CAPÍTULO 1. INTRODUCCIÓN

consultas.

Al final se muestran las conclusiones del presente trabajo y las recomendaciones del autor. Además, se proponen algunas líneas de investigación como continuación del presente trabajo.

Capítulo 2

Las Tormentas y los Relámpagos

Durante este capítulo se da una introducción al balance de energía que tiene la tierra y los medios por los que se intenta mantener ese balance, dando origen al clima. Después se describen los frentes de tormentas que se originan en las latitudes medias y las características de su formación. También se describen los medios por los cuales las nubes adquieren cargas eléctricas, lo cual permite el desarrollo de los relámpagos. Se revisan brevemente las características de los relámpagos y los métodos utilizados para su detección. Al final del capítulo se describe la importancia de la detección de los relámpagos y su impacto sobre las actividades humanas.

2.1. Introducción

Antes de entrar de lleno en la teoría sobre los relampagos es necesario aclarar y delimitar la definición de ciertos términos de uso común a lo largo del capítulo. Una breve inspección a la Real Academia Española (RAE)[4] arroja las siguientes definiciones:

Rayo

1. m. Cada una de las líneas que parten del punto en que se produce una determinada forma de energía y señalan la dirección en que esta se propaga.
2. m. Línea de luz que procede de un cuerpo luminoso [...]

3. m. Chispa eléctrica de gran intensidad producida por descarga entre dos nubes o entre una nube y la tierra.

[...]

Relámpago

1. m. Resplandor vivísimo e instantáneo producido en las nubes por una descarga eléctrica.

2. m. Fuego o resplandor repentino.

[...]

Trueno

1. m. Estruendo, asociado al rayo, producido en las nubes por una descarga eléctrica.

[...]

De lo anterior se puede establecer que cuando se usa el término *rayo* se está hablando de la descarga eléctrica, la cual produce luminosidad, y que ocurre entre dos nubes o entre una nube y la tierra. Por otro lado, cuando se usa la palabra '*relámpago*' se hace referencia al destello instantáneo de luz visible, que es producido por el *rayo*. Así mismo con '*trueno*' se hace referencia al sonido que es producido por el '*rayo*'. Por lo anterior se puede llegar a establecer que '*rayo*' hace énfasis al proceso físico y luminoso, mientras que '*relámpago*' hace énfasis en aquellas características visibles que lo componen. Por otro lado, '*trueno*' pone principal atención a aquellas características audibles. De todo lo anterior se puede concluir que los tres términos pueden ser aplicados para describir una descarga eléctrica producida por las nubes durante una tormenta, cada uno de ellos poniendo especial énfasis sobre alguna característica en particular.

Así pues, a lo largo del capítulo se utilizara el término '*relámpago*' para hacer referencia a una descarga eléctrica producida por las nubes durante una tormenta. Esto debido a que es la palabra más extendida entre la literatura, y además su definición principal es la más utilizada para referirse a las descargas eléctricas de las nubes, lo cual evita confusiones con la aplicación del término.

2.2. El balance de energía de la tierra

El sol, como la estrella en el centro del sistema solar, es un poderoso reactor de fusión nuclear que produce una tremenda cantidad de energía. El

CAPÍTULO 2. LAS TORMENTAS Y LOS RELÁMPAGOS

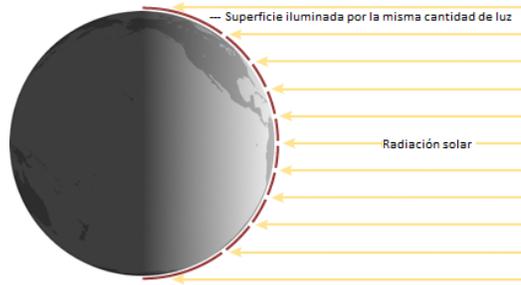


Figura 2.1: La absorción de radiación solar depende de la latitud. (NASA, ilustración de Robert Simmon[7])

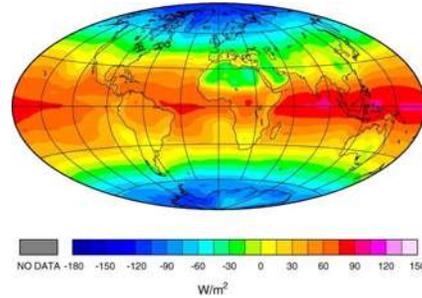


Figura 2.2: Promedio anual de radiación neta sobre la tierra. (Meerkötter y Vázquez-Navarro, 2012)

sol emite energía por radiación desde su fotosfera, una capa de varios miles de kilómetros sobre su superficie. La tierra está localizada a una distancia promedio de 149 596 mil kilómetros del sol. Así, solo una pequeña parte de la energía radiactiva emitida por el sol hacia el espacio logra alcanzar la tierra.¹

La radiación del sol que alcanza a irradiar la tierra en la parte superior de la atmósfera es conocida como la constante solar (S_o). Dicha constante solar es en promedio de $1367 W/m^2$ de energía. Sin embargo, dado que la tierra tiene forma esférica, la radiación solar no se proyecta con el mismo ángulo en todas las latitudes. Así, mientras nos movemos hacia latitudes altas la misma energía del sol es distribuida sobre grandes áreas, resultando en menos energía absorbida por área (Figura 2.1). Esta es la principal razón por la cual los trópicos son más cálidos que las latitudes más altas [6] [5].

El patrón de irradiación neta que tiene la tierra, resultado de la diferencia de radiación de onda corta absorbida y la radiación de onda larga desprendida, es presentado en la Figura 2.2. Las superficies que producen valores negativos de radiación neta representan flujos netos que están en constante aumento de calor. En contraste, valores positivos máximos son distribuidos a lo largo de los océanos ecuatoriales, donde la absorción de radiación solar es el factor dominante. Sin embargo, la gran diferencia entre la radiación neta ecuador-polo, resultante del desequilibrio en la distribución del calor neto, es bastante notable.

¹ Para un ejemplo ilustrativo, la energía solar capturada por la tierra sobre un periodo de 1 000 años es igual a la energía producida por el sol en solo 14 segundos [5] .

El desequilibrio del calor neto entre el ecuador y el polo actúa como el principal mecanismo que impulsa los procesos de compensación que ocurren en el sistema de la tierra. Estos procesos se presentan como los flujos circulantes a gran escala de la atmósfera y los océanos. El exceso de energía en los trópicos resulta en grandes flujos cálidos que transportan el calor hacia los polos por medio de la atmósfera y los océanos, transportando la energía desde las bajas latitudes hacia las latitudes más altas. Corrientes oceánicas transportan aguas cálidas lejos del ecuador. En la atmósfera, la transferencia de exceso de calor ocurre primariamente a través de la circulación meridional de la atmósfera. Dicha circulación es acompañada por varios procesos dinámicos, comúnmente experimentados como fenómenos del *clima* tales como vientos y turbulencias, cambios de temperatura, procesos de formación de nubes, eventos de precipitación, etc. En ciencias del clima este gran complejo es descrito como *el motor de calor de la tierra* (heat engine of the Earth)[5].

2.3. Formación de tormentas en latitudes medias

Los frentes de Tormentas ocurren en el límite de diferentes masas de aire (frentes), normalmente aire polar frío-seco encontrándose con aire tropical cálido-húmedo. Entre mayor diferencia entre las masas de aire (temperatura y humedad) mayor es la inestabilidad atmosférica que se desarrolla, y mayor es la intensidad de esas tormentas. Esta intensidad se hace visible por sus frecuentes eventos de relámpagos y algunas veces como TLEs² de tipo sprite³.

CAPÍTULO 2. LAS TORMENTAS Y LOS RELÁMPAGOS

Cada tormenta tiene dos frentes que separan el aire polar frío del aire tropical cálido (Figura 2.3). El frente cálido (representado por semicírculos) es caracterizado por el avance del aire caliente, mientras el frente frío (representado por triángulos) está caracterizado por el avance del aire frío entre las regiones de aire cálido. Debido a que el aire frío y seco es más denso que el aire cálido y húmedo, el aire frío en la parte baja del frente frío atraviesa por debajo del aire cálido forzándolo a ascender ligeramente por encima (Figura 2.5). El rápido ascenso de aire caliente resulta en fuertes tormentas convectivas algunas veces asociadas con una intensa actividad de relámpagos. Por el otro lado, a lo largo del frente cálido, el aire caliente lentamente asciende sobre el denso aire frío, resultando en una amplia región de llovizna y de lluvia junto con una débil actividad eléctrica. Por lo anterior, durante las tormentas en las áreas de latitud media las regiones del frente frío deberían proveer las mejores condiciones para una actividad intensa de relámpagos, observación de sprites y quizás observaciones de otros TLEs.

Grandes inestabilidades en la atmósfera también pueden ocurrir a lo largo de zonas frontales que dividen el aire cálido-húmedo del aire frío-seco alrededor de celdas de alta presión. El flujo alrededor de una celda de alta presión es en dirección de las manecillas del reloj en el hemisferio norte, mientras que cuando el flujo es alrededor de un sistema de baja presión tiene una dirección antihorario (contra las manecillas del reloj). El aire es estable en el centro de la alta presión, resultando en un clima despejado, caliente y húmedo en los meses de verano. En los límites de una región de alta presión, la mezcla de ai-

²Eventos luminosos transitorios (TLE –Transient Luminus Event) es el nombre colectivo dado a una amplia variedad de fenómenos ópticos, relacionados con los relámpagos, que ocurren sobre un amplio rango de altitudes en la estratosfera y la mesosfera y la baja termosfera, desde los 15 hasta los 110 km, en conjunción con la actividad troposférica [8].

Diferentes tipos de TLEs han sido documentados y clasificados [9] [8] [10] [11]. Los primeros en estudiarse y por tanto más entendidos son: elves (Destellos rojizos de duración breve, inducidos por relámpagos, que se extienden sobre 300km lateralmente), sprites (Destello rojizo que inicia en la baja ionosfera y presenta un rápido descenso de altitud, terminando en luminiscencia azul, presentan diferentes morfologías), halos (breves luminosidades incandescentes-rojizas con extensiones radiales de 40-70 km). Los menos frecuentes y que se desarrollan a mayor altitud son más difíciles de estudiar: blue jets (Luminosidad azul que descienden lentamente desde la parte superior de la nube hasta alcanzar altitudes de 40 km, presenta morfología de una fuente), gigant jets (Descarga eléctrica, con movimiento ascendente, entre la parte superior de la nube y la baja ionosfera, presenta luminosidad amarilla a blanca).

³ TLE más común. Véase [10] [11] [9].

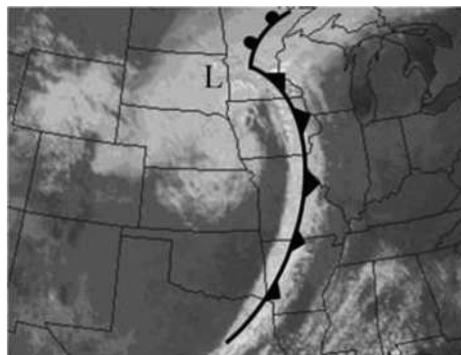
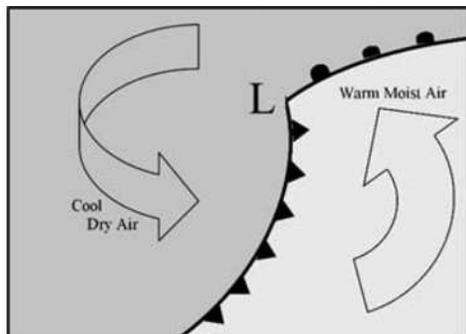


Figura 2.3: Representación esquemática de un sistema de baja presión con el frente frío y el frente cálido mostrados con triángulos y semicírculos respectivamente (Price, 2006).

Figura 2.4: Imagen satelital mostrando la formación de una tormenta a largo del frente frío. (Price, 2006).

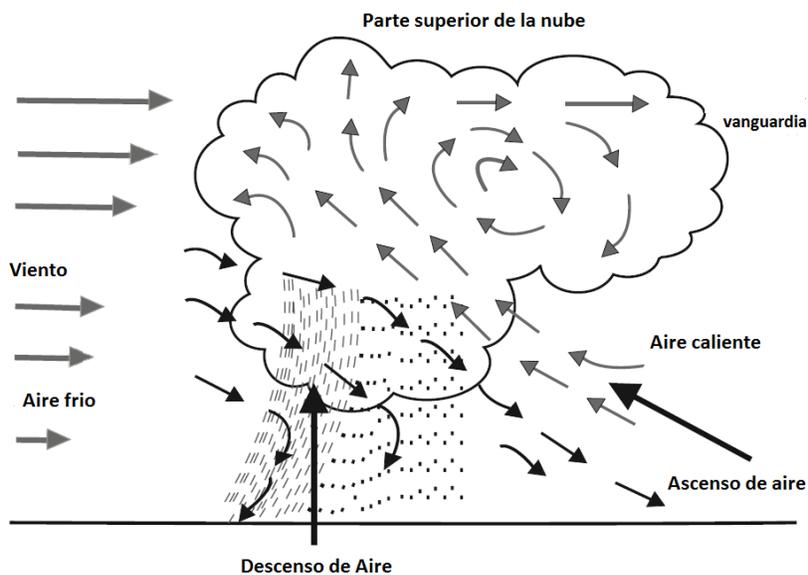


Figura 2.5: Sección transversal a través de dos frentes mostrando el desarrollo de la tormenta. (Coray, 2015).

CAPÍTULO 2. LAS TORMENTAS Y LOS RELÁMPAGOS

re seco frío y aire húmedo cálido a lo largo del frente polar permite el inicio de una convección⁴ durante la tarde. Esta convección algunas veces produce un anillo de tormentas alrededor del centro de alta presión el cual deberá rotar junto con las manecillas del reloj alrededor de la celda de alta presión. Esto en ocasiones puede producir el patrón común *anillo de fuego*⁵, el cual a veces se convierte en un sistema convectivo de mesoescala (MCS –Mesoscale Convective System)⁶ el cual puede producir grupos de tormentas llamados complejos convectivos de mesoescala (MCC –Mesoscale Convective Complexes)⁷. Estos grandes complejos de tormentas son productores asiduos de relámpagos, teniendo varias observaciones de sprites ocurriendo sobre esas tormentas en Estados Unidos de América (EUA) (Lyons et al en[10]).

Por lo anterior, las tormentas pueden ocurrir en dos entornos diferentes. Primero, las tormentas formadas por masas de aire tropical resultantes del calor diurno en la superficie de la tierra. Estas tormentas de la tarde ocurren principalmente en los trópicos pero también ocurren durante meses del verano en altitudes medias cuando inestabilidades (estáticas) pueden desarrollarse en las tardes de los días cálidos. Segundo, las tormentas frontales ocurren principalmente desde latitudes medias hasta latitudes altas donde

⁴Propagación de una magnitud física (e.g. calor) en un medio fluido por diferencias de densidad.[4] En meteorología, se refiere al movimiento ascendente de aire caliente, por diferencia de densidad al expandirse, el cual puede formar nubes tipo cumulonimbus, típica nube de tormenta que produce lluvia intensa y relámpagos [12] [4] [13].

⁵El *anillo de fuego* (Ring of Fire) es originalmente un término geológico que describe la localización de fallas geológica y volcanes como si estuvieran sobre la orilla de un anillo. En meteorología, las tormentas son sinónimos con el fuego, es decir, un anillo de tormentas.

⁶Un sistema convectivo de mesoescala (MCS –Mesoscale Convective System) se refiere a regiones de tormenta más grandes que una tormenta individual pero más pequeñas que un ciclón extratropical. Predominan la nubosidad y la lluvia, que normalmente persisten durante varias horas, tienden a formarse cerca de frentes climatológicos (frentes fríos o frentes cálidos) y tienen un tamaño en el orden de 100 km o más. MCS es un término más generalizado para referirse a los sistemas que no satisfacen los estrictos criterios de tamaño, forma, o duración de un MCC.

⁷ Un complejo convectivo de mesoescala (MCC –Mesoscale Convective Complexes) es un tipo único de sistema convectivo de mesoescala, el cual es definido por las características observadas en imágenes infrarrojas satelitales. Presenta un área de la parte superior de la nube de 100,000 km² o más con temperaturas iguales o menores que -32°C, o un área de la parte superior de a nube de 50,000 km² con temperatura menores o iguales que -52°C. Las especificaciones de tamaño se deberán mantener por 6 horas o más. Su máxima excentricidad (eje menor/eje mayor) es igual o mayor que 0.7 cuando presenta su máxima extensión, teniendo forma casi redonda. Son comúnmente formadas durante la noche y prestan abundante lluvia, viento, hielo, relámpagos y posibles tornados.

diferentes tipos de masas de aire interactúan y resultan en inestabilidades (baroclínicas⁸) a lo largo de frentes fríos, frentes cálidos y frentes estacionarios. Frentes de tormentas pueden ocurrir a cualquier hora del día, sobre los continentes o el océano, y durante el verano y el invierno. Lo que es una necesidad para el desarrollo de esas tormentas es una gran diferencia de densidades entre masas de aire adyacentes. Esta diferencia de densidad puede ser causada por una diferencia de temperaturas, una diferencia de humedad o por una combinación de ambas. En general estas inestabilidades son grandes en los meses de verano en las latitudes medias, sin embargo sobre los océanos relativamente cálidos la inestabilidad puede ser también grande en los meses de invierno.

Debería ser notado que en adición a las formas de producir tormentas mencionadas arriba, también podemos observar tormentas debido a forzamiento orográfico (ascenso forzado del aire sobre la topografía). Montañas, cordilleras e islas fuerzan al aire a fluir en ascenso y pueden iniciar inestabilidades que disparan la formación de tormentas.

2.4. Formación de nubes de tormentas y relámpagos

La luz del sol calienta el suelo, y el suelo calienta el aire que está en contacto con él. El aire se calienta gradualmente, se expande y se vuelve más ligero que el aire circundante, con lo cual comienza a ascender. La presión atmosférica disminuye por el calor y provoca que el aire ascendente se expanda aún más. Debido a la expansión de la masa de aire su temperatura interna disminuye, provocando que la humedad interna se condense en millones de gotas de agua microscópicas.

Una columna de aire ascendente en una atmósfera inestable es el primer estado de una nube de tormenta, llamado *estado cumulus* (Figura 2.6a). Conforme el aire asciende, la humedad se condensa más rápidamente y la temperatura continua bajando. Cuando la temperatura alcanza los 0°C, algunas de las pequeñas partículas de agua se congelan, formando cristales de

⁸ En meteorología, una atmósfera baroclínica es aquella donde la densidad depende tanto la temperatura como la presión; en contraste con una atmósfera barotrópica, para la cual la densidad depende únicamente de la presión. Las zonas barotrópicas por lo general son encontradas en latitudes centrales, o trópicos, mientras que las áreas baroclínicas son encontradas en los límites de regiones de latitud media y las regiones polares.

CAPÍTULO 2. LAS TORMENTAS Y LOS RELÁMPAGOS

hielo, y algunas otras se mantiene en su forma líquida como *gotas de agua superfría*. Cuando dos de estas partículas colisionan las gotas de agua superfría se congelan directamente en los cristales de hielo, haciéndolos crecer en tamaño. Aquellas partículas que crecen de esta forma son llamadas *partículas graupel*⁹. Conforme las partículas graupel crecen, el aire ascendente no puede elevarlas debido su peso y entonces comienzan a caer. Algunas partículas de hielo son transportadas hacia arriba y hacia abajo por corrientes de aire durante largo tiempo de tal forma que crecen debido a las colisiones con otras partículas[12].

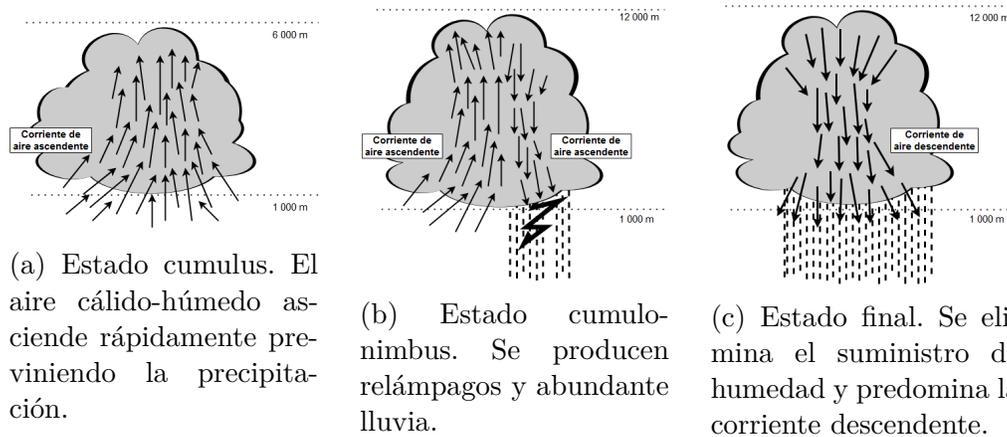
Incluso aunque hay numerosos procesos de electrificación en las nubes, actualmente es ampliamente aceptado que la principal causa de la electrificación de nubes de tormenta son procesos de carga no inductiva, originados por la colisión de los diferentes tipos de partículas de hielo que se forman en las nubes de tormenta. Dichas colisiones ocurren entre las partículas ascendentes, pequeños cristales de hielo y gotas de agua superfría, y las partículas descendentes, los graupels formados por la conglomeración de cristales de hielo y de las gotas superfrías. Debido a que los diferentes tipos de partículas tienen diferentes propiedades según su superficie, tal como superficies irregulares y/o impurezas, sus proporciones de crecimiento son diferentes. Esto causa que las capas casi líquidas en la superficie sean de diferente firmeza y la movilidad de los iones OH^- (Anión de oxígeno-hidrógeno)¹⁰ y H^+ (Cation de hidrógeno)¹¹ sean diferentes. Así, durante el contacto de las colisiones una transferencia de carga neta puede ocurrir entre las partículas, causado por el intercambio de iones en la superficie de las capas casi-líquidas. El signo del flujo de la carga depende críticamente de la temperatura del ambiente y la cantidad de agua líquida contenida, la cual determina la proporción de crecimiento y, así, la temperatura (punto de congelamiento) de los graupels. Bajo las condiciones normales de una tormenta, los graupels tienen una carga negativa y los cristales de hielo tienen una positiva.

⁹ El termino graupels, que proveniente del alemán, se refiere a la precipitación que se forma cuando gotas de agua superfría son colectadas y congeladas durante la caída de copos de nieve, esto es, es una mixtura de lluvia y nieve. Tienen una frágil estructura semisólida (se desintegran si son tocadas) y formas rectas, a diferencia del granizo que tiene una estructura sólida y una forma más definida. También son llamados hielo suave o pelotillas de nieve.

¹⁰ Anión de oxígeno-hidrógeno (OH^-) es un ion con carga negativa y que formado por un átomo de oxígeno y otro de hidrógeno.

¹¹ Cation de hidrógeno (H^+) es un ion de hidrógeno con carga positiva. También llamado hidrón.

CAPÍTULO 2. LAS TORMENTAS Y LOS RELÁMPAGOS



(a) Estado cumulus. El aire cálido-húmedo asciende rápidamente previniendo la precipitación.

(b) Estado cumulonimbus. Se producen relámpagos y abundante lluvia.

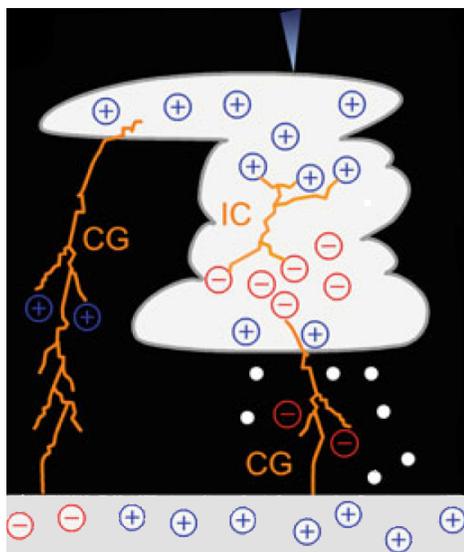
(c) Estado final. Se elimina el suministro de humedad y predomina la corriente descendente.

Figura 2.6: Los diferentes estados que presenta una nube de tormenta durante su ciclo de desarrollo.

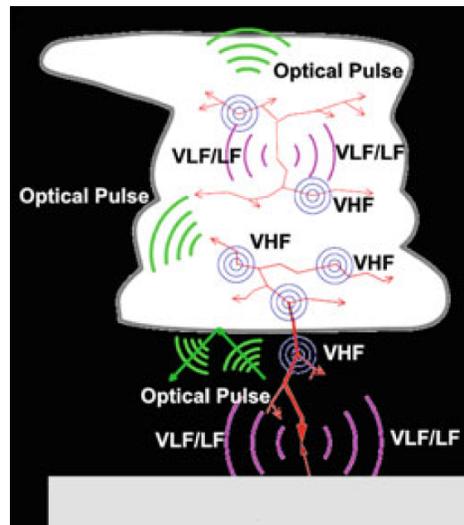
Las diferentes velocidades de caída de los graupels, partículas de hielo y los cristales de hielo, causan una separación de carga dentro de la nube de tormenta. Dicha separación de cargas da lugar a la bien conocida estructura de dipolo, con un centro de carga negativo (graupels y partículas de hielo) en niveles intermedios y un centro de carga positivo en los niveles superiores (cristales de hielo). Esta estructura de dipolo idealizada, es algunas veces acompañada por capas adicionales con cargas menores. Algunas veces centros de baja carga positiva se forman cerca de la base de la nube, en la región de aire ascendente o en las regiones estratiformes más extendidas junto al nivel de licuefacción. Por otro lado, una delgada capa de carga negativa puede ser encontrada en la parte superior de la nube. En la Figura 2.7 se muestra un esquema que ilustra las principales estructuras de carga de una nube y los diferentes tipos de descargas que se pueden producir.

Cuando una nube alcanza este estado de carga, es llamada *cumulonimbus* o una nube de tormenta desarrollada. Durante este estado, la caída de graupels se incrementa gradualmente y comienzan a empujar hacia abajo al aire circundante. Esto origina una columna de aire descendente que se opone al movimiento del aire ascendente (Figura 2.6b). Cuando los graupels alcanzan los niveles más cálidos se licuan, pasando del estado sólido al líquido, y dan lugar a las gotas de lluvia. Es en este estado donde una nube es capaz de generar descargas eléctricas, esto es, relámpagos.

Por otro lado, cuando la cantidad de carga acumulada por la nube es



(a) Principales regiones de carga y tipos de descargas producidas.



(b) Diferentes emisiones ópticas y electromagnéticas producidas durante una descarga.

Figura 2.7: Esquema sobre los tipos de relámpagos y sobre los pulsos emitidos durante la descarga. (Höller et al.,2012).

CAPÍTULO 2. LAS TORMENTAS Y LOS RELÁMPAGOS

lo suficientemente fuerte, una descarga ocurre entre dos o más de los centros de carga dentro de la nube (IC –Intra-cloud)¹², entre la nube y la tierra (CG –Cloud-to-Ground) o entre la nube y el aire circundante (Figura 2.7). También ocurren diversos procesos de descargas entre la nube y la ionosfera, conocidos como eventos luminosos transitorios TLE. Los procesos de descarga de un relámpago es acompañado por un colapso del campo eléctrico. La fase inicial de estos procesos es llamada la *ruptura inicial* (initial breakdown). La mayoría de estos procesos y sus descargas asociadas son generadas por partículas cósmicas, las cuales pueden originar avalanchas de electrones al interactuar con la atmósfera. Esto puede dar lugar a la formación de un canal ionizado relativamente grande durante las fases tempranas del desarrollo de la descarga, permitiendo la emisión de una onda larga de radiación electromagnética con una frecuencia muy baja (VLF –Very Low Frequency)¹³ y una onda de frecuencia baja (LF –Low Frequency)¹⁴, las cuales se originan a partir del pulso del relámpago [12][14].

Un *líder* es un canal de conducción eléctrica, formado por una canal de aire ionizado, entre regiones con cargas opuestas. Cuando un canal de aire ionizado se empieza a formar, los líderes son atraídos por las regiones que tienen una polaridad opuesta al origen del líder. Una descarga negativa ocurre cuando un líder del canal negativo alcanza una región cargada positivamente. Una descarga positiva, ocurre a través de la propagación de un líder cargado positivamente dentro de la región cargada negativamente. Este proceso de liderazgo es llamado líder bipolar. El líder se propaga a pasos discretos de unos pocos metros o cientos de metros y tiende a irradiar en longitud (VHF –Very High Frequency)¹⁵. Una vez que un líder hace contacto con una región de la nube que está suficientemente cargada, una impetuosa corriente de descarga (IC) se manifiesta en el canal existente de la nube y un pulso de VLF es generado. Este proceso es referido como *flujo de retorno* (recoil

¹²El termino IC es utilizado generalmente para referirse a aquellas descargas que ocurren entre diferentes centros de carga, dentro de la misma nube o entre nubes diferentes.

¹³El termino *frecuencia muy baja* (VLF –Very Low Frequency) se utiliza para denominar a la banda del espectro electromagnético (ondas de radio) que ocupa el rango de frecuencias de 3KHz a 30kHz, con longitudes de onda desde 100 km hasta 10 km.

¹⁴ El termino *frecuencia baja* (LF –Low Frequency) se utiliza para denominar a la banda del espectro electromagnético que ocupa el rango de frecuencias de 30kHz a 300 kHz, con una longitud de onda desde 10 km hasta 1 km.

¹⁵ El termino *frecuencia muy alta* (VHF –Very High Frequency) se utiliza para denominar a la banda del espectro electromagnético que ocupa el rango de frecuencias de 30MHz a 300kHz, con longitudes de onda desde 10 m hasta 1 m.

CAPÍTULO 2. LAS TORMENTAS Y LOS RELÁMPAGOS

streamer) o *proceso k* (k-process).

El inicio de un relámpago no puede tomar lugar sólo entre los principales centros de carga positiva y carga negativa dentro de la nube, sino también entre el principal centro de carga negativa y un centro de carga positiva diferente del primario. Este tipo de descarga puede iniciar una subsecuente descarga a tierra (CG). El líder negativo desciende cada vez más hacia tierra y finalmente conecta con la tierra, iniciando una progresiva *descarga de retorno* (return stroke) en la dirección opuesta. Estando en contacto con tierra, la descarga de retorno produce un fuerte pulso de VLF/LF, el cual es generalmente el proceso de manifestación más importante durante una descarga. Posterior a la primera descarga de retorno del relámpago, se pueden desarrollar subsecuentes procesos de descargas:

- Descargas intranube (IC) que pueden extender la descarga de retorno hacia el interior de la nube, llamadas *procesos de unión* (junction processes) o Procesos J (J-Processes).
- Se pueden desarrollar procesos K, que son pulsos sobrepuestos al lento proceso J.
- Se puede iniciar una o más subsecuentes descargas a tierra (CG) a partir de los líderes de vanguardia en los existentes canales ionizados.

Todos estos procesos son luminosos y dan pie a una emisión de luz óptica de diferentes intensidad y duración. Algunos relámpagos hacia tierra (CG) desarrollan una prolongada corriente continua final, la cual es perfectamente visible por algunos cientos de milisegundos. La descarga completa podría durar hasta 2 segundos y típicamente extenderse horizontalmente por unos pocos cientos de kilómetros. Las principales fuentes de emisión son ilustradas en la Figura 2.7b.

Con el incremento de la precipitación la intensidad del aire descendente se incrementa. La actividad de relámpagos toma lugar antes de que un fuerte descenso de aire sea establecido, ya que esto último indica el estado final de la nube. La corriente de aire descendente trasporta aire frío de las altas altitudes hacia el suelo que combinado con la lluvia enfrían el aire del suelo, lo cual interrumpe la formación de la columna de aire ascendente (Figura 2.6c). Al perder su fuente de aire caliente ascendente, la nube deja de crecer, la precipitación disminuye y los restos de la nube se evaporizan en la atmósfera y se dispersan[12].

Este proceso de formación de las nubes de tormenta dura aproximadamente 30 minutos. Sin embargo, durante una tormenta múltiples nubes se pueden formar paralelamente, de tal forma que cuando una de ellas se disipa ya existe otra en su lugar.

2.5. Principios para la detección de relámpagos

Los diferentes componentes de un relámpago emiten radiación en un amplio margen del espectro electromagnético. Los rangos de radiofrecuencia usados más comúnmente para la detección de descargas a tierra (CG) son las bandas de frecuencias muy bajas (VLF), frecuencias bajas (LF) y frecuencias muy altas (VHF). Las ondas se propagan como ondas terrestres a través de la superficie de la tierra o como ondas aéreas las cuales pueden ser reflejadas en la ionosfera. Las ondas largas pueden ser detectadas a más grandes distancias que las ondas cortas dado que estas son menos atenuadas. Así mismo, su pico de intensidad está conectado a una frecuencia que rondan 10 kHz.

Las redes de detección de VLF pueden ser usadas incluso para detección global con alcances del orden de 1 000 km o más. El rango de VLF/LF es usado para redes a escala de ciudad o continente con sensores de rango de algunas decenas hasta algunos cientos de kilómetros. Son comúnmente utilizadas antenas de dipolo eléctrico o antenas de bucle magnético. La onda corta de VLF requiere una red de sensores de corto alcance del orden de decenas de km, de tal forma que ellas solo pueden estar detectando su línea de visión. Su ventaja es que pueden detectar los detalles de la estructura de ramificación de la descarga, pero no pueden notar la alta corriente de energía producida por la descarga de retorno, las descargas IC y los procesos K. Otros sistemas de detección usan métodos de interferómetros¹⁶ o detección óptica de descargas.

¹⁶ Dispositivos que hacen uso de técnicas que consisten en combinar la luz (u otras ondas electromagnéticas) proveniente de diferentes receptores, telescopios o antenas de radio para obtener una imagen de mayor resolución aplicando el principio de superposición.

2.6. Importancia de la detección de relámpagos

Debido a que los relámpagos ocurren en toda la tierra y a que son fenómenos naturales impredecibles, estos pueden provocar accidentes relacionados con la alta intensidad de la descarga producida. Su poder destructivo es evidente por la alta energía generada desde la nube hacia los canales de descarga terrestres y por la corriente generada durante la descarga. Los relámpagos de nube a tierra (CG) pueden causar daños relacionados con el impacto directo y con el impacto indirecto (cercano) a estructuras características, incluyendo bosques, edificios, vehículos y personas. Los accidentes provocados por la caída de rayos pueden causar daño a la industria productiva, así como a la población en general. Este poder es un peligro para las personas, las estructuras, las instalaciones dentro de la estructura, sobre la estructura o cercanos a la estructura, los bosques y las actividades humanas.

La caída de relámpagos puede afectar de diferente manera dependiendo del área de impacto, y cada área tiene riesgos específicos [15] [16] [17] . A continuación se resumen algunos de los peligros que pueden representar los impactos de relámpagos.

- **Daños a la salud**

Estos van desde la pérdida de vidas humanas hasta lesiones severas o importantes debido a las quemaduras provocadas por la intensidad del impacto.

- **Daños a estructuras e instalaciones**

Un impacto de un relámpago hacia edificios u otras estructuras, no sólo puede resultar en daño a la propia estructura sino también en daños al equipo eléctrico y/o electrónico que se encuentre al interior. Efectos colaterales de los daños y las fallas ocasionadas pueden ser extendidos hacia las proximidades de la estructura, afectando el entorno circundante.

- **Daños a bosques**

El impacto de un relámpago en un bosque puede provocar el inicio de un incendio, debido a la intensa energía que se desprende durante la descarga. El costo social de un incendio forestal incluye los relacionados con los trabajos para delimitar el área del incendio, la supresión del

incendio y los posibles trabajos de limpieza posteriores. Así también se puede contemplar la pérdida de recursos madereros, ya sea para consumo doméstico o industrial, y la pérdida de áreas de recreación social. Finalmente también se tiene una pérdida del hábitat de la flora y la fauna, o la pérdida de flora endémica.

■ **Transmisión y distribución de electricidad**

Los relámpagos son una importante variable en la administración y operación de infraestructura eléctrica, en particular para los sistemas de transmisión y distribución que proveen energía a consumidores residenciales, comerciales, institucionales y industriales. Periodos de interrupción del servicio y pérdidas de potencia de la energía afectan a estos consumidores, y podrían dañar sus aparatos eléctricos y electrónicos, perjudicando el correcto desarrollo de sus actividades.

Para reducir el daño provocado por los relámpagos es necesario tomar medidas de protección, las cuales podrían reducir significativamente el riesgo tanto como para vidas humanas, instalaciones eléctrico-electrónicas cercanas a estructuras, áreas forestales o pérdidas económicas [18] [19] [20]. La detección, observación y análisis de relámpagos de forma temporal y espacial durante el desarrollo de las tormentas permite evitar o disminuir considerablemente el daño causado por su impacto. Como resultado final del análisis se pueden implementar diversos sistemas auxiliares: alarmas de impactos cercanos a poblados, alarmas de riesgo de incendios forestales, búsqueda de fallas eléctricas por relámpagos, estimación de la fuerza de la tormenta, seguimiento de tormentas, alarmas preventivas para impactos cercanos a líneas eléctricas, etc. Para poder realizar un análisis de riesgos ante las descargas de relámpagos, es necesario contar con los siguientes datos:

- Localización espacial y temporal de los impactos de los relámpagos detectados.
- Características de los relámpagos detectados.
- Densidad de las descargas de relámpagos en la región de estudio.
- Localización de las áreas con riesgo de recibir impactos de relámpagos.
- Tipo de estructuras a proteger y sus características.
- Características del equipo y los servicios dentro de la estructura.

CAPÍTULO 2. LAS TORMENTAS Y LOS RELÁMPAGOS

Tabla 2.1: Características mínimas de un relámpago.

Característica	Descripción
Tiempo de ocurrencia	Tiempo global de referencia en que el relámpago ocurrió.
Lugar de ocurrencia	Posición global de referencia donde ocurrió el relámpago.
Tipo	Tipo de relámpago detectado: CG o IC.

Los tres primeros puntos se enfocan en las características y comportamiento de los relámpagos. El primer punto puede ser cubierto por cualquier sistema de detección de relámpagos, pero no permitiría diferenciar entre tipos de relámpagos (CG/IC). El segundo punto requiere de un equipo y un procesamiento más especializado, debido a que se requiere determinar la polaridad y el tipo de descarga detectada. Estos datos brindan una mayor visión sobre aquellos relámpagos que pueden ser potencialmente más peligrosos, los relámpagos a tierra (CG). Para el tercer punto se necesita cubrir mínimo el primer punto, y almacenar los datos de las detecciones durante suficiente tiempo para establecer un análisis de relevancia.

En la Tabla 2.1 se presentan las características mínimas que se necesitan para poder representar completamente un relámpago que cubra estos primeros puntos. Las primeras dos características, tiempo y lugar de ocurrencia, son suficientes para poder identificar de forma única a un relámpago. La última característica, tipo, nos permite identificar aquellos relámpagos tipo CG, los cuales son los más peligrosos para objetos en tierra.

Para los últimos tres puntos se necesita información sobre el donde y el que se quiere proteger. Esta información es indispensable para poder decidir los tipos de protecciones necesarias, y los procedimientos de respuesta frente a impactos de relámpagos. Sin embargo, estos últimos tres puntos no tiene mayor alcance si es que no se cuenta con información sobre relámpagos detectados. Por lo anterior es primordial poder contar con datos de detecciones de relámpagos antes de poder avanzar en la implementación de medidas preventivas.

2.7. Comentarios finales

Durante este capítulo se han revisado los diferentes términos que pueden ser aplicados para definir una descarga eléctrica producida por una nube de tormenta. De entre las múltiples definiciones en el presente trabajo es utilizado el término *relámpago*; el cual representa el fenómeno que se quiere expresar y además reduce otras posibles interpretaciones, con lo cual se evitan ambigüedades. Además, se reviso brevemente el origen del clima; el cual surge como un efecto secundario del balance de energía de la tierra. Se describe el entorno en que se pueden formar las tormentas: por el aire caliente diurno, por interacción entre diferentes frentes de aire y por forzamiento orográfico. Así mismo, se describen los mecanismos de formación de las nubes de tormenta (cumulonimbus), las cuales se forman por el ascenso de aire caliente y su interacción con el aire frío. Se presentan los procesos de electrificación que permiten a las nubes tener carga; las cargas son originadas por la colisión de partículas en el interior de la nube. También se describe el proceso de desencadenamiento de los relámpagos, el cual se origina cuando se forma un canal que conecta dos centros de carga opuestos. Posteriormente se presentan los principios utilizados para la detección de relámpagos, usando los pulsos electromagnéticos de VLF, LF y VHF que son generados durante una descarga de un relámpago. Después, se resumen los peligros que representan los relámpagos para las actividades humanas, los datos necesarios para poder estudiarlos y, así, poder reaccionar de manera oportuna ante la ocurrencia de estos fenómenos naturales. Así mismo, se enumeran los datos necesarios para poder establecer un análisis sobre los relámpagos en una región de interés, y se describen cuales son las características mínimas para poder representar un relámpago detectado de forma única.

Capítulo 3

Detección de Relámpagos: Antenas Boltek

Durante el capítulo anterior se revisó la naturaleza de los relámpagos, y cuáles son las formas de detectarlos. En el presente capítulo se revisará la detección de relámpagos para el caso particular de las antenas Boltek. Primero se describen los tipos de antenas que ofrece la empresa Boltek, y se presentan las características de aquellas utilizadas durante este trabajo. Después, se describen los componentes de hardware necesarios para lograr la detección de relámpagos. Se muestran de forma breve los diferentes tipos de interfaz software disponibles y las características que proporciona cada uno. Posteriormente se resumen las diferentes opciones disponibles, ofrecidas por NexStorm, para acceder a los datos de las detecciones de los relámpagos. Finalmente, se muestra el formato en que se encuentran los registros para la opción de acceso FlashGate-Strikes.

3.1. Introducción

La empresa Boltek se dedica a proveer sistemas detectores de relámpagos para empresas y también para usuarios particulares. Las antenas Boltek para detección de relámpagos permiten desplegar un mapa, sobre el escritorio de una laptop o PC (Personal Computer), en el cual se pueden visualizar los relámpagos detectados. Los datos son detectados en tiempo real, es decir, que cuando una descarga eléctrica es producida por un relámpago, esta es detectada por la antena, procesada en el dispositivo de pre-procesamiento, y

CAPÍTULO 3. DETECCIÓN DE RELÁMPAGOS: ANTENAS BOLTEK

Tabla 3.1: Comparación de características entre las diferentes antenas Boltek.

Característica	Antena			
	EFM-100	StormTracker	LD-250	LD-350
Detección de corto alcance	✓	✓	✓	✓
Alta exactitud en corto alcance	✓	X	X	X
Detección de largo alcance	X	✓	✓	✓
Detección direccional	X	✓	✓	✓
Clasificación de tipo y polaridad	X	✓	X	✓
Operación móvil	X	X	✓	X

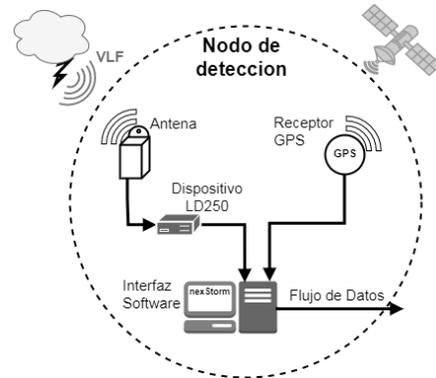
desplegada sobre el mapa por la interfaz software [21]. Todo esto se realiza dentro de los milisegundos de que un relámpago ocurre. Boltek ofrece varios tipos de antenas de detección, para PC y para laptops, de corto y de largo alcance, y para puestos de detección estacionarios y móviles. En la Tabla 3.1 se pueden ver la comparación entre los diferentes principales tipos de antenas que ofrece Boltek.

En la DEP de la FIE se cuenta con tres antenas Boltek para la detección de relámpagos. Dos antenas tipo StormTracker, las cuales tienen una interfaz PCI (Peripheral Component Interconnect)¹, y una antena tipo LD250, la cual cuenta con un dispositivo externo de pre-procesamiento. Ambos productos, tanto StormTracker-PCI y LD-250, son antenas de amplio rango que permiten detectar los relámpagos. La única diferencia entre estos dos productos es que la antena LD-250 permite montarse sobre un vehículo y realizar una detección móvil, mientras que StormTracker-PCI es para detección estacionaria. Otra diferencia crucial es que LD-250 no provee información sobre el tipo y polaridad de los relámpagos detectados.

¹ La Interconexión de Componentes Periféricos (PCI –Peripheral Component Intercon-

CAPÍTULO 3. DETECCIÓN DE RELÁMPAGOS: ANTENAS BOLTEK

Figura 3.1: Un *nodo* es todo el conjunto de dispositivos necesarios para producir un flujo de datos sobre las características de los relámpagos detectados.



Cada antena viene con un paquete de detección, el cual conforma todo el sistema necesario para la detección de relámpagos. Dicho paquete está compuesto por la antena de detección, el dispositivo de pre-procesamiento y la interfaz software[21]. También se tienen que tomar en cuenta los elementos adicionales que permiten el funcionamiento correcto de las antenas: PC donde reside la interfaz software, sistema operativo, software adicional de procesamiento, GPSs² que proveen tiempo, reguladores de energía, etc. Estos elementos interaccionan entre sí, cada uno de ellos cumpliendo una función específica, con el fin de poder completar la tarea de detección y poder desplegar los relámpagos detectados al usuario. Se puede abstraer a todo este conjunto de dispositivos necesarios para la detección como un *nodo de detección*, de aquí en adelante nombrado solamente como *nodo* para acortar.

nect) es un bus estándar de computadoras para conectar dispositivos externos directamente a la placa base. Estos dispositivos pueden ser circuitos integrados ajustados o tarjetas de expansión que se ajustan en los conectores.

² El Sistema de Posicionamiento Global (GPS – Global Positioning System) es un sistema que permite determinar en toda la Tierra la posición de un objeto con una precisión en metros. Para determinar las posiciones en el globo, el sistema GPS se sirve de 24 a 32 satélites y utiliza la trilateración. Para obtener la posición de un objeto, se solicita información de los satélites y se calcula el tiempo que tardan en llegar las señales al equipo. De esta forma es posible calcular la posición relativa a los satélites y las coordenadas reales del punto de medición. Además, el reloj interno del GPS se sincroniza con los relojes atómicos que llevan a bordo los satélites, consiguiendo una gran exactitud de tiempo. El sistema fue desarrollado, instalado y empleado por el Departamento de Defensa de los Estados Unidos.

3.2. Hardware

La antena de detección viene empaquetada para su instalación en interiores y exteriores, con el fin de evitar los efectos del clima. La antena tiene la capacidad de búsqueda por radio dirección (RDF)³, por lo cual detecta las señales de VLF y de LF que son producidas por los relámpagos y permite determinar la dirección desde donde provienen las señales. La antena manda estos datos de detección hacia el dispositivo de pre-procesamiento para la determinación de las características del relámpago [22].

Las antenas de Boltek con las que se cuentan tienen un alcance de 480 km (300 millas). Cabe mencionar que dicho rango de alcance puede variar dependiendo del entorno de la antena (altura, obstáculos, interferencias, etc.). De esta forma, podría ser un rango inferior si la antena no está instalada en un entorno conveniente o podría ser superior si el entorno es extremadamente conveniente. Los requisitos para la instalación de la antena (altura, sujeción, entorno, etc.) son cubiertas en [23] y [24].

El *área de detección* de un nodo se refiere al área circundante dentro de la cual un nodo tiene la capacidad de detectar las descargas de los relámpagos. El área de detección de un nodo es igual al alcance de la antena utilizada, esto debido a que esta última es quien recibe las señales de los relámpagos. Un nodo puede detectar un relámpago r , sólo si, este ocurre dentro del área de detección del nodo j . Lo contrario no es necesariamente cierto. Si el evento r ocurre dentro del área de detección de j , entonces r **podría no** ser detectado por j . Esto es debido a que las señales usadas para la detección pueden degradarse o distorsionarse por factores del entorno. El área de detección es una importante característica de un nodo, pues determina cual es su alcance para detectar relámpagos. Otra característica importante es la localización del nodo, esto es, el lugar donde está instalado, debido a que dicha localización es el pivote central del área de detección. Conforme la localización del nodo cambie, el área superficial de detección también cambiará, y por tanto, la cantidad y distribución de los datos detectados.

El dispositivo de pre-procesamiento de StormTracker-PCI es una tarjeta PCI, la cual se inserta en la ranura PCI libre de una PC[23]. Para el caso del LD-250 su dispositivo de pre-procesamiento es una unidad externa, la cual se conecta a la computadora a través de un puerto USB o un puerto RS232

³ Búsqueda por radio dirección (RDF –Radio Direction Finding) es la determinación de la dirección desde la cual una señal fue transmitida. Estas pueden referirse a señales de radio u otras formas de comunicación inalámbrica.

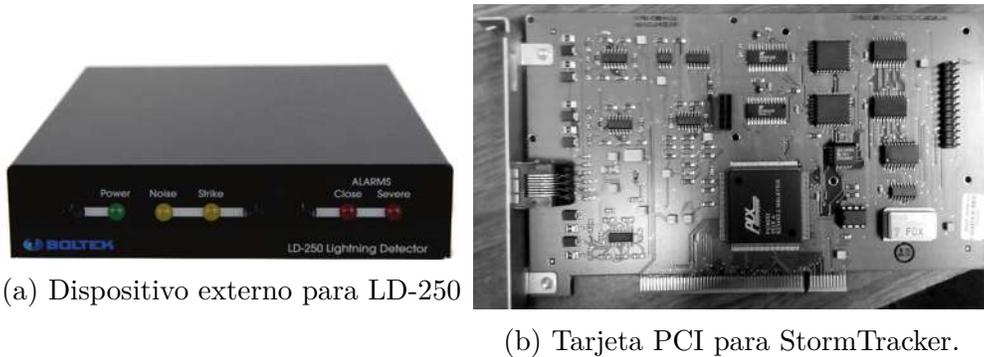


Figura 3.2: Dispositivos de pre-procesamiento para antenas Boltek

(COM)[24]. El dispositivo de pre-procesamiento se encarga de determinar las características del relámpago: distancia, tipo y polaridad. La fuerza de la señal recibida por la antena se utiliza para calcular la distancia aproximada de la descarga del rayo, otros procesamientos son realizados para determinar el tipo y la polaridad [21] [22].

Para un desglose completo sobre la instalación del hardware, las antenas, el dispositivo de pre-procesamiento, el cable de conexión y otros, véase [23] y [24]. Para ver sobre la instalación de los drivers necesarios véase el Apéndice B: Drivers de Antenas Boltek.

3.3. Software

Una interfaz software es un programa que sirve como remplazo o complemento de una interfaz física para interactuar con las antenas Boltek. Boltek distribuye junto con sus antenas una interfaz software que permite configurar las opciones de la antena instalada, y también permite visualizar un mapa donde se despliegan las detecciones de relámpagos[21]. La interfaz software también es encargada de realizar un procesamiento adicional a los datos recibidos desde el dispositivo, para mejorar la exactitud de la distancia del relámpago y reducir los efectos de las variaciones de rayo a rayo debido a la energía generada durante la descarga [22]. Las antenas Boltek pueden utilizar alguno de los siguientes software como interfaz:

- StormTracker
- NexStorm Lite (versión reducida)

CAPÍTULO 3. DETECCIÓN DE RELÁMPAGOS: ANTENAS BOLTEK

- NexStorm (versión completa)

El software NexStorm Lite es una versión reducida de NexStorm [25]. NexStorm Lite es freeware, lo cual significa que puede ser descargado de manera gratuita desde la página oficial de Astrogenic Systems[26], o desde la página oficial de Boltek bajo la sección de descargas del producto correspondiente[27]. Anteriormente StormTracker era la interfaz software por defecto que se incluía junto a las antenas Boltek, en especial para StormTracker-PCI. Sin embargo recientemente Boltek está comenzando a reemplazar StormTracker por NexStorm Lite [26]. Ambos software ofrecen las mismas funcionalidades elementales: análisis del tipo de descarga, visualización de la posición de la descarga detectada, filtro de ruido, capacidad para cambiar el mapa de fondo y configurar alarmas para proximidad de descargas. Sin embargo NexStorm Lite ofrece ciertas ventajas sobre StormTracker como son:

- Mejor clasificación y visualización de las descargas en el mapa.
- Seguimiento de grupos de tormentas.
- Visualización de estadísticas y gráficos sobre la actividad reciente.
- Posibilidad de reproducir la actividad registrada desde un archivo.
- Permitir mostrar mapas de densidad de descargas.

NexStorm (versión completa) [25] es software de pago y por tanto solo puede ser adquirido mediante la compra directa con sus proveedores. También puede ser adquirido al comprar alguna de las antenas de Boltek, pues viene incluido como la interfaz software por defecto. NexStorm tiene las mismas funciones que NexStorm Lite y además añade funciones adicionales de exportación de los datos generados con aplicaciones externas. Por lo anterior, *si se quiere acceder a los datos de la detección de las descargas, es necesario utilizar NextStorm*. Esto debido a que es la única interfaz software que da opciones de exportación y de acceso a los datos de los relámpagos detectados por las antenas. A continuación se resumen las funciones adicionales que ofrece NexStorm:

- Acceso a los datos procesados mediante Windows Shared Memory.
- Captura y exportación automática de la interfaz de usuario, o del mapa de graficación de descargas.
- Extracción y exportación de archivos de datos automáticamente.

- Utilidades adicionales para manipular mapas personalizados de fondo en el área de graficación de descargas capturadas.
- Utilidades adicionales para exportación automatizada de archivos localmente y mediante FTP.
- Acceso a la red StrikeStar⁴; comunidad que es una red de detección de relámpagos.
- Actualizaciones del software de por vida.

Para que los programas descritos previamente trabajen correctamente es necesario instalar los drivers correspondientes. Si se utilizan dispositivos externos que se conectan mediante USB (Universal Serial Bus) al computador, generalmente no es necesaria la instalación de ningún driver pues son detectados automáticamente. Cuando se utilice un dispositivo de pre-procesamiento tipo PCI, será necesario instalar los drivers correspondientes para la PC detecte la tarjeta correctamente. La versión de la interfaz software utilizada (StormTracker/NexStorm Lite/NexStorm) determina que versión del driver será necesario instalar. Con NexStorm es necesario instalar el driver versión 5. Para mayor información sobre la instalación de los drivers véase el Apéndice B: Drivers de Antenas Boltek.

3.4. Opciones de Acceso a los Datos

En [28] se describen varias formas que tiene NexStorm para compartir los datos capturados por las antenas. Las opciones disponibles para acceder dichos datos se resumen a continuación:

Capturas de Pantalla (screenshots): El usuario puede activar esta opción para tomar capturas de pantalla de toda la interfaz o solo para el área de graficación de las descargas detectadas. Las capturas funcionaran incluso si la aplicación esta minimizada.

Se puede configurar el formato de la captura (BMP, PNG o JPG), el nombre de la captura, el directorio destino y el intervalo entre capturas (minutos). Con la opción *Exportación de Capturas y Datos Binarios* se pueden exportar hacia otra carpeta las capturas guardadas.

⁴www.strikestarus.com

Exportación de Capturas y Datos Binarios (data export): Con esta opción se pueden generar archivos binarios que contengan los datos de los relámpagos detectados. Los archivos tienen la información primordial manteniendo un tamaño reducido y pueden ser interpretados correctamente al expandir la información en su destino. Los archivos binarios son generados automáticamente a intervalos preestablecidos en la configuración. Se puede utilizar *Exportar Datos* o *Copia de Datos* para exportar las capturas de pantalla y/o los datos binarios generados. *Exportar Datos* permite subir los datos hacia un servidor mediante FTP (File Transfer Protocol). *Copia de Datos* permite realizar una exportación entre carpetas de acceso local.

Se pueden configurar los datos que se desean exportar: solo capturas, solo datos binarios o alternar (capturas y datos). También se puede establecer el intervalo entre exportaciones (1 a 120 minutos), la máxima antigüedad de los datos exportados (horas) y se puede activar la inclusión de un reporte sobre el seguimiento de tormentas (TRAC report) a los datos generados. Para la opción de exportación FTP se pueden configurar el servidor FTP, puerto, modo (pasivo o activo), nombre, password y directorio destino. Para la opción de exportación entre carpetas (Copia de Datos) se pueden configurar el directorio destino.

Pos-almacenamiento de archivos (post-store): Los archivos que contienen los registros de las detecciones realizadas durante todo el día son conocidos como *archivos diarios*. Los archivos diarios son finalizados a la media noche de cada día, y se crea un nuevo archivo para el siguiente día. El usuario puede activar esta opción para mover un archivo de datos completo hacia un directorio diferente.

Se puede configurar el directorio destino para almacenar los archivos, pero sólo serán movidos después de ser finalizados. Es decir, sólo serán movidos después de la media noche.

Flujo de Datos Compartido (FlashGate): El usuario puede activar la opción FlashGate, la cual permite a aplicaciones externas poder acceder a un flujo de datos en tiempo real. Dicho flujo de datos proporciona acceso directo a los datos de los relámpagos detectados (Strike data) y a los datos sobre el Seguimiento de tormentas (TRAC data). La opción solo se puede activar, para permitir el acceso; o desactivar, para denegar el acceso y conservar los recursos del sistema.

CAPÍTULO 3. DETECCIÓN DE RELÁMPAGOS: ANTENAS BOLTEK

Tabla 3.2: Comparación de ventajas y desventajas entre los diferentes tipos de acceso a los datos de detección ofrecidos por NexStorm.

Tipo	Ventajas	Desventajas
Screenshots	Funciona con aplicación minimizada. Configuración de intervalo entre capturas y carpeta destino. Exportación mediante <i>Data Export</i> .	No podemos interactuar con los datos directamente (imagen).
Data export	Archivos binarios compactos. Muchas opciones sobre la exportación. Soporta exportación local y FTP.	Intervalo mínimo de exportación de 1 minuto.
Post-store	Datos del día concentrados en un archivo. Se proporcionan estadísticas del día.	Exportación sólo al final del día. Sólo soporta la exportación local.
FlashGate	Acceso en tiempo real según la disponibilidad de nuevos datos.	Datos efímeros. Complejidad de acceso a datos.

El flujo de datos es establecido a través de IPC (Inter-Process Communication) de Windows. Los datos son almacenados en una sección de memoria compartida del sistema. Esta sección de memoria es sobrescrita cada vez que nuevos datos estas disponibles. Las aplicaciones que deseen acceder al flujo deberán utilizar la API (Application Programming Interface) de Windows para solicitar el acceso.

Debido al grado de configuración que posee cada opción de acceso a los datos, cada una presenta alguna ventaja. Así mismo, la falta de alguna configuración puede traer desventajas al momento de manipular los datos. En la Tabla 3.2 se resumen las ventajas y desventajas de cada una de las opciones.

Con la opción *screenshots* no se puede interactuar con los valores de los datos directamente, o se sería necesario un procesamiento diferente. La opción *post-store* permite tener todos los datos del día concentrados, pero tendríamos que esperar al final del día (medianoche) para poder acceder a los datos. Por otro lado la opción *Data Export* permite tener una representación mas compacta de los datos, y con un menor tiempo de espera para acceder

a los datos. Sin embargo con la opción *FlashGate* se puede tener acceso a los datos de forma casi instantánea cada vez que se detectan nuevos relámpagos, a pesar de que es necesario un procedimiento adicional para acceder a dichos datos. Por lo anterior, es conveniente el utilizar la opción *FlashGate* para acceder a los datos, y procesarlos tan pronto como estén disponibles.

3.5. Formato de los datos en FlashGate

NexStorm provee dos espacios de memoria para compartir los datos generados, uno para los relámpagos detectados (IPC-Strikes) y otro para los reportes del seguimiento de tormentas (IPC-TRAC). No es de interés para el presente trabajo los datos generados por TRAC (Thunderstorm Rangin and Acquisition subprocess)⁵ por lo tanto solo se presenta el formato de los datos para IPC-Strikes. Antes de poder utilizar la opción de *FlashGate* esta se debe activar desde NexStorm en la opción *FlashGate IPC* bajo el menú *Options* de la aplicación.

Los datos son representados como una cadena de caracteres (string) con longitud de 1024 caracteres. Dicha cadena de caracteres contiene diversos campos separados por coma (.). Cada campo representa un dato en específico sobre el relámpago detectado; al conjunto de todos los campos se le llama registro. En la tabla 3.3 se presentan cada uno de los campos que componen un registro ([28] pp 53). Existen tres tipos registros que se pueden presentar:

1. Un *dato*, producto de una detección efectiva de un relámpago.
2. Un *heartbeat*, un registro producido por NexStorm cada 1.5 min para indicar que sigue activo el acceso a FlashGate.
3. *Ruido*, producido por una incorrecta detección de un relámpago, la cual es clasificada como una señal no relacionada con un relámpago.

La forma de diferenciar entre estos tres tipos registros es por medio del valor de sus campos. Es ruido si los campos 6, 7, 8 o 9 tienen un valor de -1. Es un pulso de vida si cualquiera de los campos, excepto 5 y 8, tiene un valor de -9 (o 0 para enteros sin signo). Si no entra en alguna de las categorías anteriores entonces es un dato. En la Tabla 3.4 se presentan ejemplos de los diferentes tipos de registros.

⁵ TRAC (Thunderstorm Rangin and Acquisition subprocess) representa un subproceso de NexStorm que permite mejorar la exactitud de la detección de los relámpagos.

CAPÍTULO 3. DETECCIÓN DE RELÁMPAGOS: ANTENAS BOLTEK

Tabla 3.3: Descripción de los diferentes campos que componen un registro de datos.

	Tipo	Nombre	Valor	descripción
1	uint	count	0-4294967294	Contador del numero de registro de un relámpago detectado.
2	uint	year	-	Año de la detección.
3	uint	month	-	Mes de la detección.
4	uint	day	-	Día de la detección.
5	uint	timestamp_secs		Segundo de la detección, medido en segundos transcurridos desde la media noche.
6	int	TRACbearing	0-3600	Angulo en grados ^a , corregidos por TRAC.
7	int	TRACdistance		Distancia en km, corregidos por TRAC.
8	int	RAWbearing	0-3600	Angulo en grados ^a , sin corregir.
9	int	RAWdistance		Distancia en km, sin corregir.
10	int	TRAC_X	0-1000	genéricos
11	int	TRAC_Y	0-1000	genéricos
12	int	RAW_X	0-1000	genéricos
13	int	RAW_Y	0-1000	genéricos
14	int	StrikeType	-1,0,1	Tipo de descarga: desconocida(-1), CG(0) o IC(1)
15	int	StrikePolarity	-1,0,1	Polaridad de la descarga: desconocida(-1), positiva(0) o negativa(1)

int: entero.

uint: entero sin signo.

^aLos campos *bearing expresan su valor con un decimal, de tal forma que un valor de 238 representa 23.8° y un valor de 3600 representa 360.0°

CAPÍTULO 3. DETECCIÓN DE RELÁMPAGOS: ANTENAS BOLTEK

Tabla 3.4: Ejemplos de los diferentes tipos de registros que se pueden presentar.

Tipo de registro	Ejemplos
Datos	512,2018,6,15,187,942,625,942,370,999,539,807,523,1,1 122,2018,6,13,77776,1050,580,1050,633,967,625,999,636,-1,-1 ^a
Heartbeats	133,0,0,0,180,-9,-9,360,-9,-9,-9,-9,-9,-9 138,0,0,0,733,-9,-9,360,-9,-9,-9,-9,-9,-9
Ruido	1379,2018,6,15,296,-1,-1,-1,-1,0,0,0,-1,-1 1461,2018,6,15,4262,-1,-1,-1,-1,0,0,0,-1,-1

^aEstos registros sólo se presentan con el dispositivo externo LD250, debido a que no se proporciona información sobre el tipo y polaridad de los relámpagos detectados.

Dado que solo existe un espacio reservado, cada vez que una nueva información está disponible se sobrescribe la última. Por lo anterior, solo se tiene acceso al último registro escrito en la sección de memoria, y es necesario realizar una copia de los datos disponibles hacia otra sección de memoria para evitar su pérdida cuando se escriban los nuevos datos. Mientras que no se produzcan nuevos datos, el registro escrito en el espacio reservado se conserva inmutable. NexStorm recomienda que se realice una lectura de la sección de memoria compartida cada 15 milisegundos, esto para evitar pérdidas de información debido a la sobrescritura. Para información adicional véase [28].

3.6. Comentarios finales

Durante este capítulo se reviso como se realiza la detección de los relámpagos mediante las antenas Boltek, y cómo se puede acceder a dichos datos. Se inicia revisado los diferentes tipos de antenas Boltek y describiendo sus características principales. Además, se presentan las antenas que se utilizaran durante el presente trabajo: LD250 y StormTracker. Se analizan y describen los diferentes componentes físicos (hardware) y lógicos (drivers y software) que son necesarios en su conjunto para poder cumplir con la tarea de detección. Todo este conjunto de dispositivos son agrupados bajo la definición de *nodo*, el cual tiene como características principales su localización y su área de detección. Mientras que la localización es el lugar donde se encuentra el nodo, el área de detección se refiere al área alrededor de su localización en la cual puede detectar relámpagos. Posteriormente, se resumen y analizan las

CAPÍTULO 3. DETECCIÓN DE RELÁMPAGOS: ANTENAS BOLTEK

diferentes opciones, ofrecidas por la interfaz software NexStorm, mediante las cuales se puede acceder a los registros de las detecciones de relámpagos realizadas por un nodo. Se muestran y comparan las ventajas y desventajas que nos ofrece cada opción, siendo la opción FlashGate la más conveniente a pesar de requerir un poco de mayor esfuerzo. Finalmente, se describe el formato de los datos presentados por FlashGate-Strikes. Se establece un *registro* como el conjunto de datos, entregados por FlashGate-strikes, que corresponden con una detección de un relámpago. Finalmente, se muestran y explican los campos que conforman un registro, y como diferenciar entre los diferentes tipos de registros (dato, ruido y heartbeat) mediante el valor de sus campos.

Capítulo 4

El Tiempo Global en Sistemas Distribuidos

Durante el presente capítulo se revisara el concepto de tiempo global en los sistemas distribuidos. Se inicia analizando los problemas que se presentan en un sistema distribuido que intenta compartir una misma referencia de tiempo. Después se revisa la relación entre el tiempo y el orden de un conjunto de eventos. Se presentan los relojes físicos, se describen sus propiedades y los fallos que pueden presentar. Posteriormente se describen las propiedades en la medición del tiempo, se presentan el tiempo global y se analiza el alcance que tiene dentro de un sistema distribuido. Se describe el funcionamiento y las características de la sincronización interna y externa. Finalmente, se presentan los formatos de tiempo externo más importantes en la actualidad, NTP (Network Time Protocol) y IEEE 1588, y se describen sus principales características. Una gran parte del contenido de este capítulo fue extraído de [29], en donde se puede encontrar una explicación más detallada.

4.1. Introducción

Un **sistema distribuido** (SD) es una colección de computadoras separadas físicamente y conectadas entre sí por una red de comunicaciones, que se perciben en su conjunto como un único sistema. Cada máquina posee sus propios componentes de hardware y software, pudiendo tener diferente cantidad y tipo de componentes, de forma que cada una de ellas se percibe como un nodo que pertenece al SD. Cuando el SD trabaja con tareas que invo-

CAPÍTULO 4. EL TIEMPO GLOBAL EN SISTEMAS DISTRIBUIDOS

lucran el valor del tiempo, es necesario que cada nodo cuente con un reloj local que permita al nodo en cuestión determinar el valor del tiempo dentro del SD. Aun cuando todas las computadoras tuvieran algún tipo de reloj para medir el tiempo, estos relojes tienen una exactitud y precisión limitada. Por lo anterior, el valor de tiempo que un nodo percibe de su reloj local podría diferir del *valor real del tiempo*. Por ejemplo, un nodo podría tener mal configurado el mes, día u otro valor de referencia local. Más aun, dos diferentes nodos que consulten su reloj local al mismo instante de tiempo (en tiempo real) podrían observar diferentes valores de tiempo entre sus valores locales, y entre sus valores locales y el valor real del tiempo. Este sesgo en la percepción del tiempo entre los nodos se podría solucionar si todos los nodos tuvieran un reloj común, al cual todos pudieran acceder para consultar el tiempo actual. Además, este reloj común tendría que entregar su valor de tiempo de forma *inmediata* a cada nodo, esto es, sin ningún retraso de tiempo entre la lectura del valor del reloj y la notificación recibida por el nodo. Tal retraso en la entrega del valor de tiempo introduce un error en la percepción de tiempo para cada nodo, y si además el retraso es diferente para cada nodo, las cosas no mejoran mucho. Otro problema con los relojes es que también podrían desviarse del tiempo real, aunque sea por una cantidad mínima (segundos, microsegundos,...), cuando operan libremente sobre grandes periodos de tiempo. Por lo cual, si logramos hacer que el reloj común entregue su valor de tiempo sin retraso, este valor todavía podría diferir del valor real de tiempo debido a la desviación.

Tradicionalmente, la sincronización de relojes fue tratada en el contexto de un grupo de nodos iguales, cada uno teniendo un reloj local equivalente, con exactitud y desviación conocida [30]. El objetivo en tal sistema era diseñar un protocolo aceptable por el cual los relojes se podrían mantener tan cerca como fuera posible al tiempo real, y con el cual la tendencia de relojes individuales de desviarse (respecto unos con otros y/o con el tiempo real) podría ser controlada. Para lograr esto, los procesos involucrados deberían intercambiar periódicamente su lectura de tiempo, ejecutando un protocolo por medio del cual un reloj de software podría ser construido teniendo sustancialmente mejores propiedades que cualquiera de los participantes individuales. Sin embargo, un problema obvio es la precisión que pueden alcanzar los algoritmos de sincronización de relojes. Los parámetros clave para cualquier protocolo de sincronización, son la latencia esperada y la latencia máxima en las comunicaciones del sistema. Puede verse que esos valores limitan la calidad de la sincronización del reloj que se puede alcanzar en el sistema,

dado que se introduce incertidumbre en los valores intercambiados entre procesos. Por ejemplo, si la latencia de la comunicación entre p y q es conocida por variar en el rango $[0, \varepsilon]$, cualquier registro de lectura de tiempo que p transmita a q podría potencialmente ser ε unidades de tiempo más antiguo que al tiempo que fue recibido por q [30] .

Otro enfoque podría no tomar en cuenta las diferencias de tiempo entre los diferentes relojes del grupo, pero podría enfocarse en al menos poder determinar el orden parcial o total de un conjunto de eventos detectados por los múltiples nodos. Desde este enfoque la sincronización de los relojes es secundaria, y se da prioridad a determinar el orden de los eventos: X sucedió antes de Y , X sucedió después de Y o X y Y ocurrieron al mismo tiempo. De esta forma se pueden determinar cuándo diferentes registros de eventos detectados por múltiples nodos pertenecen a un solo evento, cuando las detecciones ocurrieron al mismo tiempo, o determinar que son eventos separados y establecer su orden de ocurrencia, la detección de e sucedió antes de la detección de f o la detección de f sucedió después de la detección de e .

4.2. El orden y el tiempo

En una aplicación distribuida común, el sistema de computadoras distribuido realiza una multitud de diferentes funciones al mismo tiempo, que normalmente son ejecutadas en diferentes nodos. Para garantizar el comportamiento consistente en un sistema distribuido completo, se deberá garantizar que todos los nodos procesen todos los eventos con el mismo orden de consistencia, preferiblemente en el mismo orden temporal en el cual los eventos ocurren en el objeto controlado y/o censado. Cuando el objeto censado por un sistema distribuido es el mundo real, es igual de importante poder establecer un orden de los eventos detectados, y además, poder identificar aquellos que ocurren simultáneamente. El desempeño de los dispositivos medidores de tiempo, los relojes, son de vital importancia cuando las estampas de tiempo son utilizadas para determinar la relación de orden.

4.2.1. El orden lógico

Diversos algoritmos han sido propuestos para poder reconstruir el orden de un conjunto de eventos que ocurren en el tiempo. Leslie Lamport propuso un modelo de tiempo lógico. Tal modelo consideraba un conjunto de procesos

que interaccionaban por medio del intercambio de mensajes y tenían un reloj global C que asigna un número $C_i(e)$ a cada evento del sistema. Sea C el conjunto de relojes C_i , tal que cada C_i es una función la cual asigna un número $C_i(a)$ a cualquier evento a que ocurre en ese proceso P_i . Teniendo las siguientes condiciones para cualesquiera dos eventos a, b :

1. si a y b son eventos en el proceso P_i , y a sucede antes que b , entonces $C_i(a) < C_i(b)$
2. Si a es el envío de un mensaje por el proceso P_i y b es la recepción de ese mensaje por parte del proceso P_j , entonces $C_i(a) < C_j(b)$.

se podía obtener un orden parcial de los eventos del sistema que ocurrían entre cada ciclo envío-recepción de mensajes.

El principal razonamiento es que, si un evento a ocurre antes que un evento b , entonces a debería suceder en un tiempo anterior que b . Por lo anterior, si un mensaje se envía en el tiempo s entonces se tiene que recibir en el tiempo r tal que $s < r$, es decir, un mensaje no se puede enviar hacia el pasado o un mensaje no se puede recibir antes de ser enviado. La sincronización del sistema es completada por un intercambio de mensajes entre todos los nodos que componen el sistema, pudiendo corregir su reloj local cada vez que se reciben un mensaje. Por todo lo anterior, el tiempo se vuelve abstracto, representado por una secuencia (representación del tiempo) que es continua e incremental. De esta forma, el número asignado a un evento representa el tiempo en que ocurrió. Así pues, el método se enfocada en el orden en el cual ocurren los eventos, de tal forma que su representación de tiempo (lógico) no se tiene relación alguna con el valor del tiempo real (físico) [31].

El método de Lamport es útil cuando los eventos son generados a través de los nodos, y se requiere establecer una relación de orden entre los eventos de todos los nodos sin importar el valor del tiempo real. Para situaciones en las que los nodos detectan un conjunto de eventos, que tienen una relación directa entre su tiempo de ocurrencia y el valor del tiempo real, existen mejores alternativas para determinar tal relación de orden.

4.2.2. El orden temporal

Si cada evento tiene asociada una estampa del tiempo real que representa el instante de ocurrencia de dicho evento, entonces todos los eventos se pueden ordenar temporalmente, dado que el tiempo es continuo. La continuidad del

CAPÍTULO 4. EL TIEMPO GLOBAL EN SISTEMAS DISTRIBUIDOS

tiempo real Newtoniano se puede modelar como una línea de tiempo que consiste de un conjunto infinito T de *instantes* (o *puntos en el tiempo*) con las siguientes propiedades:

1. T es un conjunto ordenado, esto es, si p y q son dos instantes cualesquiera, entonces p es simultáneo con q , o p precede a q , o q precede a p , donde esas relaciones son mutuamente excluyentes. Al orden de instantes sobre la línea de tiempo le llamamos el *orden temporal*.
2. T es un conjunto denso. Esto significa que al menos hay un q entre p y r , si y solo si p no es el mismo instante que r , donde p, q , y r son instantes.

Una sección de la línea de tiempo entre dos instantes es llamada **duración**. En este modelo, un **evento** toma lugar en un instante de tiempo y no tiene duración. Si dos eventos ocurren al mismo *instante*, entonces se dice que los dos eventos ocurrieron *simultáneamente*. Los instantes están totalmente ordenados; sin embargo, los eventos están solo particularmente ordenados, dado que los eventos simultáneos no tienen una relación de orden. Los eventos pueden ser totalmente ordenados si algún criterio adicional es introducido para ordenar los eventos que ocurren simultáneamente. Por ejemplo, el número de nodo donde ocurrió el evento.

4.2.3. Los relojes físicos

Un **reloj digital físico** es un dispositivo para la medición del tiempo. Este contiene un *contador* y un *mecanismo de oscilación física* el cual periódicamente genera un evento para incrementar el contador. El evento periódico que incrementa el contador interno es llamado el *microtick* del reloj. La duración entre dos microticks consecutivos de un reloj digital físico es llamada la **granularidad** (granularity) del reloj. La granularidad de un reloj puede ser medida solo si se encuentra disponible un reloj con una granularidad más fina. La granularidad de cualquier reloj digital tiende a experimentar un error durante el proceso de digitalizar la medición del tiempo.

En subsecuentes definiciones, se usara la siguiente notación: Los relojes son identificados por números naturales $1, 2, 3, \dots, n$. Las propiedades de los relojes se expresaran con el identificador del reloj como un superíndice junto con su valor de microtick, o tick, como un subíndice. Por ejemplo, el microtick i del reloj k es denotado por $microtick_i^k$.

CAPÍTULO 4. EL TIEMPO GLOBAL EN SISTEMAS DISTRIBUIDOS

Asumamos que tenemos un **reloj de referencia global Z** con una frecuencia que está perfectamente en sincronía con el estándar de tiempo internacional. También, asumamos que tiene frecuencia muy grande (digamos 10^{15} microticks/segundo), de tal forma que su granularidad sea demasiado pequeña (10^{-15} segundos). Al tener un granularidad tan pequeña, el error de digitalización del reloj de referencia es considerado un efecto de segundo orden y por lo tanto es ignorado.

Supongamos que cada vez que ocurre un evento e , instantáneamente se registra el estado del reloj de referencia como el tiempo de ocurrencia del evento, generando así, una **estampa de tiempo** para e . Sea un reloj k y un evento e , $k(e)$ denota la generación de la estampa de tiempo para el evento e usando el reloj k . Dado que Z es el reloj de referencia global, $Z(e)$ es llamada la *estampa de tiempo absoluta* del evento e .

La **duración entre dos eventos** es medida por el conteo de los microticks del reloj de referencia que ocurren durante el intervalo entre esos dos eventos. La *granularidad* g^k de un reloj k puede ahora ser representada por el número nominal n^k , donde n es la cantidad de microticks que ocurren en el reloj de referencia Z entre 2 microticks del reloj k .

El orden temporal de los eventos que ocurren entre cualesquiera dos microticks consecutivos del reloj de referencia, esto es, dentro de la granularidad g^z , no puede ser restablecido desde sus estampas de tiempo absolutas. Este es el límite fundamental de la medición del tiempo.

La **desviación** (drift) de un reloj físico k entre el microtick i y el microtick $i + 1$ es la frecuencia proporcional entre este reloj k y el reloj de referencia, al instante del microtick i . La desviación es determinada por la medición de un ciclo del reloj k utilizando el reloj de referencia Z y dividiendo eso entre el número nominal n^k microticks de referencia en un ciclo del reloj k .

$$drift_i^k = \frac{Z(\text{microtick}_{i+1}^k) - Z(\text{microtick}_i^k)}{n^k} \quad (4.1)$$

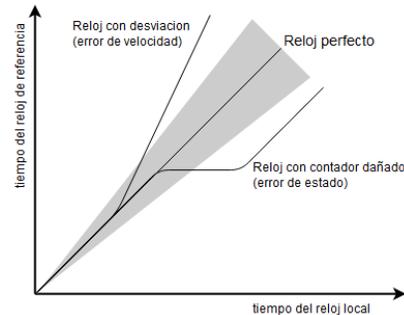
Dado que un buen reloj tiene una desviación muy cercana a 1, por conveniencia de la notación la velocidad de desviación p_i^k es introducida como:

$$p_i^k = \left| \frac{Z(\text{microtick}_{i+1}^k) - Z(\text{microtick}_i^k)}{n^k} - 1 \right| \quad (4.2)$$

Un reloj perfecto deberá tener una velocidad de desviación de 0. Los relojes reales tienen una velocidad de desviación variable, la cual es influenciada

Figura 4.1: Modos de falla de un reloj físico (kopetz, 2011)

Un buen reloj con una acotada velocidad de desviación p se mantiene dentro del área sombreada.



por condiciones del entorno (temperatura, voltaje aplicado, edad del cristal, etc.). Debido a que cada reloj tiene una velocidad de desviación diferente de cero, los relojes que se ejecutan libremente, esto es, aquellos que nunca se han re-sincronizado, abandonan cualquier limitación de intervalo de tiempo entre microticks después de un tiempo finito, incluso si son completamente sincronizados al inicio de su ejecución.

Los dos **tipos de fallas** que puede presentar un reloj físico (Figura 4.1), y que afectan directamente su exactitud, son:

Falla de valor: El contador podría ser dañado de tal forma que el valor del contador se vuelve erróneo.

Falla de desviación: La velocidad de desviación propia del reloj podría alejarse demasiado de la desviación de velocidad especificada (área sombreada) debido a que el reloj aumenta su contador más rápido (o más lento) que su especificación de fábrica.

4.3. La medición del tiempo

Un sistema distribuido está compuesto por varios nodos, cada uno con su propio reloj para medir el tiempo, y deberán mantener sincronizado su tiempo. Si los relojes locales de los nodos están perfectamente sincronizados con el reloj de referencia Z , entonces debería ser fácil el reconstruir el orden temporal de los eventos a partir de sus estampas de tiempo. En un sistema distribuido débilmente acoplado donde cada nodo tiene su propio oscilador local (reloj local), una completa sincronización de relojes no es posible. Por esto es necesario establecer criterios que nos permitan medir la calidad de

sincronización que logran alcanzar los relojes locales. Aunque no se puede lograr una completa sincronización, aún es posible establecer una noción de tiempo consensuada entre todos los nodos. El concepto de tiempo global es entonces introducido en el SD como una débil noción de tiempo que funciona como referencia universal.

4.3.1. Precisión y exactitud

El **error de desviación** (offset) al microtick i entre dos relojes j y k con la misma granularidad es definida como:

$$offset_i^{jk} = \left| Z(microtick_i^j) - Z(microtick_i^k) \right| \quad (4.3)$$

El error de desviación denota la diferencia de tiempo entre los respectivos microticks de los dos relojes, medidos en el número de microticks del reloj de referencia.

Dado un conjunto de n relojes $1, 2, \dots, n$, el error de desviación máximo entre cualesquiera dos relojes del conjunto

$$\Pi_i = \max_{\forall 1 \leq j, k \leq n} \{offset_i^{jk}\} \quad (4.4)$$

es llamada la **precisión** Π del conjunto al microtick i . EL valor máximo de Π_i , sobre un *intervalo de interés* es llamado la *precisión* Π del conjunto. La precisión denota el *error de desviación máximo* entre cualesquiera dos relojes del conjunto durante un intervalo de interés. La precisión es expresada en el número de microticks del reloj de referencia.

Debido a la velocidad de desviación de cualquier reloj físico, los relojes de un conjunto tenderán a desviarse entre ellos si no son re-sincronizados periódicamente, para mantener una precisión acotada entre ellos. Al proceso de resincronización mutua dentro del conjunto de relojes, para mantener una precisión acotada, es llamado **sincronizan interna**.

Al error de desviación que presenta un reloj k con respecto al reloj de referencia Z al microtick i se le conoce como la **exactitud** (accuracy) de k al microtick i , denotada como $accuracy_i^k$. El máximo error de desviación sobre todos los microticks i en un intervalo de interés es llamado la exactitud del reloj k , denotado como $accuracy^k$. La exactitud denota el máximo error de desviación de un reloj comparado con una referencia de tiempo externa en un intervalo de interés.

Para mantener un reloj de forma acotada dentro de un intervalo del reloj de referencia, este deberá ser periódicamente re-sincronizado con un tiempo de referencia externo. Este proceso de resincronización de un reloj con una referencia de tiempo externa es llamado **sincronización externa**. Si todos los relojes de un conjunto son sincronizados externamente con una exactitud E , entonces el conjunto es también internamente sincronizado con una precisión de al menos $2E$. Lo contrario no es verdadero. Un conjunto de relojes internamente sincronizados deberá desviarse del tiempo externo si nunca son re-sincronizados con el tiempo base externo.

4.3.2. Tiempo global

Supongamos que tenemos un conjunto de nodos, cada uno con su propio reloj físico local C^k que se desempeña con una granularidad g^k . Asumamos que todos los relojes están sincronizados internamente con precisión Π , esto quiere decir que para cualesquiera dos relojes j, k , y todos los microticks i

$$\left| Z(\text{microtick}_i^j) - Z(\text{microtick}_i^k) \right| < \Pi \quad (4.5)$$

Entonces es posible seleccionar un *subconjunto de microticks* de cada reloj local k para la generación de la implementación local de una noción global de tiempo. Tal selección de los microticks locales i es llamado un *macrotick*, o un *tick* de tiempo global. Por ejemplo, cada diez microticks de un reloj local k pueden ser interpretados como un tick global, macrotick t_i^k , de este reloj (Figura 4.2). Para especificar el suceso de un macrotick i , sin importar en cual reloj ocurre, se denotara como t_i sin un superíndice. Un tiempo global es una *noción abstracta*, la cual es *aproximada* a partir de la selección apropiada de microticks de los relojes locales de un conjunto.

Condición de aceptabilidad (Reasonableness Condition)

El tiempo global t es llamado *acceptable* si todas las implementaciones locales de tiempo global satisfacen la **condición de aceptabilidad**:

$$g > \Pi \quad (4.6)$$

donde g es la granularidad global.

Esta condición de aceptabilidad asegura que el error de sincronización está acotado por un *macrociclo*, esto es, está acotado por la duración entre

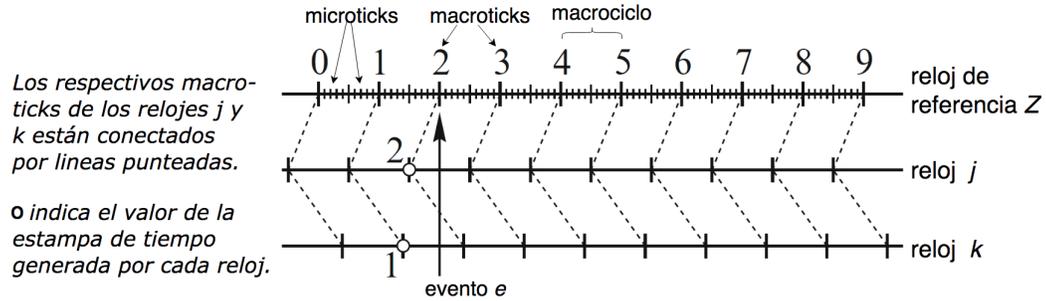


Figura 4.2: Estampas de tiempo generadas para un único evento detectado desde dos diferentes nodos (kopetz, 2011).

dos macroticks. Si esta condición de aceptabilidad es satisfecha, entonces para un único evento e que es observado por cualesquiera dos relojes del conjunto tenemos que:

$$|t^j(e) - t^k(e)| \leq 1 \quad (4.7)$$

esto es, las estampas de tiempo global para el evento e puede diferir por máximo un tick. *Esto es lo mejor que podemos obtener.* Debido a que es imposible sincronizar perfectamente los relojes, y a la granularidad de cualquier tiempo digital, siempre existe la posibilidad de que ocurra la siguiente secuencia de eventos:

1. El reloj j marca su tick i .
2. El evento e ocurre.
3. El reloj k marca su tick i .

En tal situación, el único evento e obtiene su estampa de tiempo por los dos relojes j y k con una diferencia de un tick (Figura 4.2): $j(e) = t_i = 2$ y $k(e) = t_{i-1} = 1$.

En la Figura 4.3 se presentan cuatro eventos: evento 17 (e_{17}), evento 42 (e_{42}), evento 67 (e_{67}) y evento 69 (e_{69}). Aunque la duración entre e_{17} y e_{42} es de 25 microticks, y la duración entre e_{67} y e_{69} es solo de 2 microticks, ambas duraciones tiende hacia la misma diferencia de un macrotick. La estampa de tiempo global para e_{69} es *menor* que la estampa de tiempo global para e_{67} , aunque e_{69} ocurrió *después* de e_{67} . Debido a la acumulación del error de sincronización y al error de digitalización, no es posible reconstruir el orden temporal de dos eventos a partir del conocimiento de que su tiempo global difiere en un tick. Sin embargo, si la estampa de tiempo de dos eventos difieren

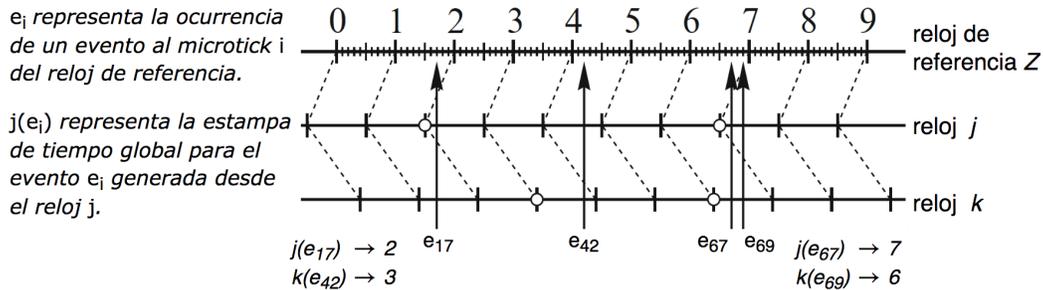


Figura 4.3: El orden temporal de dos eventos que presentan un tick de diferencia en sus estampas de tiempo (kopetz, 2011).

por dos ticks, entonces el orden temporal puede ser reconstruido debido a que la suma del error de sincronización y de digitalización es siempre menor que dos ciclos de reloj en un sistema con un tiempo global base aceptable.

Esta limitación fundamental en la medición del tiempo restringe la *confiabilidad* del modelo de computadora digital de un subsistema físico controlado. El tiempo base de la parte física de un sistema cibernético-físico es continuo, mientras el tiempo base en la parte de computación es discreto. Siempre que dos eventos en el subsistema físico ocurren juntos, comparados con la granularidad del tiempo global, no es posible el reconstruir el orden temporal físico de los eventos de forma exacta y confiable dentro del sistema computacional. La única manera de esquivar este dilema es la provisión de un tiempo global base con una granularidad más pequeña, de tal manera que los errores temporales son reducidos.

4.4. Sincronización de tiempo

Para que todos los nodos que pertenecen al sistema distribuido mantengan una referencia de tiempo equivalente, es necesario que sean sincronizados con otras fuentes de tiempo. La sincronización se puede realizar entre los miembros que pertenecen al SD, intercambiando la percepción de tiempo al interior del conjunto. También se puede sincronizar al conjunto de nodos contra una referencia de tiempo externa, revisando el valor real del tiempo desde una fuente externa al SD y corrigiendo los relojes locales en concordancia. Además es necesario realizar una resincronización periódica, con el fin de acotar el error de desviación que presenten los relojes locales.

4.4.1. Sincronización interna

La **sincronización interna** asegura que los ticks globales de un conjunto de relojes ocurren dentro de la precisión Π . La sincronización se deberá lograr independientemente de la velocidad de desviación que presente cada reloj del conjunto. Cada nodo de un SD tiene un reloj local que incrementa su contador (tick) con una frecuencia determinada por los parámetros físicos propios del oscilador del reloj. Un subconjunto de microticks del reloj local, llamado *macrotick*, es interpretado como un tick del tiempo global para ese nodo. El subconjunto de microticks (g) deberá de cumplir con la Ecuación 4.6. Cada vez que el nodo alcanza un *macrotick*, se incrementa el contador local de tiempo global de ese nodo.

Debido a que la disponibilidad de un apropiado tiempo global base es crucial, la sincronización de reloj no debería depender de la exactitud de un único reloj, pues el comportamiento malicioso de un nodo podría dar lugar a inconsistencias de tiempo. Para mejorar la consistencia de tiempo entre los relojes y reducir el impacto de nodos maliciosos, se pueden insertar rondas adicionales de intercambio de mensajes de tiempo. Sin embargo, la sincronización para un conjunto de relojes solo puede ser garantizada si se cumple la condición (Lamp85 en [29]):

$$N \geq (3k + 1) \tag{4.8}$$

donde N es el número total de relojes, y k es el número de nodos maliciosos. Cuando la información de todos los nodos es tomada en cuenta para determinar de forma consensuada el valor del tiempo del conjunto, se dice que el conjunto es **tolerante a fallos**.

4.4.2. Sincronización externa

La **sincronización externa** enlaza el tiempo global de un grupo de nodos a un estándar de tiempo externo. Para este propósito es necesario acceder a un *proveedor de tiempo*, esto es, un recurso de tiempo externo que periódicamente reporta el tiempo de referencia actual en forma de un *mensaje de tiempo*. Este mensaje de tiempo deberá disparar un evento de sincronización en un nodo designado que pertenezca al grupo y deberá identificar este evento de sincronización en una escala de tiempo conocida. Tal escala de tiempo deberá estar basada en una medida de tiempo ampliamente aceptada, por ejemplo, el segundo físico, y deberá relacionar el evento de sincronización con

CAPÍTULO 4. EL TIEMPO GLOBAL EN SISTEMAS DISTRIBUIDOS

un origen de tiempo definido, *la época*. La interfaz del nodo con un proveedor de tiempo es llamada una *pasarela de tiempo*.

La pasarela de tiempo periódicamente publica mensajes de tiempo conteniendo un evento de sincronización, así como la información para realizar esta sincronización en la escala TAI¹. La pasarela de tiempo deberá sincronizar el tiempo global de su grupo con el tiempo recibido desde la fuente de tiempo externa. Esta sincronización es unidireccional, y por esto, asimétrica. Esto puede ser utilizado para ajustar la velocidad los relojes sin la preocupación de que surjan efectos de inestabilidad. Si algún otro grupo es conectado al grupo *primario* por medio de una pasarela de tiempo secundaria, entonces, la sincronización unidireccional funciona de la misma forma. La pasarela de tiempo secundaria considera el tiempo sincronizado del primer grupo como su tiempo de referencia, y sincroniza el tiempo global del segundo grupo.

Mientras la sincronización interna es una actividad cooperativa entre todos los miembros de un grupo, la sincronización externa es un proceso autoritario: la pasarela de tiempo fuerza su percepción del tiempo externo sobre todos sus subordinados. Desde el punto de vista de la tolerancia a fallos, un régimen autoritario introduce un problema: si la autoridad envía un mensaje incorrecto, entonces todos sus *obedientes* subordinados deberán comportarse incorrectamente. Sin embargo, para la sincronización externa de relojes la situación está bajo control debido a la inercia del tiempo. Una vez que un grupo ha sido sincronizado, el tiempo global base dentro del grupo actúa como *monitor* de la pasarela de tiempo. Un mensaje de sincronización externo solo deberá ser aceptado si su contenido es suficientemente cercano al punto de vista de tiempo externo que posee el grupo. La pasarela de tiempo tiene solo una autoridad limitada para corregir la velocidad de desviación del grupo.

¹ El Tiempo Atómico internacional (TAI –Temps Atomique Internationale) es un estándar atómico de alta precisión para medir el tiempo con un reloj atómico. La invención del reloj atómico en 1948 hizo posible medir el tiempo de manera más precisa e independiente de los movimientos de la Tierra. Se definió al segundo físico como el tiempo que necesita el átomo de cesio 133 para efectuar exactamente 9.192.631.770 transiciones. Los diversos laboratorios que tienen un reloj atómico registran el número de transiciones que marcan los relojes, y el promedio de esas marcas produce el Tiempo Atómico Internacional. .

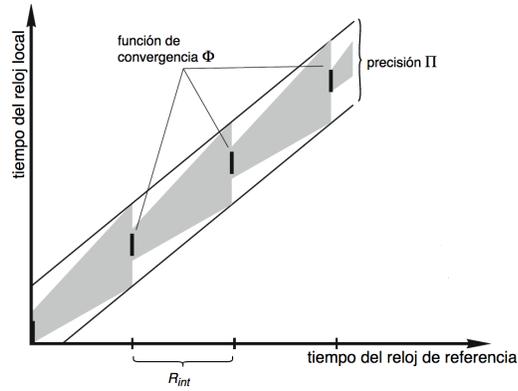


Figura 4.4: La condición de resincronización (kopetz, 2011).

Todos los relojes sincronizados correctamente deberán operar dentro del área gris, manteniendo una precisión Π .

4.4.3. La Condición de resincronización

Los contadores de tiempo global (macroticks) de cada nodo dentro del conjunto deberán ser periódicamente resincronizados para poder establecer un tiempo global base con la precisión deseada. Si un reloj pierde precisión, entonces la precisión del tiempo global del grupo, la cual depende directamente de sus integrantes, también pierde precisión. Si los relojes no son corregidos periódicamente estos tenderán a alejarse de la *referencia de tiempo*, debido a la velocidad de desviación que posee cada reloj, haciendo que el tiempo global pierda precisión. El periodo de tiempo entre cada resincronización es llamado el *intervalo de resincronización* R_{int} ; es durante este intervalo que los relojes corren libremente. Al final de cada intervalo de resincronización, los relojes se encuentran ajustados con una mejor consistencia de tiempo entre ellos. La *función de convergencia* Φ denota el error de desviación de los valores de tiempo inmediatamente después de la sincronización. Después de la resincronización, los relojes se alejan del tiempo de referencia, debido a su velocidad de desviación particular, hasta que son resincronizados al final del próximo intervalo de resincronización R_{int} (Figura 4.4). El error de desviación Γ indica la máxima desviación acumulada de cualesquiera dos relojes durante el periodo de resincronización R_{int} , durante el cual los relojes corren libremente. El error de desviación Γ depende de la longitud del intervalo de resincronización R_{int} y la velocidad de desviación máxima ρ del reloj:

$$\Gamma = 2\rho R_{int} \quad (4.9)$$

Un conjunto de relojes se considera sincronizado si la siguiente *condición de sincronización* es mantenida entre la función de convergencia Φ , el error

de desviación Γ y la precisión Π :

$$\phi + \Gamma \leq \Pi \tag{4.10}$$

Supongamos que al final del intervalo de sincronización, los relojes se han desviado tanto que se encuentran al límite del intervalo de precisión Π (Figura 4.4). La resincronización deberá restablecer la consistencia del tiempo de los relojes de tal forma que la cantidad de desviación durante el próximo intervalo de resincronización, donde corren libremente, no permitirá que un reloj del conjunto abandone el límite del intervalo de precisión.

4.4.4. Corrección de estado y corrección de velocidad

Durante la resincronización de un reloj se compara el valor de tiempo percibido localmente contra el valor de tiempo recibido externamente, de otros nodos o de un proveedor de tiempo, y se determina un valor de corrección para aplicar al contador local. El valor de corrección puede ser aplicado de dos formas:

- **Corrección de estado:** El valor es aplicado directamente al valor del tiempo local de forma inmediata. Por ejemplo, si el reloj local está atrasado 5 segundos respecto a la referencia de tiempo, entonces inmediatamente adelantamos el reloj local por 5 s, logrando corregir el reloj.
- **Corrección de velocidad:** La velocidad del reloj es modificada, para que el reloj avance más rápido o más lento durante el próximo intervalo de resincronización, de tal forma que el reloj mejore su consistencia de tiempo con el resto del conjunto. Por ejemplo, si el reloj local está atrasado 5 segundos respecto a la referencia del tiempo, entonces aumentamos la velocidad del reloj para que cuente los segundos más rápido, logrando que durante la próxima resincronización el reloj este más cerca del tiempo de referencia.

La corrección de estado es simple para aplicar, pero tiene la desventaja de generar discontinuidad del tiempo base. Si a los relojes se les aplica un adelanto del tiempo, entonces estaremos *saltando* en el tiempo, y software que dependa de un flujo de tiempo continuo podría no funcionar de forma correcta. Si a los relojes se les aplica un retroceso de tiempo y el mismo

valor nominal de tiempo es alcanzado dos veces, entonces, perniciosas fallas pueden ocurrir con software que depende del valor del tiempo. Por ejemplo, un sistema que realice transacciones todos los días a las 12:00:00 pm; si a las 11:59:59 el reloj es adelantado mínimo 2 segundos, la transacción no se realizara, y si a las 12:00:01 es atrasado mínimo 1 segundo, la transacción se realizara dos veces.

Por lo anterior, es preferible implementar la corrección de velocidad estableciendo un límite para el máximo valor de velocidad permitida, de tal forma que el error en la medición de intervalos es limitada. El tiempo global base resultante mantiene la propiedad de ser continuo a pesar de la resincronización. La corrección de velocidad puede ser implementada en el dominio digital, cambiando el número de microticks en alguno de los macroticks, o en el dominio analógico, ajustando el voltaje aplicado al cristal del oscilador. Para evitar una velocidad de desviación común de todo el conjunto de relojes, el promedio de todos los términos de corrección aplicados a los relojes deberá ser cercano a cero.

4.5. Formatos de tiempo externo

En los últimos años, un número de formatos de tiempo externo han sido propuestos para la sincronización externa de los relojes. El más importante es el estándar para el formato de tiempo propuesto en el Network Time Protocol (NTP). Este formato de tiempo con una longitud de 8 Bytes contiene dos campos: un campo de 4 Bytes para segundos completos, donde los segundos son representados acorde a el Tiempo Universal Coordinado (UTC)², y un campo para fracciones de segundo, donde la fracción de un segundo es representada como una fracción binaria con una resolución cercana a 232 ps. En enero 1 de 1972 a la media noche el reloj de NTP fue establecido a 2,272,060,800 segundos, esto es, el número de segundos desde Enero 1 de 1900 a las 00:00 hrs. El tiempo de NTP no es de tipo cronoscópico debido a que está basado en UTC el cual tiene que acomodarse para el segundo intercalar.

² El Tiempo Universal Coordinado (UTC) es el principal estándar de tiempo por el cual el mundo regula los relojes y el tiempo. Es uno de los varios sucesores estrechamente relacionados con el tiempo medio de Greenwich (GMT). El UTC se obtiene a partir del Tiempo Atómico Internacional. Debido a que la rotación de la Tierra es constante pero no estable y se retrasa con respecto al tiempo atómico, UTC se sincroniza con el tiempo

CAPÍTULO 4. EL TIEMPO GLOBAL EN SISTEMAS DISTRIBUIDOS

El estándar NTP sigue, aunque débilmente, la convención establecida por la industria telefónica, el nivel de cada servidor en la jerarquía está definido por un número de *estrato* (stratum). Los servidores primarios tiene asignado el estrato uno. Para cada uno de los servidores secundarios que se encuentran a por debajo de los primarios se les asigna un numero de estrato más grande, en una unidad, que su nivel predecesor. Conforme el número de estrato incrementa, su exactitud se degrada dependiendo de la congestión de la red, el camino utilizado, y la estabilidad del reloj local del sistema (Mills 2010 en [32]).

NTP es utilizado en el Internet de hoy en día con computadoras que van desde PC personales hasta super-computadoras. Provee una exactitud generalmente en el rango desde 0.1 ms, con computadoras y conexiones LANs rápidas, hasta unas pocas decenas de milisegundos con el Internet intercontinental. En general, la exactitud que se puede alcanzar con NTP depende de varios factores externos [32][33]. En [32] se presentan la arquitectura, el protocolo y los algoritmos utilizados por NTP, así como también los resultados de investigación y publicaciones relacionadas.

Otro formato de tiempo es el protocolo de tiempo de precisión (PTP –Precision Time Protocol) definido en el estándar IEEE 1588. En este formato de tiempo la época inicia en enero 1 de 1970 a las 00:00 hrs o es definida por el usuario. Los segundos completos son contados en concordancia con TAI, mientras que las unidades de la fracción de segundo son los nanosegundos. Esto causa un cambio abrupto en la representación cada vez que un segundo completo es alcanzado. El IEEE 1588 está diseñado para sistemas cuyas especificaciones no puedan ser satisfechas de forma apropiada por cualquiera de los dos protocolos dominantes, NTP y GPS. Es adecuado para entornos industriales que cuenten con redes pequeñas y bien diseñadas. Permite una sincronización precisa de los relojes en redes con capacidad de multidifusión, como Ethernet. Puede alcanzarse una precisión de nanosegundos utilizando en los paquetes de datos las estampas de tiempo generadas por el hardware.

Una estampa de tiempo generada por software pueden alcanzar una precisión desde 10 y 100 microsegundos, dependiendo del entorno[33], y puede ser implementado fácilmente utilizando dispositivos de red comunes, como los switchs estándar. Por otro lado, las estampas de tiempo generadas por hardware pueden alcanzar precisiones de 30 y 8 nanosegundos. Sin embargo, para

medio de Greenwich (obtenido a partir de la duración del día solar), al que se le añade o quita un segundo intercalar cuando resulta necesario.

poder obtener esta resolución es necesario que tanto el servidor de tiempo como el cliente tengan dispositivos para poder generar estampas de tiempo por hardware. Esto significa que se necesita comprar hardware especializado por cada dispositivo que se quiera sincronizar. Las pasarelas de red, tal como switches, deberán estar configuradas correctamente para poder proveer la capacidad de sincronización superior y se deberá tener una topología de red estable. PTP es altamente dependiente de la configuración de la red, pues si las configuraciones no son correctas, la sincronización de PTP sólo alcanzara una precisión de NTP.

4.6. Comentarios finales

En el trascurso del capítulo se reviso las bases para la medición del tiempo desde un reloj local, y se analizó la forma en que se percibe el tiempo desde un sistema compuesto por un conjunto de relojes. El tiempo global representa la percepción de tiempo para un conjunto de relojes, permitiendo tener una misma noción de tiempo para todos los nodos. Se concluye que para poder establecer un tiempo global aceptable g , la precisión Π de los relojes del conjunto deberán cumplir con: $g > \Pi$. Además, se establece que para poder establecer una consistencia entre las diferentes percepciones del tiempo de los distintos relojes que conforman el sistema es necesario sincronizarlos internamente, entre ellos, o sincronizarlos externamente, con fuentes externas de tiempo. Si se utiliza la sincronización externa con una precisión E entonces también se obtiene una sincronización interna con precisión $2E$. Se concluye así que los relojes que pertenezcan al conjunto deberán ser resincronización periódicamente para asegurar que no se desvíen demasiado de la referencia, esto es, se mantengan bajo una precisión acotada. Además, también es preferible realizar un corrección de velocidad a los relojes, debido a que nos permite mantener un tiempo continuo. Finalmente se describen dos principales protocolos para la sincronización de relojes a través de una red, el NTP y IEEE 1588. Mediante estos protocolos se pueden usar algoritmos de sincronización que mantengan los relojes de los nodos acotados a una determinada precisión. NTP es el protocolo más utilizado, y por tanto el más probado en diferentes entornos, mientras que el IEEE 1588 es más reciente y más preciso, pero también el menos extendido. IEEE 1588 surgió como una alternativa a NTP con el objetivo de mejorar la precisión proporcionada, hasta nanosegundos, pero también necesita mayores requisitos de hardware

CAPÍTULO 4. EL TIEMPO GLOBAL EN SISTEMAS DISTRIBUIDOS

y, una topología red estable y predecible. Tanto NTP como IEEE 1588 son utilizados actualmente dependiendo del grado de precisión que se requiera y de la capacidad para establecer un entorno aceptable de funcionamiento.

Capítulo 5

Sistema Propuesto

En el presente capítulo se describe de forma general sobre el funcionamiento del sistema. Se inicia con un resumen sobre los conceptos clave que se han revisado previamente. Se describe el contexto necesario para poder recolectar los datos, la disponibilidad del acceso a FlashGate y la configuración necesaria para obtener una buena precisión en las estampas de tiempo. Se presenta el algoritmo de recolección de los registros, y el procesamiento local de los registros. Después se describe la etapa de procesamiento de los registros, es decir, su alineación. Se analizan los criterios que se necesitaran utilizar para poder alinearlos y se presenta el algoritmo de alineación propuesto. Finalmente se presenta el formato de almacenamiento de la información resultante, y se describen los diferentes tipos de consultas disponibles.

5.1. Introducción

En el Capítulo 2 se revisó la naturaleza de los relámpagos y sus características intrínsecas. Un **relámpago** se define como una descarga producida por un centro de carga dentro de las nubes, adquirido durante el ciclo de desarrollo de la nube (Sección 2.4), y algún otro centro de carga, dentro de la misma nube o de otra nube (IC), o con la tierra (CG). Un **evento** sucede durante un instante de tiempo y no tiene duración (Subsección 4.2.2). Debido a que un relámpago es producido por una descarga entre dos centros de carga en un instante de tiempo determinado, podemos concluir que *un relámpago es un evento*. Al ser un evento, el instante de tiempo en que sucede un relámpago es una de las principales características que lo permiten identificar

dentro de un conjunto. La posición geográfica donde ocurre el relámpago es otra de las características que permite diferenciar a los relámpagos, así pues, aunque varios relámpagos pueden ocurrir simultáneamente necesariamente tendrán que ocurrir en diferentes lugares. Utilizando estas dos características se puede identificar de forma única a cada relámpago, y si además se agrega el tipo de descarga generada (IC/CG), es posible establecer las características mínimas para su análisis (Tabla 2.1).

Las antenas Boltek pueden detectar las descargas producidas por relámpagos (Capítulo 3), y permiten acceder a los registros de dichas detecciones por medio del software NexStorm (Sección 3.4). Un **nodo**, representa todo el conjunto de hardware y software necesarios para lograr la detección efectiva de relámpagos. Un **registro** (Sección 3.5) contiene la información de relevancia capturada durante una detección de un relámpago intentando representar sus principales características, es decir, un registro intenta representar al relámpago detectado. Dependiendo de las características del equipo utilizado para la detección (Tabla 3.1), la cantidad de información que es proporcionada por un registro puede variar (Tabla 3.3). El **área de detección** es el área efectiva en que el nodo tiene la capacidad de detectar los eventos de relámpago, dicho alcance está determinado por el alcance de la antena de detección del nodo (Sección 3.2). Debido a un *error de detección*, producto de la distorsión y/o degradación de la señal percibida por la antena de detección y al ruido del entorno, un nodo puede detectar una posición diferente a la posición real del relámpago o no detectarlo en absoluto. Debido a lo anterior, diferentes nodos tendrían diferentes conjuntos de detecciones realizadas según la calidad de las señales que perciben. Además, los relojes utilizados por los diferentes nodos podrían presentar diferentes comportamientos que alteran el tiempo que representan (Subsección 4.2.3).

Si se cuenta con múltiples nodos de detección y además sus área de detección se solapan (Figura 5.1), entonces cualquier evento de relámpago r que ocurra dentro de dichas zonas solapadas será detectado por máximo, el número de nodos que cubren esa área. Por todo lo anterior se puede concluir que dado un conjunto de nodos de detección D y un evento de relámpago r , las características de r , temporales y espaciales, registradas por cada $d \in D$ serán diferentes entre sí. Esto quiere decir que, cada nodo que detecte un evento de relámpago tendrá su propia percepción sobre el lugar y el tiempo en que ocurrió dicho evento. Debido a esto, cuando un conjunto de nodos con área solapada detecta un único evento, se obtendrán un conjunto de registros que intentan representar al mismo evento pero que tendrán discrepancias en

CAPÍTULO 5. SISTEMA PROPUESTO

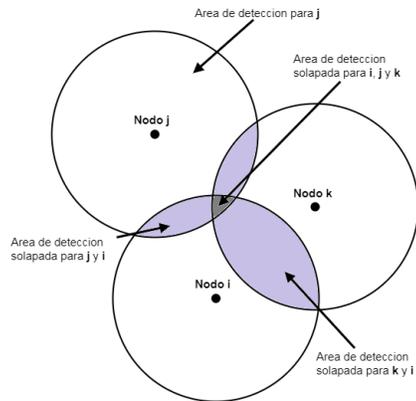


Figura 5.1: Tres nodos mostrando el solapamiento de sus áreas de detección, representadas por un círculo.

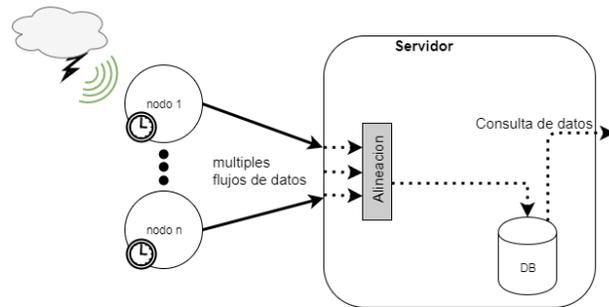


Figura 5.2: Diagrama del sistema propuesto mostrando el flujo de datos de los registros.

su información de posición y tiempo de ocurrencia. Por lo anterior, no es posible identificar a priori aquellos registros de detecciones que pertenecen a un mismo evento y diferenciar aquellos registros que no pertenecen al mismo evento. Una **alineación** de los registros de detecciones de relámpago deberá ser necesaria. Es decir, se deberá poder identificar y agrupar aquellos registros que pertenezcan a un mismo evento, de tal forma que cada grupo formado identificara un evento de relámpago y cada registro del grupo representara una detección del evento de relámpago por uno nodo.

Se cuenta con tres nodos para la detección de relámpagos, y se desea poder alinear los registros producidos por los nodos de tal forma que sea posible procesar los eventos capturados como un solo flujo de datos. Los nodos utilizan antenas Boltek para la detección de relámpagos, dos antenas con dispositivo tipo PCI y una antena con dispositivo LD250, dichas antenas tienen un alcance de 480 km (300 millas). Cada nodo se encuentran en diferentes puntos de Michoacán: Morelia, Yurecuaro y Lázaro Cárdenas. Debido a la cercanía de sus instalaciones y al alcance de las antenas, los tres nodos tienen sus áreas de detección solapadas.

El sistema que se propone a continuación, tiene como objetivo concentrar y alinear los registros de todos los nodos disponibles (Figura 5.2). El primer paso para poder realizar tal concentración de datos, es recolectar los datos desde cada nodo. Es necesario acceder a los datos capturados por cada

nodo, los cuales podrían necesitar algún procesamiento local, y posteriormente deben ser enviados hacia el servidor que se encargara de concentrar los datos. Conforme los datos lleguen al servidor, se deberán procesar para alinear aquellos registros que pertenecen a un mismo evento. Dicho procesamiento tomara en cuenta la información contenida en cada registro, esto es, las características capturadas de los relámpagos; y además también deberá tomar en cuenta el posible error introducido dentro de los datos. Una vez que los registros son procesados satisfactoriamente y se encuentran alineados, se almacenaran para su futuro análisis, consulta y/o procesamiento posterior. Por ultimo, una vez teniendo los datos debidamente almacenados, es necesario proveer una forma para realizar su consulta desde aplicaciones externas, permitiendo la utilización de la información obtenida para los objetivos que se crean necesarios.

El sistema propuesto fue implementado utilizando un servicio web RESTful como base, un servidor web en Node.js y una base de datos en MongoDB. La utilización de RESTful como la base arquitectónica nos permite publicar un servicio web, donde podemos establecer rutas de entrada de datos y rutas de consulta de datos. Se escogió la plataforma Node.js como el entorno de ejecución del sistema propuesto, debido a su alto desempeño en tareas concurrentes y baja demanda de recursos. Para almacenar los resultados de la alineación se utilizó una base de datos de MongoDB, la cual tiene un buen desempeño en operaciones de I/O. Tanto Node.js como MongoDB permiten trabajar con el lenguaje JavaScript, por lo cual tenemos un entorno unificado y completo. En el Apéndice C: Tecnologías Utilizadas se describen más a fondo estas tecnologías.

5.2. La recolección de datos

El primer paso para iniciar la recolección de los registros es obtener el acceso a dichos datos. Cada nodo tiene que estar instalado y configurado correctamente para que se pueda producir un flujo de registros de detección. El acceso a los registros es proporcionado a través del software NexStorm, el cual añade una estampa de tiempo a cada dato capturado. Sin embargo, dicha estampa de tiempo tiene una precisión limitada a segundos, la cual no es lo suficientemente precisa para nuestros fines, por lo que tendremos que agregarle una estampa de tiempo más precisa. Además, también es necesario asegurar que dicha estampa de tiempo sea igual independientemente del

nodo, esto es, que si un evento e es detectado por el nodo j y el nodo k entonces la estampa de tiempo generada por j y k deben ser iguales. Para cumplir con esta restricción de tiempo se utilizara NTP sobre cada nodo, de tal forma que la precisión de cada reloj sea la mejor posible. A si mismo se cuenta con dispositivos GPS, los cuales pueden cooperar con NTP para mejorar la precisión del reloj local, y es deseable que se configuren como una fuente de tiempo.

5.2.1. Acceso a los registros de detección

Para poder acceder a los registros de detección de los relámpagos es necesario contar con un nodo de detección completo y funcional. En [23] se describe ampliamente sobre la instalación del equipo perteneciente al dispositivo PCI; y en [24] se puede encontrar lo mismo para el equipo externo LD250. En ambas fuentes se explican formas de instalar la antena de detección, su correcta orientación y otras recomendaciones a tomar en cuenta.

La orientación de la antena es un factor importante a tomar en cuenta, pues dependiendo de su orientación la posición de los relámpagos detectados también cambian. Si la antena presenta una desviación de θ grados, sus detecciones presentaran esta misma desviación. Otra factor importante a tomar en cuenta es la corrección de rango de la antena, característica proporcionada por NexStorm. Mediante la corrección de rango se puede modificar la percepción de la antena respecto a la distancia de cada relámpago detectado ([28] pp. 29). Como resultado, las tormentas podrían posicionarse más cerca o más lejos de la localización del nodo dependiendo de la configuración. Tanto la corrección de rango como la orientación de la antena caen fuera del alcance del presente documento, pero tienen un papel importante, pues determina la precisión de nuestra fuente de datos. Sin embargo, aunque la antena tenga una incorrecta orientación y/o no tenga una buena corrección de rango, aun se producirán los datos de las detecciones de relámpagos realizadas.

Para poder acceder a los registros de las detecciones, es necesario tener instalado NexStorm como la interfaz software de la antena. Así mismo, es necesario que sea la versión completa de NexStorm debido a que es el único software que nos permite acceder a los registros de las detecciones de relámpagos. Por otro lado, también se debe verificar que la opción de FlashGate se encuentre activada en *Options/FlashGate IPC/IPC strike*.

5.2.2. Instalando y configurando NTP

Los varios sistemas operativos Unix/Linux son probablemente los mas flexibles en términos de soporte para NTP. En estos entornos se facilita la configuración de *servidores de tiempo* primarios (estrato 1), servidores de mayor estrato y clientes finales. El principal programa NTP es el demonio `xntpd/ntpd`, los cuales soportan la integración de drivers para relojes de referencia. La flexibilidad de su configuración y la oportunidad de una personalización mas detallada son los puntos fuertes del soporte NTP en estos entornos[34]. Para Microsoft Windows la historia es un poco diferente.

Microsoft Windows también ofrece soporte para sincronización de tiempo, la cual es implementada a través del servicio Windows Time (W32Time). El servicio de tiempo de Windows (W32Time) tiene la habilidad de sincronizar el reloj de la computadora con un servidor NTP. Por defecto no soportan funcionar como servidores primarios (estrato 1), sin embargo es capaz de operar en un nivel de estrato 2 o superior, ofreciendo servicio de tiempo a otros dispositivos a través de la red. Una de sus configuraciones es el tiempo entre peticiones de para sincronizar el reloj local contra una recurso de tiempo externo, el cual esta establecido por defecto a 1 hora o a 7 días entre peticiones[35]. Aunque por defecto tiene un tiempo entre sincronizaciones demasiado grande, todavía es posible modificar dicho valor con una granularidad en segundos. Sin embargo, W32Time no ofrece una alta precisión ya que en algunas versiones esta basado en SNTP (Simple Network Time Protocol), pues solo se asegura que la exactitud de valor del reloj pueda ser utilizado para la validación de red y tiempo de acceso a recursos[36]. Debido a esto, *el servicio W32Time no puede asegurar mantener sincronizado el tiempo del reloj en el rango de 1 a 2 segundos. Tal precisión esta fuera de sus especificaciones de diseño del servicio W32Time* [37]. Esto ultimo es valido al menos para versiones anteriores a Windows Server 2016 y Windows 10. Estas nuevas versiones contienen una implementación completa de NTP, y pueden garantizar una precisión de hasta de milisegundos bajo ciertas condiciones.

Por todo lo anterior, se concluye que el servicio para sincronizar tiempo ofrecido por Windows (W32Time) no cumple con nuestros requisitos, esto debido a que algunos nodos utilizan maquinas con Windows XP y otras cuentan con un GPS. Por esto es necesario instalar una aplicación de terceros para conseguir sincronizar con una mayor precisión. Se decidió utilizar la implementación NTP procedente de `www.ntp.org` [38], la cual tiene una versión portada para funcionar en Windows [39]. Dicha implementación ofrece varias

ventajas sobre W32Time y sobre otras aplicaciones, como son:

- Es multiplataforma, y cuenta con integración como servicio de Windows.
- Utilización de múltiples servidores NTP.
- Intervalo entre peticiones auto-balanceada según necesidad de resincronización (cd 1-17 min).
- Permite el uso de recursos de tiempo propios (GPSs, receptores de radio, etc.).
- Es de código abierto, y está acorde al estándar NTP.

Para un manual de referencia sobre la instalación de NTP en Windows y la descripción de otras ventajas véase [40].

Incorporación con un GPS

Una vez que NTP está instalado y funcionando, este se encargara de mantener sincronizado nuestro reloj local. NTP resincronizara nuestro reloj local cada cierto tiempo, seleccionando como fuente de tiempo principal el servidor que mantenga un mejor desempeño y un menor retardo de comunicación. Seleccionara el mejor servidor que se encuentre disponible de entre aquellos proporcionados en su configuración y descartara aquellos que se desvíen demasiado del tiempo. Sin embargo, si se cuenta con un GPS entonces es posible, y deseable, utilizarlo como una referencia tiempo que ayude a mantener y mejorar la precisión de nuestro reloj.

NTP versión 4 soporta casi cuatro docenas de relojes de referencia satelitales, de radio y de modems telefónicos. Dichos relojes de referencia son soportados de la misma manera que un servidor NTP ordinario, y usan los mismos algoritmos de filtrado, selección, agrupamiento y combinación [41]. La utilización de un driver se especifica mediante una dirección de la forma *127.127.t.u*, donde *t* es el tipo de driver y *u* es un numero en el rango 0-3 para distinguir múltiples instancias del mismo driver. En nuestro caso se utilizó el driver tipo 20: Generic NMEA GPS Receiver (NMEA).

En [42] se encuentra un manual, que incluye solución a problemas, sobre la instalación y utilización de un GPS genérico o un GPS con PPS (Pulse-per-second Signal) como fuente de tiempo para NTP. Sin embargo, tal como se menciona en [43], fue necesario determinar una buena estimación para el

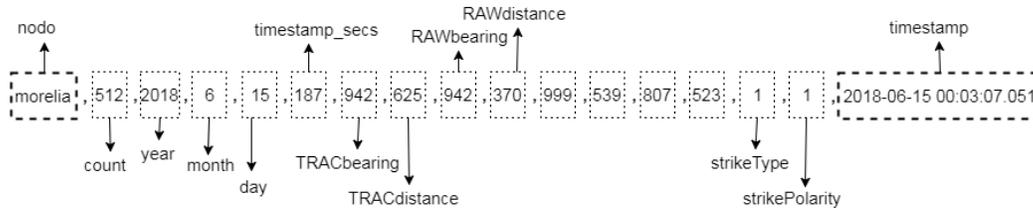


Figura 5.3: Los nuevos campos *nodo* y *timestamp* son agregados a un registro tipo dato.

intervalo de retraso (fudge factor) *time2*. Esto debido a que se presentaba un retraso promedio de 500 mili-segundos en la obtención del tiempo desde los GPSs. La configuración final utilizada en los nodos se muestra en Apéndice D: Archivo de Configuración de NTP.

5.2.3. Algoritmo de recolección

El **algoritmo de recolección** es el código encargado de recolectar los registros desde los nodos y enviarlos al servidor para su procesamiento. Cada nodo deberá tener su propia instancia del algoritmo de recolección. El algoritmo de recolección (Algoritmo 5.1) se puede dividir principalmente en los siguientes pasos:

1. Acceder a la memoria compartida de FlashGate.
2. Leer los datos disponibles y validarlos.
3. Procesar aquellos datos que pasen la validación.
4. Enviar los datos procesados hacia el servidor.
5. Cerrar el acceso a la memoria compartida.

La validación de los datos se refiere a determinar si existe un nuevo dato, por medio del campo *count*, y a determinar el tipo de registro (línea 4). Aunque todos los tipos de registros pueden proveer cierta información sobre el nodo, para lograr la alineación de registros de eventos de relámpagos solamente son necesarios registros tipo dato. Por lo anterior, solo se procesaran y enviaran al servidor los registros tipo dato y se ignoraran el resto. Durante la etapa de procesamiento, al registro se le agregan dos nuevos campos (Figura 5.3): *nodo* y *timestamp*. El primer campo es el identificador del nodo

en el que se realizó la captura del registro. El segundo campo es una nueva estampa de tiempo más precisa, tomada del reloj local. Por último, una vez que se concluye el procesamiento del nuevo registro, este es enviado al servidor como un paquete único (línea 6).

Algoritmo 5.1 Algoritmo de recolección.

Entrada: Nombre de la sección compartida mem_{name}^a .

Salida: Un nuevo registro reg_{send} por cada detección realizada.

```

1: Adquirir acceso a  $mem_{name}$ 
2: repetir
3:    $reg \leftarrow getReg(mem_{name})^b$ 
4:   si  $reg^{count} \neq last^{count}$  and  $isData(reg) = \text{cierto}$  entonces
5:      $reg_{send} \leftarrow procesar(reg)$ 
6:      $enviar(reg_{send})$ 
7:      $last \leftarrow reg$ 
8:   fin si
9:   Esperar 15 mls antes de verificar la disponibilidad de nuevos registrosa
10: fin repetir
11: Cerrar acceso a  $mem_{name}$ 

```

^aEspecificado por el fabricante de NexStorm en [28] pp 53.

^b $getReg(mem)$:= retorna el registro reg desde la sección compartida mem .

En [44] se puede encontrar un ejemplo ilustrativo sobre la utilización del API de Windows para solicitar acceso a una sección de memoria compartida. Por otro lado, en el CD-ROM de instalación de NexStorm se incluye un ejemplo, escrito en VB, sobre como realizar el acceso al FlashGate y validar los datos ([28] pp 53). Además, en [3] se pueden encontrar dos aplicaciones que hacen uso del FlashGate de NexStorm para la recolección de datos. La aplicación *readerNSms* está escrita en Java, y utiliza la librería JNA (Java Native Access) para comunicarse con el API de Windows. Por otro lado, la aplicación *localNSms* está escrita completamente en C y sólo utiliza la librería de Windows (windows.h).

5.3. Procesamiento y almacenamiento

En la etapa de concentración, los datos simplemente arriban al servidor y están disponibles para su procesamiento antes de ser almacenados. Durante el procesamiento es necesario alinear aquellos registros que pertenecen a un mismo evento. Para lograr dicha alineación se toman en cuenta los datos de los registros y se utilizan criterios para seleccionar los registros que se tienen que agrupar. Cada grupo formado representara un evento de relámpago detectado por el conjunto de antenas. Un grupo estará compuesto por las características de los registros que conforman el grupo, y por las características de formación del grupo. Los grupos formados, junto con sus características, son almacenados para posteriores consultas y/o procesamientos. Por otro lado, también se provee una forma de consulta para nuestros datos generados, tomando en cuenta los diversos usos que podrían darle a dichos datos.

5.3.1. Concentración y procesamiento: RESTful y Node.js

Los nodos necesitan enviar constantemente los registros de las detecciones al servidor, por esto es necesario contar con una puerta de entrada en el servidor para poder recibir los datos. Esto se puede lograr de diferentes maneras, para nuestro caso, escogimos los servicios RESTful. Este método fue escogido debido a su facilidad en implementación y mantenimiento, además de tener un escalamiento relativamente bueno. Por otro lado, también permite interaccionar entre diferentes computadoras a través de la red utilizando el ampliamente probado estándar HTTP. REST (REpresentational State Transfer) representa el estilo arquitectónico utilizado; Express, un paquete de Node.js, contiene funcionalidades que permiten implementar tal estilo. El punto de entrada de los datos es especificado como una URL (Uniform Resource Locator). Los nodos se deberán comunicar con el servidor a través de mensajes HTTP (Hypertext Transfer Protocol): el cuerpo del mensaje contendrá el registro, en formato JSON (JavaScript Object Notation), que se pretende mandar al servidor. Dicho mensaje es enviado hacia la URL, definida previamente, utilizando el método PUT. De esta forma todos los nodos podrán enviar sus registros hacia una sola dirección de Internet.

Del lado del servidor se recibirán los registros provenientes de los diferentes nodos, y se procesaran conforme arriben. Debido a que Node.js trabaja de forma asíncrona, todos los registros que se reciban sera procesados de forma

paralela. Durante el procesamiento, el algoritmo de alineación determinara cuales registros serán agrupados bajo un único evento y cuales representan diferentes eventos. Una vez formados los grupos, donde cada grupo representa un evento (relámpago), estos son almacenados para su persistencia y su consulta.

5.3.2. Criterios de alineación

Un primer criterio que nos permite diferenciar entre registros de eventos de relámpago que pertenecen a diferentes eventos, es el origen del nodo que los detecto. Cada registro que arribe al servidor se puede clasificar dependiendo del nodo que lo envió, de tal forma que al conjunto de todos los registros enviados por un nodo j le llamamos **flujo de datos** de j . Cuando se tienen varios nodos de detección, hay múltiples flujos de datos que llegan al servidor. Si suponemos que cada registro r que pertenece al flujo de datos de j , r^j , es una verdadera y única representación de un evento dentro de j , es decir, un nodo de detección j no puede detectar múltiples veces un evento de relámpago, entonces no es posible que varios registros del nodo j representen el mismo evento de relámpago. Por lo anterior, solo tiene sentido agrupar y etiquetar como un único evento de relámpago aquellos registros provenientes de diferentes nodos.

La forma más obvia de verificar si dos registros de relámpagos pertenecen a eventos diferentes o a un mismo evento, es comparando sus propiedades intrínsecas. Un buen criterio se obtiene verificando las propiedades tipo (CG/IC) y polaridad (+/-) de ambos registros de relámpagos, si son diferentes entonces los registros pertenecen a eventos diferentes, si son iguales entonces es altamente probable que ambos registros pertenecen al mismo evento. Tal como se explicó al principio del capítulo, aunque las propiedades temporales (tiempo que ocurrió) y espaciales (posición donde ocurrió) de un relámpago son suficientes para identificarlo, no pueden ser utilizadas directamente debido al error de detección de los nodos.

Supongamos que tenemos un sistema de detección de eventos de relámpagos con múltiples nodos, y todos los nodos se encuentran sincronizados con un tiempo global g . Si dos diferentes nodos detectan un evento de relámpago y generan dos registros con estampas de tiempo que difieren en menos que dos ticks globales ($2g$), entonces podemos asociar dicha diferencia de tiempo al error de sincronización (Subsección 4.3.2). Por lo anterior, cuando dos registros están mas cerca que dos ticks globales, podemos considerarlos

generados por el mismo evento. Por otro lado, cuando dos registros están separados por dos ticks globales o más, podemos considerarlos generados por diferentes eventos. Sin embargo, podría ocurrir una situación como la mostrada en la Figura 5.4. El evento e_{16} es detectado por el nodo j en el tick 2 y por el nodo m en el tick 1, y el evento e_{42} es detectado solo por el nodo k en el tick 3. Si solo tomamos en cuenta las observaciones de j y m , o de j y k , podemos concluir que corresponden al mismo evento debido a que existe una diferencia de un tick. Si solo tomamos en cuenta las observaciones de k y m , podemos concluir que evidentemente corresponden a dos eventos diferentes debido a que tienen una separación mayor a dos ticks. Así pues, dependiendo de los agrupamientos de registros que se hagan, podemos llegar a conclusiones diferentes.

Este problema se presenta cuando dos eventos diferentes están separados por una distancia menor o igual que tres ticks globales, esto es, $\leq 3g$. Supongamos que dos eventos e_a y e_b ocurrieron con una diferencia de $3g$, donde e_a obtiene una estampa de tiempo adelantada en un tick (debido a Ecuación 4.6) y e_b obtiene una estampa de tiempo atrasada en un tick (ídem). En tal situación, los registros resultantes obtendrán una diferencia final de un tick entre sus estampas de tiempo, y por lo tanto se concluiría que fueron originados por un mismo evento, siendo la realidad diferente (Figura 5.4). La única forma de esquivar este problema es tener una mejor granularidad durante la obtención de las estampas de tiempo, esto es, mejorar la precisión de los relojes utilizados por los nodos. Esta inconsistencia no ocurre cuando dos eventos están separados por una distancia igual o mayor que $4g$. Lo anterior es debido a que si una estampa se atrasa y otra se adelanta, al igual que en el caso anterior, al final se obtendría una diferencia de 2 ticks entre las estampas de tiempo, permitiendo diferenciar correctamente los dos diferentes eventos. De todo lo anterior podemos concluir que, cuando dos registros de eventos tienen una diferencia de 1 tick es *probable* que pertenezcan al mismo evento, y cuando tienen una diferencia de > 2 ticks son *definitivamente* dos eventos diferentes.

Cuando un único evento de relámpago puede ser detectado múltiples veces por diferentes nodos, cada registro de detección generado tendrá pequeñas diferencias debido al error de detección. Para alinear estos registros se tendrán que formar grupos de registros que pertenezcan a un mismo evento de relámpago. Cada grupo representara un relámpago detectado por el conjunto de antenas, y cada registro dentro de un grupo representa una detección de ese relámpago por alguna de las antenas. Además todos los grupos que

grupo g , y donde

$$\begin{aligned} \delta_g(reg) &= \left\{ 1 - \left(K \frac{\Delta d}{2rang} + M \frac{\Delta t}{2g} \right) \right\} \\ \Delta d &= |reg^{posicion} - g^{posicion}| \\ \Delta t &= |reg^{timestamp} - g^{timestamp}| \end{aligned} \quad (5.3)$$

donde $rang$ es la máxima desviación espacial que presentan las detecciones de los nodos, g es el tick global, K y M son los pesos para los valores de posición y tiempos, respectivamente. Δd es el error de desviación para la distancia, esto es, la diferencia entre la posición según el registro de detección y la posición según el grupo. Δt es el error de desviación para el tiempo, tomando en cuenta la estampa de tiempo (timestamp) del registro y del grupo.

Mediante este criterio adicional el grupo seleccionado será el que mejor se ajuste a la información del registro tanto en tiempo como en posición. Por ejemplo, con una selección de los valores $K = M = 0.5$ obtenemos un peso del 50% para cada parámetro, se seleccionara el grupo mas cercano en tiempo y en posición. Sin embargo, para poder utilizar completamente este criterio es necesario que la alineación de la antena y la corrección de rango estén correctamente configuradas, de otra forma el parámetro Δd se vuelve impreciso y malicioso. En tal caso, se le debería dar un bajo peso al parámetro de posición o en su defecto no tomarlo en cuenta. Si el parámetro Δd no es tomado en cuenta, entonces la Ecuación 5.3 se convierte en

$$\delta_g(reg) = \Delta t \quad (5.4)$$

En este caso, el criterio de la Ecuación 5.2 seleccionara el grupo que se encuentra mas cercano temporalmente a reg_{new} .

5.3.3. Algoritmo de alineación

El **algoritmo de alineación** es el código encargado de realizar la alineación de los registros de detecciones de relámpagos. Este algoritmo deberá tener acceso a los grupos formados, los registros que componen cada grupo y sus características. El algoritmo de alineación (Algoritmo 5.2) se puede descomponer en las siguientes partes:

1. Seleccionar un único grupo k que cumpla con los criterios para aceptar el nuevo registro.

- Si no existe ningún grupo, se creara un nuevo grupo vacío.
- Si existen múltiples grupos, se seleccionara el más cercano temporalmente.

2. Añadir el nuevo registro al grupo seleccionado k .

Para filtrar los grupos, y poder seleccionar solo uno de ellos, se utilizan los criterios descritos previamente. El primer filtro (línea 5: $T(group, reg)$) selecciona aquellos grupos que se encuentren a una distancia temporal menor de g respecto del nuevo registro. De esta manera, un grupo j con una estampa de tiempo de s_j podría anexar dos registros, uno con una estampa de s_i y otro con una de s_k , donde se cumple que

$$[(s_j - s_i) < g] \wedge [(s_k - s_j) < g] \wedge [(s_k - s_i) < 2g] \quad (5.5)$$

tal que $s_i < s_j < s_k$. El segundo filtro (línea 5: $C(group, reg)$) selecciona aquellos grupos que tengan características intrínsecas idénticas al nuevo registro. El tercer filtro (línea 8) selecciona aquellos grupos que aún no contengan un registro que provenga del mismo nodo de donde es el nuevo dato. Con esto se evita que se formen grupos que contengan únicamente registros de un solo nodo. Los filtros anteriores no se tienen que aplicar necesariamente en ese orden. Si después de aplicar estos filtros aún se tienen más de un grupo prospecto, entonces se utilizara el criterio adicional de la Ecuación 5.2.

Cada grupo tiene asignada una estampa de tiempo ($g^{timestamp}$) que representa el pivote central de los registros pertenecientes al grupo. Dicha estampa de tiempo central representa el tiempo estimado cuando ocurrió el evento, y es calculada como:

$$t_p = t_o + \frac{t_n - t_o}{2} \quad (5.6)$$

donde t_o es la estampa de tiempo del registro más antiguo dentro del grupo, t_n es la estampa de tiempo del registro más reciente dentro del grupo y t_p es el pivote del grupo. Cuando un nuevo registro r es asignado a un grupo vacío (sin registros), entonces la estampa de tiempo del registro, t_{reg} , es utilizada como la estampa de tiempo del grupo, t_p . Cuando un nuevo registro es asignado a un grupo con al menos un elemento, entonces un nuevo t_p tiene que ser calculado. Si t_r está dentro del rango $t_o - t_n$, entonces t_p no se modifica. Si t_r está fuera del rango $t_o - t_n$, entonces t_p sera modificado (Ecuación 5.6). Los valores de t_o y t_n representan respectivamente el límite inferior y superior temporal del grupo. Así pues todo el grupo representa una ventana de tiempo

Algoritmo 5.2 Algoritmo de alineación.

Entrada: Un nuevo registro reg y un conjunto de grupos alineados $groups_{align}$.

Salida: Un conjunto de grupos alineados $groups_{align} | \exists! g : reg \in g \wedge g \in groups_{align}$

```

1:  $E$     {precisión externa alcanzada por los nodos}
2:  $g \leftarrow 2E + 1$     {tiempo global del sistema}
3:  $t_o \leftarrow (reg^{timestamp} - g)$ 
4:  $t_n \leftarrow (reg^{timestamp} + g)$ 
5:  $prospects \leftarrow groups_{select} := \{group \mid group \in groups_{align} \wedge T(group, reg)^1[1] \wedge C(group, reg)^2[2]\}$ 
6: para todo  $group$  en  $prospects$  hacer
7:   para todo  $register$  en  $group$  hacer
8:     si  $register^{node} = reg^{node}$  entonces    {3er criterio}
9:        $prospects \leftarrow (prospects - G) \mid G := \{group\}$ 
10:    fin si
11:  fin para
12: fin para
13: si  $|prospects| = 0$  entonces
14:    $g_{select} \leftarrow G \mid G := reg$ 
15:    $g_{select}^{timestamp} \leftarrow reg^{timestamp}$     {establecer tiempo}
16:    $g_{select}^{position} \leftarrow posición(reg)$     {calcular posición}
17: si no, si  $|prospects| \geq 1$  entonces
18:   si  $|prospects| = 1$  entonces
19:      $g_{select} \leftarrow prospects$ 
20:   si no, si  $|prospects| > 1$  entonces
21:      $g_{select} \leftarrow g \mid \delta_g = \min_{\forall i \text{ in } prospects} \{\delta_i(reg)\}$     {(Ecuación 5.2)}
22:   fin si
23:    $g_{select}^{timestamp} \leftarrow (t_o + \frac{t_n - t_o}{2})$     {calcular tiempo (Ecuación 5.6)}
24:    $g_{select}^{position} \leftarrow posición(g_{select})$     {calcular posición}
25: fin si
26: Guardar  $reg$  en base de datos
27: Guardar  $g_{select}$  en base de datos

```

^a $T(group, reg) := t_o < group^{timestamp} < t_n$ (1er criterio)

^b $C(group, reg) := group^{type} \equiv reg^{type} \wedge group^{polarity} \equiv reg^{polarity}$ (2do criterio)

que va desde t_o hasta t_n , y debido a que t_p es utilizado durante el primer filtro, ningún grupo puede tener asignados dos registros que estén a una distancia mayor de $2g$.

Por todo lo anterior, se puede concluir que un grupo deberá tener al menos las siguientes características:

- Una estampa de tiempo (t_p), que representa el tiempo cuando ocurrió el evento de relámpago.
- Una posición absoluta (latitud y longitud), que representa el lugar donde ocurrió el relámpago.
- Un tipo, que representa el tipo de evento de relámpago (CG/IC).
- Una polaridad, que representa la dirección del relámpago (+/-).
- Un conjunto de registros de detecciones, que representan las diferentes detecciones realizadas por los nodos, teniendo un registro por cada nodo.

En la presente sección ya se explicó cómo aproximar la primera característica. La segunda característica es tratada en [1]. El tercer y cuarto punto son proporcionados directamente por las antenas de detección. La característica final se logra durante la recolección de los diferentes registros y su procesamiento para alinearlos. Aunque estas son las características principales, es posible anexar características secundarias que ayuden durante el proceso de alineación u otros campos que se consideren necesarios.

5.3.4. Almacenamiento y consulta: MongoDB y RESTful

Durante esta etapa los datos ya procesados son almacenados para su persistencia, en este caso, son almacenadas en una base de datos de MongoDB. Así mismo, es conveniente almacenar los registros recolectados de los nodos antes de ser procesados, con el fin de poder consultar los registros integrantes de cada grupo. El almacenamiento es necesario si se requiere hacer un seguimiento, análisis o consulta de los registros recolectados por los diferentes nodos. En resumen, es necesario almacenar los registros para cualquier aplicación que no sea trivial.

Los campos que comprenden un registro, tal como los entrega NexStorm, y los nuevos campos que le agregamos, nodo y timestamp, determinan su

CAPÍTULO 5. SISTEMA PROPUESTO

formato de almacenamiento en la base de datos. Por lo anterior los registros recibidos por el servidor serán almacenados en una estructura como la mostrada en el Listado 5.1. En la sección anterior se describieron las características principales para representar un relámpago detectado dentro del sistema. Por lo tanto cuando se requieran almacenar los relámpagos, su formato de almacenamiento sera parecido al mostrado en el Listado 5.2.

```
1  datastrike{
2    node: {type: String },
3    count: {type: Number},
4    year: {type: Number},
5    month: {type: Number},
6    day: {type: Number},
7    timestamp_secs: {type: Number},
8    TRACbearing: {type: Number},
9    TRACdistance: {type: Number},
10   RAWbearing: {type: Number},
11   RAWdistance: {type: Number},
12   TRAC_X: {type: Number},
13   TRAC_Y: {type: Number},
14   RAW_X: {type: Number},
15   RAW_Y: {type: Number},
16   strikeType: {type: Number, enum:[0,1,-1]},
17   strikePolarity: {type: Number, enum:[0,1,-1]},
18   timestamp: {type: Date}
19 }
```

Listado 5.1: Estructura de un documento que representa un registro.

```
1  strike{
2    timestamp: {type: Date},
3    latitud: {type: Number},
4    longitud: {type: Number},
5    strikeType: {type: Number, enum:[0,1,-1]},
6    strikePolarity: {type: Number, enum:[0,1,-1]},
7    data: [{type: ObjectId, ref: 'datastrike'}]
8  }
```

Listado 5.2: Estructura de un documento que representa un grupo.

Una vez que tenemos los datos almacenados, es necesario proveer un método de consulta que nos permita acceder a dichos datos. Una forma de acceder

a tales datos sería directamente usando una conexión con la base de datos de MongoDB. Sin embargo, utilizando esta forma tenemos que conceder permisos de acceso para cada usuario, y además tenemos que verificar que los datos no sean alterados sin permiso. Esta tarea no es sencilla cuando se quiere conceder acceso a un gran número de usuarios, y por otro lado los usuarios necesitan conocer la forma de crear y administrar sus conexiones.

Una vez más, existen varias formas de poder desplegar y/o consultar información desde una base de datos. Aquí proponemos utilizar los servicios RESTful, debido a la facilidad con que podemos publicar y consultar nuestros datos. Estableciendo un flujo de salida para nuestros datos mediante una URL, podemos compartir aquellos datos que se deseen y además podemos limitar el tipo de datos a compartir o darle un pre-procesamiento antes de pasarlos al usuario. El usuario solo tendrá que utilizar a un URL para poder consultar los datos disponibles que nosotros publicamos. Se proponen tres tipos de salida de datos:

- **Datos de registros:** Para la consulta de los registros sin procesamientos y sin alinear.
- **Datos de alineación:** Para la consulta de la información mínima de los grupos resultantes de la alineación.
- **Datos de Grupos:** Para la consulta de los grupos almacenados y todas sus características, incluyendo los registros que integran cada grupo.

El primer tipo de consulta mostrara los registros tal como fueron recibidos por el servidor, sin ningún procesamiento adicional incluyendo la alineación. Esta consulta permite trabajar con los registros originales de los nodos, con la posibilidad de utilizarlos para otros fines. El segundo tipo de consulta permitirá revisar los resultados de la alineación, sin incluir información adicional sobre cómo se realizó dicha alineación. Para proyectos de despliegue y visualización el segundo tipo es más que suficiente, si se requiere ver los resultados de la alineación este es la consulta que se debería usar. Esta consulta es la pre-definida cuando se requiera un flujo de registros que se encuentren alineados. El tercer tipo de consulta muestra toda la información de los grupos formados durante la alineación: las características del grupo, las características del evento de relámpago y las características de cada registro que pertenece al grupo. Esta consulta podría ser utilizada para validación de los grupos formados durante la alineación, ya que nos permite ver todas las características involucradas durante la formación de los grupos.

5.4. Comentarios finales

Durante este capítulo se presentó una descripción del sistema propuesto. Primero se describió la etapa de recolección de datos, la cual se involucra desde la adquisición de datos hasta su concentración en el servidor. Se describieron las condiciones necesarias para poder obtener registros que contengan datos precisos y fiables. Esto es importante debido a que la calidad de los resultados obtenidos depende de la información de los registros utilizados como entrada. Para poder mejorar la precisión de los relojes locales se tiene que instalar NTP, esto debido a que el servicio de Windows ofrece una precisión demasiado baja incapaz de cumplir con nuestros requerimientos. Así mismo se incorporó la utilización de un GPS junto con NTP para mejorar aun mas la precisión local. También se presentó el algoritmo de recolección, encargado de concentrar los registros relevantes desde los diferentes nodos hacia un servidor. Después se dio una descripción sobre la etapa de procesamiento, durante la cual se realiza propiamente la alineación de los registros de relámpagos en el servidor. Se presentó el análisis sobre los criterios utilizados para poder alinear los eventos de relámpagos: sus estampas de tiempo, sus características intrínsecas (tipo y polaridad) y el flujo de datos de origen. Adicionalmente también se explico como se puede utilizar una heurística adicional para desempatar la elección de un grupo. Dicha heurística busca minimizar el error de detección de posición y tiempo dentro de los grupos. También se presentó el algoritmo utilizado para realizar la alineación, se describió como se forman los grupos, los campos que contendrán y como se calculara el valor de cada campo. Finalmente se presenta el formato que sera utilizado para almacenar lo datos recibidos desde lo nodos y los nuevos grupos formados, almacenando así tanto la fuente de datos como los resultados de la alineación. Además se describen los tipos de consultas que se podrían realizar: consulta de los registros sin procesar, consulta de la representación de los relámpagos, y consulta de los grupos formados. Si bien aquí se plantea la utilización de node.js como servidor, de MongoDB como base de datos y, de RESTful como arquitectura de servicio, como base para la implementación de la alineación de registros de relámpagos, no es necesario utilizar tan particulares tecnologías. Si se requiere, la implementación puede ser portada total o parcialmente a otras tecnologías de manera fácil, e.g., utilizar PostgreSQL como base de datos en lugar de MongoDB.

Capítulo 6

Pruebas y Resultados

Durante el presente capítulo se presentan las pruebas realizadas, y los resultados obtenidos, de los diferentes componentes del sistema propuesto. Se inicia verificando la precisión de los relojes locales que fue alcanzada mediante NTP. Se describe la recolección de registros desde los diferentes nodos y el conjunto de datos que se utilizó como entrada del sistema. Después se analiza el comportamiento del algoritmo de alineación ante la utilización de diferentes parámetros y se comentan los resultados obtenidos. Así también, se analiza y describe la mejor forma para escoger el mejor valor de los parámetros utilizados por el algoritmo de alineación. Finalmente, se presentan los resultados que se obtuvieron al realizar diversas consultas.

6.1. Precisión de relojes locales

La recolección de los registros desde cada nodo es un parte primordial de nuestro sistema. El sistema propuesto utiliza los registros de relámpagos producidos por los nodos como información de entrada. Dependiendo de la exactitud o precisión que presenten los datos contenidos dentro de los registros, así también será la calidad de los resultados obtenidos en la alineación.

NTP nos permite recolectar estadísticas sobre la desviación que presenta nuestro reloj local respecto a las referencias de tiempo externas utilizadas durante la resincronización[45]. Para activar la recolección de estadísticas es necesario activar la opción en el archivo de configuración de NTP, y especificar el directorio destino donde almacenarlas (Véase Apéndice D: Archivo de Configuración de NTP). Las estadísticas son almacenadas en archivos de

CAPÍTULO 6. PRUEBAS Y RESULTADOS

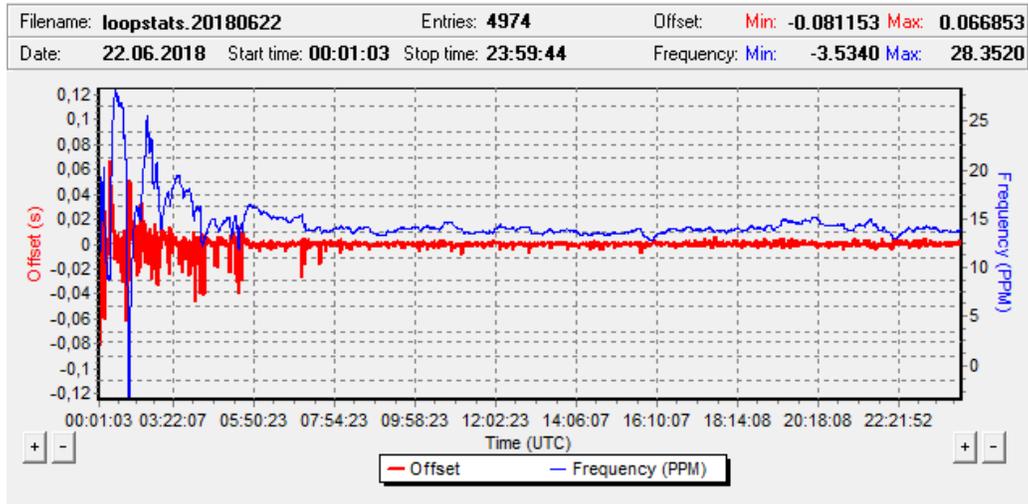


Figura 6.1: Gráfica sobre el error de desviación del reloj local a través del tiempo, posterior a la instalación de NTP.

texto, una línea por cada registro generado. La aplicación *NTP Time Server Monitor*[46] proporciona una interfaz gráfica para interactuar con el servicio de NTP. Entre otras opciones, nos permite visualizar de forma gráfica los archivos de estadísticas generados por NTP.

En la Figura 6.1 se muestra una gráfica obtenida utilizando *NTP Time Server Monitor* y un archivo de estadísticas generado por NTP en uno de los nodos. El archivo contiene estadísticas recolectadas durante el día posterior a la instalación de NTP en el nodo. En la gráfica se visualiza, en color rojo, la desviación (offset) que presenta el reloj local respecto de la referencia de tiempo (ordenada) a través del tiempo (abscisa). Los valores positivos representan un adelanto en el tiempo del reloj local respecto a la fuente de tiempo tomada como referencia. Los valores negativos representan un atraso en el tiempo del reloj local respecto a la referencia. En la gráfica se puede apreciar claramente que conforme el tiempo avanza la precisión del reloj local tiende a mejorar. Durante las primeras horas de la mañana el reloj tiene un alto error de desviación, ≈ 80 mili-segundos, después de varias horas se acerca hasta un precisión de ≈ 40 milisegundos para finalmente mantenerse con una precisión de ≈ 10 milisegundos al final del día. El resto de los nodos, los cuales también tienen instalado NTP, presentan gráficas muy parecidas a la mostrada aquí.

Con esto, podemos concluir que efectivamente nuestros nodos mantienen la mejor precisión posible y tienden a corregir cualquier desviación que presente nuestro reloj local respecto a la fuente de tiempo tomada como referencia. Además, debido a que NTP influye directamente en el reloj local de la computadora, la precisión que se logre alcanzar afecta indirectamente a cualquier aplicación que haga uso del reloj local. De esta forma no solo mantenemos una buena precisión para nuestra aplicación sino también para posibles aplicaciones secundarias que también requieran de una buena precisión. Tal es el caso de *readerNSms* y *localNSms*, las cuales toman una estampa de tiempo del reloj local durante la captura de un nuevo registro desde la memoria compartida IPC-Strikes [3]. Así pues mediante la instalación y correcta configuración de NTP se logra mejorar la precisión de todo el ecosistema de aplicaciones dentro del nodo.

6.2. Recolección de registros

En una situación ideal, cada nodo deberá tener instalada una aplicación que verifique por la disponibilidad de nuevos registros de relámpagos y conforme encuentre nuevos registros los envíe hacia el servidor para su procesamiento. Cada vez que se realiza la detección de un relámpago por alguno de los nodos, se genera la estampa de tiempo del registro y es enviado al servidor inmediatamente. Cada registro que llega al servidor es procesado de inmediato para alinearlo y luego es almacenado tanto el registro recibido como los resultados del procesamiento de alineación. Aplicaciones externas podrían consultar en todo momento la información almacenada, siempre obteniendo los datos más actuales disponibles.

El proceso anterior es perfecto en condiciones normales de funcionamiento del sistema, sin embargo para la verificación del propio sistema es necesario poder utilizar datos reales de entrada y además poderlo hacer bajo demanda. De otra forma tendríamos que esperar por una tormenta, que genere relámpagos, para poder realizar la validación del correcto funcionamiento. Además, de esta forma no se podría realizar un correcto análisis de comportamiento, pues cada vez se obtendrían diferentes datos de entrada. Por lo anterior se creó una aplicación auxiliar, que utiliza datos reales obtenidos mediante *readerNSms* y *localNSms* [3], para poder realizar las pruebas del sistema.

La aplicación auxiliar replica la función de los nodos (enviar registros hacia el servidor) y no interviene directamente con el servidor, logrando así

que la entrada de datos sea transparente. Los registros son obtenidos desde varios archivos de texto, con un registro por línea y un archivo por nodo, y los envía hacia la dirección del servidor para su procesamiento de alineación. De esta forma podemos realizar pruebas bajo demanda al sistema, y verificar los diferentes comportamientos del sistema para el mismo conjunto de datos.

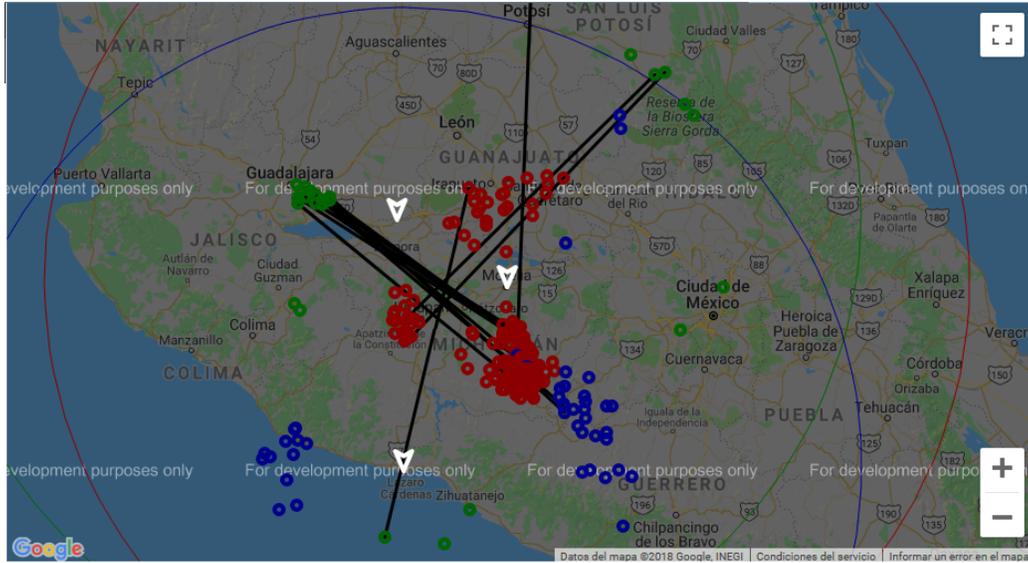
6.3. Algoritmo de Alineación

El algoritmo de agrupación utiliza solo un parámetro para realizar la clasificación y agrupación de los diferentes registros que se reciben, el tick global del sistema g . Una buena forma de ver el comportamiento del sistema es revisar el tipo de agrupaciones que resultan del algoritmo de alineación. Sin embargo, ver un conjunto de datos en bruto con la información de cada grupo no proporciona ninguna comprensión intuitiva. Por esto se procedió a mostrar gráficamente la distribución espacial de los registros alineados.

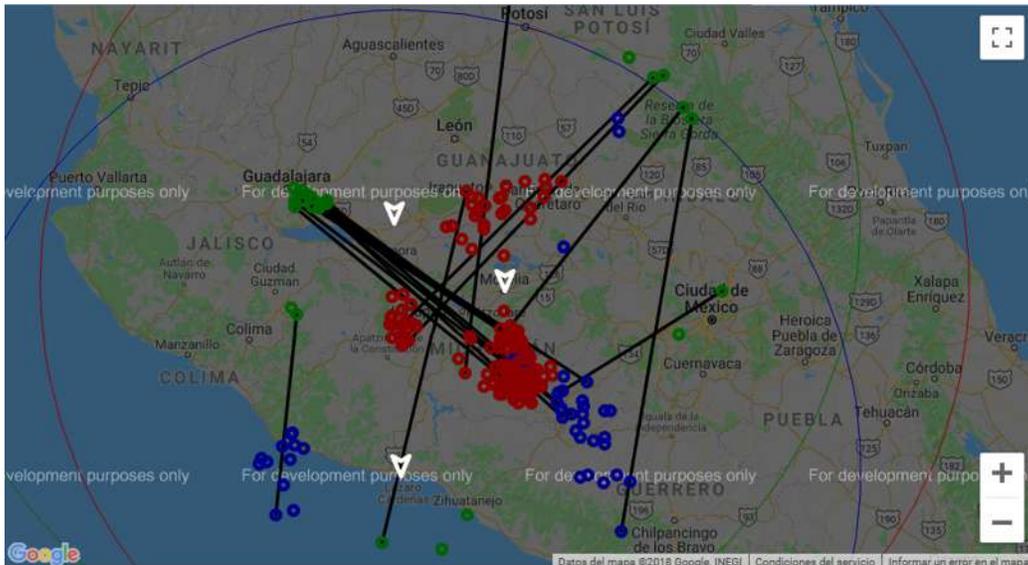
Para visualizar gráficamente los grupos realizados se utiliza una consulta de los *Datos de Grupos*. Esto nos permite obtener tanto las características de los grupos como los registros que conforman cada grupo, siendo perfecta para realizar una validación de la alineación. Cada registro de detección fue posicionado en su lugar de ocurrencia, según la antena que lo detecto, y aquellos registros que pertenecen al mismo grupo fueron unidos por una línea. De esta forma se puede apreciar visualmente de un vistazo donde se encuentran los datos de los diferentes grupos y ver sus características. Los grupos y los registros fueron desplegados sobre un mapa de Google Maps, las coordenadas (latitud y longitud) de cada registro fueron calculadas utilizando el algoritmo de Vincenty [47] [48] .

En la Figura 6.2 se presentan las imágenes sobre los diferentes comportamientos que presenta el algoritmo de alineación para diferentes valores del tick global g . Cuando el valor de g es bajo, el número de grupos que se forman aumenta. Esto debido a que g controla el espacio de tiempo necesario entre diferentes registros para poder considerarlos diferentes. En un caso extremo, si g fuera 0, indicaría que todos nuestros relojes son precisos y por tanto solo agruparía registros con exactamente la misma estampa de tiempo. Conforme g aumenta, el número de grupos formados disminuye. Esto debido a que el criterio de tiempo utilizado para seleccionar los registros a agrupar se relaja, permitiendo que exista una diferencia entre las estampas de tiempo de los registros que conforman cada grupo.

CAPÍTULO 6. PRUEBAS Y RESULTADOS



(a) $g = 20$



(b) $g = 80$

Figura 6.2: Comportamiento del algoritmo de alineación ante diferentes valores del tick global g .

Como criterio adicional se utilizó la Ecuación 5.4. Esto debido a que nos es imposible acotar el error de desviación de posición de los registros de detecciones, y no existen referencias de que se haya realizado previamente. Por lo cual se considera que el valor del parámetro de posición para la Ecuación 5.3 no aporta suficiente confiabilidad, y por tanto es desestimado. Al utilizar solo el parámetro temporal como criterio adicional, se obtiene un método de alineación que depende únicamente del tiempo.

6.4. Selección de parámetros de alineación

Tal como se vio en la sección anterior, conforme el valor del parámetro g varía el comportamiento del algoritmo de alineación también lo hace. Por lo tanto, es necesario establecer correctamente el valor de g con el fin de obtener el mejor desempeño del algoritmo, y así agrupar correctamente los registros de relámpago pertenecientes al mismo evento. La cuestión es: ¿Cuál es el mejor valor para g ?

Nuestros relojes están sincronizados externamente mediante NTP, contra servidores de tiempo y contra GPS. Tal como se vio en la Subsección 4.3.1, cuando un conjunto de relojes son sincronizados externamente con una exactitud E , entonces el conjunto está también internamente sincronizado con una exactitud de al menos $2E$. Siendo la exactitud E definida como

$$E = \max_{\forall 1 \leq k \leq n} \{accuracy^k\} \quad (6.1)$$

donde k representa cada uno de los relojes del conjunto de nodos, y n el número total de relojes dentro del conjunto. Además, en la Subsección 4.3.2 se vio que el tick global g tiene que ser mayor que la precisión del conjunto de todos los relojes locales Π (Ecuación 4.6). Por todo lo anterior se tiene que cumplir la condición

$$g > 2E \quad (6.2)$$

para que el tick global sea aceptable.

Para poder obtener la exactitud externa E es necesario conocer la exactitud que presenta cada nodo del sistema. Tal como vimos previamente, *NTP Time Server Monitor* nos permite ver de forma gráfica el comportamiento del reloj de un nodo. Además nos proporciona información adicional sobre la desviación (*offset*) máxima y mínima que presenta nuestro reloj local en el intervalo mostrado (Figura 6.1). Estos valores indican cual es la máximo

adelanto y el máximo atraso, respectivamente, que presente nuestro reloj en el intervalo de tiempo mostrado. De estos dos valores debemos seleccionar el que tenga mayor valor absoluto, dicho valor representara la exactitud (*accuracy*) de nuestro nodo. Una vez teniendo los valores de exactitud de todos los nodos, tendremos que seleccionar el mayor valor absoluto de entre todos ellos. Este valor es la exactitud externa E de nuestro conjunto de nodos.

Para obtener el valor del tick global g , es necesario respetar la Ecuación 6.2. Teniendo el valor de E , solo tendremos que encontrar un valor mayor que $2E$ para designar a g . Para lograr esto podemos hacer que

$$g = 2E + 1 \tag{6.3}$$

De forma ideal, mediante este procedimiento se obtendría el mejor valor para g . Sin embargo, los relojes pueden presentar comportamientos esporádicos que hacen que su precisión aumente o disminuya durante breves intervalos de tiempo. Tales fenómenos se deben a factores externos propios del entorno donde se encuentra el reloj (temperatura, presión, humedad, etc.) o de características intrínsecas del propio reloj (demanda de uso, voltaje, antigüedad, calidad, etc.). También podrían ser producto de una fuente de tiempo maliciosa, o de un retraso durante el intercambio de mensajes de tiempo. La implementación de NTP intentara evitar las fuentes de tiempo maliciosas y seleccionar aquellas que tengan un buen desempeño, pero no siempre es posible evitar los efectos colaterales. Así pues, se podría seleccionar una fuente de tiempo maliciosa como fuente de referencia, se sincronizaría el reloj local y, posteriormente, cuando se detectara la inconsistencia de sus mensajes de tiempo seria desechada. Una nueva referencia de tiempo, utilizada anteriormente o recién descubierta, sera elegida para sincronizar el reloj local. Durante el breve intervalo en que la referencia maliciosa desvía el reloj local hasta que este es corregido nuevamente, la precisión del nodo puede alejarse demasiado.

En la Figura 6.3 se puede ver de forma gráfica aquellos intervalos de tiempo en que los relojes tiende a desviarse demasiado de la referencia de tiempo. En general el reloj de la Figura 6.3a mantiene una buena exactitud por debajo de 5 mili-segundos. Sin embargo durante algunos momentos su precisión (*offset*) decae fuera del promedio. El primer pico de desviación muestra una exactitud de ≈ 13 mili-segundos, mientras que los posteriores picos muestran una exactitud de ≈ 8 mili-segundos. En el reloj de la Figura 6.3b se pueden apreciar variaciones mas extremas. El reloj pasa de una exactitud de \approx mili-

CAPÍTULO 6. PRUEBAS Y RESULTADOS

segundos hasta una de ≈ 68 ms durante el primer pico de desviación, y hasta ≈ 40 ms durante el segundo pico.

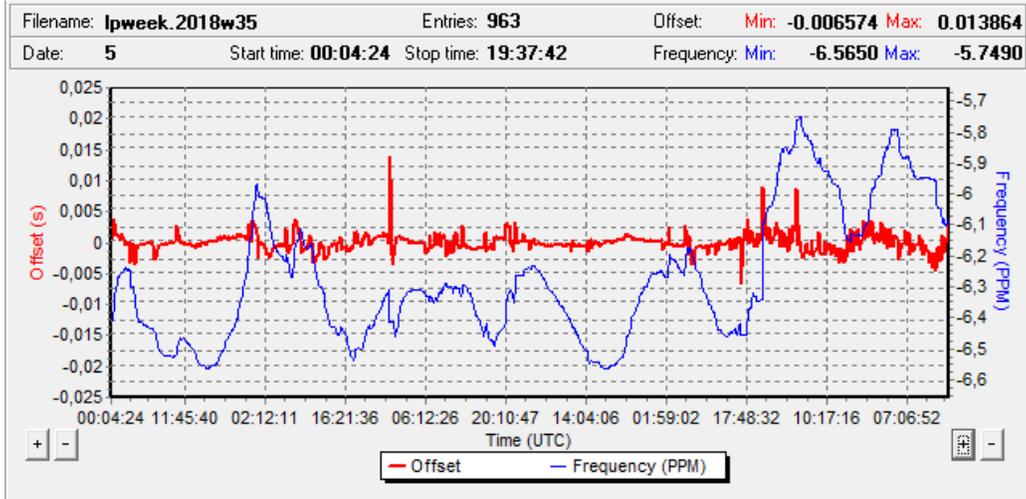
Este tipo de comportamiento tiene que ser tomado en cuenta al momento de seleccionar el valor de g . Se podría utilizar siempre el máximo error de desviación que se presente en el reloj como la exactitud del nodo, sin embargo también se podría utilizar el valor promedio de desviación. Por ejemplo, para el caso 6.3a podríamos utilizar el valor de 14 ms como la exactitud de un nodo. De esta forma se acota los picos de desviación que se presentan y se asegura que nuestro nodo *siempre* tendría un valor correcto de precisión por debajo de 14 ms. Por otro lado también podríamos ignorar los picos de desviación que se presenten en nuestros nodos, y utilizar el valor promedio de exactitud. En este caso, para el mismo ejemplo, se asignaría el valor de 5 ms como la exactitud del nodo en lugar de utilizar el valor del pico. De esta forma no podríamos asegurar que siempre se tendrá una exactitud acotada. Pero se podría asegurar que durante la *mayor parte del tiempo* se obtendría la mejor exactitud posible, y durante breves intervalos perderíamos dicha exactitud.

Por otra lado, nos es imposible predecir con antelación cuando nuestros relojes tendrán picos de desviación y que tanta precisión se perdería debido a esos picos. Por lo anterior, tiene sentido ignorar los picos de desviación demasiado extremos y utilizar el valor promedio de la exactitud que presenten los nodos. Se recomienda considerar el valor de exactitud promedio que presente cada nodo para seleccionar el tick global, siempre y cuando dicha precisión se mantenga durante la mayor parte del tiempo (Figura 6.3b). Si la precisión del nodo tiende a variar demasiado y/o de forma muy errática, es mejor establecer un valor de exactitud que acote por arriba la mayoría de dichas variaciones (Figura 6.3a).

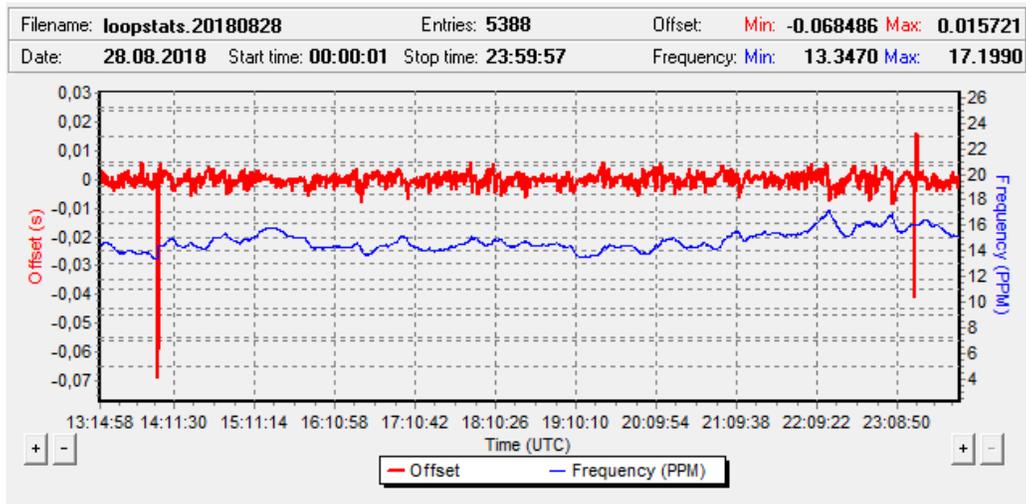
Para poder utilizar la Ecuación 5.3 como criterio adicional tenemos que conocer la posición aproximada del relámpago. Para obtener dicha posición debemos establecer un mecanismo para estimar la posición del relámpago a partir de la información contenida en los registros que componen el grupo[1]. Además, se debe asegurar que las antenas tenga configurada la corrección de rango de forma correcta. Estas tareas se encuentran fuera del alcance del presente trabajo, y para la segunda condición no se encontró documentación que avale la implementación de la configuración.

Si el error espacial en las detecciones no se encuentra acotado, es decir, no sabemos cual es la máxima desviación espacial que se presenta en los registros de las detecciones de un mismo evento, entonces solo podemos intentar

CAPÍTULO 6. PRUEBAS Y RESULTADOS



(a) Reloj con precisión de 5 ms que presenta picos de desviación.



(b) Reloj con precisión de 10 ms con picos de desviación.

Figura 6.3: Gráficas sobre la exactitud del relojes (rojo) que muestran picos de desviación.

adivinar ciegamente el valor del parámetro *rang*. Tenemos que asignar un valor al parámetro espacial *rang* de tal forma que, podamos alinear aquellos registros que pertenecen al mismo evento pero fueron separados espacialmente por el error de detección y, podamos ignorar aquellos registros que evidentemente se encontraban separados desde un inicio. Si seleccionamos valor demasiado pequeño, entonces aquellos registros que presenten un error de desviación mayor que el seleccionado serán ignorados para ser agrupados. Si por el contrario seleccionamos un valor de desviación espacial demasiado grande, entonces registros que originalmente estaban separados, y por tanto pertenecían a diferentes eventos, serán agrupados como un mismo evento. Por lo anterior la utilización del parámetro de localización se comportara de forma maliciosa, podría no alinear detecciones que pertenezcan al mismo relámpago, pero alinearía detecciones que originalmente pertenecían a diferentes relámpagos

Por lo anterior, la utilización del parámetro de localización como criterio adicional es en última instancia, inútil, debido a que no podemos saber cuándo estamos realizando una alineación de baja calidad y cuando estamos realizando una sobre-alineación. Teniendo en cuenta que no podemos acotar el error de detección espacial, tendríamos que elegir un valor demasiado bajo para K , el peso del parámetro de posición, y valor alto para M , el peso del parámetro de tiempo; o en última instancia solo utilizar el valor temporal. En caso de que se pueda determinar el error de desviación espacial presente en las detecciones de los nodos, una configuración $K = M = 0.5$ resultaría conveniente. En las pruebas realizadas solo se utilizó el valor temporal como parámetro adicional. Debido a esto, nuestro sistema utilizó únicamente parámetros temporales para lograr la alineación.

Tal como se mencionó anteriormente, si los relojes de los nodos pierden precisión el algoritmo de alineación también lo hará, y si el error de posición en los registros es muy alto también afectara, aunque en menor cantidad, los resultados de la alineación. Debido a los impredecibles picos de desviación que presentan los relojes sincronizados, no es posible tener la precisión acotada todo el tiempo. Y por esto se tendrá que llegar a un acuerdo entre perder desempeño del algoritmo de alineación durante breves intervalos de tiempo o degradar la precisión del nodo todo el tiempo.

6.5. Consulta de datos

Cada uno de los tipos de consulta (datos de registros, datos de alineación, datos de grupos) proporcionan un tipo de dato específico dependiendo del fin con que se quieran utilizar. Esto permite seleccionar aquella consulta que mejor se acomode con nuestras necesidades, y con el tipo de datos que queramos trabajar. Los resultados de cada consulta se encuentran en formato JSON, lo cual nos permite trabajar con los datos de forma casi directa en el navegador.

En la Sección 6.3 de este capítulo, se utilizó la consulta de grupos para visualizar los datos de forma gráfica. De esta forma nos permito interaccionar tanto con los grupos como con los registros que los conforman y poder mostrar características de ambos. Para lograr esto se utilizó JavaScript, AJAX y JQuery. El código para tal consulta de datos se muestra en el Listado 6.1. La forma de realizar una consulta diferente, sea para datos de registros o para datos de alineación, es similar y puede lograrse modificando únicamente la dirección URL de la consulta. Una vez que el servidor responde a nuestra petición, los datos son almacenados localmente y puestos a nuestra disposición para su manipulación.

```
1   var urlbase_raws="http://localhost:3000/raws/"
2   var urlbase_strikes="http://localhost:3000/strikes/"
3   var urlbase_groups="http://localhost:3000/groups/"
4   urlquery_raw=url_raws+url_query_parameters;
5   urlquery_stk=url_strikes+url_query_parameters;
6   urlquery_grp=url_groups+url_query_parameters;
7   var urlquery_get=urlquery_grp;
8
9   //solicitando una consulta de datos
10  $.get (
11    urlquery_get,
12    function (data,stat){
13      if (stat == "success"){           // si la consulta
14        fue exitosa                    fue exitosa
15        setGroupMarks (data,mainMap); //graficando grupos
16      }else{                           //si la consulta no
17        fue exitosa                    fue exitosa
18        $("#console").text (' Error (AJAX.get):'+stat);
19    }
20  )
```

CAPÍTULO 6. PRUEBAS Y RESULTADOS

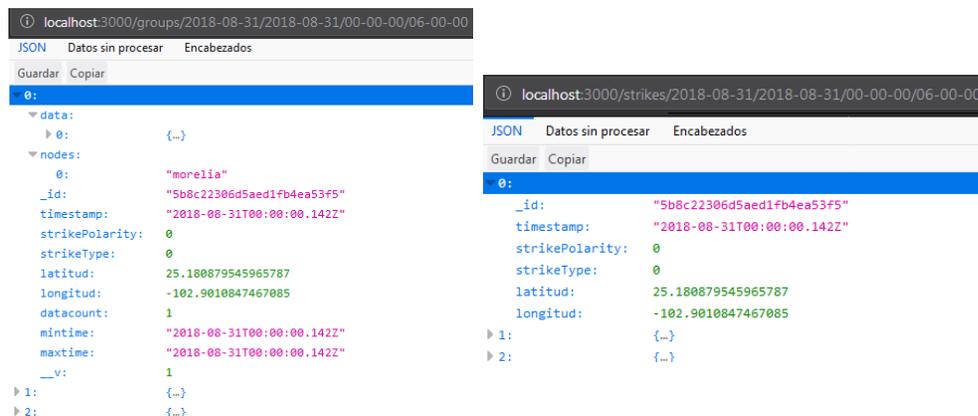


Figura 6.4: Resultados de realizar las consulta de datos utilizando únicamente el navegador Firefox.

Listado 6.1: Consulta de grupos utilizando AJAX y JQuery.

En el Figura 6.4 se muestra el resultado de utilizar únicamente el navegador para realizar las consultas. Debido a que el navegador no tiene programada ninguna función especial para tratar con los datos recibidos, simplemente los despliega al usuario sin ningún formato.

6.6. Comentarios finales

Durante este capítulo se ha verificado que la precisión alcanzada por cada uno de los nodos es en efecto mejor que la que se tenía previamente antes de instalar NTP. Además, cada vez que se presenta una desviación respecto a la referencia del tiempo NTP tiende a corregirla, manteniendo así siempre la mejor precisión posible. También se explico sobre el funcionamiento de la aplicación auxiliar, la cual tiene como fin el ayudar a realizar pruebas bajo demanda del funcionamiento del sistema. Dicha aplicación esta creada para funcionar de forma transparente como entrada de datos, y así simular el comportamiento de los nodos. Así mismo se analiza el comportamiento que tiene el algoritmo de alineación ante diferentes tipos de parámetros, concluyendo que es necesario seleccionar un buen valor del parámetro g para

CAPÍTULO 6. PRUEBAS Y RESULTADOS

obtener buenos resultados. Se describió una forma de determinar los mejores parámetros para nuestro algoritmo, principalmente el valor del tick global g , mediante el cual se obtienen resultados aceptables. Con lo anterior se puede concluir que el algoritmo propuesto siempre realizara el mejor proceso de alineación, según la calidad que presenten los datos dentro de los registros proporcionados. Por esto es de vital importancia asegurarse que los nodos estén configurados correctamente, para que puedan proporcionar los datos mas exactos sobre las detecciones. Por ultimo también se presenta la forma en que realizo la consulta de datos utilizando tecnologías del lado del navegador. De esta forma se comprobó que el acceso a los datos funcionan correctamente y ademas funcionan como encapsulamiento para el gestor de base de datos, permitiendo seleccionar la cantidad y tipo de información que se desea compartir.

Capítulo 7

Conclusiones y Trabajos Futuros

En este capítulo se presentan las conclusiones generales a las que se llegaron después de completar el desarrollo del sistema propuesto. Se analizan los resultados obtenidos y se presentan sugerencias para su implementación. Además se discuten el alcance y las limitaciones que presenta nuestra solución. Así también, se presentan algunas de las posible líneas de desarrollo futuro que pueden ser seguidas.

7.1. Conclusiones generales

Durante este trabajo se presentó una propuesta para alinear diferentes registros de relámpagos detectados desde múltiples nodos. Para presentar esta propuesta primero se han analizado el origen y las características de los relámpagos. Se ha descrito la función de detección de relámpagos que presentan las antenas Boltek, y el formato de los registros que se proporcionan para el acceso de aplicaciones externas. Después se describió como la utilización de un tiempo global base ayuda a tener un percepción de tiempo sincronizada para un sistema distribuido, esto es, para un conjunto de computadoras que interaccionan entre si pero que se encuentran separadas a través de la red. Una vez descritas las bases teóricas, presentamos el sistema propuesto para lograr la alineación de registros de relámpagos, siendo el algoritmo de alineación el núcleo del sistema. Finalmente probamos nuestro sistema, logrando la alineación de detecciones y mostrando un ejemplo de consulta de registros

alineados.

El algoritmo de alineación se basa principalmente en el tiempo global que presenta el sistema y en las estampas de tiempo que contiene cada registro de entrada. El tiempo global de nuestro sistema nos permite determinar cuándo dos registros de eventos de relámpago sucedieron en diferentes instantes de tiempo y cuando sucedieron al mismo instante de tiempo. Cuando dos o más registros de relámpagos suceden al mismo instante de tiempo, es probable que pertenezcan al mismo evento de relámpago, si suceden en diferentes instantes de tiempo, entonces obviamente son dos eventos diferentes. Para poder establecer un buen tiempo global base es necesario que los relojes de los nodos pertenecientes al sistema tengan una buena precisión. Debido a que la implementación del algoritmo de sincronización con que cuenta el sistema por defecto no ofrece suficiente precisión, se utilizó una implementación de NTP para mejorar la precisión de nuestros relojes.

Para poder determinar si diferentes registros sucedieron durante el mismo tiempo, se utiliza la estampa de tiempo que contiene cada registro. Debido a que originalmente las antenas Boltek no proporcionan suficiente precisión temporal de los nuevos registros de detección, tuvimos que agregar información adicional a cada registro proporcionado por los nodos. Los campos que se agregaron a los registros fueron: el nodo de origen y una estampa de tiempo más precisa. El campo sobre el nodo de origen, nos permite utilizar un criterio de alineación adicional, mediante el cual podemos mejorar la calidad de la alineación. Con esto logramos obtener las condiciones mínimas necesarias para que nuestro algoritmo de alineación trabaje correctamente.

Adicionalmente también se intentó utilizar un criterio de alineación que tomara en cuenta la desviación espacial dentro del error de detección. Sin embargo, no existen trabajos previos que demuestren que el error de desviación espacial de los relámpagos detectados por el conjunto de antenas está correctamente acotado. Por lo anterior no es posible utilizar el criterio espacial para ayudar en la alineación, debido a que dicho parámetro podría interferir con la correcta alineación causando una inconsistencia en los resultados. Así pues, con el fin de comprobar el funcionamiento de nuestro sistema al final solo se utilizaron criterios temporales para determinar la correcta alineación.

Nuestro algoritmo de alineación utiliza las estampas de tiempo de los registros, dichas estampas son tomadas desde cada nodo que provee el registro. Cada estampa de tiempo es tomada de los relojes locales, y los relojes locales están siendo sincronizados con referencias de tiempo externas. Por lo anterior, los resultados de alineación que se puedan obtener dependen de la

precisión que puedan alcanzar los relojes de los nodos. Si se quiere mejorar la calidad de la alineación entonces, necesariamente se tendrá que mejorar la precisión de los relojes utilizados. Cuando la precisión de los relojes se encuentre por debajo del error de digitalización del nodo o por debajo del intervalo de disponibilidad de nuevos datos (< 15 ms), el algoritmo de alineación obtendrá el mejor desempeño posible. Sin embargo el alcanzar esta precisión y además mantenerla todo el tiempo no es completamente posible, principalmente por factores externos que influyen en el desempeño del reloj. Los nodos que utilizan un GPS como fuente de tiempo adicional tienen un mejor desempeño al mantener una buena precisión, sin embargo todavía presentan picos de desviación esporádicos.

Por todo lo anterior, podemos concluir que el algoritmo propuesto tiene un desempeño del **mejor esfuerzo**. Es decir, el algoritmo realizara siempre el mejor desempeño tomando en cuenta la precisión que tengan los nodos y los parámetros de los criterios utilizados. Además, debido a los picos de desviación de precisión imprevisibles el algoritmo tendrá intervalos de tiempo en que la alineación se degradaría hasta que se corrija la precisión desviada. Una solución podría ser acotar los picos de desviación por medio del tiempo global, sin embargo, haciendo esto también degradaríamos la alineación todo el tiempo, en lugar de solo durante breves intervalos. Por lo anterior se tiene que llegar a un acuerdo de que tanto es admisible degradar la calidad de la alineación durante breves intervalos de tiempo y, que tan grandes y frecuentes serán dichos intervalos.

7.2. Trabajos futuros

- Añadir tolerancia a fallos del sistema, detección de nodos caídos, de nodos maliciosos y detecciones erróneas. Mediante la detección de nodos caídos y de nodos maliciosos es posible tomar acciones para corregir o desechar aquellos nodos que no aporten información de relevancia.
- Realizar una comparación entre los registros de detecciones de relámpagos que proporcionan las antenas contra otra fuente de datos externa. Esto con el fin de validar los datos obtenidos por el conjunto de nodos y poder realizar las configuraciones necesarias para mejorar la exactitud de los datos capturados.
- Generación de mapas de densidad sobre la frecuencia de relámpagos

CAPÍTULO 7. CONCLUSIONES Y TRABAJOS FUTUROS

en las zonas de detección. Permitiendo conocer las áreas donde ocurren con más frecuencia los relámpagos y las áreas que presentan menor frecuencia, y además, conocer el valor de frecuencia de relámpagos para una zona en particular.

- Análisis estadístico de datos recolectados sobre largos intervalos tiempo, con el fin de poder determinar un modelo que represente el desarrollo de tormentas sobre la región.

Apéndice A: Detección de Relámpagos desde el Espacio

Las detecciones de relámpagos desde el espacio proveen la mayor área de estudio sobre la ocurrencia de los relámpagos. Además tiene la ventaja de una eficiente y uniforme detección de relámpagos sobre los continentes y los océanos. La detección desde el espacio también permite detectar la radiación óptica de descargas dentro de la nube (IC) y de descargas de nube a tierra (CG). Este conjunto total de descargas, refleja más completamente la actividad eléctrica de las tormentas. Por el otro lado, la separación de las descargas nube a tierra por medio de únicamente medios ópticos es difícil y podría requerir datos adicionales y nuevos algoritmos.

Los principales instrumentos utilizados para lograr tales detecciones fueron el OTD y el LIS. Estos instrumentos realizan una detección óptica de relámpagos utilizando capas de CCD (Charge-Couple Device) y procesando las imágenes capturadas en tiempo real. Para lograr identificar los relámpagos se hace uso de un filtro que detecta una línea espectral particular. Posteriormente se hace un procesamiento espacial y temporal sobre las imágenes capturadas con el fin de determinar la longitud y la duración del relámpago detectado. Un análisis temporal sobre los grupos formados permite detectar el desarrollo y comportamiento de la tormenta.

Instrumentos de detección

Desde la mitad de 1990 dos instrumentos ópticos han sido operados por la Administración Nacional de la Aeronáutica y del Espacio (NASA –National Aeronautics and Space Administration) a bordo de satélites de órbita baja. Ambos instrumentos tienen incorporadas cámaras de amplio ángulo detectando la señal óptica de los relámpagos [14] [49].

APÉNDICE A: DETECCIÓN DE RELÁMPAGOS DESDE EL ESPACIO

El detector óptico de transitorios (OTD –Optical Transient Detector) estuvo operando a bordo del satélite Microlab-1, el cual tenía una inclinación de órbita de 70° a una altitud de 735 km sobre el suelo y una órbita periódica sobre los 100 min. El OTD obtuvo datos sobre un periodo de 5 años comprendido de 1995 al 2000, y tenía una eficiencia de detección del 40 % al 65 % dependiendo de condiciones externas, como luminosidad y radiación[50]. Los relámpagos fueron detectados con una matriz de capas de CCD¹ con dimensiones de 128x128, esto corresponde a una vista instantánea con un rango de 1250 km².

El Sensor de Imágenes de Relámpagos (LIS –Lightning Imaging Sensor) fue lanzado en diciembre de 1997 a bordo del satélite para la Misión de Medición de Lluvia Tropical (TRMM –Tropical Rainfall Measuring Mission) y terminó su misión después de 17 años de producir datos científicos[51]. El LIS, que utilizó al OTD como prototipo, tenía una eficiencia de detección de aproximadamente 90 %. Debido a la órbita del satélite el área de observación está limitada a las latitudes $\pm 35^\circ$. La altitud de 350 km y el periodo de la órbita de 92 min fueron cambiados en agosto del 2001 durante la corrección del TRMM a 402 km y 93 min.

El sensor LIS es funcionalmente idéntico al sensor OTD, pero posee severas mejoras en sensibilidad, diseño y electrónica. El rango de vista instantáneo para el LIS es cerca de 670 km² con un tiempo de visualización local de 93 segundos. LIS también usa una matriz CCD de 128x128 píxeles permitiendo hasta 5 km de píxeles de longitud por lado. El tiempo de resolución es de 2 ms (tiempo de integración de un frame) [52] [49] [14].

FORTE (Fast On-Orbit Recording of Transient Events) es un satélite artificial multi-sensor, proyecto conjunto entre el laboratorio nacional de Los Alamos (LANNL –Los Alamos National Laboratory) y los laboratorios nacionales de Sandia. El satélite fue lanzado en 1997 en una órbita circular a 825 km de altitud, con una inclinación de 70° . Junto a un sistema de localización óptico para relámpagos, con el mismo diseño que el OTD y LIS, se encuentra un fotodiodo de banda amplia con una resolución de 15 μ s y un receptor de VHF, lo cual permite la combinación de las observaciones de radio frecuencia y observaciones ópticas de los relámpagos [49].

¹ Un dispositivo de carga acoplada (CCD –Charge-Couple Device) es un circuito integrado que contiene un número determinado de condensadores enlazados o acoplados, permitiendo mover la carga eléctrica capturada hacia un área diferente. Es una tecnología revolucionaria para la adquisición de imágenes con alto desempeño.

Metodología de localización y detección

Los instrumentos del OTD y del LIS fueron diseñados para detectar la breve señal óptica transitoria de los relámpagos. La típica señal óptica desde el relámpago es un pulso de luz que contiene líneas espectrales características, formadas por los gases del aire que son excitados e ionizados, con una duración menor a $500 \mu s$ y un tamaño espacial sobre los 10 km. La línea espectral más usada para la detección óptica de relámpagos desde el espacio es del oxígeno atómico OI(1) encontrada cerca del espectro infrarrojo a 777.4 nm.

La señal óptica del relámpago tiene que ser discriminada bajo condiciones de un fuerte ruido de fondo, el cual es producido por el reflejo de la radiación solar sobre las superficies de la nube. Primero, la señal pasa el filtro centrado en la línea espectral 777.4 nm para captar la energía del relámpago y eliminar la energía solar reflejada. Después, el tamaño espacial de los pixeles de detección y el tiempo de integración son optimizados para capturar el evento completo, y eliminar una fragmentación espacial y temporal del relámpago. Al final, un procesamiento de eventos en tiempo real es aplicado con el fin de detectar el breve pulso luminoso del relámpago, pero que presenta un fondo que varía lentamente en luminosidad [49] [52].

Los datos ópticos obtenidos por la matriz de pixeles son clasificados en áreas, destellos, grupos y eventos, basados en su proximidad en tiempo y espacio. Se agrupan pixeles adyacentes para formar grupos, diferentes grupos pueden fusionarse dependiendo de su cercanía temporal y espacial. Cada grupo corresponde a un relámpago detectado. Aquellos eventos que tienen cercanía espacial son agrupados en *áreas*, las cuales representan tormentas activas. Esto con el fin de que este agrupamiento refleje tanto como sea posible el entendimiento convencional de relámpagos y tormentas. Eventos que no corresponden a un relámpago, causados por ruido electrónico, partículas de alta energía o reflexiones especulativas, son introducidos en los datos y pueden ser reconocidos por su distintiva distribución espacio-temporal [49].

Apéndice B: Drivers de Antenas Boltek

Cuando se usa el dispositivo de pre-procesamiento en modo externo, por lo general no es necesario instalar ningún driver para la detección del puerto RS232 o del puerto USB. Esto debido a que el sistema operativo los puede reconocer automáticamente sin ningún inconveniente (Windows XP SP3 o superior). Los drivers para esta conexión se pueden encontrar en [27] bajo la sección del producto correspondiente.

Cuando se usa el dispositivo de pre-procesamiento en forma PCI, es necesario instalar los drivers correspondientes para que la interfaz software pueda detectar el dispositivo y acceder a los datos. Si se cuenta con el CD-ROM proporcionado por el fabricante, se recomienda realizar la instalación desde este. Si no se cuenta con el CD-ROM proporcionado por el fabricante, hay ciertas cuestiones a tomar en cuenta durante la instalación manual de dichos drivers. Los drivers para el dispositivo PCI pueden ser encontrados en [27] bajo la sección StormTracker-PCI.

Boltek ofrece dos diferentes versiones de drivers para la tarjeta receptora PCI: Driver V5 (Antigua versión 5) y Driver V9 (Nueva versión 9). La diferencia entre estos dos drivers es la interfaz software que se utilizara para acceder al dispositivo PCI. Para el nuevo software StormTracker V9.01 y NexStorm-Lite es necesario utilizar la nueva versión del driver (V9). Si se requiere ejecutar un software más antiguo es necesario utilizar la antigua versión del driver (V5). Para nextStorm (versión completa) será necesario instalar el driver versión 5. *Las dos versiones de drivers pueden ser instaladas conjuntamente sin que causen conflicto entre ellas.*

Para la instalación del driver V5 son necesarios los siguientes archivos:

- instalador driver V3.
- archivo *StormPCI.inf*

APÉNDICE B: DRIVERS DE ANTENAS BOLTEK

- archivo *windrvr.sys* (actualización V5)

Antes de ejecutar el instalador del driver (V3) es necesario colocar el archivo *StormPCI.inf*, descargado desde la página de Boltek, en la ruta:

`C:\Windows\Inf\`

Este archivo es utilizado por el instalador del driver V3 durante el proceso de instalación. Si no se coloca el archivo en la ruta correspondiente, la instalación finalizara con errores. Una vez completada la instalación de la versión V3 podemos proceder a actualizar hacia el versión V5.

Para actualizar al driver versión 5 utilizamos el archivo *windrvr.sys* descargado desde la web de Boltek. Abrimos la ruta:

`C:\Windows\system32\drivers\`

y reemplazamos el archivo *windrvr.sys* existente por el nuevo archivo *windrvr.sys* proporcionado. Posteriormente reiniciamos la PC para que Windows detecte los cambios. Con esto completamos la actualización del driver V3 hacia el driver V5.

La instalación del driver V9 está completamente empaquetada en su instalador y no son necesarios archivos extras. Solo es necesario ejecutar el instalador y seguir las instrucciones hasta completar la instalación. Al finalizar la instalación es necesario reiniciar la PC para que los cambios tengan efecto.

Apéndice C: Tecnologías Utilizadas

Antes de describir que es REST, es necesario dar un repaso sobre que son los servicios web y cual fue su origen. Los servicios web se encuentran actualmente en auge. Sin embargo, aunque su origen fue estimulado aun hoy en día no se cuenta con ningún estándar de implementación definitivo. Durante el periodo en que se empezaron a definir los primeros servicios web, REST surgió como una de las muchas alternativas disponibles para implementarlos. REST especifica una arquitectura que utiliza el estándar de HTTP como base para construir un servicio web.

El framework Node.js nos permite implementar un servidor web ligero y muy eficiente. Node.js puede realizar trabajos de forma concurrente con relativa facilidad, y puede manejar un alto número de peticiones simultaneas. También puede soportar tareas de I/O de manera asíncrona mediante callbacks. El principal lenguaje para realizar un implementación en Node.js es JavaScript.

MongoDB es un manejador de base de datos orientado a documentos, que permite trabajar con formatos BSON. Es altamente escalable horizontalmente, pudiendo tener clusters de bases de datos y replicación entre los diferentes clusters. También se encuentra optimizado para mejorar las peticiones de escritura y lectura hacia la base de datos. Por lo anterior MongoDB además de ser redundante y a prueba de fallos, también tiene una alta disponibilidad de los datos. MongoDB también utiliza el lenguaje JavaScript como base para realizar las consultas de los datos almacenados. Mediante la conjunción de Node.js y MongoDB se puede lograr un entorno unificado bajo el lenguaje JavaScript y que además tiene un alto desempeño para tareas que requieran alta demanda.

Servicios Web

En un servicio Web una tecnología Web tal como HTTP (originalmente diseñada para la comunicación humano-máquina) es utilizada para comunicaciones máquina-máquina, más específicamente para transferencia de formatos de archivo comprendidos por máquinas tal como XML (eXtensible Markup Language) y JSON (JavaScript Object Notation). La principal motivación de los servicios web fue transformar la Web de un sistema estático a uno dinámico. Es estática en el sentido de que contiene aplicaciones que interactúan con humanos y que no pueden ser reutilizadas en diferentes contextos y escenarios. Los servicios web pretenden ofrecer interoperatividad e integración de funcionalidades entre diferentes máquinas y dispositivos, en última instancia de forma automática. Aunque, esto último aun está lejos de ser alcanzado plenamente en la actualidad [53].

Un servicio Web comúnmente provee una interfaz basada en Web, y orientada a objetos mutables, de un servidor de base de datos. Esto permite la publicación de recursos(objetos) y sus operaciones a través de Internet, permitiendo su acceso remoto, y promoviendo un desacoplamiento que permite la interoperatividad entre servicios y servidores. Dicho servicio Web también podría ser utilizado como una interfaz para el usuario final.

Los primeros servicios Web fueron construidos para el transporte de llamadas a procedimientos remotos (RPC –Remote Procedure Calls) sobre la Web. La idea comenzó a despegar rápidamente y resultó en una gran colección de estándares, comenzando con SOAP (Simple Object Access Protocol) y WSDL (Web Services Description Language). Sorprendentemente, esos estándares fueron definidos con poca consideración de la práctica contemporánea; algunas veces antes de que existiera alguna implementación por estandarizar. El resultado final de esta estandarización prematura fue confusión, en lugar del orden que usualmente brinda un estándar. En respuesta, comenzó a aparecer un estilo de servicios Web alternativo, construido acorde con las reglas de la Web contemporánea. Esos servicios Web (después llamados RESTful) están madurando, o más precisamente, se está re-aprendiendo a usar los estándares de la Web, ya utilizados y probados, para construir servicios Web. Ambos estilos son utilizados ampliamente, y además, las interacciones y comparaciones entre ellos comienzan a alcanzar un estado constructivo, por lo cual se espera que se aprenda de la experiencia mutua y maduren aún más [54] [55].

Los servicios RESTful están más centrados en los datos, mientras que los servicios WS-* están más centrados en funcionalidad. Por otro lado, las

aplicaciones web son similares a los servicios RESTful, pero se diferencian en que las primeras fueron hechas para consumo humano mientras que los segundos lo fueron para consumo máquina, esto es, los usuarios del servicio son otros servicios u otras aplicaciones (p.ej. aplicaciones web) [56].

REST y RESTful

REST (REpresentational State Transfer) se refiere al estilo arquitectónico que aprovecha la tecnología y los protocolos de la Web existentes para crear servicios Web, por otro lado, RESTful se utiliza para referirse a aquellos servicios Web que implementan tal arquitectura. REST es un estilo arquitectónico que define un conjunto de restricciones para ser usadas para la creación de servicios Web. REST fue introducido y definido en 2000 por Roy Fielding. Él identificó cuatro principios de REST (las cuales llamo restricciones)[54]:

1. Identificación de recursos mediante una Uniform Resource Identifier (URI).
2. Manipulación de recursos a través de sus representaciones.
3. Mensajes auto-descriptivos.
4. Hiper-media como el motor del estado de la aplicación (HATEOAS –Hypermedia as the engine of application state).

Un *recurso* es cualquier cosa que puede ser asignada como un objetivo de hipertexto (p.ej. un archivo, un script, una colección de recursos). En respuesta a la petición de un recurso el cliente recibe una *representación* de ese recurso, la cual podría diferir del formato mantenido por el servidor. Los recursos son manipulados por medio de *mensajes* que tienen significados estandarizados; en la Web esos mensajes son los métodos HTTP. El cuarto principio significa que el estado de cualquier interacción cliente-servidor es mantenida en la hiper-media que ellos intercambian, esto es, los enlaces (Hyperlinks), o URI (Uniform Resource Identifier), que intercambian. Cualquier información del estado del recurso es pasada entre el cliente y el servidor en cada mensaje, por lo cual, en ambos lados no es necesario mantener un seguimiento del estado actual.

Un servidor RESTful provee acceso a los diferentes recursos, mientras que el cliente RESTful accede y modifica los recursos usando los métodos del

protocolo HTTP. Los siguientes cuatro métodos son comúnmente usados en una arquitectura basada en REST:

- **GET** - Usado para proveer acceso de solo lectura a un recurso.
- **PUT** - Usado para crear un nuevo recurso.
- **DELETE** - Usado para remover un recurso.
- **POST** - Usado para actualizar un recurso existente o crear un nuevo recurso.

node.js

Node.js es una framework multiplataforma, y de código abierto, del lado del servidor(pero no limitado) útil para la construcción de aplicaciones rápidas y altamente escalables. Node.js es una plataforma que está construida sobre v8, la implementación de JavaScript utilizada por el navegador Web Chrome diseñado por Google. Node.js está diseñado para ser bueno con aplicaciones de uso intenso de I/O, utilizando la arquitectura orientada a eventos no-bloqueante [57] [58].

Mientras que Node.js puede ejecutar funciones de forma síncrona, es más común la ejecución de operaciones de forma asíncrona. Una *callback* es un equivalente asíncrono de una función. Una función callback es ejecutada en la terminación de una tarea dada. Esto significa que conforme se ejecuta la aplicación se pueden lanzar instrucciones con una callback registrada para su ejecución después del retorno de la función. Mientras se espera por el retorno, la siguiente función o evento en la aplicación puede ser encolada para su ejecución inmediata. Node.js hace un amplio uso de callbacks, lo cual le permite pasar a la siguiente operación disponible mientras espera por la terminación de una tarea pendiente [58].

Node.js también provee un amplia librería de varios módulos JavaScript con los cuales se simplifica el desarrollo de las aplicaciones Web. Los módulos aprovechan las operaciones asíncronas siempre que es posible, y algunas veces incluyen opciones tanto síncronas como asíncronas. Se pueden encontrar módulos que frecuentemente son usados mientras se desarrolla una aplicación. Para propósitos de búsqueda, instalación y administración de módulos de Node, se cuenta con NPM (Node Package Manager) [57] [59].

Express.js

Express es un minimalista y flexible framework para aplicaciones Web basadas en Node.js. Provee un robusto conjunto de características para desarrollar aplicaciones Webs y móviles, facilitando el rápido desarrollo de aplicaciones Web [60]. Las siguientes son algunas de las características principales de framework Express:

- Permite la instalación de middlewares² para manejar las peticiones HTTP.
- Permite la definición de tablas de enrutamiento, la cual es usada para ejecutar diferentes acciones basadas en el método HTTP y URL.
- Permite renderizar dinámicamente paginas HTML, mediante el paso de argumentos a plantillas.

MongoDB

MongoDB es un sistema de base de datos NoSQL³ orientado a documentos, desarrollado bajo el concepto de código abierto, que provee un fácil desarrollo, alto desempeño y alta escalabilidad. En lugar de guardar los datos en tablas como se hace en las base de datos relacionales, MongoDB guarda estructuras de datos en documentos similares a JSON con un esquema dinámico (MongoDB utiliza una especificación llamada BSON), haciendo que la integración de los datos en ciertas aplicaciones sea más fácil y rápida [61]. Entre algunas de las principales características que ofrece MongoDB están:

- Esquema dinámico.
- Fácil de escalar horizontalmente (Sharding).

² Una función que es invocada por la capa de enrutamiento de Express antes de pasar la petición al manejador final, esto es, situándose entre una petición sin procesar y el procesamiento final .

³ NoSQL hace referencia a una amplia clase de sistemas de gestión de bases de datos que difieren del modelo clásico de SGBDR en aspectos importantes, siendo el más destacado que no usan SQL como lenguaje principal de consultas. Los datos almacenados no requieren estructuras fijas como tablas, normalmente no soportan operaciones JOIN, ni garantizan completamente ACID (atomicidad, consistencia, aislamiento y durabilidad), y habitualmente escalan bien horizontalmente.

APÉNDICE C: TECNOLOGÍAS UTILIZADAS

Tabla 7.1: Comparación de conceptos en SGBDR y MongoDB.

SGBDR	MongoDB
Base de Datos(Database)	Base de Datos(Database)
Tabla(Table)	Colección(Collection)
Fila (Tuple/Row)	Documento(Document)
Columna (Column)	Campo(Field)
Uniones de tablas(Joins)	Documentos embebidos y referencias a documentos (referencias <code>_id</code>)
Llave primaria	Llave primaria (Llave por defecto <code>_id</code> proporcionada por MongoDB automáticamente)

- Alto desempeño por reducción de I/O.
- Alta disponibilidad por replicación (redundancia).
- Conversiones/mapeos de objetos de aplicaciones hacia objetos de la base de datos de forma intuitiva.
- Sin uniones de información complejas (No Joins).

MongoDB trabaja bajo el concepto de colecciones y documentos. Una **Base de Datos** es un contenedor físico que alberga colecciones. Cada base de datos tiene su propio conjunto de archivos en el sistema de archivos del sistema operativo. Una **colección** es un contenedor lógico de documentos. Las colecciones no tienen un esquema definido, esto es, los documentos dentro de la colección pueden tener diferentes campos (cantidad, tipo y contenido). Generalmente, todos los documentos en una colección son similares o de propósito relacionado. Un **documento** es un conjunto de pares *clave-valor*. Los documentos tienen esquemas dinámicos, es decir, los documentos dentro de una misma colección no necesitan tener el mismo conjunto de campos o la misma estructura. En la Tabla 7.1 se pueden ver de forma comparativa los conceptos entre un RDBMS tradicional y MongoDB [62] [63].

Apéndice D: Archivo de Configuración de NTP

```
1 # NTP Network Time Protocol
2 # please check http://www.ntp.org for additional
   documentation and background information
3
4 #for more details see http://news.meinberg.de/244
5 restrict default noquery nopeer nomodify notrap
6 restrict -6 default noquery nopeer nomodify notrap
7
8 # allow status queries and everything else from localhost
9 restrict 127.0.0.1
10 restrict -6 ::1
11
12 # Use drift file
13 driftfile "C:\Tools\NTP\etc\ntp.drift"
14
15 # Used the driver 20: Generic NMEA GPS Receiver
16 # see https://www.eecis.udel.edu/~mills/ntp/html/refclock.html
17 server 127.127.20.4 minpoll 4 prefer # NMEA serial port
18
19 # Used NTP servers from the ntp pool project
20 # see http://www.pool.ntp.org
21 #Mexico pool
22 server 0.mx.pool.ntp.org
23 server 1.north-america.pool.ntp.org
24 server 3.north-america.pool.ntp.org
25 #USA pool
26 server 0.us.pool.ntp.org iburst minpoll 6 maxpoll 7
27 server 1.us.pool.ntp.org iburst minpoll 6 maxpoll 7
28 server 2.us.pool.ntp.org iburst minpoll 6 maxpoll 7
```

APÉNDICE D: ARCHIVO DE CONFIGURACIÓN DE NTP

```
29  server 3.us.pool.ntp.org
30
31  # Options for generate statistics
32  # See https://www.meinbergglobal.com/download/ntp/docs/ntp\_cheat\_sheet.pdf
33  enable stats
34  stats loopstats
35  statsdir C:\Tools\NTP\stats
36  filegen loopstats file lpweek type week
```

Listado 7.1: Configuración utilizada para NTP.

Apéndice E: Algoritmo de Alineación en JavaScript

```
1  /* Procesando los datos recibidos */
2  var new_register= req.body;
3  new_register.timestamp=new Date(new_register.timestamp);
4  var ceiling=new Date();
5  var floor=new Date();
6  var nix=(new_register.timestamp).getTime();
7  ceiling.setTime(nix+g);
8  floor.setTime(nix-g);
9
10 strike.find(
11   {$and: [
12     {timestamp: {$gte: floor}},
13     {timestamp: {$lte: ceiling}},
14     {$or: [
15       {$and:[
16         {strikeType: new_register.strikeType},
17         {strikePolarity: new_register.strikePolarity}
18       ]},
19       {$and:[
20         {strikeType: -1},
21         {strikePolarity: -1}
22       ]}
23     ]}
24   ]}
25   ,function(err,result){
26     if(!err){
27       var select_group;
28       /*eliminacion de grupos que tienen asignado un dato
29         del mismo nodo*/
30       result=result.filter(function(value){
```

APÉNDICE E: ALGORITMO DE ALINEACIÓN EN JAVASCRIPT

```
30     for(var i=0;i<value.nodes.length;i++){
31         if( value.nodes[i] == new_register.node ){
32             /* ya existe un dato de este nodo dentro del
33                grupo */
34             return false;
35         }
36     }
37     /*se paso la prueba
38     return true;
39 }));
40
41 if( result.length==0){ /* Si ningun grupo fue
42     encontrado */
43     select_group=new strike();
44     select_group.create(new_register);
45
46     /* si se encontraron uno o mas grupos */
47 }else if(1 <= result.length){
48
49     /* si solo es un grupo, lo seleccionamos */
50     if(result.length==1){
51         select_group=result[0];
52
53     /*si quedan mas de un grupo reducimos los grupos
54     prospectos */
55 }else if( 1 < result.length ){
56
57     //buscamos el grupo mas cercano en tiempo
58     var near_index=0;
59     for(var i=0;i<result.length;i++){
60         if(
61             Math.abs(result[i].timestamp-new_register.
62                 timestamp)
63             <
64             Math.abs(result[near_index].timestamp-
65                 new_register.timestamp)
66         ){
67             near_index=i;
68         }
69     }
70     select_group=result[near_index];
71 }
72
73 /* agregado informacion el nuevo registro*/
74 select_group.nodes.push(new_register.node);
```

APÉNDICE E: ALGORITMO DE ALINEACIÓN EN JAVASCRIPT

```
70     select_group.datacount=select_group.datacount+1;
71
72     /* buscando tiempos minimos y maximo contra el
73     nuevo dato */
74     if(new_register.timestamp<select_group.mintime){
75         select_group.mintime=new_register.timestamp;
76         //tiempo intermedio entre dos estampas de
77         tiempo
78         select_group.timestamp=new Date();
79         select_group.timestamp.setTime(
80             select_group.mintime.getTime() +
81             ( (select_group.maxtime.getTime()-
82               select_group.mintime.getTime())/2 )
83         );
84     }
85
86     if(select_group.maxtime<new_register.timestamp){
87         select_group.maxtime=new_register.timestamp;
88         //tiempo intermedio entre dos estampas de
89         tiempo
90         select_group.timestamp=new Date();
91         select_group.timestamp.setTime(
92             select_group.mintime.getTime() +
93             ( (select_group.maxtime.getTime()-
94               select_group.mintime.getTime())/2 )
95         );
96     }
97
98     }//end if 1<= lenght
99
100    /* Almacenando los datos recibidos */
101    //creando nuevo registro
102    var new_data= new data(new_register);
103    select_group.save(function(err, group){
104        new_data.save(function(err, register){
105            if(!err){
106                // registro guardado!
107                select_group.data.push(register._id);
108                select_group.save(function(err,data){});
109            }else{
110                // error al guardar grupo!
111                console.err(err);
112            }
113        });
114    });
115    });
```

APÉNDICE E: ALGORITMO DE ALINEACIÓN EN JAVASCRIPT

```
110
111     }else{//end !err finds groups
112         console.err(err);
113     }
114 });//end find
```

Listado 7.2: Algoritmo de alineación escrito en JavaScript, con Node.js y MongoDB.

Bibliografía

- [1] G. González, “Visualización de descargas multi-sensadas usando trilateración sobre google maps,” M. Eng. thesis, Facultad de Ingeniería Eléctrica, Universidad Michoacana de San Nicolás de Hidalgo, Morelia, Michoacan, Dic. 2015.
- [2] C. Viveros, “Visualización de trayectorias de tormentas utilizando ventanas de tiempo y agrupamiento espacial basado en k-means,” M. Eng. thesis, Facultad de Ingeniería Eléctrica, Universidad Michoacana de San Nicolás de Hidalgo, Morelia, Michoacan, Dic. 2015.
- [3] (2018, Apr.) Proyectos Lab Electrica DEP FIE. Lab Electrica DEP FIE. [Online]. Available: 148.216.38.50/proyectos/
- [4] (2014, Jun.) Diccionario de la lengua española (22.a ed.). Real Academia Española. [Accessed: Mar. 14 2018]. [Online]. Available: <http://www.rae.es/rae.html>
- [5] R. Meerkötter and M. Vázquez-Navarro, *Earth’s Radiation Budget: The Driver for Weather and Climate*, ser. Research Topics in Aerospace. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, ch. 2, pp. 55–67. [Online]. Available: https://doi.org/10.1007/978-3-642-30183-4_4
- [6] C. Price, “Global thunderstorm activity,” in *Sprites, Elves and Intense Lightning Discharges*, ser. NATO Science Series II: Mathematics, Physics and Chemistry, M. Fullekrug, E. A. Mareev, and M. J. Rycroft, Eds., vol. 225. Dordrecht: Springer Netherlands, Jul. 2006, pp. 85–99.
- [7] (2009, Jan.) Climate and earth’s energy budget. NASA. [Online; accessed 19-June-2018]. [Online]. Available: <https://earthobservatory.nasa.gov/Features/EnergyBalance>

BIBLIOGRAFÍA

- [8] V. P. Pasko, Y. Yair, and C.-L. Kuo, “Lightning related transient luminous events at high altitude in the earth’s atmosphere: Phenomenology, mechanisms and effects,” *Space Science Reviews*, vol. 168, no. 1, pp. 475–516, Jun. 2012. [Online]. Available: <https://doi.org/10.1007/s11214-011-9813-9>
- [9] W. A. Lyons, “The meteorology of transient luminous events - an introduction and overview,” in *Sprites, Elves and Intense Lightning Discharges*, ser. NATO Science Series II: Mathematics, Physics and Chemistry, M. Fullekrug, E. A. Mareev, and M. J. Rycroft, Eds., vol. 225. Dordrecht: Springer Netherlands, Jul. 2006, pp. 19–56.
- [10] W. A. Lyons, CCM, M. A. Stanley, J. D. Meyer, T. E. Nelson, S. A. Rutledge, timothy L. Lang, and S. A. Cummer, *The Meteorological and Electrical Structure of TLE-Producing Convective Storms*. Dordrecht: Springer Netherlands, 2009, ch. 17, pp. 387–416. [Online]. Available: https://doi.org/10.1007/978-1-4020-9079-0_17
- [11] M. J. Rycroft, “Introduction to the physics of sprites, elves and intense lightning discharges,” in *Sprites, Elves and Intense Lightning Discharges*, ser. NATO Science Series II: Mathematics, Physics and Chemistry, M. Fullekrug, E. A. Mareev, and M. J. Rycroft, Eds., vol. 225. Dordrecht: Springer Netherlands, Jul. 2006, pp. 1–18.
- [12] V. Cooray, *An Introduction to Lightning*. Uppsala, Sweeden: Springer, 2015.
- [13] (2018, May) convection - Wikipedia. [Online; accessed 19-June-2018]. [Online]. Available: <https://en.wikipedia.org/wiki/Convection#Weather>
- [14] U. F. Hartmut Höller, Hans-Dieter Betz and K. Schmidt, *Lightning Detection*, ser. Research Topics in Aerospace. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, ch. 20, pp. 331–346. [Online]. Available: https://doi.org/10.1007/978-3-642-30183-4_20
- [15] B. Mills, D. Unrau, L. Pentelow, and K. Spring, “Assessment of lightning-related damage and disruption in canada,” *Natural Hazards*, vol. 52, no. 2, pp. 481–499, Feb 2010. [Online]. Available: <https://doi.org/10.1007/s11069-009-9391-2>

BIBLIOGRAFÍA

- [16] W. P. Roeder, B. H. Cummins, K. L. Cummins, R. L. Holle, and W. S. Ashley, “Lightning fatality risk map of the contiguous united states,” *Natural Hazards*, vol. 79, no. 3, pp. 1681–1692, Dec 2015. [Online]. Available: <https://doi.org/10.1007/s11069-015-1920-6>
- [17] Z. Nie, W. Zhao, P. Zhang, Y. Wen, and W. Chen, “Power network lightning accident quick inquiry with the lightning location system,” *Geo-spatial Information Science*, vol. 11, no. 1, pp. 43–48, Mar 2008. [Online]. Available: <https://doi.org/10.1007/s11806-007-0153-6>
- [18] R. Zoro and H. Laksmiwati, “Improvement of lightning protection system for transmission and distribution lines in high lightning density area,” in *2017 6th International Conference on Electrical Engineering and Informatics (ICEEI)*, Nov 2017, pp. 1–5.
- [19] D. Aranguren, J. González, A. Cruz, J. Inampué, H. Torres, and P. S. Pérez-Tobón, “Lightning strikes on power transmission lines and lightning detection in colombia,” in *2017 International Symposium on Lightning Protection (XIV SIPDA)*, Oct 2017, pp. 273–278.
- [20] B. Denov and R. Zoro, “Lightning protection system for oil and gas installation case on cilacap area at central java, indonesia,” in *2017 6th International Conference on Electrical Engineering and Informatics (ICEEI)*, Nov 2017, pp. 1–7.
- [21] (2007, Nov.) Boltek. lightning detection systems. Boltek. [Online]. Available: www.boltek.com/
- [22] Faq boltek. Boltek. [Online]. Available: www.boltek.com/boltek-faq
- [23] (2014, Nov.) Manual boltek pci. Boltek. [Online]. Available: www.boltek.com/manual-pci.pdf
- [24] (2018, Nov.) User manual boltek ld250. Boltek. [Online]. Available: <http://www.boltek.com/LD-250%20User%20Manual%2002232018.pdf>
- [25] (2017, Apr.) Nexstorm. Astrogenic Systems. [Online]. Available: www.astrogenic.com/nexstorm.html
- [26] (2017, Apr.) Nexstorm lite. Astrogenic Systems. [Online]. Available: www.astrogenic.com/nexstorm_lite.html

BIBLIOGRAFÍA

- [27] Downloads boltek. Boltek. [Online]. Available: www.boltek.com/downloads/
- [28] *NexStorm – User’s Manual*, 2nd ed., Astrogenic Systems, Jul. 2004.
- [29] H. Kopetz, *Global Time*, 2nd ed., ser. Real-Time Systems. New York, Dordrecht Heidelberg, London: Springer, 2011, ch. 3, pp. 51–78.
- [30] K. P. Birman, *Reliable Distributed Systems. Technologies, Web Services, and Applications*. New York: Springer, 2005, ch. 23, pp. 493–508.
- [31] L. Lamport, “Time, clocks, and the ordering of events in a distributed system,” *Commun. ACM*, vol. 21, no. 7, pp. 558–565, Jul. 1978. [Online]. Available: <http://doi.acm.org/10.1145/359545.359563>
- [32] (2012, May) Network Time Synchronization Research Project. Network Time Synchronization Research Project. [Online]. Available: <https://www.eecis.udel.edu/~mills/ntp.html>
- [33] (2008, 8) NTP accuracy - netcraftsmen. NetCraftsmen. [Online]. Available: <https://www.netcraftsmen.com/ntp-accuracy/>
- [34] P. Rybaczky, *Expert Network Time Protocol: An Experience in Time with NTP*, J. Sumsers, Ed. Apress, 2005.
- [35] (2018, Jul.) Windows Time Service Tools and Settings - Microsoft Docs. Microsoft. [Online]. Available: <https://docs.microsoft.com/en-us/windows-server/networking/windows-time-service/windows-time-service-tools-and-settings>
- [36] (2018, Jul.) Windows Time Service - Microsoft Docs. Microsoft. [Online]. Available: <https://docs.microsoft.com/en-us/windows-server/networking/windows-time-service/windows-time-service-top>
- [37] (2018, Jul.) High Accuracy W32time Requirements. Microsoft. [Online]. Available: <https://blogs.technet.microsoft.com/askds/2007/10/23/high-accuracy-w32time-requirements/>
- [38] (2018, Jul.) External Time RelatedLinks <Main <NTP. Network Time Foundation. [Online]. Available: <http://support.ntp.org/bin/view/Main/ExternalTimeRelatedLinks>

BIBLIOGRAFÍA

- [39] (2018, Jul.) Meinberg NTP Software Downloads. Meinberg. [Online]. Available: <https://www.meinbergglobal.com/english/sw/ntp.htm>
- [40] (2018, Feb.) Installing NTP on Windows. satsignal. [Online]. Available: <http://www.satsignal.eu/ntp/setup.html>
- [41] (2014, Mar.) Reference Clock Support. [Online]. Available: <https://www.eecis.udel.edu/~mills/ntp/html/refclock.html>
- [42] (2018, May) An NTP Stratum-1 clock using a GPS 18 LVC and Windows 2000/XP/Windows-7. satsignal. [Online]. Available: <http://www.satsignal.eu/ntp/NTP-on-Windows-serial-port.html>
- [43] (2018, Jul.) Configuring NMEA Refclocks <Support <NTP. Network Time Foundation. [Online]. Available: <https://support.ntp.org/bin/view/Support/ConfiguringNMEARefclocks>
- [44] (2018, May) Creating Named Shared Memory — Microsoft Docs. Microsoft. [Online]. Available: <https://docs.microsoft.com/es-mx/windows/desktop/Memory/creating-named-shared-memory>
- [45] (2008) NTP Cheat Sheet: ntpd Configuration File. Meinberg. [Online]. Available: https://www.meinbergglobal.com/download/ntp/docs/ntp_cheat_sheet.pdf
- [46] (2018, Jul.) NTP Time Server Monitor. Meinberg. [Online]. Available: <https://www.meinbergglobal.com/english/sw/ntp-server-monitor.htm>
- [47] T. Vincenty, “Direct and inverse solutions of geodesics on the ellipsoid with application of nested equations,” *Survey Review*, vol. 23, pp. 88–93, Apr. 1975. [Online]. Available: https://www.ngs.noaa.gov/PUBS_LIB/inverse.pdf
- [48] C. Veness. (2018) Vincenty solutions of geodesics on the ellipsoid in javascript — movable type scripts. Movable Type Scripts. [Online]. Available: <https://www.movable-type.co.uk/scripts/latlong-vincenty.html>
- [49] U. Finke, “Optical detection of lightning from space,” in *Lightning: Principles, Instruments and Applications*, H. D. Bentz, U. Schumann, and P. Laroche, Eds. Dordrecht: Springer Netherlands, 2009, ch. 12, pp. 271–286. [Online]. Available: https://doi.org/10.1007/978-1-4020-9079-0_12

BIBLIOGRAFÍA

- [50] (2007, Nov.) Ghrc: Optical transient detector. NASA. [Accessed: Mar. 22 2018]. [Online]. Available: <https://lightning.nsstc.nasa.gov/otd/>
- [51] (2018, May) Trmm mission comes to an end after 17 years. NASA. [Accessed: Mar. 14 2018]. [Online]. Available: <https://pmm.nasa.gov/trmm/mission-end>
- [52] H. J. Christian, “Optical detection of lightning from space,” in *Proceedings of the 11th International Conference on Atmospheric Electricity*, Guntersville, Alabama, Jun 7-11 1999, pp. 175–718.
- [53] D. Fensel, F. M. Facca, E. simperl, and I. Toma, *Web Services*. Verlag Berlin Heidelberg: Springer, 2011, ch. 4, pp. 37–66.
- [54] P. Adamczyk, P. H. Smith, and R. E. Johnson, *REST and Web Services: In Theory and Practice*. Berlin Heidelberg London: Springer, 2011, ch. 2, pp. 35–57.
- [55] M. Sehrish and K. Do-Hyeum, “A comparison of RESTful vs. soap web services in actuator networks,” *IEEE*, 2017.
- [56] D. Fensel, F. M. Facca, E. simperl, and I. Toma, *Web2.0 and RESTful Services*. Verlag Berlin Heidelberg: Springer, 2011, ch. 5, pp. 67–86.
- [57] B. A. Syed, *Beginning Node.js*. Apress, 2014.
- [58] (2018, 8) Node.js. Node.js Foundation. [Online]. Available: <https://nodejs.org/es/>
- [59] C. Gackenhaimer, *Node.js Recipes*. Apress, 2013.
- [60] (2017, 8) Express - node.js web application framework. Node.js Foundation. [Online]. Available: <http://expressjs.com/>
- [61] E. Plugge, P. Membrey, and T. Hawkins, *The Definitive Guide to MongoDB*. Apress, 2010.
- [62] (2018, 8) MongoDB for giant ideas. mongodb. MongoDB Inc. [Online]. Available: <https://www.mongodb.com/>

BIBLIOGRAFÍA

- [63] J. Benymol and S. Abraham, “Exploring the merits of NoSQL: A study based on mongodb,” in *2017 International Conference on Networks Advances in Computational Technologies (NetACT)*, 7 2017, pp. 266–271. [Online]. Available: [10.1109/NETACT.2017.8076778](https://doi.org/10.1109/NETACT.2017.8076778)