



**UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN**

**DESARROLLO DE SOFTWARE PARA ESTIMAR VELOCIDADES
DE LOS BUSES DEL TRANSANTIAGO**

MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL EN COMPUTACIÓN

MAURICIO ANDRÉS ZÚÑIGA GAJARDO

**PROFESOR GUÍA:
MARCELA ADRIANA MUNIZAGA MUÑOZ**

**MIEMBROS DE LA COMISIÓN:
NANCY VIOLA HITSCHFELD KAHLER
JUAN FERNANDO ALVAREZ RUBIO**

**SANTIAGO DE CHILE
ABRIL DE 2010**

Resumen

La Universidad de Chile y la agencia que regula la operación del transporte público de Santiago, Transantiago, acordaron en el marco de un convenio crear y desarrollar una metodología y software capaz de procesar la información provista por los buses de este sistema para entregar una estimación de las velocidades comerciales en algunos tramos de calles de Santiago. En la actualidad este tipo de información es provista a través de observación en terreno con recursos humanos o utilizando software con metodologías demasiado globales que entregan resultados con un alto margen de error. Esto significa altos costos, poca información y mediana o baja confiabilidad.

El objetivo general del presente trabajo de título es diseñar y construir una herramienta capaz de estimar las velocidades comerciales en la red de buses de Transantiago, utilizando los datos GPS, con el objeto de proveer información relevante para un diagnóstico global que contribuya a la mejora del servicio. Por velocidad comercial se entiende la velocidad de avance a lo largo de la ruta o en un tramo de esta, la cual es percibida por el pasajero.

El cálculo de velocidades debió ser realizado de forma tal que tomara en cuenta la agregación temporal y espacial necesaria para obtener un promedio. En este trabajo se decidió tomar la velocidad representativa de un elemento de una grilla espacio-tiempo, determinando los tiempos y distancias de cada bus en ese elemento de la grilla para luego sumarlos y obtener el cociente entre la suma de las distancias sobre la de los tiempos. En forma paralela se diseñaron e implementaron variados métodos de visualización de esta información.

El resultado final de este trabajo de título es un software capaz de calcular las velocidades y frecuencias de las distintas líneas de buses que pertenecen al sistema de Transantiago y que además permite el diagnóstico de problemas a través de un novedoso método de visualización de la información basado en imágenes que representan la velocidad de una línea en cierto tramo de ruta en cierto periodo de tiempo.

En la actualidad la información generada por este software, que se traduce en archivos visualizables en Google-Earth, ya es utilizada por Transantiago para estudiar la situación actual de cada línea respecto a su velocidad comercial, y la herramienta se proyecta como un real apoyo a la toma de decisiones que tienen un alto impacto social.

Agradecimientos

A mi amor Carla por su apoyo incondicional.

A mi hijo Sebastián por la motivación.

A mis padres Silvia y Hernán porque sin ellos hubiese sido imposible.

A mi hermana María José y a mi sobrina Catalina por sus apoyos.

A mis tías Ana María y Edu por insistir en la perseverancia.

A mis amigos por recordarme terminar.

Índice

Resumen	2
1. Introducción.....	7
1.1 Motivación y justificación.....	8
1.2 Objetivos.....	9
1.3 Contenido de memoria.....	10
2. Antecedentes.....	11
2.1 Transantiago	11
2.2 Demanda.....	17
2.3 Estado del arte	19
2.4 Terminología	21
3. Metodología.....	22
3.1 Metodología de Trabajo.....	22
3.2 Metodologías para estimar las velocidades.	24
4. Diseño e Implementación	32
4.1 Herramientas de apoyo	32
4.2 Base de Datos	35
4.3 Software estimación de velocidades.....	47
4.4 Interfaces	66
5. Resultados.....	79
5.1 Visualización	79
5.2 Base de datos	85
5.3 Evaluación de servicio.....	86
5.4 Comparaciones entre servicios	89
6. Conclusiones.....	92
6.1 Trabajo futuro	93
7. Bibliografía	97
8. Anexos	99
8.1 Definición de requerimientos	99
8.2 Scritp SQL.....	101

Índice de Figuras

Figura 1 : División de zonas en Santiago.	11
Figura 2 : Servicios Troncales (Izquierda), servicios locales (derecha).....	12
Figura 3 : Red de Transantiago.....	13
Figura 4 : Paradas locales y troncales, señalética (derecha).....	14
Figura 5 : Estación de Trasbordo.....	14
Figura 6 : Estación modal, fotografía (izquierda) esquema (derecha)	15
Figura 7 : Histograma de cantidad de transacciones cada 30 minutos.	17
Figura 8 : Demanda, 6:45 (izquierda-arriba), 8:15 (derecha-arriba), 15:00 (izquierda-abajo) y 18:00(derecha-abajo).	18
Figura 9: Método I media de velocidades instantáneas por tramo y por periodo.....	24
Figura 10: Método II, media de velocidades calculadas con dos pulsos.....	25
Figura 11: Método III	26
Figura 12: Método IV para un Bus	26
Figura 13: Error de estimación por curvas.	27
Figura 14: Error de estimación por escala.	27
Figura 15: Cálculo de las velocidades para el método V	28
Figura 16: Datos antes de proyectarse (diagrama superior), datos proyectados (diagrama inferior).	29
Figura 17: Esquema método VII	30
Figura 18: Casos especiales método VII	31
Figura 19: Herramientas de apoyo	34
Figura 20: Datos originales en base de datos.	37
Figura 21: Tabla original de asignación de servicio.....	38
Figura 22: Ruta Excel (izquierda-superior) y tabla base de datos (derecha-inferior)	41
Figura 23: Estructura final de la tabla de posicionamiento GPS de los buses (abajo), tabla original (arriba)	45
Figura 24: Diagrama de flujo de la rutina que calcula las velocidades para el Método VII (final).	47
Figura 25: Formato de archivo de parámetros.....	50
Figura 26: Ruta original para el servicio 506 sentido Ida.	52
Figura 27: Exceso de nodos en la ruta (Abajo), comportamiento erróneo (Arriba).....	53
Figura 28: Ejemplo método de simplificación de polígonos.....	54
Figura 29: Ejemplo ruta original (azul) y ruta simplificada (verde).....	55
Figura 30: Ejemplo de aplicación de algoritmo de tramificación (ruta 506).....	55
Figura 31 : Conversión de coordenadas LL a UTM.....	60
Figura 32: Buses fuera de ruta.....	62
Figura 33: Emisiones GPS que saltan espacialmente kilómetros por segundo.	63
Figura 34 : Interfaz Google-Earth	68
Figura 35: Ruta coloreada según magnitud de velocidades comerciales.	72
Figura 36: Ejemplo de gráfico creado con la librería ChartDirector.	73
Figura 37: Representación de las velocidades medias por tramo (abscisa) y por periodo (ordenada) creada utilizando gnuplot desde el software en C++.	74
Figura 38: Ruta del servicio 506, del periodo 8:00 a 8:30.	80
Figura 39: Ruta coloreada, gráfico de velocidad-observaciones, información adicional al hacer click con el mouse en un nodo de la ruta.....	80
Figura 40: La Gráfico para el servicio 506.....	81
Figura 41: Despliegue de la información de cada patente para el tramo 38.....	82

Figura 42: Dibujo de todas las trayectorias de los buses involucradas con algún periodo-sección de la ruta.	83
Figura 43: Trayectorias de los buses proyectadas a la ruta (servicio 506).	84
Figura 44: Base de datos con velocidades para todos los servicios-tramo-periodo.	85
Figura 45: Grafico de velocidades representativas (índice) por cada servicio en cada periodo del día.	86
Figura 46 : Las figuras representan todos los servicios, la imagen de la izquierda corresponde a Septiembre 2008, la del centro a Marzo 2009 y la de la derecha a Abril 2009.	87
Figura 47: Histograma de las velocidades representativa para las tres semanas procesadas.	89
Figura 48: Comparación entre las medias de todos los servicios entre Marzo y Septiembre.	90
Figura 49: Comparación de las medias de los índices entre semanas similares, Marzo y Abril del 2009.	90
Figura 50: Los puntos amarillos representan la existencia de un paradero en esa posición.	94
Figura 51: Subidas de los pasajeros en los paraderos donde fueron realizadas.	95
Figura 52: Posiciones de los semáforos. Cada punto en el mapa es un semáforo.	95
Figura 53: Frecuencias e intervalos de tiempo de buses por paradero.	96

Índice de Tablas

Tabla 1: Operadores de Transantiago.	16
Tabla 2: Periodos de un día laboral	19
Tabla 3: Métodos de escala de colores continuas.	76
Tabla 4: Escala de colores usada para las velocidades.	79

1. Introducción

Hoy en día el transporte público en Santiago se propone como una opción eficiente, en términos de uso de energía y del espacio urbano. No obstante, esto no significa que sea la opción elegida por la mayoría de las personas. Las razones para elegir un modo u otro son variadas: comodidad, tiempos de viaje, caminata, tiempos de espera. Hasta ahora muchas personas siguen siendo usuarios de transporte público, y una de las principales preocupaciones del gobierno es que esto se mantenga, de esta forma uno de los objetivos para lograr esto es mantener una buena calidad de servicio en el sistema de transporte público para que los usuarios no emigren a otras alternativas.

Transantiago es un moderno sistema de transporte público, el cual opera a través de distintas empresas, llamadas operadores. Una de las características que distingue este sistema de otros es la forma de pago de los usuarios, esta consiste en el uso de una tarjeta personal RFID (identificación por radio frecuencia) llamada tarjeta BIP, la cual es cargada con dinero por el usuario en centros de pago proporcionados por Transantiago. Por otra parte en cada lugar donde el usuario suba a un bus o estación de metro existen máquinas especiales llamadas validadores donde se acerca la tarjeta para realizar la transacción de pago y autorizar el acceso.

Transantiago recibe y almacena los datos generados automáticamente por la operación del sistema, estos consisten principalmente en las transacciones BIP y las posiciones GPS de los buses cada 30 segundos. Con estos últimos datos hoy se obtiene además la asignación de servicio de los buses, información clave para que Transantiago pueda fiscalizar que los operadores cumplan con sus responsabilidades.

La División de Ingeniería de Transporte del departamento de Ingeniería Civil de la Universidad de Chile en conjunto con Transantiago realizó un convenio para generar investigación y desarrollo utilizando los datos descritos anteriormente. Este acuerdo consiste en la entrega de datos y horas hombre de los especialistas por parte de Transantiago y la investigación y desarrollo por parte de la universidad el cual debe ser finalmente transferido. Actualmente se investigan variados temas tales como patrones de viajes, perfiles de carga, matriz origen/destino, entre otros.

Dentro del marco de esta alianza, durante el verano del 2008, se dio inicio a un estudio con el objetivo de estimar y diagnosticar las velocidades de un servicio del Transantiago utilizando la información provista por el GPS. Este trabajo consiste en la investigación y el desarrollo realizados para construir una herramienta capaz de determinar las velocidades de los buses para cada uno de sus servicios.

1.1 Motivación y justificación

Una de las características relevantes de todo sistema de transporte público es su calidad de servicio la cual se evalúa mediante varios atributos, tales como, tiempos de espera, cantidad de trasbordos, tiempos de viaje, comodidad en el viaje, entre otros. Hoy en día cada uno de estos índices es medido a través de encuestas y trabajo en terreno, lo que implica una pequeña muestra de datos, no necesariamente confiables, y con un alto costo asociado.

La velocidad comercial es un importante factor en la operación de sistemas de transporte público, ya que representa una medida directa de la calidad del servicio prestado a los usuarios de y afecta considerablemente el costo del sistema. Monitorear el rendimiento del sistema por medio de la velocidad comercial prestada por los servicios de autobús es por supuesto muy importante, sin embargo, en una densa red se convierte en una tarea muy difícil debido a la cantidad de información necesaria para aplicar un procedimiento de monitoreo. La introducción de la tecnología GPS en los autobuses supera esta dificultad en términos de disponibilidad de la información, pero a su vez trae consigo el reto de procesar enormes cantidades de información en forma sistemática.

Conociendo todos o algunos de estos atributos los expertos pueden evaluar si el servicio cumple con ciertas condiciones mínimas o en caso contrario tener una idea de cuál es la causa por la que no se cumplen estas condiciones. De esta manera el conocimiento de esta información es imprescindible para tomar decisiones adecuadas en la evolución de un sistema de transporte público.

La principal motivación de este estudio nace de la necesidad de conocer las velocidades comerciales de los servicios de Transantiago, procesando los datos provistos por los GPS de los buses, de tal manera de extraer y proveer una información más completa, ágil, económica y fidedigna que la actual.

Una segunda motivación consiste en generar información valiosa para las diversas investigaciones realizadas en la División de Ingeniería de Transporte, esta consistiría en las velocidades comerciales y flujo vehicular en distintos tramos de las calles de Santiago. Esta información ya ha sido utilizada en investigaciones en el área de micro simulación donde se compararon las velocidades obtenidas por este estudio con las simuladas, con el objetivo de validar las simulaciones.

Una tercera motivación se centra en enfrentar el desafío de manejar bases de datos de gran tamaño en forma óptima para responder con tiempos de cálculos dentro de los márgenes necesarios para ser útiles a nivel operativo.

Por lo tanto, es imprescindible disponer de una plataforma y software que sea capaz de extraer la información de las velocidades comerciales sólo utilizando los datos digitales provistos por Transantiago, de tal manera de alimentar con información real a las diversas investigaciones de la universidad y a los expertos de Transantiago.

1.2 Objetivos

El objetivo general de este Trabajo de Titulación es diseñar y desarrollar un software que permita calcular, visualizar y diagnosticar las velocidades comerciales de los buses de Transantiago, utilizando los datos entregados por Transantiago.

Y a su vez cumplir con los siguientes objetivos específicos:

1. Implementar distintos métodos para el cálculo de las velocidades y compararlos para definir qué método es el más indicado para el caso de estudio.
2. Exportar los datos entregados por Transantiago a una base de datos.
3. Generar código fuente reutilizable para otras investigaciones, por ejemplo, manejo de rutas y visualización de datos, entre otros.
4. Diseñar el software con la flexibilidad necesaria para integrar otros tipos de información, por ejemplo, transacciones bips y semáforos.
5. Crear documentación del software, documentación de código fuente, documentación de arquitectura y tutoriales para su uso.
6. Utilizar herramientas de código abierto para la realización del software, para agilizar el desarrollo evitando gestiones para la adquisición de herramientas comerciales, como también para eliminar costos.

1.3 Contenido de memoria

El contenido de este informe se distribuye en seis capítulos principales los cuales describen y detallan todas las etapas de este trabajo de título.

El primer capítulo consiste en la introducción al tema de este trabajo explicando donde está localizado, en qué consiste y cuáles son los objetivos a resolver.

El segundo capítulo explica y detalla el ambiente donde se desarrolla la problemática a resolver, una breve explicación del sistema de transporte público y los actuales métodos usados para estimar las velocidades comerciales.

El tercer capítulo muestra cada una de las metodologías usadas en este trabajo, tanto la metodología de trabajo como las metodologías creadas para resolver el problema de estimación de velocidades.

El cuarto capítulo contiene todo el detalle del diseño e implementación de cada uno de los módulos de software creados, este se divide en tres subcapítulos los que describen las bases de datos, estimación de velocidades e interfaces en forma separada.

El quinto capítulo muestra todos los resultados obtenidos en este trabajo, tanto de los módulos descritos en el capítulo cuatro por separado como también los resultados obtenidos funcionando en conjunto. Además contiene análisis de los resultados de la herramienta y comparaciones entre resultados de distintos datos.

El sexto y último capítulo muestra las conclusiones obtenidas con este trabajo y el trabajo futuro para mejorar esta herramienta.

2. Antecedentes

Este capítulo tiene por objetivo explicar con mayor detalle el ambiente donde se ubica el problema, las variables que lo componen, la situación actual de la estimación de velocidades y por último explicar la terminología del área.

2.1 Transantiago

En su sitio web Transantiago se define como un proyecto que hoy busca mejorar la cobertura del transporte público de Santiago disminuyendo el número de transbordos, bajando los tiempos de espera y aumentando la oferta de recorridos. A través de un rediseño del sistema, se espera implementar un nuevo transporte de calidad para los usuarios.

2.1.1 Zonas y Servicios

El sistema está compuesto por el servicio de buses y Metro, y se organiza en base a zonas y rutas, la Figura 1 muestra como se distribuyen las distintas zonas sobre la ciudad de Santiago.



Figura 1 : División de zonas en Santiago.



Figura 2 : Servicios Troncales (Izquierda), servicios locales (derecha)

La Figura 2 muestra los dos tipos de servicios que existen en las zonas, troncales y locales, como se aprecia cada tipo de servicio y zona tiene asignado un color identificativo.

Los servicios troncales conectan una zona con otra, cruzando los ejes importantes de la ciudad. El Metro de Santiago también opera como un servicio troncal más de Transantiago. Los Troncales se identifican como buses blancos con franjas verdes en sus costados, y tienen un número asignado de tres dígitos (101, 309, etc.).

Los servicios locales son recorridos específicos dentro de cada zona de servicio y su finalidad es acercar a los lugares de acceso a un recorrido troncal. Los locales se identifican con el color de su zona correspondiente y tienen asignada la letra de esta zona y un número de dos dígitos (B12, F14, etc.).

2.1.2 Rutas

Todos estos servicios conforman una gran y compleja red de transporte público que podemos observar globalmente en la Figura 3, donde se aprecia cada una de las zonas del sistema diferenciadas con colores y las rutas que las componen. Cada una de estas rutas es cubierta por uno o más servicios que pueden ser de tipo troncal o local.

Cada servicio tiene asignada su ruta correspondiente, pero en algunos casos esto no es rígido, en otras palabras, algunos servicios varían su ruta por diversas razones, desvíos por construcciones, calles que cambian de sentido según horario, ferias libres, entre otros.

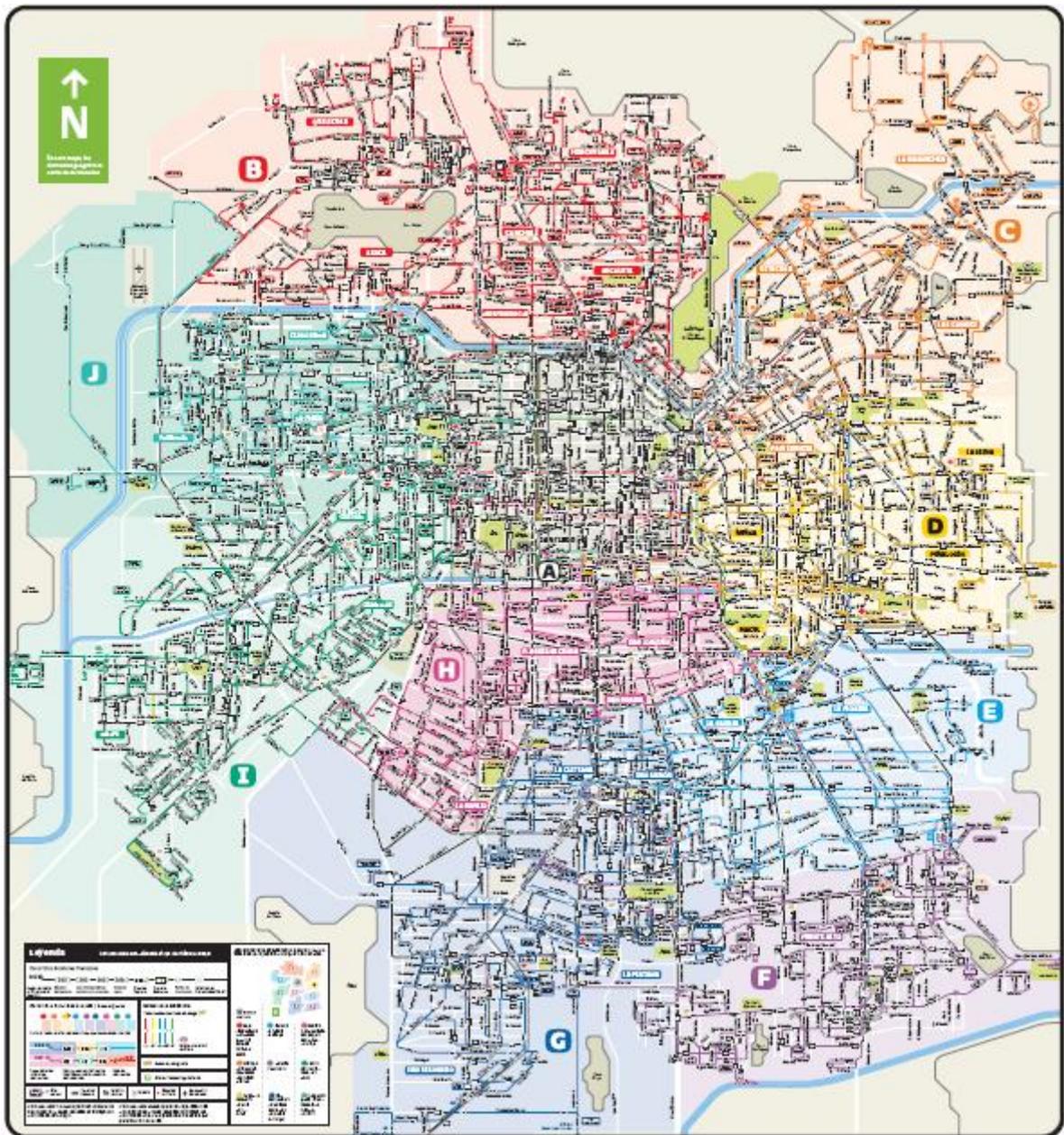


Figura 3 : Red de Transantiago

2.1.3 Paradas o paraderos

Otro elemento importante dentro de este sistema son los paraderos o lugar donde el público toma los servicios o hace trasbordos. Estos lugares se dividen en tres tipos que se describen a continuación:

Paradas locales y troncales: Corresponde a puntos autorizados para que los usuarios puedan subir o descender de los buses. Una misma parada puede ser utilizada indistintamente para servicios troncales o locales. Cuentan con una señal vertical con los datos de la parada y los recorridos que se detiene en ese punto. En el caso de las paradas que cuentan también con refugio, existe un panel de información que contiene mapas de los recorridos Transantiago de la zona en que se encuentra instalado y datos útiles para el usuario en su viaje. La Figura 4 muestra una fotografía de una de estas paradas y la señalética que utilizan.



Figura 4 : Paradas locales y troncales, señalética (derecha)

Estaciones de Traslado: Puntos para facilitar y hacer más expedita la transferencia de pasajeros entre los distintos servicios, ya sea entre buses troncales y locales o con Metro. En Santiago hay 36 estaciones de traslado. Cada una está compuesta por una o más paradas y tienen señalética que indica los servicios disponibles en cada uno de estos puntos. La Figura 5 muestra una imagen de una de estas paradas y un esquema de cómo están conformadas.

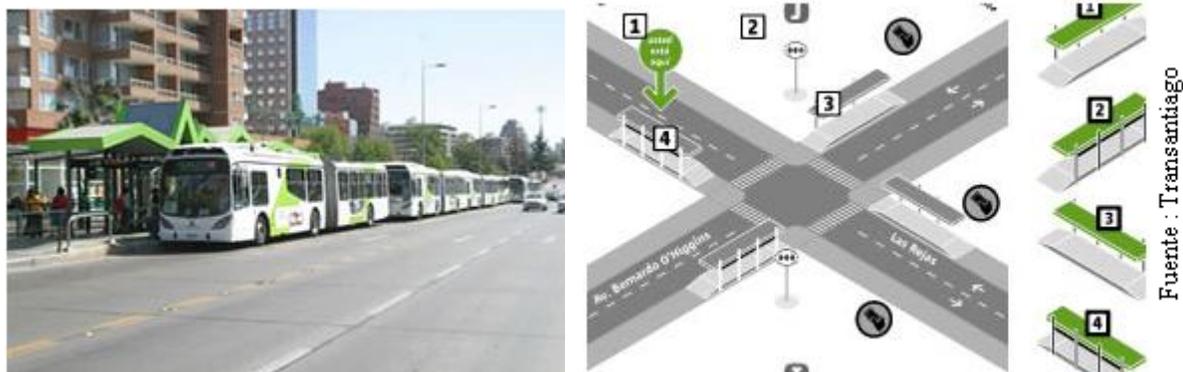


Figura 5 : Estación de Traslado.

Estación de Intercambio Modal: Es posible hacer transbordo entre distintos medios de transporte: buses, Metro, taxis, colectivos, entre otros. Esta estación se ubica en La Cisterna. La Figura 6 muestra una fotografía de la estación junto con un esquema.



Fuente : Transantiago

Figura 6 : Estación modal, fotografía (izquierda) esquema (derecha)

2.1.4 Tipos de vías

Dentro del sistema no todas las vías por donde pasan los buses tienen igual características, esto es muy importante porque estas características tienen directa implicancia en las condiciones de tráfico que debe enfrentar cada servicio. A continuación una breve explicación de cada una de estas vías.

Vías segregadas: Son pistas para el uso de buses que se encuentran físicamente separadas del resto de la calle a través de tachones.

Vías exclusivas: Son ejes con uso exclusivo para la circulación de buses Transantiago entre las 7:30 hrs y las 10:00 hrs., de lunes a viernes.

Pistas Sólo Buses: Son ejes de la ciudad con pistas demarcadas exclusivas para buses.

2.1.5 Operadores

Cada uno de los servicios que participan en el sistema es operado por una empresa concesionaria, cada una de estas empresas es comúnmente llamada operadora u operador. Es importante saber que uno de los objetivos de Transantiago es poder evaluar a cada una de estas empresas tanto fiscalizando la realización de las tareas acordadas como también midiendo el nivel de calidad de servicio, puesto que existen incentivos en dinero por cumplir algunas metas de buena calidad de servicio, y uno de los parámetros con los cuales se mide esta calidad es la velocidad comercial.

La Tabla 1 muestra cada uno de los operadores del sistema actual con el número de buses y servicios con el que participa en el sistema, algunos manejan más de una zona por lo que en la tabla aparecen por separado. Como se observa la distribución de buses no es homogénea, depende de la demanda de las zonas que operan, es importante tener presente en los análisis que la muestra de observaciones GPS variará según zona.

Operador	Nº Buses	Nº Servicios
A5 - Zona G	137	100
Buses Gran Santiago S.A. (T3)	805	141
Buses Gran Santiago S.A. (Zona B)	274	114
Buses Metropolitana S.A.	493	44
Comercial Nuevo Milenio S.A. (Zona I)	397	95
Comercial Nuevo Milenio S.A. (Zona J)	230	130
Express de Santiago Uno S.A.	897	169
Inversiones Alsacia S.A.	615	51
Redbus Urbano S.A.	286	83
STP Santiago S.A. (Zona D)	193	55
STP Santiago S.A. (Zona F)	274	126
Subus Chile S.A.	792	166
Testing Sonda	4	0
Transantiago	505	243
Trans Araucarias S.A.	164	38
Union de Transporte S.A.	180	42

Tabla 1: Operadores de Transantiago.

2.2 Demanda

Un factor muy relevante en cuanto a influencia en el tráfico y calidad de servicio es la demanda de pasajeros, esta varía considerablemente según la hora con las mayores alzas entre 6:30 AM – 8:30 AM (Punta Mañana) y 17:30 PM – 20:30 PM (Punta Tarde). La Figura 7 muestra como se distribuyen las transacciones bips durante un día laboral cada treinta minutos.

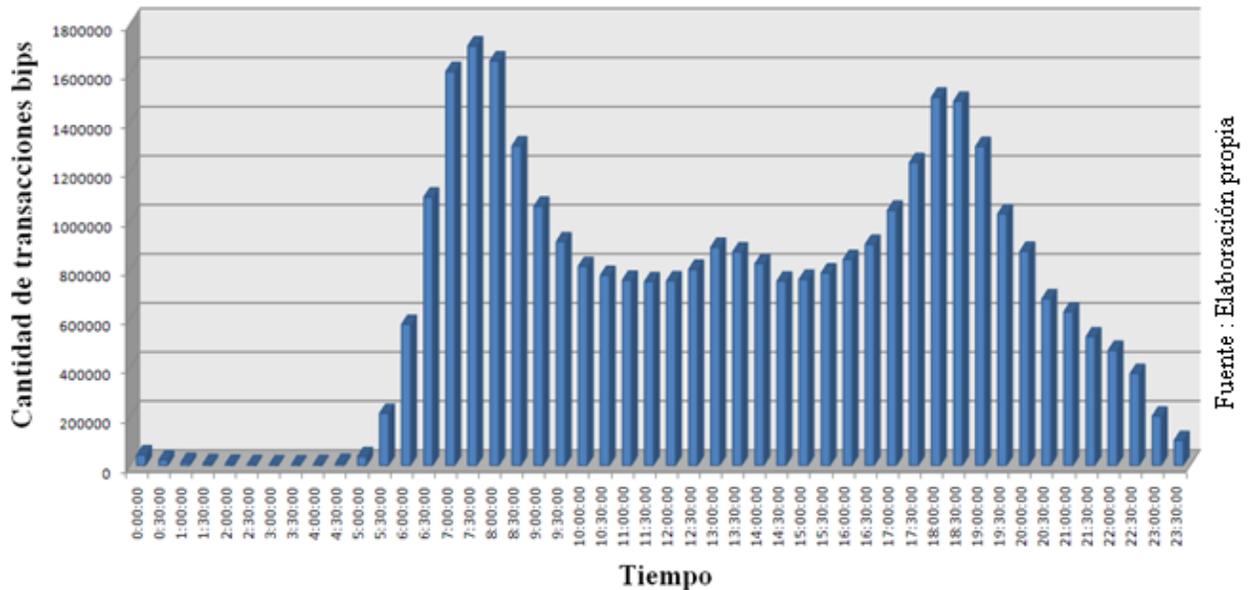


Figura 7 : Histograma de cantidad de transacciones cada 30 minutos.

Este fenómeno que aparece en los horarios donde la población se dirige a sus lugares de trabajo y cuando vuelven a sus hogares es la razón de la congestión en las condiciones de tráfico y por ende en la baja de las velocidades del sistema.

Un punto importante es la distribución espacial de esta demanda que también depende del tiempo, y a la hora de diagnosticar un problema se debe considerar. La Figura 8 muestra como se presenta la demanda muy temprano en la mañana y como varía a medida que corre el tiempo, cada círculo corresponde a una parada y su tamaño es proporcional a la cantidad de transacciones que son realizadas en ella en un periodo de tiempo de quince minutos.

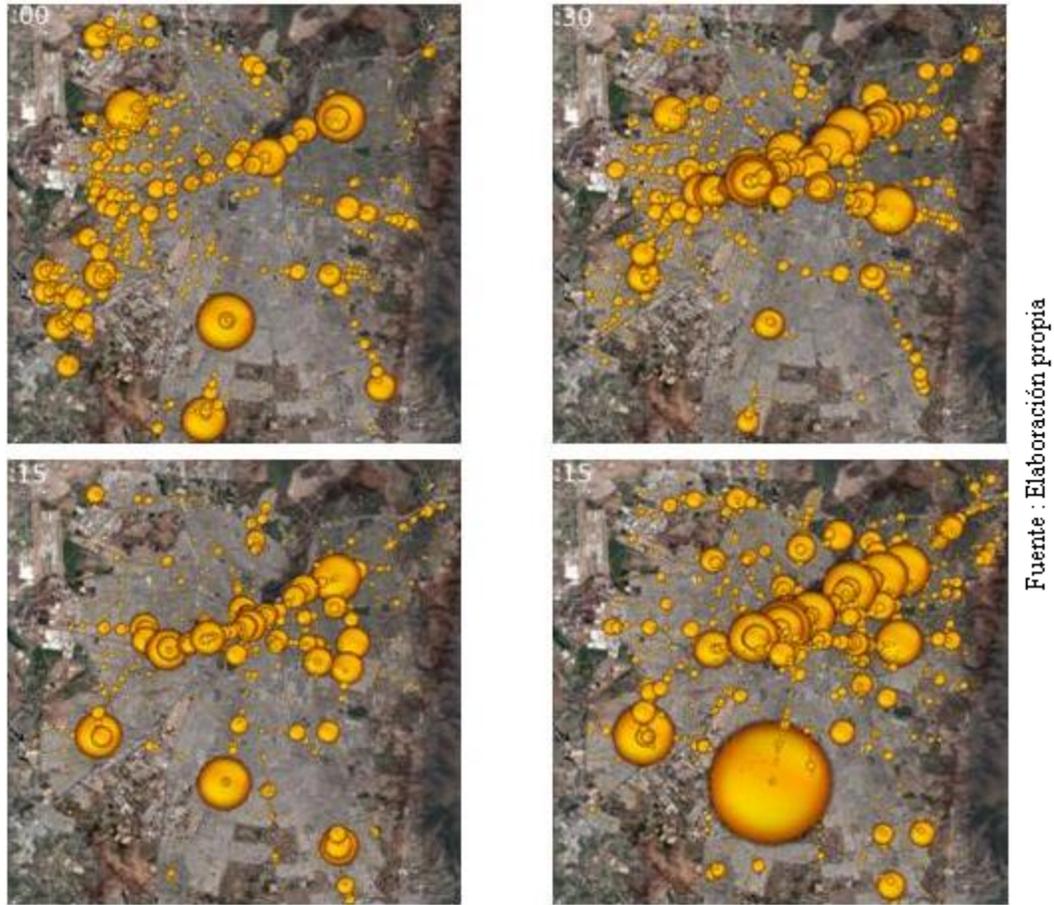


Figura 8 : Demanda, 6:45 (izquierda-arriba), 8:15 (derecha-arriba), 15:00 (izquierda-abajo) y 18:00(derecha-abajo).

Como se aprecia, la demanda es un elemento del sistema que tiene una gran variabilidad y eso dificulta bastante cualquier estudio que involucre al transporte público. En este trabajo en particular es necesario tener claro este fenómeno para poder diagnosticar problemas con las velocidades ya que es una de las principales causales de las bajas velocidades por las condiciones del tráfico y los tiempos de transferencia de pasajeros en las paradas.

Entendiendo en términos generales el comportamiento de la demanda de pasajeros Transantiago ha construido una clasificación temporal del día en periodos con cierta similitud en la demanda, la Tabla 2 muestra estos módulos durante un día laboral.

Nombre Periodo	Inicio	Término	Nº Horas
01 - Pre Nocturno	0:00	0:59	1
02 - Nocturno	1:00	5:29	4.5
03 - Transición Nocturno	5:30	6:29	1
04 - Punta Mañana	6:30	8:29	2
05 - Transición Punta Mañana	8:30	9:29	1
06 - Fuera de Punta Mañana	9:30	12:29	3
07 - Punta Mediodía	12:30	13:59	1.5
08 - Fuera de Punta Tarde	14:00	17:29	3.5
09 - Punta Tarde	17:30	20:29	3
10 - Transición Punta Tarde	20:30	21:29	1
11 - Fuera de Punta Nocturno	21:30	22:59	1.5
12 - Pre Nocturno	23:00	23:59	1

Tabla 2: Periodos de un día laboral

2.3 Estado del arte

Hoy en día las velocidades del sistema son estimadas con distintas estrategias que van desde lo más práctico como medir en terreno a las más sofisticadas herramientas de software, en esta sección se entregará una pequeña descripción de estos métodos utilizados por Transantiago y a su vez una reseña de otros métodos utilizados en el mundo.

2.3.1 Área de investigación

El uso de datos GPS para monitorear la velocidad es una tendencia creciente. Varios autores ya han aplicado algunos métodos con los datos GPS con diferentes enfoques. Por ejemplo, Storey y Holtom (2003) [12] presentan una aplicación en el Reino Unido utilizando datos históricos de diferentes tipos de vehículos. Jiang y Li (2001) [6] presentan un estudio del efecto de las carreteras en áreas de trabajo (en la autopista de Indiana) en la velocidad utilizando un vehículo de pruebas equipado con GPS, también se estiman las tasas de aceleración y desaceleración, usando intervalos de un segundo. Greaves y Figliozzi (2008) [4] usaron una encuesta GPS en vehículos comerciales en Melbourne, Australia para obtener los perfiles de velocidad de los camiones. Sin embargo, un factor común es que todos los autores concluyen que los datos GPS proporciona una fuente fiable de información con un gran potencial para el control de velocidad. Storey y Holtom (2003) [12] concluyen que las medidas de la velocidad GPS son más precisas que las medidas de vehículos observadores en movimiento, mientras que Jiang y Li (2001) [6] concluyen que son más precisos que los obtenidos por los contadores de tráfico. Por otra parte, Gurushinge et al (2002) [10] reporta los resultados de un estudio empírico realizado en un campo de sondeo, que concluye que la exactitud de los datos obtenidos a través de GPS fueron superiores a la de los mismos datos obtenidos mediante mediciones convencionales.

2.3.2 Velocidad por asignación de servicio

Uno de los procesos que realiza Transantiago con sus datos es la asignación de servicio, esto consiste en asignarle a cada bus del sistema el servicio realizado. Esto es construido a través de un algoritmo diseñado por Transantiago que utilizando la información GPS más las rutas es capaz de determinar qué servicio realizo un bus y en que periodo de tiempo. Con esta última información es sólo cuestión de calcular el cuociente entre el largo de la ruta y el tiempo en recorrerla para obtener una velocidad media del servicio.

Una de las desventajas de este método es que la velocidad no puede ser observada en detalle espacialmente a lo largo de la ruta (por tramos), ni tampoco es trivial definir en qué periodo de tiempo del día corresponde porque el inicio de un servicio puede estar en un periodo de tiempo como por ejemplo punta mañana y al terminar en otro como medio día.

2.3.3 Velocidad calculada en terreno

Otro método utilizado es subiendo medidores a bordo de los buses. Esto es caro y normalmente se toman muestras muy pequeñas (típicamente 3) y dada la alta dispersión de las velocidades resultan poco confiables. Es prácticamente imposible medir de esta forma todo el sistema a toda hora en todo lugar, además es importante hacer estas mediciones constantemente.

2.3.4 Velocidad a través de software Etraus

Etraus es un software de modelos de simulación estratégica, divide a Santiago en aproximadamente 600 zonas y es capaz de entregar velocidades pero los resultados a nivel micro no son confiables, por lo que no es utilizado para decisiones a ese nivel. Además los resultados no consideran variables como vías transversales o demanda de pasajeros que se suben en los paraderos.

2.3.5 Velocidad online

El sinóptico, nombre del sistema de visualización online de posición de buses con el que cuenta Transantiago, tiene implícitas estimaciones de velocidad pero ellas se usan por ahora sólo para visualizar en tiempo real, y la velocidad es calcula entre dos emisiones GPS, lo que no es útil para determinar velocidades medias en algún tramo específico o en algún periodo determinado.

2.4 Terminología

Antes de comenzar con la descripción de este estudio es necesario conocer un conjunto de conceptos y notaciones usadas en el área de transporte público, a continuación se explican los más relevantes.

Término	Descripción
Ruta	Conjunto ordenado de posiciones que definen un camino, en este trabajo estarán siempre sobre calles de Santiago
Servicio	Ruta que siguen un conjunto de buses, comandados por un operador
Operador	Empresa encargada de administrar y operar un conjunto de buses que realizan uno o más servicios
Tramo	Segmento o sección de una ruta, compuesto por 2 puntos sobre la ruta
Periodo	Periodo de tiempo usado para dividir un día en varias partes, durante este trabajo se usarán periodos de 30 minutos
Trayectoria	Conjunto de posiciones GPS que describen una muestra de la trayectoria seguida por un bus
Sentido	Sentido de avance sobre una ruta
Troncal	Tipo de servicio que recorre vías de gran envergadura, las principales calles de Santiago
Alimentador	Tipo de servicio que tiene por objetivo acercar a los usuarios a vías con recorridos troncales.
Punta Mañana	Periodo de tiempo en el día donde se registran grandes afluencias de usuarios del sistema en la mañana, corresponde al periodo donde las personas viajan hacia sus lugares de trabajo (7:00 AM a 9:00 AM).
Punta Tarde	Periodo de tiempo en el día donde se registran grandes afluencias de usuarios en la tarde, corresponde al periodo donde las personas vuelven a sus hogares (5:30 PM a 7:30 PM)

3. Metodología

3.1 Metodología de Trabajo

Una de las principales características que define este estudio es su metodología de desarrollo. Esta puede ser comparada con un desarrollo de software iterativo incremental, dado que cada vez que se realizaba una reunión con los expertos del área de transporte para mostrar los avances se definían nuevos requerimientos o se redefinían otros para mejorar la calidad de los cálculos o las maneras de visualización de la información.

Es importante comentar que las reuniones estuvieron compuestas en su mayoría por tres profesores del departamento de transporte y un representante de Transantiago, el cual velaba para que la información resultante fuera de utilidad para su posterior análisis de parte de los especialistas de Transantiago.

Por cada iteración, compuesta por la investigación, desarrollo y posterior aprobación de parte de los expertos, se construyó un prototipo el cual consiste en un sencillo software ejecutable desde una consola Linux, quien realiza todas las operaciones necesarias para calcular las velocidades y un archivo compatible con Google-Earth, que muestra la información necesaria para analizar las velocidades de un servicio específico.

En términos más globales podemos dividir el trabajo en tres grandes etapas, la primera constituida por el desarrollo necesario para procesar los datos originales entregados por Transantiago, la segunda compuesta por el primer prototipo del software donde se construyó gran parte de las funciones y clases que manejan todo lo que no involucra al cálculo de velocidades, por ejemplo, un módulo o clase para poder dibujar sobre Google-Earth, una clase para la transformación de coordenadas y manejo de rutas, entre otras. Y por último una tercera etapa, donde incrementalmente se modificó la metodología e implementación del cálculo de la velocidad, hasta llegar finalmente a un método satisfactorio.

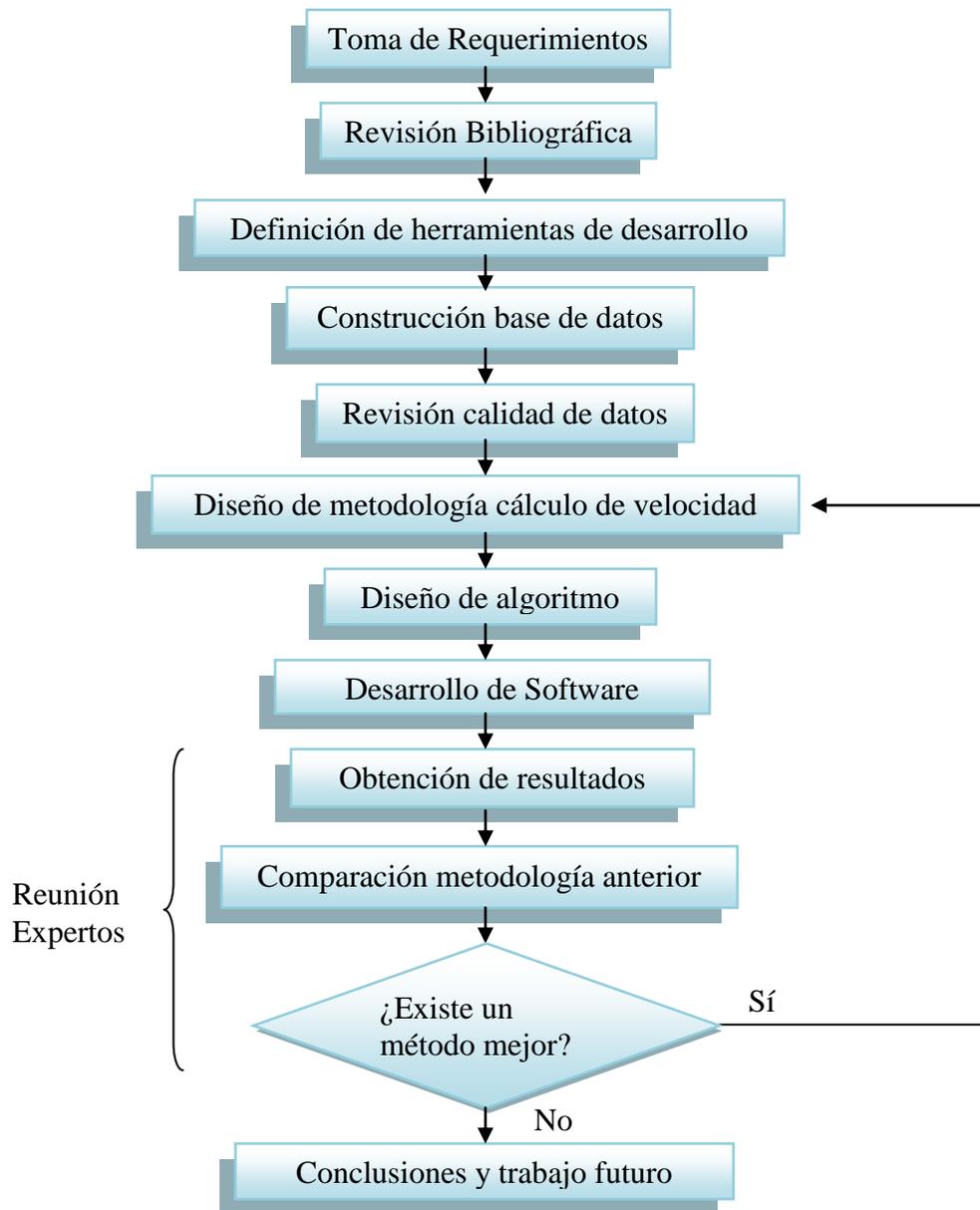


Diagrama 1: Metodología de trabajo

El Diagrama 1, describe las distintas etapas que componen la metodología de trabajo, la etapa inicial compuesta por requerimientos globales, revisión bibliográfica, herramientas de apoyo, pre proceso de datos, la etapa media donde iterativamente se fueron probando métodos específicos para el cálculo de las velocidades y la etapa final donde hasta la fecha se discuten otros elementos que pueden integrarse a esta herramienta.

3.2 Metodologías para estimar las velocidades.

En esta sección se detalla cada uno de los métodos diseñados para calcular velocidades, ordenados cronológicamente según su desarrollo. Es importante notar que la creación de cada una depende de los resultados de las anteriores.

3.2.1 Método I (Agrupación de velocidades instantáneas)

Uno de los datos que proviene del instrumental GPS de los buses es su velocidad instantánea. En la actualidad Transantiago no confía en esta información puesto se han visto muestras que parecen estar fuera de los rangos lógicos. Por otra parte, aunque estas observaciones fuesen exactas dan la impresión de ser poco representativas para establecer una velocidad media en un tramo de una ruta, hay que notar que es probable que existan muchas observaciones con velocidad cero por subidas de pasajeros o paradas por semáforos u otras en periodos de aceleración o desaceleración.

La metodología diseñada para transformar estas observaciones en una velocidad representativa de un tramo de una ruta consiste en agruparlas por cercanía, utilizando distancia euclidiana al tramo más cercano y luego promediarlas. La Figura 9 describe visualmente el cálculo de las velocidades medias a través de este método.

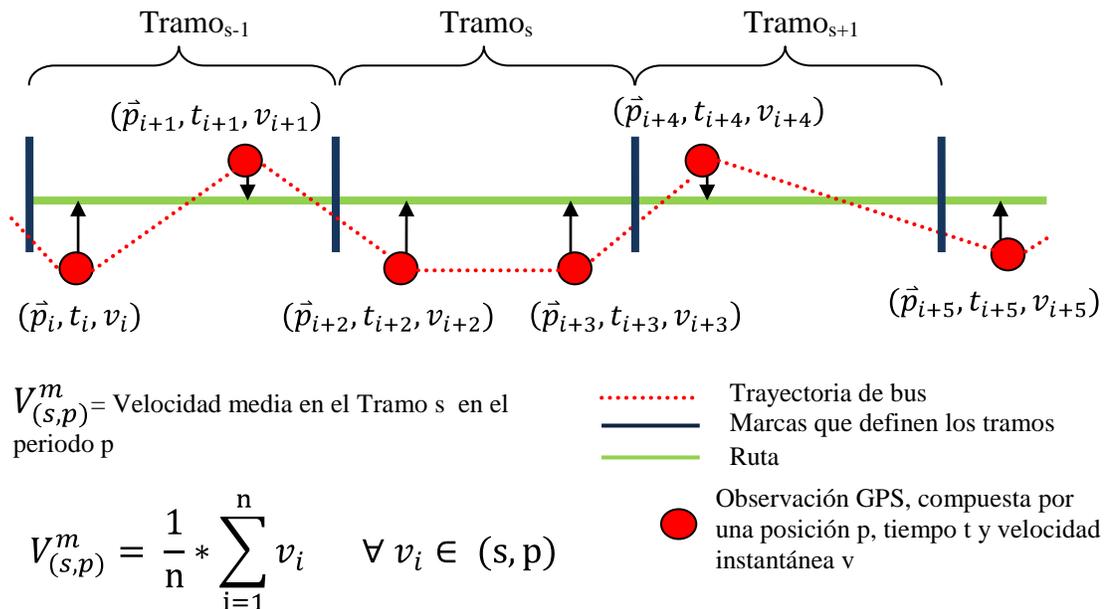


Figura 9: Método I media de velocidades instantáneas por tramo y por periodo.

3.2.2 Método II (Agrupación de velocidades calculadas)

Este método es prácticamente igual al descrito anteriormente con la única salvedad que no se utiliza la velocidad provista por el GPS sino una calculada con la distancia y tiempo entre registros del bus, la Figura 10 muestra como se aplica este cálculo.

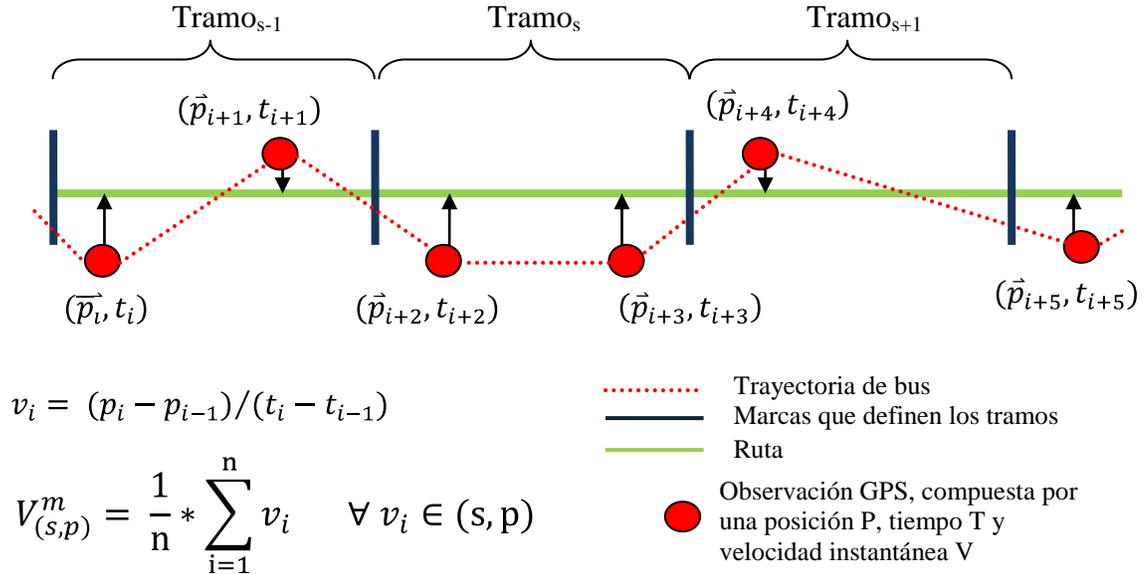
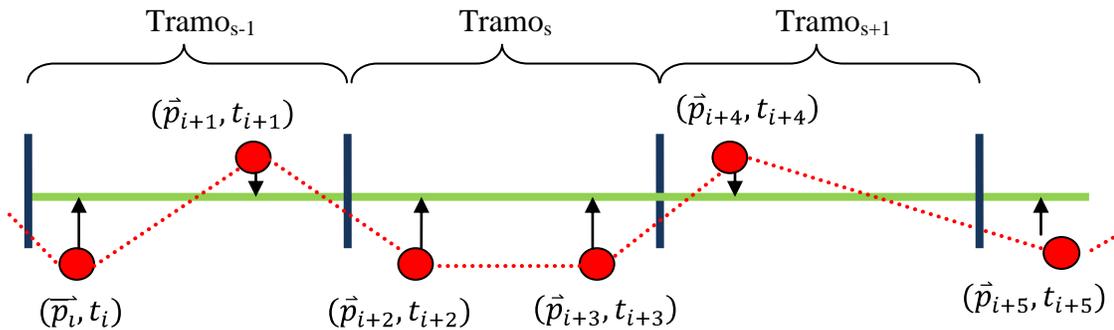


Figura 10: Método II, media de velocidades calculadas con dos pulsos.

3.2.3 Método III (Agrupación de velocidades calculadas variación método II)

Este método utiliza el mismo sistema de asociación de observaciones a los tramos con la diferencia que la velocidad en cada observación se calcula como el promedio entre las velocidades obtenidas con las observaciones anterior y posterior, para explicar mejor este cálculo la Figura 11 lo detalla visualmente.



$$v_i = \frac{(p_i - p_{i-1}) / (t_i - t_{i-1}) + (p_{i+1} - p_i) / (t_{i+1} - t_i)}{2}$$

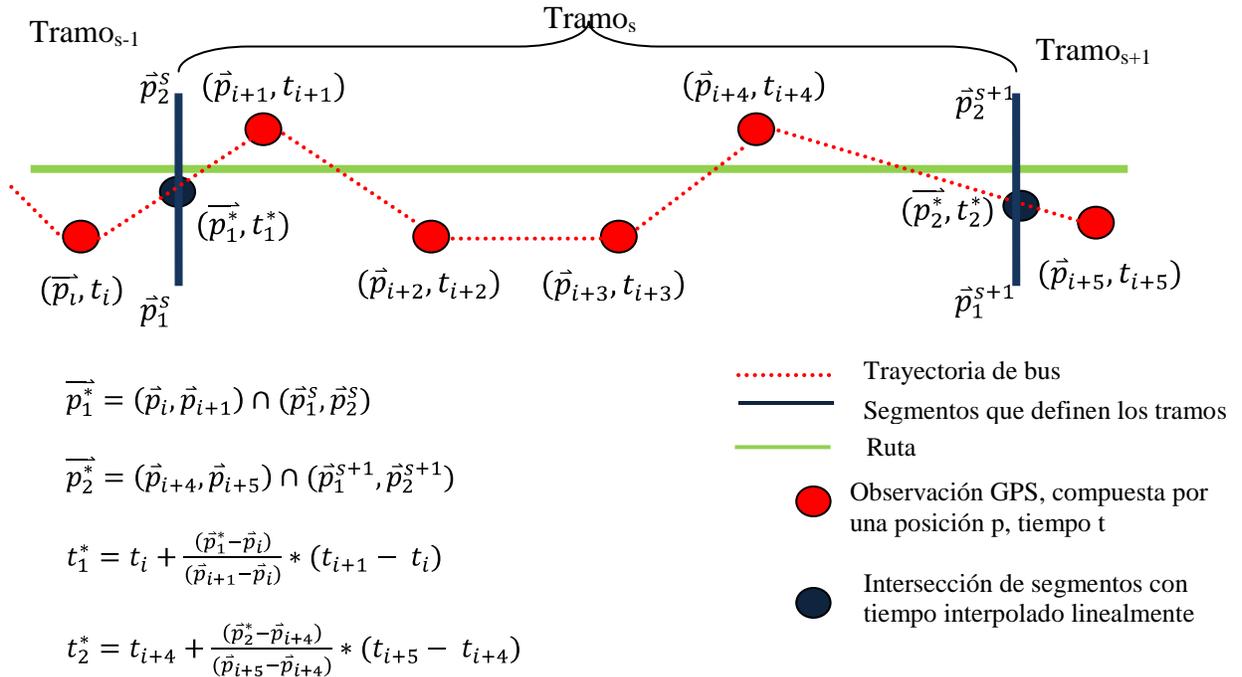
$$V_{(s,p)}^m = \frac{1}{n} * \sum_{i=1}^n v_i \quad \forall v_i \in (s, p)$$

Figura 11: Método III

3.2.4 Método IV (Interpolación temporal inicio-fin de tramo)

Esta metodología consiste en calcular las velocidades estimando los tiempos de entrada y salida de un bus sobre los tramos de la ruta, de esta manera la velocidad queda como el cociente entre la longitud del tramo sobre la diferencia de estos tiempos estimados.

Teniendo en cuenta que el tiempo exacto en que un bus pasa por el inicio o final de un tramo no es un dato con el que se cuenta, se estimará interpolando linealmente entre los tiempos de las observaciones anterior y posterior a un inicio de tramo y de la misma forma en el final del tramo. La Figura 12 muestra visualmente este procedimiento:



$$\vec{p}_1^* = (\vec{p}_i, \vec{p}_{i+1}) \cap (\vec{p}_1^s, \vec{p}_2^s)$$

$$\vec{p}_2^* = (\vec{p}_{i+4}, \vec{p}_{i+5}) \cap (\vec{p}_1^{s+1}, \vec{p}_2^{s+1})$$

$$t_1^* = t_i + \frac{(\vec{p}_1^* - \vec{p}_i)}{(\vec{p}_{i+1} - \vec{p}_i)} * (t_{i+1} - t_i)$$

$$t_2^* = t_{i+4} + \frac{(\vec{p}_2^* - \vec{p}_{i+4})}{(\vec{p}_{i+5} - \vec{p}_{i+4})} * (t_{i+5} - t_{i+4})$$

$V_{(s,p)}^b =$ Velocidad del bus b en el periodo p y tramo s

$$V_{(s,p)}^b = \frac{\|Tramo_s\|}{(t_2^* - t_1^*)} \rightarrow V_{(s,p)}^m = \frac{1}{n_b} * \sum_b V_{(s,p)}^b \quad \forall b / \exists (\vec{p}_1^* \wedge \vec{p}_2^*)$$

Figura 12: Método IV para un Bus

El punto donde se estima el tiempo de entrada al tramo resulta de intersectar el segmento normal a la ruta con el segmento de la trayectoria del bus, luego con este punto se obtienen las proporciones de distancias y con estas finalmente se estima el tiempo en ese punto.

En este método es necesario resolver dos problemas, el primero es producido en las curvas y el segundo es definir las implicancias de la relación de tamaño entre los segmentos de las trayectorias del bus y los tramos de la ruta. La Figura 13 muestra un ejemplo de los problemas en las curvas y la Figura 14 lo que se debe tener en consideración con respecto a los tamaños.

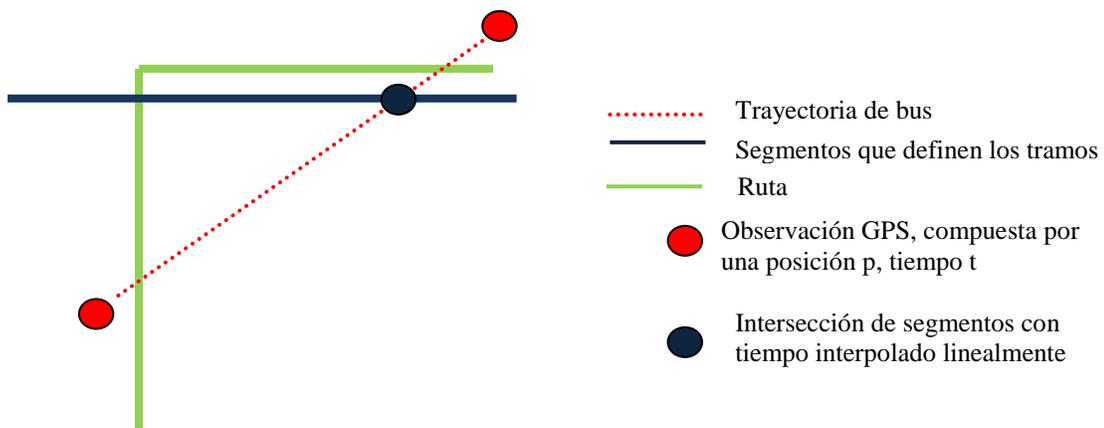


Figura 13: Error de estimación por curvas.

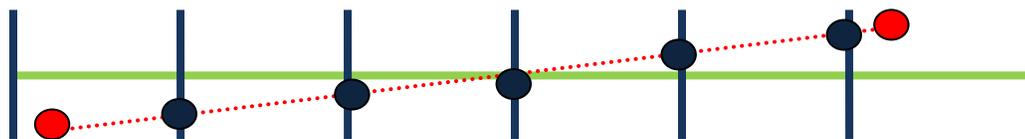
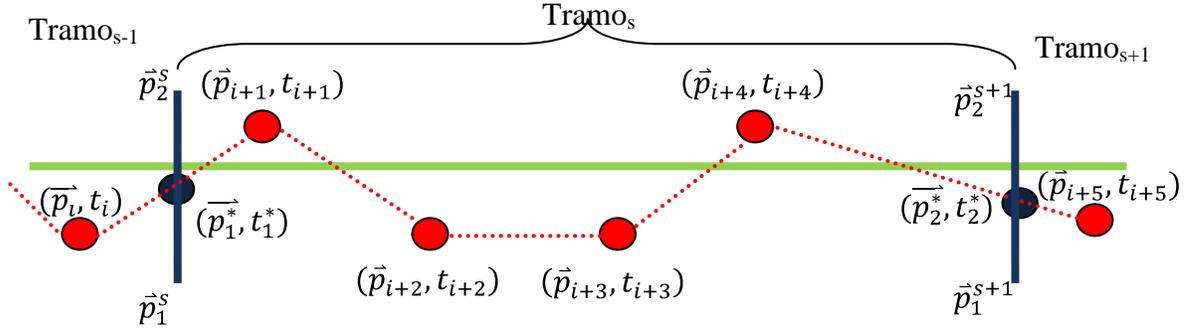


Figura 14: Error de estimación por escala.

3.2.5 Método V (1era variación método IV)

Esta variación del método IV consiste en no interpolar los tiempos, sino que utiliza los tiempos obtenidos de la observación GPS, tiempo antes de entrar al tramo y tiempo después de salir del tramo, la idea de hacer esto es poder comparar con el método IV y estudiar sus diferencias. La Figura 15 muestra un ejemplo de esto.



$V_{(s,p)}^b =$ Velocidad del bus b en el periodo p y tramo s

$$V_{(s,p)}^b = \frac{\|Tramo_s\|}{t_{i+5} - t_i}$$

$$V_{(s,p)}^m = \frac{1}{n_b} * \sum_b V_{(s,p)}^b \quad \forall b / \exists (\vec{p}_1^* \wedge \vec{p}_2^*)$$

- ⋯ Trayectoria de bus
- Segmentos que definen los tramos
- Ruta
- Observación GPS, compuesta por una posición p , tiempo t
- Intersección de segmentos con tiempo interpolado linealmente

Figura 15: Cálculo de las velocidades para el método V

3.2.6 Método VI (2da variación método IV)

Para poder eliminar el problema de las curvas o cualquier otro problema generado por geometría este método propone proyectar las observaciones GPS a la ruta de tal manera que se eliminan las intersecciones y se reemplazan por el concepto de distancia sobre la ruta, en otras palabras, se considera que un bus entra o sale de un tramo en la ruta cuando los límites del tramo están entre estas observaciones proyectadas. La Figura 16 a continuación ejemplifica mejor esta metodología.

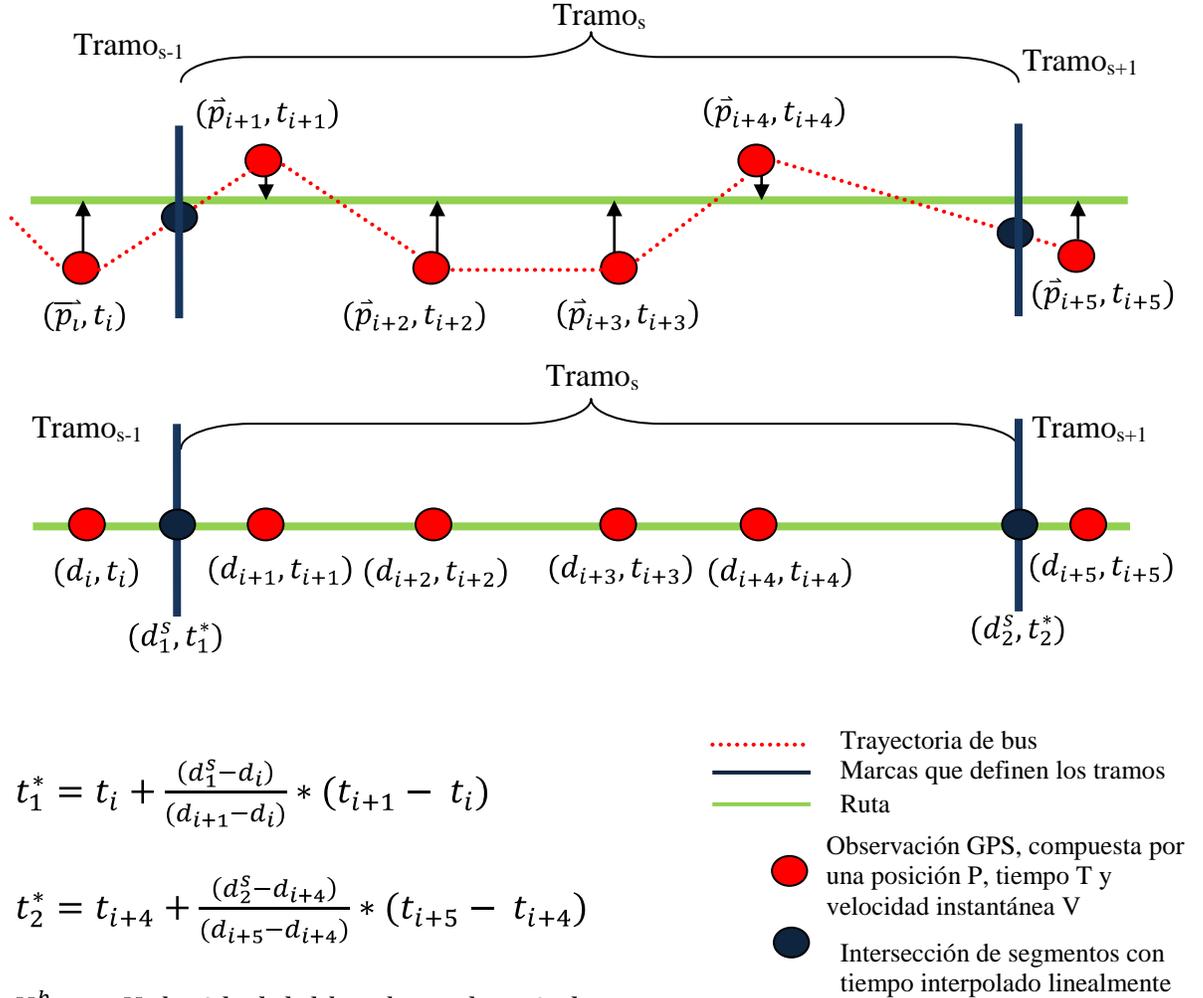


Figura 16: Datos antes de proyectarse (diagrama superior), datos proyectados (diagrama inferior).

3.2.7 Método VII

El ya descrito método VI presenta un problema de pérdida de información entre periodos. Esto porque las trayectorias de buses son extraídas por bloque o periodo de tiempo lo significa que si un bus no alcanza a cruzar un tramo antes que el periodo se acabe ese tramo no utiliza la información.

Por esta razón se propone un nuevo método que utiliza el mismo concepto de interpolación de tiempos en la entrada y salida de los tramos, pero agrega una nueva interpolación de posición por tiempo para el caso de trayectorias de buses que quedan entre periodos. Además no se promediarán las velocidades sino que se sumarán por separados los tiempos y distancias que recorre cada bus en cada periodo y tramo. Calculándose finalmente la velocidad comercial representativa como el cociente entre la sumatoria de distancias y la sumatoria de tiempos.

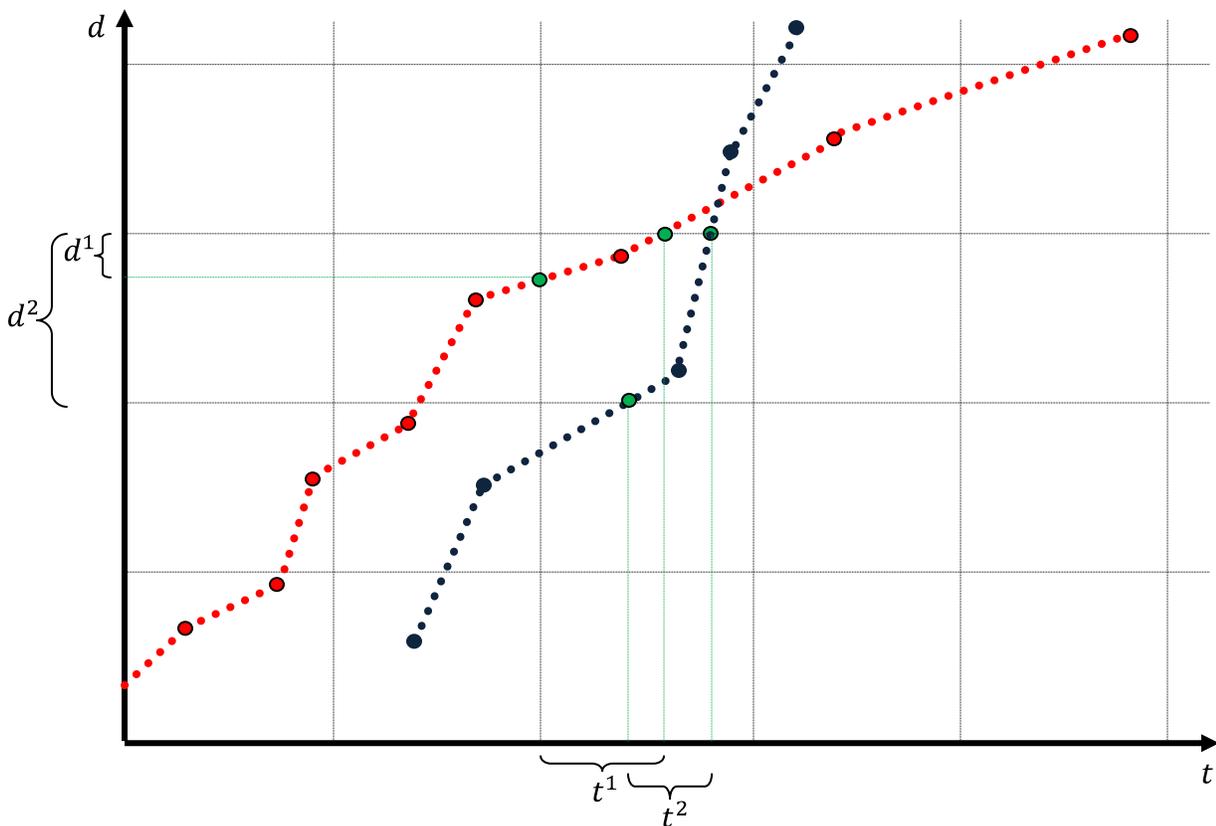


Figura 17: Esquema método VII

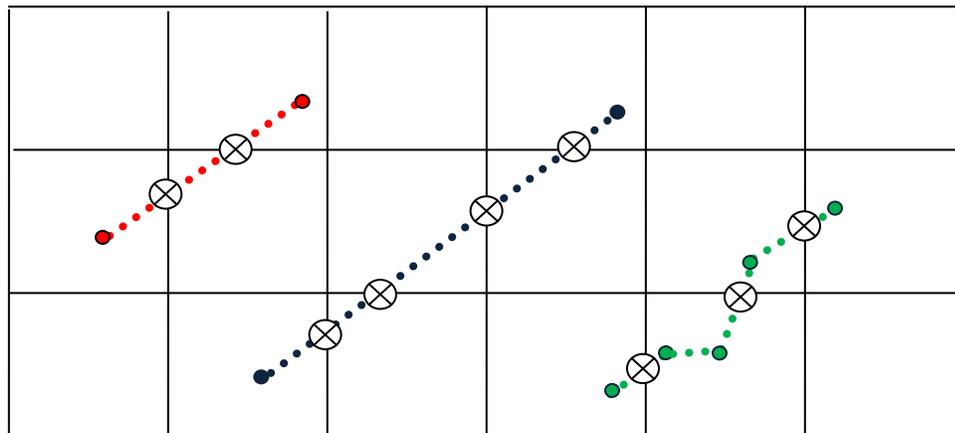
La Figura 17 muestra un ejemplo con dos trayectorias de buses sobre una grilla distancia-tiempo. Las trayectorias de los buses, azul y roja, son proyectadas sobre la ruta (distancia/tiempo). Los puntos generados por las intersecciones entre las trayectorias y la grilla definen los tiempos y distancias de entrada y salida de un tramo o de un periodo del día.

La Ecuación 1 describe finalmente como es realizado el cálculo donde la velocidad representativa $V_{(s,p)}^m$ es el cociente entre la sumatoria de todas las distancias d de los buses b en el tramo s en el periodo p y la sumatoria de los tiempos t de los mismos buses en el mismo tramo s y periodo p .

$$V_{(s,p)}^m = \sum_b d_{(s,p)}^b / \sum_b t_{(s,p)}^b$$

Ecuación 1: Velocidad comercial representativa de un tramo-periodo.

Para la determinación de los tiempos y distancias existen varios casos desde el punto de vista geométrico que fueron detectados al testear el método y modificaron el algoritmo original para una implementación más robusta. La Figura 18 muestra estos casos, que consisten en intersecciones de un sólo segmento de trayectoria (formado por dos pulsos GPS) con varias intersecciones, esto implica que es necesario detectar esta situación y partir el segmento para poder identificar el tiempo y distancia que corresponde para cada cuadro de la grilla, casos de color rojo y azul, por otra parte el caso de color verde muestra el comportamiento del 95% de los datos donde una intersección corresponde a un segmento.



⊗ Intersección trayectoria inicio-fin tramo-periodo

Figura 18: Casos especiales método VII

Además en una escala despreciable están los casos donde el pulso está sobre una línea de tiempo o división de tramo o dos o más pulsos en la misma posición sobre una línea de la grilla, pero el método también los considera.

4. Diseño e Implementación

En este capítulo se detalla el proceso de diseño e implementación del software del presente trabajo, todo lo involucrado en esta sección es posible dividirlo en cuatro partes importantes.

1. Herramientas de Apoyo
2. Base de Datos
3. Software de estimación de velocidad
4. Interfaces

Cada una de estas partes, a excepción de la primera, se componen a su vez de dos partes una de diseño y otra de desarrollo explicando con precisión cada proceso de construcción del módulo en cuestión.

4.1 Herramientas de apoyo

Una de las características importantes de este trabajo es la utilización de herramientas de fácil gestión, esto se tradujo en la elección de herramientas de código abierto (open-source¹) que no requirieron gestiones de largo plazo para obtenerlos, ni costo para su obtención, de tal manera de acelerar el proceso de investigación y desarrollo. Esto por la sencilla razón que cualquier herramienta que implique algún costo debe pasar por un lento proceso de compra. Además se eligió crear una plataforma completamente gratuita para facilitar su futura implementación y puesta en marcha.

Como sistema operativo se escogió Linux Ubuntu 8.10 y 9.04, esto por mi experiencia en esta distribución y conocimiento de diversas herramientas de desarrollo que he utilizado por alrededor de 4 años en problemas de similares características, manejo de datos masivos, cálculo intensivo y visualización.

Para el procesamiento de datos en bruto enviados por Transantiago se utilizaron las siguientes herramientas *awk*, *sed*, *grep*, estos comandos vienen integrados en la consola de cualquiera distribución de Linux. Con ellos se pueden crear script que procesan texto, modifican, pegan y cortan archivos, entre otros. A continuación una pequeña reseña de cada uno:

¹ http://es.wikipedia.org/wiki/Código_abierto

Awk²: Es un lenguaje de programación diseñado para procesar datos basados en texto, ya sean ficheros o flujos de dato. Usa ampliamente el tipo de datos de listas asociativas.

Sed³: Es un editor de texto, que parsea y edita archivos a través de su lenguaje de programación, el programa lee línea a línea un archivo aplicando las operaciones que se le hayan especificado.

Grep⁴: Comando utilizando para hacer búsquedas de patrones de caracteres sobre archivos de texto.

El lenguaje de programación utilizado para el desarrollo del software fue C++, elegido por ser un lenguaje de programación versátil que brinda la posibilidad de desarrollar un software con cualquier característica deseada y compatible con innumerables librerías científicas.

Como IDE para trabajar sobre los diversos lenguajes de programación se utilizó Kdevelop3, el cual provee diversas vistas para distintos lenguajes, para el caso de este estudio se utilizó C++ y Scripting. La elección de este IDE respecto a otros no requiere razones técnicas puesto que sólo se trabajó con lenguajes comúnmente utilizados y no existieron otro tipo de requerimientos que necesitarán algo más del IDE.

Para el almacenamiento de los datos de Transantiago se decidió utilizar PostgreSQL8.3, en conjunto con el software de administración pgAdminIII para visualizar la estructura de la base de datos y testear consultas antes de integrarlas al software. Para la comunicación con la base de datos desde C++ se utilizó la librería libpqxx, con el objetivo de simplificar la codificación, dado que utilizar las librerías de PostgreSQL directamente hubiese implicado mayor desarrollo y tiempo.

Para la generación de gráficos automáticos se utilizó Gnuplot 4.2, esto por mi experiencia utilizándolo en proyectos anteriores lo cual se tradujo en una rápida codificación. Además se utilizó una librería comercial en C++ (ChartDirector) para realizar cierto tipo de gráficos pero sólo se usó en modo de prueba, es decir, modo sin pago y lo más probable es que sea reemplazado por Gnuplot 4.2 en el futuro.

Finalmente para la interacción y visualización global de los resultados se optó por usar Google-Earth dado que provee una API de dibujo sencilla y completa, y principalmente porque posibilita visualizar geográficamente teniendo el mapa de Santiago sin costo y en cualquier plataforma, característica muy importante a la hora de revisar resultados para Transantiago.

² <http://es.wikipedia.org/wiki/AWK>

³ <http://en.wikipedia.org/wiki/Sed>

⁴ <http://en.wikipedia.org/wiki/Grep>

Durante el desarrollo también se utilizaron herramientas de ofimática como por ejemplo OpenOffice 3.0 para observar tablas de resultados y gráficos rápidos, y finalmente Office 2007 para la construcción del presente informe.

La Figura 19 muestra un resumen de las herramientas utilizadas en este trabajo.

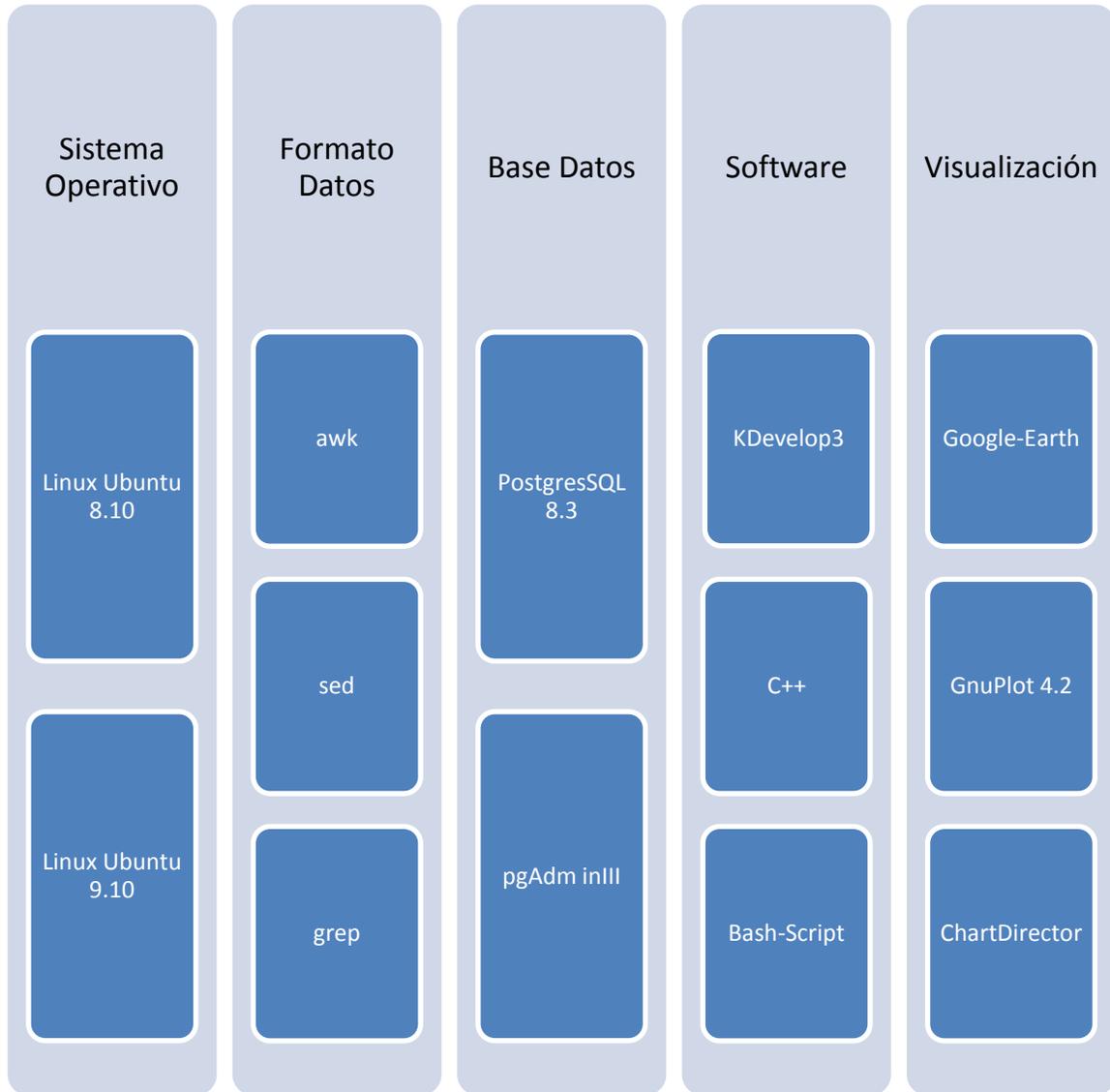


Figura 19: Herramientas de apoyo

4.2 Base de Datos

En esta sección se describen cada uno de los pasos y procesos realizados para lograr construir una base de datos con la información necesaria para la estimación de las velocidades a partir de los datos originales entregados por Transantiago.

En resumen el proceso global realizado en esta etapa consistió en transformar el formato original de los datos para poder cargarlos como tablas en PostgreSQL y finalmente enriquecerlos con algunos campos extras necesarios para etapas posteriores a través de programas construidos en C++.

4.2.1 Pre-procesamiento de datos

La primera etapa de este trabajo consistió en investigar y estudiar los datos que proveyó Transantiago, estos son formados por un conjunto de archivos con distintos formatos que describen una semana de operación de todo el sistema, la primera de Septiembre del 2008. Para este trabajo en particular sólo participan tres tipos de tablas de datos, descritos a continuación:

Tipo de Dato	Descripción
Posicionamiento	Posición de los buses
Asignación de Servicio	Servicio que realizan los buses en cierto periodo de tiempo
Rutas	Caminos georeferenciados que recorre cada servicio

Otros tipos de datos que existen, pero que no participaron en este estudio, son por ejemplo, las transacciones bips de los usuarios, los paraderos de buses, las estaciones de metro, zonas pagas...entre otros.

El objetivo de esta etapa es lograr llevar todos estos datos repartidos en variados archivos y formatos a una única base de datos, de tal forma de facilitar el posterior trabajo con ellos. A continuación se realiza una descripción en mayor detalle de los datos y de su posterior procesamiento para lograr llevarlos a una base de datos.

Cabe notar que estos procesos iniciales sufrieron modificaciones a lo largo del trabajo a medida que aparecieron nuevos requerimientos, en este capítulo se describirá la versión final y se explicará brevemente el por qué de algunos detalles que pudiesen parecer innecesarios.

4.2.1.1 Posicionamiento

Los datos de las posiciones de los buses fueron entregados en 14 archivos separados por día, AM y PM. Cada registro en estos archivos está compuesto por los siguientes campos:

1. **Unidad de negocio:** Nombre del operador del bus
2. **Patente:** Patente del bus a quien corresponden los datos
3. **Ignición:** Variable (SI/NO) que corresponde a si el bus tenía el motor apagado o no
4. **Tiempo GPS:** Fecha y hora de la toma del registro
5. **Latitud:** Coordenada correspondiente a la latitud del bus en ese instante
6. **Longitud:** Coordenada correspondiente a la longitud del bus en ese instante
7. **Velocidad GPS:** Velocidad instantánea del bus en ese instante de tiempo

En una primera reunión Transantiago explicó el significado de cada uno de estos campos y se definieron algunos detalles como la resta de 4 horas al tiempo para sincronizar con la hora de Santiago y el tipo de datum⁵ para las coordenadas geográficas WGS84, zona 19H, esto último refiere al estándar utilizado para medir latitud y longitud. En la actualidad existen distintos estándares.

Teniendo en cuenta todo lo anterior se dio paso al primer procesamiento de los archivos concatenándolos en uno para simplificar su carga a una sencilla tabla en una base de datos creada en PostgreSQL. Al intentar cargar este archivo en la base de datos se encontraron variados problemas de formato, a continuación se describen las transformaciones necesarias para lograr cargar con éxito los datos.

1. Eliminación de los encabezados
2. Formato de fecha: 01/09/2008 12:00:00.000000 → 2008-09-01 12:00:00
3. Los campos que son SI o NO fueron cambiados por 1 y 0 respectivamente
4. Se cambiaron las comas por puntos como separador de decimales
5. Se transformó el encoding⁶ del archivo a UTF-8.

La Figura 20 muestra los primeros registros después del proceso de cambio de formato, el cual tomó 50 minutos para procesar los 80 millones de registros. Todo este proceso fue realizado en un bash-script en Linux utilizando el comando *sed* para los cambios de formato y el comando *iconv* para el cambio de encoding. El código fuente se encuentra en el anexo de este informe.

⁵ <http://es.wikipedia.org/wiki/Datum>

⁶ http://en.wikipedia.org/wiki/Character_encoding

	concesionario character varyi	patente characte	mtc integer	tipogp charac	tiempogps timestamp without	latitud double precision	longitud double precision	ignicion characte	velocidadgps double precisi
1	Transantiago	AA-0016	6709		2008-09-03 05:55:58	-33.5886306762695	-70.6734008789063	Si	5.9264
2	Transantiago	AA-0016	6709		2008-09-03 05:59:29	-33.5889205932617	-70.6731185913086	Si	6.8524
3	Transantiago	AA-0016	6709		2008-09-03 05:59:59	-33.5889587402344	-70.6734848022461	Si	7.9636
4	Transantiago	AA-0016	6709		2008-09-03 06:00:29	-33.5888175964355	-70.6742477416992	Si	8.8896
5	Transantiago	AA-0016	6709		2008-09-03 06:00:59	-33.5886535644531	-70.6750869750977	Si	9.8156
6	Transantiago	AA-0016	6709		2008-09-03 06:01:29	-33.588451385498	-70.6760101318359	Si	9.8156
7	Transantiago	AA-0016	6709		2008-09-03 06:01:59	-33.5882606506348	-70.6769027709961	Si	9.8156
8	Transantiago	AA-0016	6709		2008-09-03 06:02:29	-33.5880889892578	-70.6777877807617	Si	4.8152

Figura 20: Datos originales en base de datos.

Una vez procesados los datos, se procedió a cargarlos en una base de datos. Para esto se diseñó, creó y cargó una tabla con los datos desde el archivo procesado. A continuación se muestra un diagrama de la tabla.

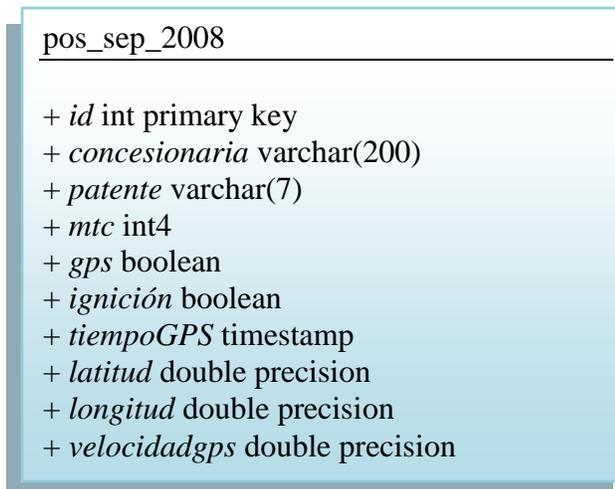


Diagrama 2: Diseño tabla de posicionamiento.

El script de creación de la tabla, cargado de datos y posterior orden (índices) se encuentra en los anexos de este informe crea_posicionamiento_sep_2008.sql. El Diagrama 3 resume el proceso descrito.



Diagrama 3: Flujo preproceso posicionamiento

4.2.1.2 Asignación de Servicio

La asignación de servicio es la información sobre qué servicio o ruta está realizando cada uno de los buses en un cierto periodo de tiempo. Estos datos provienen de un proceso realizado internamente por Transantiago, a través de un algoritmo. El resultado es agrupado en archivos Excel por cada día de la semana.

El primer paso para llevar estos datos a la base de datos, fue manualmente cambiar el formato de archivo .xls a .csv utilizando las opciones de Excel 2007. Luego aplicar diversos cambios de formatos a las fechas y números descritos a continuación:

1. Eliminación de los encabezados
2. Formato de fecha: 01/09/2008 12:00:00.000000 → 2008-09-01 12:00:00
3. Fusión de archivos diarios
4. Se cambiaron las comas por puntos como separador de decimales
5. Integración de un nuevo campo que diferencia alimentadores de troncales
6. Se transforme el encoding⁷ del archivo a UTF-8.

La Figura 21 muestra algunos registros para ejemplificar estos datos. El proceso de transformaciones descritas anteriormente fue llevado a cabo utilizando las mismas herramientas ya descritas en la sección de posicionamiento.

	tipo_bus character	un character	operador character	servicio character	patente character	sentido character	inicio time with	intervalo time with	control time with	fin time with	distancia double pr	tiempo time with	velocidad double pr	capacidad smallint
1	alimentado	A3	Union de Tr	E17	ZW-2024	Ida	23:02:23	00:25:29	23:15:52	23:24:53	7690	00:22:30	20.5	60
2	alimentado	A4	STP Santiag	F01	UY-3169	Regreso	19:26:22	00:03:07	19:36:52	19:47:52	7680	00:21:30	21.4	77
3	troncal	T2	Subus Chile	206	SA-6080	Ida	06:57:44	00:00:25	07:15:13	07:40:14	20330	00:42:30	28.7	77
4	troncal	T4	Express de	401c	TL-6865	Ida	06:37:39	00:02:32	07:18:40	07:28:10	22000	00:50:31	26.1	77
5	alimentado	A6	Comercial M	103	TJ-9685	Ida	17:47:26	00:21:49	18:27:56	18:59:55	20990	01:12:29	17.4	77
6	troncal	T2	Subus Chile	225	BJFD-89	Ida	12:41:54	00:04:43	13:28:54	13:53:54	27710	01:12:00	23.1	91
7	troncal	T2	Subus Chile	227	BJFG-84	Ida	20:05:02	00:05:07	20:55:02	21:06:31	20390	01:01:29	19.9	91

estado character	fecha date	día smallint	tipo_día character	periodo character	cada_5_m time with	cada_30_m time with
OK	2008-09-03	3	Laboral	12 - Pre Noc	23:00:00	23:00:00
OK	2008-09-03	3	Laboral	09 - Punta T	19:25:00	19:00:00
OK	2008-09-01	1	Laboral	04 - Punta T	06:55:00	06:30:00
OK	2008-09-04	4	Laboral	04 - Punta T	06:35:00	06:30:00
OK	2008-09-03	3	Laboral	09 - Punta T	17:45:00	17:30:00
OK	2008-09-05	5	Laboral	07 - Punta T	12:40:00	12:30:00
OK	2008-09-05	5	Laboral	09 - Punta T	20:05:00	20:00:00

Figura 21: Tabla original de asignación de servicio

⁷ http://en.wikipedia.org/wiki/Character_encoding

Dos campos extras fueron agregados a esta tabla que son *fecha_inicio* y *fecha_fin* formados por la composición del campo *fecha* más *inicio* y *fin*. El objetivo de estos dos campos es estandarizar la comparación entre tiempos con otras tablas. Para crear estos dos campos se utilizaron sólo sentencias SQL y se muestran a continuación:

```
ALTER TABLE as_sep_2008 ADD COLUMN fecha_inicio timestamp;  
ALTER TABLE as_sep_2008 ADD COLUMN fecha_fin timestamp;
```

```
UPDATE as_sep_2008 SET fecha_inicio = fecha + inicio;  
UPDATE as_sep_2008 SET fecha_fin = fecha + fin;
```

El Diagrama 4 muestra la tabla creada en la base de datos para almacenar estos datos y el Diagrama 5 muestra visualmente los pasos realizados para construir esta tabla.

as_sep_2008
+ <i>id</i> int primary key
+ <i>tipo_bus</i> varchar(100)
+ <i>un</i> varchar(50)
+ <i>operador</i> varchar(200)
+ <i>servicio</i> varchar(50)
+ <i>patente</i> varchar(10)
+ <i>sentido</i> varchar(10)
+ <i>inicio</i> time
+ <i>interval_salida</i> time
+ <i>control</i> time
+ <i>fin</i> time,
+ <i>distancia</i> double precision
+ <i>tiempo</i> time
+ <i>velocidad</i> double precision
+ <i>capacidad</i> smallint
+ <i>estado</i> varchar(10)
+ <i>fecha</i> date
+ <i>dia</i> smallint
+ <i>tipo_dia</i> varchar(100)
+ <i>periodo</i> varchar(200)
+ <i>cada_5_min</i> time,
+ <i>cada_30_min</i> time
+ <i>fecha_inicio</i> timestamp
+ <i>fecha_fin</i> timestamp

Diagrama 4: Diseño tabla que almacena los datos de asignación de servicio.



Diagrama 5: Proceso asignación de servicio.

El script de creación de la tabla, cargado de datos y posterior orden (índices) se encuentra en los anexos de este informe (crea_as_sep_2008.sql).

4.2.1.3 Rutas

Las rutas son los caminos que definen a cada servicio. Esta información proviene del diseño hecho por Transantiago utilizando herramientas tipo CAD, de tal forma que para llevar estos datos a una tabla en la base de datos se realizó un proceso de varias etapas, tanto automáticas como manuales.

En esta etapa de pre-procesamiento de datos, las rutas son utilizadas para calcular la distancia de los buses a estas, con el fin de determinar buses que están fuera de ruta, y además obtener el avance lineal de cada posición GPS. Estos dos datos son de vital importancia para el software de velocidades pero su utilización quedará mejor explicada en los capítulos siguientes.

Transantiago proporcionó esta información a través de archivos construidos en Transcad. Una de las primeras labores de esta investigación consistió en llevar esta información a una tabla en la base de datos.

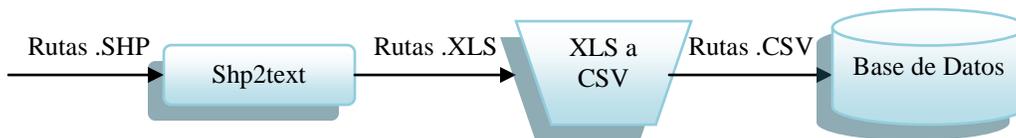


Diagrama 6: Proceso seguido para llevar los archivos shapes de las rutas a la base de datos

Para crear la tabla de rutas se utilizó el procedimiento descrito en el Diagrama 6. Donde Shp2text⁸ es un software que transforma archivos de formato .shp a archivos Excel. Este software fue sugerido por Transantiago y fue ejecutado en ambiente Windows dado que no pudo ser compilado en Linux Ubuntu.

⁸ <http://www.obviously.com/gis/shp2text/>

El archivo excel de salida fue transformado manualmente a .csv utilizando la opción de “guardar como” de Excel. Se observó la necesidad de insertar un nuevo campo que represente el orden original de los nodos de las rutas, el cual se llamó correlativo.

Los datos que describen las rutas siguen el formato de una poli línea, es decir, un conjunto de nodos coordinados (Latitud-Longitud) ordenados según el sentido del camino, además existe un campo que tiene el nombre del servicio y otro con el tipo de servicio (Local-Troncal). El Diagrama 7 muestra la estructura de la tabla final de las rutas en la base de datos.

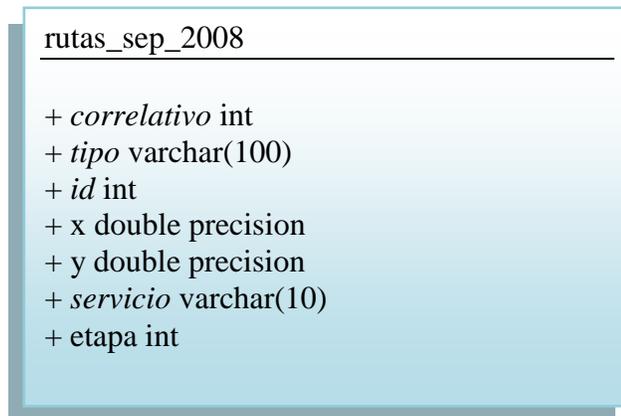


Diagrama 7: Tabla de rutas

Finalmente se creó una tabla en la base de datos llamada rutas donde se insertaron todos los registros del archivo csv. El script que crea la tabla, carga e indexa los datos está en el archivo crea_rutas.sql en los anexos de este informe. Una vez que los datos fueron ingresados se indexó la tabla con el objetivo de mejorar el desempeño en consultas que requieran una ruta de un servicio específico. La

Figura 22 muestra un extracto de los datos en Excel y el resultado en la base de datos.

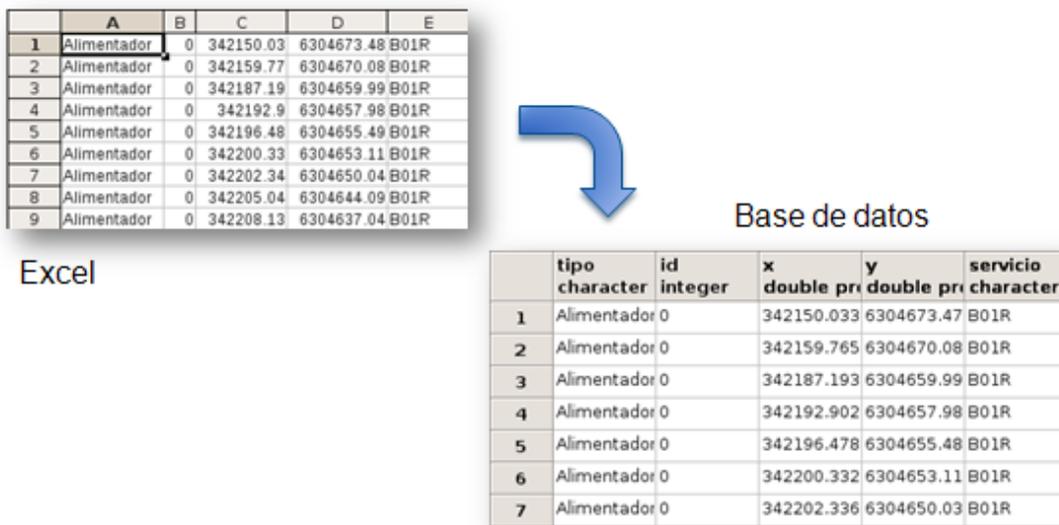


Figura 22: Ruta Excel (izquierda-superior) y tabla base de datos (derecha-inferior)



Diagrama 8: Proceso para rutas originales

4.2.2 Software para enriquecer el Posicionamiento GPS

Una vez almacenados los datos en sus respectivas tablas, queda un último procesamiento antes de calcular las velocidades. Este último proceso consiste en relacionar toda esta información en una única tabla de posicionamiento y agregar campos extras como coordenadas en metros, diferencias de tiempos y distancias entre registros, información con respecto a la ruta, entre otros.

Para realizar lo descrito en el párrafo anterior se construyó un software en C++ que hace todos estos cálculos y relaciones, generando como salida una nueva tabla que contiene los campos de la anterior tabla de posicionamiento más los nuevos campos descritos.

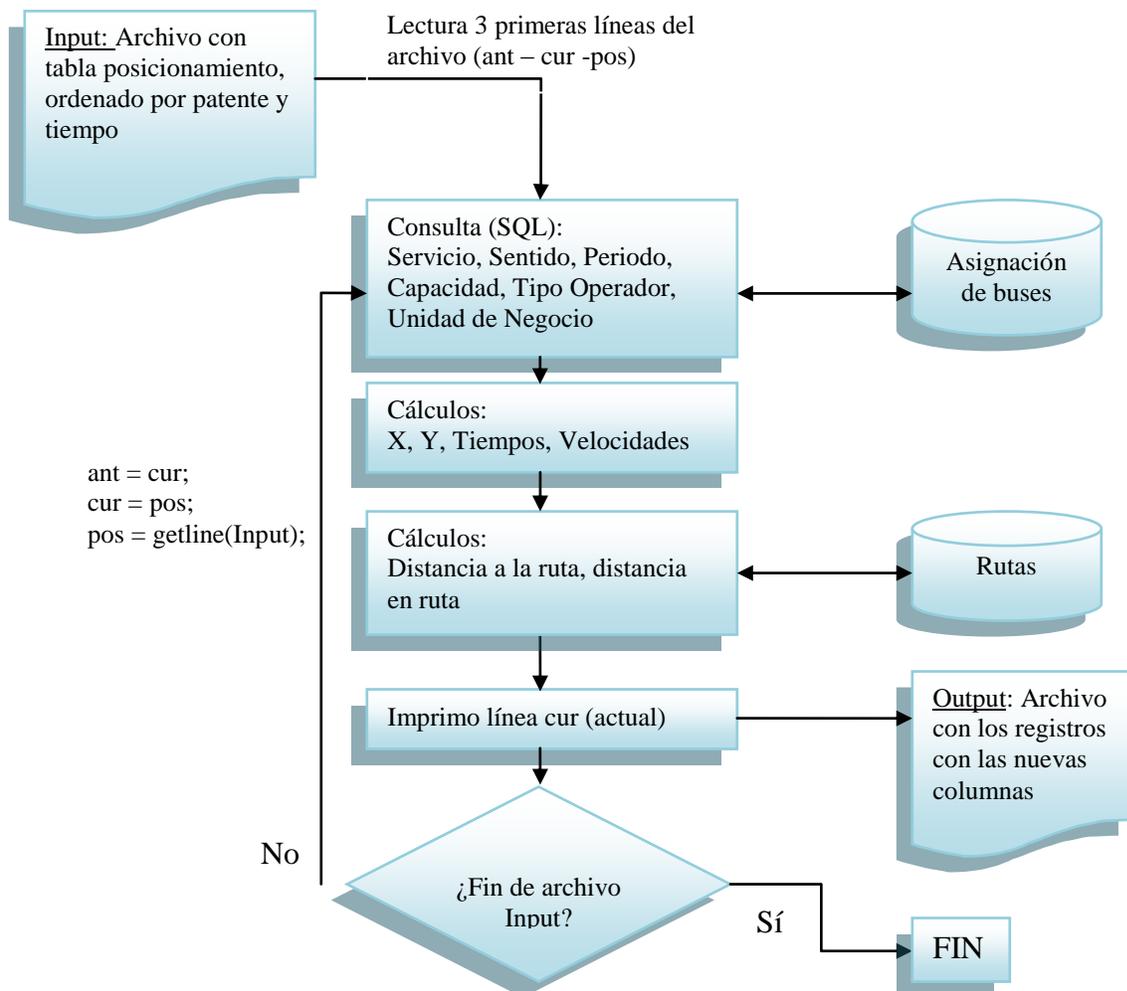


Diagrama 9: Flujo del software procesaPosicionamiento

El Diagrama 9 muestra el flujo del funcionamiento global del programa, el que principalmente recorre la tabla original de posicionamiento registro a registro, manteniendo siempre tres registros en memoria (anterior,actual,posterior), de tal forma de permitir los cálculos de distancia y tiempo del registro actual con los demás. Además, el programa consulta por cada registro el servicio que tiene asignado en la base de datos, transforma las coordenadas, calcula las distancias hacia la ruta y la distancia en la ruta.

La estructura general del software está compuesta por distintas clases que manejan cada uno de los ítems ya descritos, por ejemplo, la clase ruta tiene métodos para cargar las rutas desde la base de datos y métodos que toman como entrada un par coordenado y entrega como salida la distancia a la ruta o en la ruta. Cada uno de estos ítems es reutilizado en el software de velocidades por lo que una mejor descripción de ellos será expuesta más adelante en este informe.

En la actualidad el software que realiza estos cálculos consiste en un comando ejecutable desde un terminal UNIX y recibe como parámetros el nombre del archivo con los datos de entrada (tabla posicionamiento original ordenado por patente y tiempo) y el nombre del archivo de salida donde quedará la tabla con los nuevos campos. Ej: `>./procesapositionamiento archivo.in archivo.out`

Alrededor de 12 horas es el tiempo que demora el programa en procesar 80 millones de registros, es importante notar lo imprescindible de los índices en las tablas de *asignación de servicio* como la tabla de *rutas* para que los tiempos se mantengan razonables.

Uno de los mayores problemas en esta etapa de desarrollo fue la implementación de las transformaciones de coordenadas, latitud-longitud => UTM (metros). Para esto se probaron tres implementaciones disponibles en la web, de las cuales se escogió la que coincidió en resultados con los datos UTM entregados por Transantiago en otras tablas de datos. Esta implementación se adaptó y encapsuló en una clase (ConvertCoordinate) que contiene dos métodos estáticos, el primero llamado LL2UTM y el segundo UTM2LL los cuales permiten pasar de un sistema de coordenadas a otro. El detalle de la implementación será mostrado en los capítulos de desarrollo del software de velocidades más adelante.

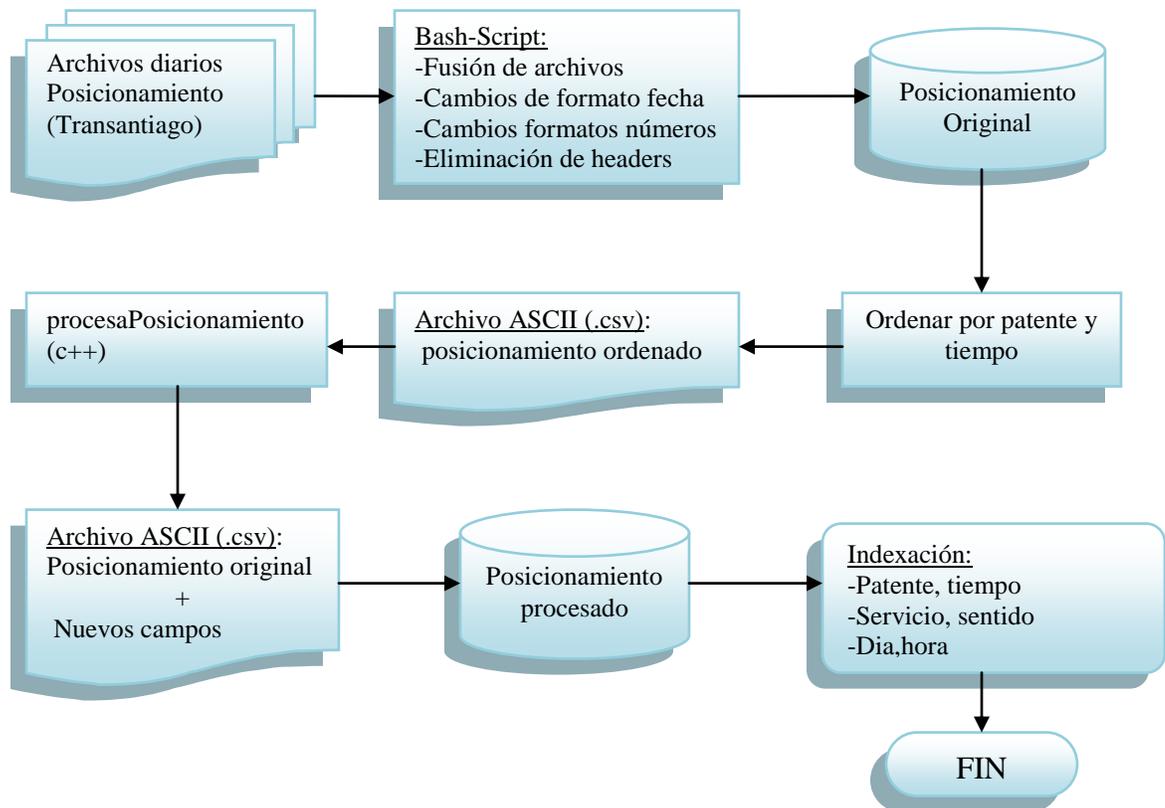


Diagrama 10: Descripción del proceso completo para llegar a la tabla final de posicionamiento GPS

Una vez procesada la tabla original de posicionamiento se obtiene entonces otra tabla de posicionamiento esta vez enriquecida con nuevos campos. El Diagrama 10 resume todas las etapas del método que procesa el posicionamiento. Comenzando con el cambio de formato de los datos originales de Transantiago para subirlos a una base de datos, luego un programa en C++ (procesaPosicionamiento) mezcla y agrega datos a una única tabla de posicionamiento la cual se vuelve a cargar en la base de datos donde se le aplican diversos indexados para las diversas consultas hechas tanto por este estudio como por otras investigaciones realizadas en el departamento de transporte.

La Figura 23 a continuación muestra la tabla inicial de posicionamiento con los datos originales de Transantiago y la tabla final después de haber pasado por todos los procesos ya descritos.

	concesionario character vary	patente character	mtc intec	tipogps character	tiempogps timestamp	latitud double	longitud double	ignicion character	tipogps character	tiempogps timestamp	latitud double	longitud double	ignicion character	velocidadgps double	precis precis
1	STP Santiago S./	BBJZ-70	4158	Sonda	2008-09-01 05:53:30	-33.509716	-70.560729	Si	Sonda	2008-09-01 05:53:30	-33.509716	-70.560729	Si	33.8916	
2	STP Santiago S./	BBJZ-70	4158	Sonda	2008-09-01 05:54:00	-33.508445	-70.561759	Si	Sonda	2008-09-01 05:54:00	-33.508445	-70.561759	Si	0	
3	STP Santiago S./	BBJZ-70	4158	Sonda	2008-09-01 05:54:30	-33.508033	-70.564781	Si	Sonda	2008-09-01 05:54:30	-33.508033	-70.564781	Si	45.9296	
4	STP Santiago S./	BBJZ-70	4158	Sonda	2008-09-01 05:55:00	-33.507415	-70.569503	Si	Sonda	2008-09-01 05:55:00	-33.507415	-70.569503	Si	51.856	
5	STP Santiago S./	BBJZ-70	4158	Sonda	2008-09-01 05:55:30	-33.506824	-70.573898	Si	Sonda	2008-09-01 05:55:30	-33.506824	-70.573898	Si	51.856	
6	STP Santiago S./	BBJZ-70	4158	Sonda	2008-09-01 05:56:00	-33.506278	-70.578430	Si	Sonda	2008-09-01 05:56:00	-33.506278	-70.578430	Si	9.8156	
7	STP Santiago S./	BBJZ-70	4158	Sonda	2008-09-01 05:56:30	-33.506084	-70.580177	Si	Sonda	2008-09-01 05:56:30	-33.506084	-70.580177	Si	36.8548	
8	STP Santiago S./	BBJZ-70	4158	Sonda	2008-09-01 05:57:00	-33.505676	-70.583999	Si	Sonda	2008-09-01 05:57:00	-33.505676	-70.583999	Si	33.8916	
9	STP Santiago S./	BBJZ-70	4158	Sonda	2008-09-01 05:57:30	-33.505428	-70.586471	Si	Sonda	2008-09-01 05:57:30	-33.505428	-70.586471	Si	44.8184	
10	STP Santiago S./	BBJZ-70	4158	Sonda	2008-09-01 05:58:00	-33.506507	-70.589225	Si	Sonda	2008-09-01 05:58:00	-33.506507	-70.589225	Si	17.9644	



	concesionario character vary	patente character	mtc intec	tipogps character	tiempogps timestamp	latitud double	longitud double	ignicion character	tipogps character	tiempogps timestamp	latitud double	longitud double	ignicion character	velocidadgps double	precis precis	servicio character	sentido character	periodo character	n_servicios integer	t_anterior integer	t_posterior integer	velocidad double	ntt integer		
1	STP Santiago S./	BBJZ-70	4158	Sonda	2008-09-01 05:53:30	-33.509716	-70.560729	Si	Sonda	2008-09-01 05:53:30	-33.509716	-70.560729	Si	33.8916		6291113.42	355040.152	D04	Regreso	03 - Transici	1	30	30	20.7886	2
2	STP Santiago S./	BBJZ-70	4158	Sonda	2008-09-01 05:54:00	-33.508445	-70.561759	Si	Sonda	2008-09-01 05:54:00	-33.508445	-70.561759	Si	0		6291252.83	354942.359	D04	Regreso	03 - Transici	1	30	30	20.4355	1
3	STP Santiago S./	BBJZ-70	4158	Sonda	2008-09-01 05:54:30	-33.508033	-70.564781	Si	Sonda	2008-09-01 05:54:30	-33.508033	-70.564781	Si	45.9296		6291294.29	354661.028	D04	Regreso	03 - Transici	1	30	30	34.1244	0
4	STP Santiago S./	BBJZ-70	4158	Sonda	2008-09-01 05:55:00	-33.507415	-70.569503	Si	Sonda	2008-09-01 05:55:00	-33.507415	-70.569503	Si	51.856		6291356.19	354221.307	D04	Regreso	03 - Transici	1	30	30	53.2867	0
5	STP Santiago S./	BBJZ-70	4158	Sonda	2008-09-01 05:55:30	-33.506824	-70.573898	Si	Sonda	2008-09-01 05:55:30	-33.506824	-70.573898	Si	51.856		6291415.57	353812.099	D04	Regreso	03 - Transici	1	30	30	49.6193	0
6	STP Santiago S./	BBJZ-70	4158	Sonda	2008-09-01 05:56:00	-33.506278	-70.578430	Si	Sonda	2008-09-01 05:56:00	-33.506278	-70.578430	Si	9.8156		6291469.66	353390.206	D04	Regreso	03 - Transici	1	30	30	51.0416	2
7	STP Santiago S./	BBJZ-70	4158	Sonda	2008-09-01 05:56:30	-33.506084	-70.580177	Si	Sonda	2008-09-01 05:56:30	-33.506084	-70.580177	Si	36.8548		6291488.76	353227.582	D04	Regreso	03 - Transici	1	30	30	19.6491	2
8	STP Santiago S./	BBJZ-70	4158	Sonda	2008-09-01 05:57:00	-33.505676	-70.583999	Si	Sonda	2008-09-01 05:57:00	-33.505676	-70.583999	Si	33.8916		6291528.61	352871.824	D04	Regreso	03 - Transici	1	30	30	42.9579	1
9	STP Santiago S./	BBJZ-70	4158	Sonda	2008-09-01 05:57:30	-33.505428	-70.586471	Si	Sonda	2008-09-01 05:57:30	-33.505428	-70.586471	Si	44.8184		6291552.60	352641.779	D04	Regreso	03 - Transici	1	30	30	27.7551	1
10	STP Santiago S./	BBJZ-70	4158	Sonda	2008-09-01 05:58:00	-33.506507	-70.589225	Si	Sonda	2008-09-01 05:58:00	-33.506507	-70.589225	Si	17.9644		6291428.88	352387.763	D04	Regreso	03 - Transici	1	30	30	33.9	0

Figura 23: Estructura final de la tabla de posicionamiento GPS de los buses (abajo), tabla original (arriba)

El Diagrama 11 muestra las tres tablas que forman la base de datos necesaria para que el software de estimación de velocidad funcione. La tabla de asignación de servicio no es utilizada posteriormente, dado que la información relevante quedó integrada a la nueva tabla de posicionamiento. El objetivo de resumir toda la información necesaria en una sola tabla es bajar los tiempos de cálculo en las consultas futuras, hay que recordar que esta información no sólo se utiliza en este estudio y al resumir y precalcular los campos una sola vez tienen un impacto significativo en el rendimiento de aplicaciones futuras.

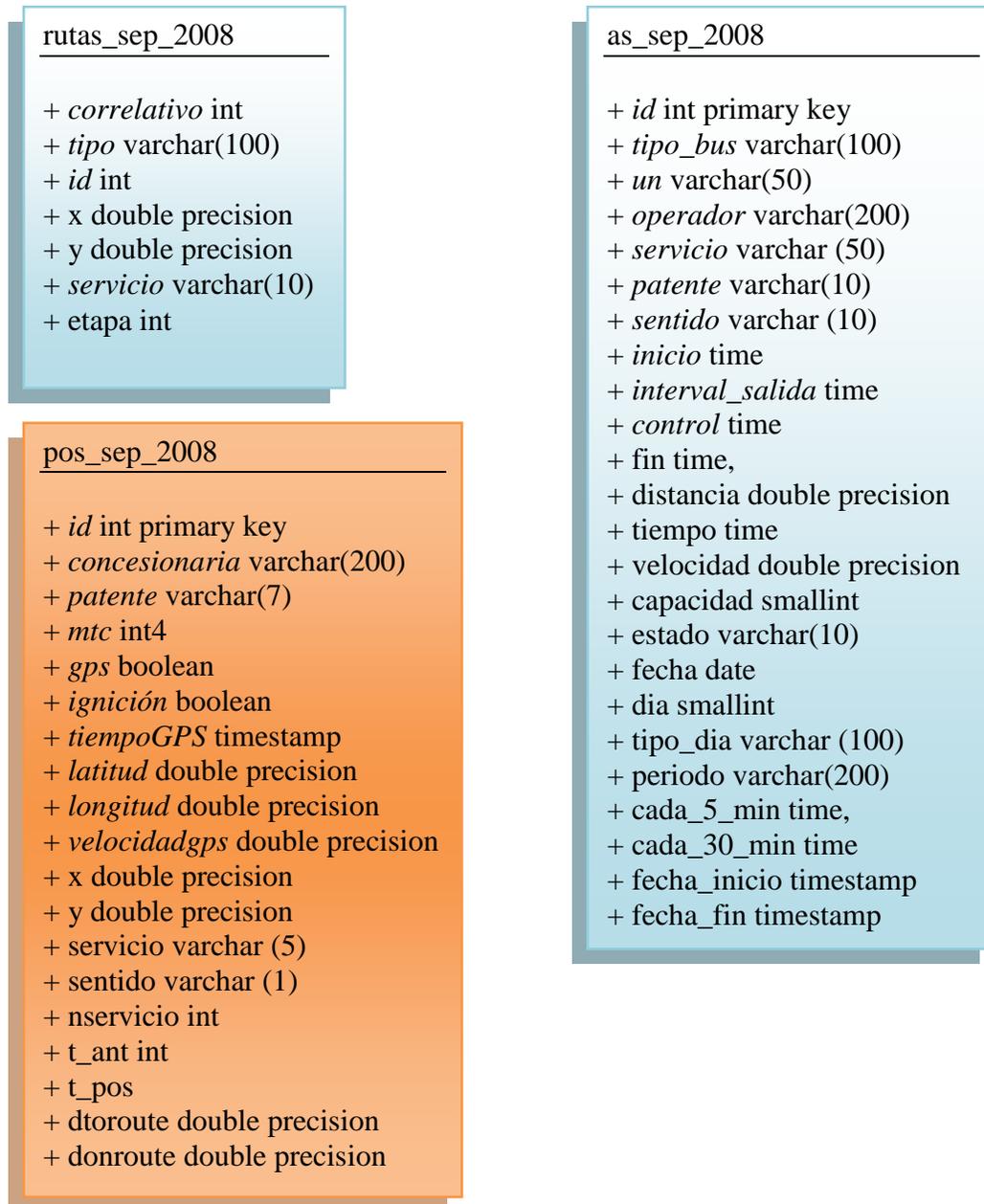


Diagrama 11: Tablas utilizadas para estimación de velocidades

4.3 Software estimación de velocidades

4.3.1 Diseño

Como se explicó en el capítulo de metodología, el diseño de este software sufrió variados cambios respecto al diseño original. La Figura 24 muestra un esquema general del diseño final del software para estimar las velocidades comerciales de un servicio-sentido en particular.

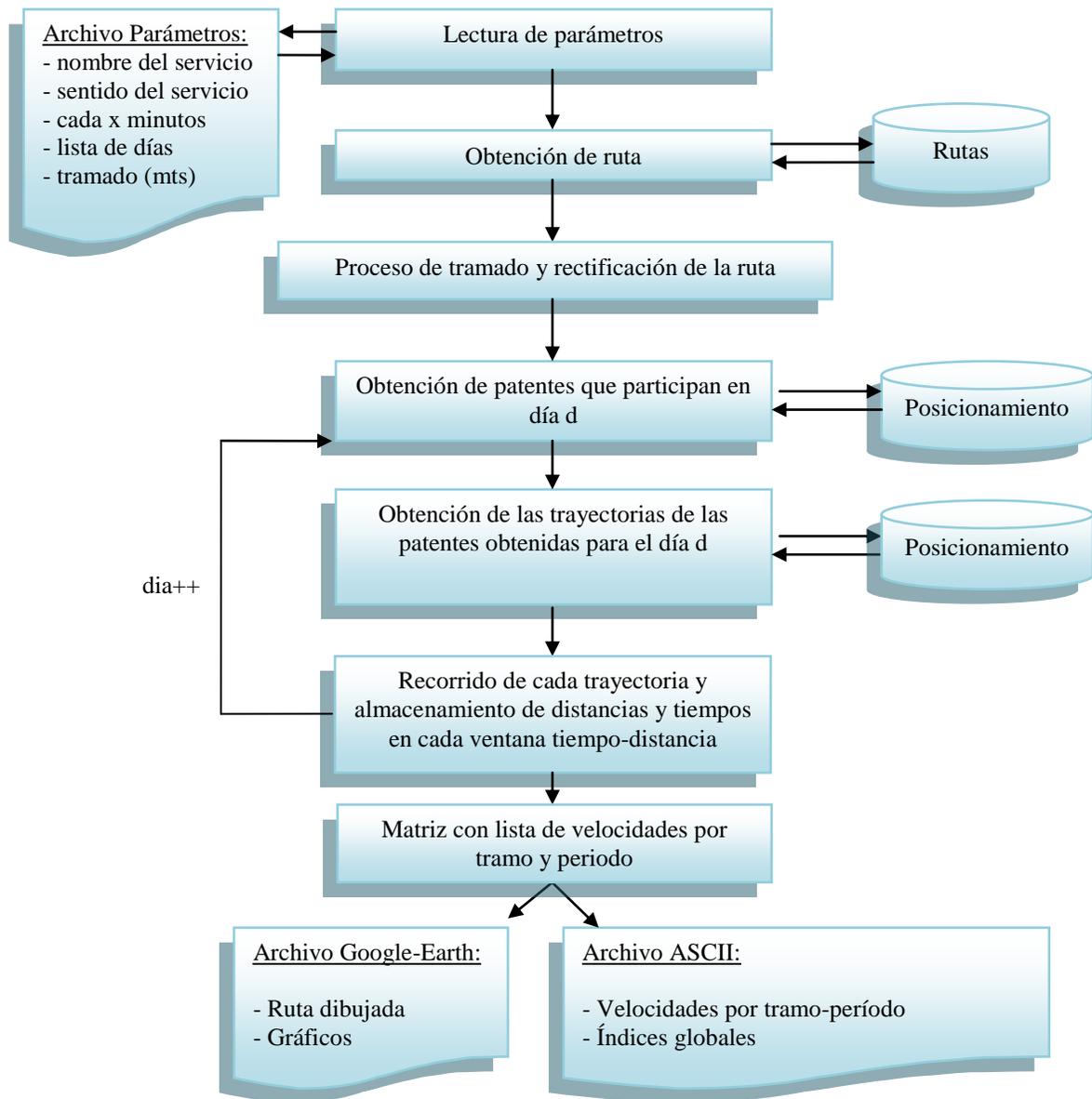


Figura 24: Diagrama de flujo de la rutina que calcula las velocidades para el Método VII (final).

El software entonces está compuesto inicialmente por un módulo que lee variados parámetros desde un archivo externo, parámetros como el nombre del servicio, qué días se evaluarán, entre otros. Todos estos parámetros son descritos con mayor detalle en el capítulo de implementación más adelante en este informe.

Una vez que los parámetros son almacenados en memoria la siguiente etapa en el programa es la obtención de la ruta del servicio especificado en los parámetros. Esto es ejecutado a través de un módulo llamado ruta, el cual además de cargar la ruta realiza diversos procesos a esta, también descritos en el capítulo de implementación.

Ya cargada la ruta y los parámetros el siguiente paso da pie al algoritmo principal del software. Este consiste en cargar las posiciones GPS de los buses. Para esto el programa consulta el listado de patentes que participan el día especificado, en otras palabras, todas las patentes que tienen datos para ese día, luego el programa obtiene por cada patente todos los datos GPS ordenados por tiempo de tal forma de establecer las distintas trayectorias recorridas por estos buses. En estos pasos diversas clases están involucradas, tales como Trayectoria, Bus, PosicionGPS...entre otras, de igual forma como ya se ha explicado, el detalle de cada uno de estos módulos es descrito en el capítulo de implementación.

Con todos los datos ya cargados el algoritmo que calcula las velocidades recorre temporalmente las trayectorias nodo por nodo, verificando en qué ventana de la grilla están, de tal forma que cuando se detectan 2 nodos adyacentes en distintas ventanas de grilla se calculan las intersecciones y se almacenan las distancias recorridas y tiempos utilizados en una lista por cada ventana de la grilla. De esta forma, una vez procesadas todas las trayectorias se suman todas las distancias en cada ventana y se dividen por la suma de todas las distancias, obteniendo así, una velocidad comercial representativa de cada ventana de la grilla.

Resumiendo, tenemos una grilla formada por tramos de distancia regulares (500 mts) en un eje y en el otro, tramos de tiempo (30 minutos), y por cada ventana formada entre 1 tramo de distancia y otro de tiempo tenemos una velocidad comercial representativa. Con esta matriz de información se da paso a la etapa de visualización de los resultados, donde el programa construye diversos gráficos y archivos compatibles con Google-Earth para poder analizar estas velocidades en detalle. También se construyen archivos con la información en formatos compatibles para cargarlos a una base de datos.

El Diagrama 12 muestra la estructura de clases que conforman el software. El diagrama muestra todas las clases sin el detalle de sus atributos y métodos por razones de espacio, tal detalle será descrito en las siguientes secciones.

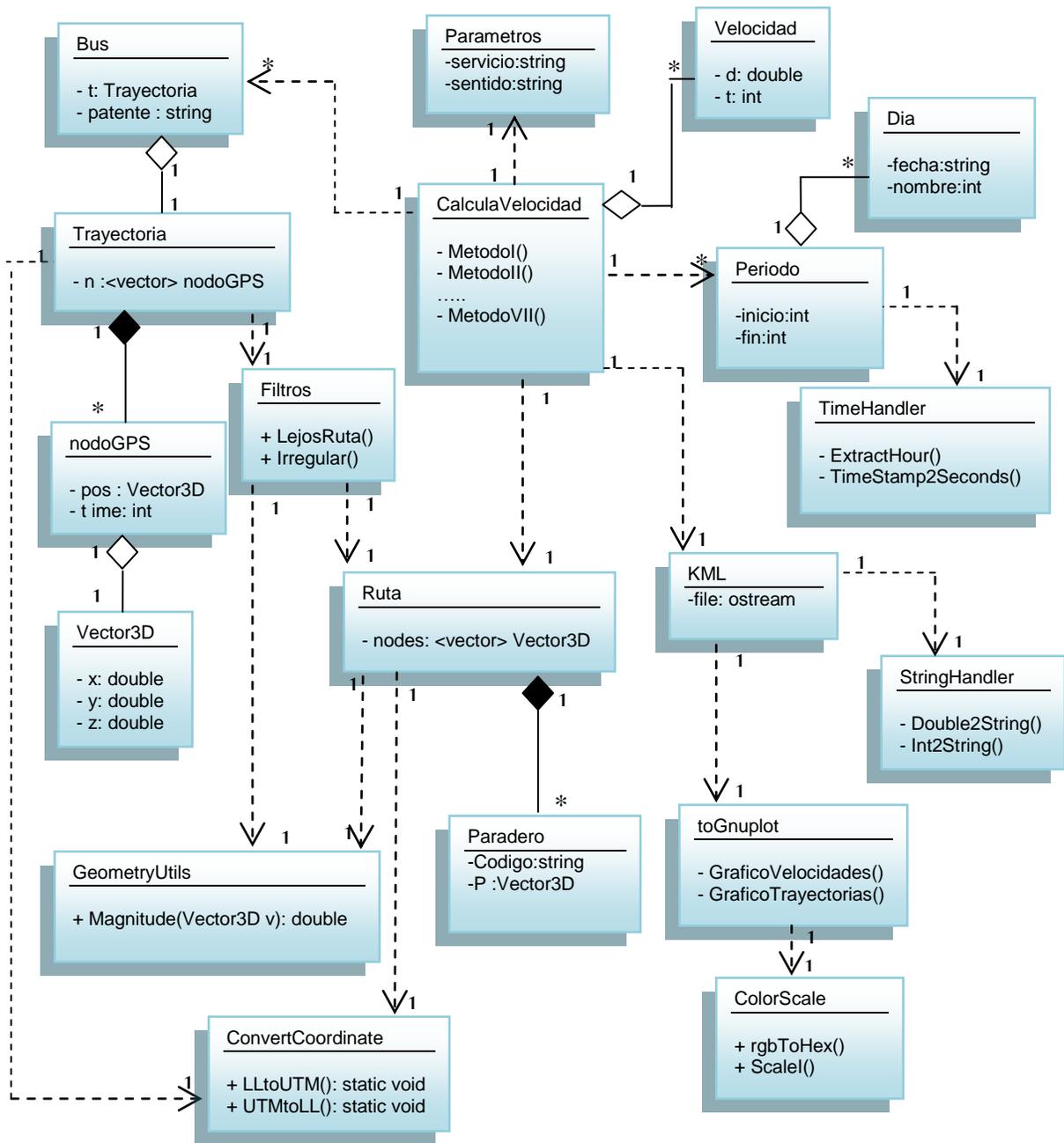


Diagrama 12: Clases resumidas y relaciones

4.3.2 Implementación

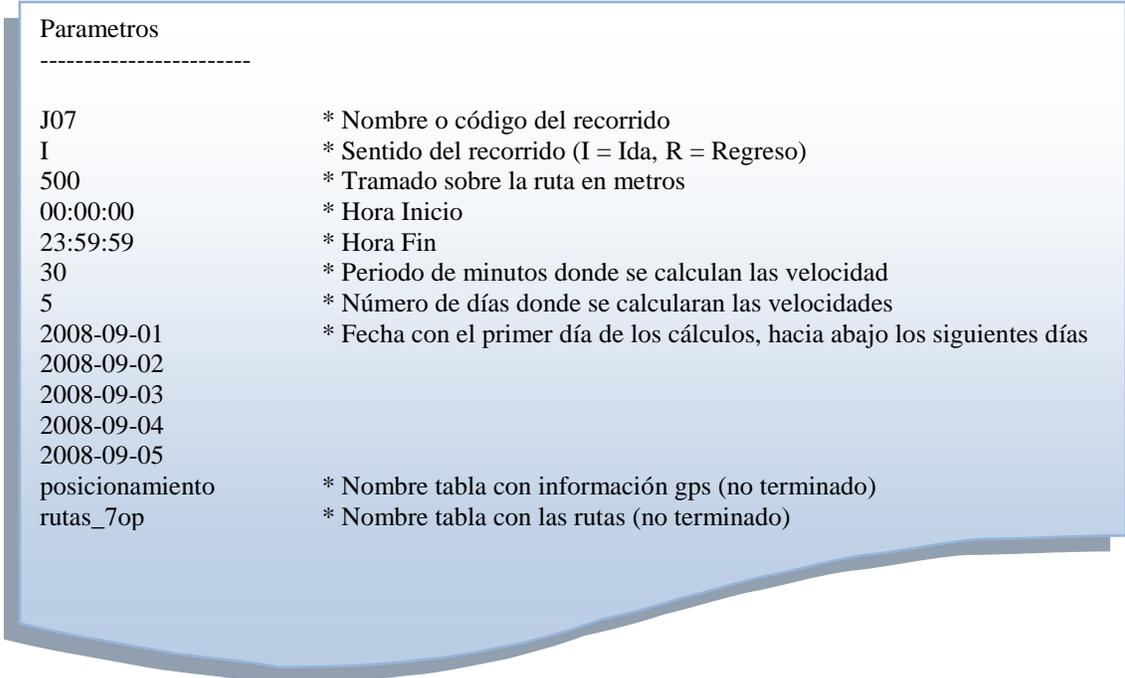
En este capítulo se detallan todos los módulos que forman el software de cálculo y diagnóstico de velocidades.

4.3.2.1 Parámetros

Uno de los primeros pasos en el desarrollo del software fue la creación de un módulo especial para la lectura y almacenamiento de parámetros. Los parámetros deducidos de los requerimientos son:

1. Nombre de servicio de buses(506,401,402)
2. Sentido del recorrido (Ida o Regreso)
3. Metros del tramado, estos son los metros que mide cada tramo (metros)
4. Hora inicio, Hora final, estos 2 parámetros marcan entre qué horas del día se realizaran los periodos de cálculo.
5. Tiempo de periodo, esta es la cantidad de minutos que conforma un periodo
6. Número de días que utilizaran para rescatar la información
7. Días que se utilizaran para los cálculos.
8. Nombre de las tablas en la base de datos donde se obtendrá la información de los buses

La Figura 25 muestra un ejemplo del formato que debe seguir el archivo de parámetros.



Parametros	

J07	* Nombre o código del recorrido
I	* Sentido del recorrido (I = Ida, R = Regreso)
500	* Tramado sobre la ruta en metros
00:00:00	* Hora Inicio
23:59:59	* Hora Fin
30	* Periodo de minutos donde se calculan las velocidad
5	* Número de días donde se calcularan las velocidades
2008-09-01	* Fecha con el primer día de los cálculos, hacia abajo los siguientes días
2008-09-02	
2008-09-03	
2008-09-04	
2008-09-05	
posicionamiento	* Nombre tabla con información gps (no terminado)
rutas_7op	* Nombre tabla con las rutas (no terminado)

Figura 25: Formato de archivo de parámetros.

El módulo parámetros fue implementado como una clase, donde los parámetros son almacenados en variables privadas y son cargadas con un método que lee el archivo de parámetros.

El Diagrama 13 detalla las variables y métodos que contiene la clase Parametros.

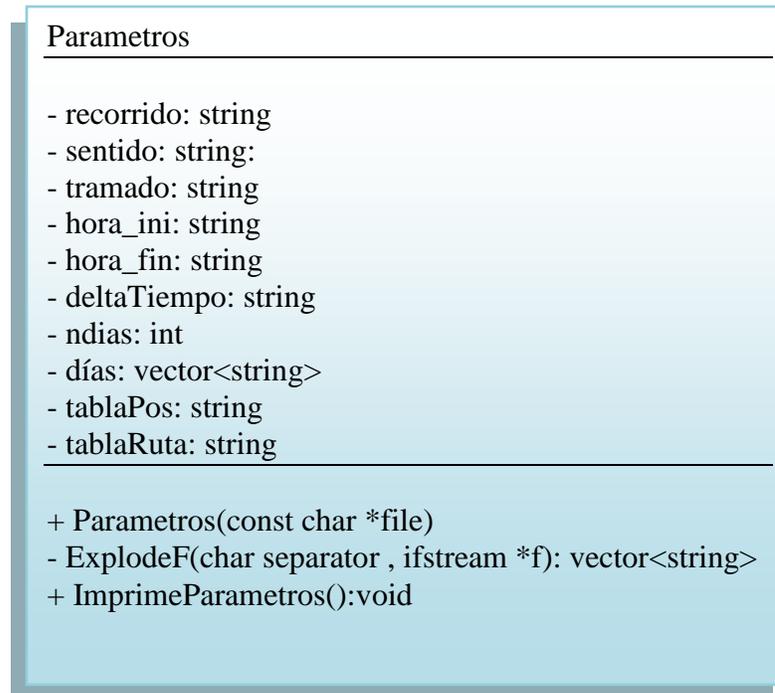


Diagrama 13: Representación de la clase Parametros

4.3.2.2 Manejo de rutas

Después del preproceso de las rutas descrito en la sección 4.2.1.3 se revisó visualmente algunas rutas al azar sobre el mapa de Santiago, para verificar que estos datos estuviesen correctamente georeferenciados. Para esto se construyó un pequeño programa que contiene rutinas para dibujar estas rutas en formato .kml (Google-Earth). La Figura 26 muestra la ruta del servicio 506-Ida.



Figura 26: Ruta original para el servicio 506 sentido Ida.

Las rutas se encontraron correctamente georeferenciadas, pero se observaron dos situaciones no esperadas:

1. Exceso de información en las rutas, es decir, se esperaban nodos sólo en los cambios de dirección de la ruta. Por el contrario existen más nodos generando redundancia de información.
2. Se encontró una situación inusual en algunas intersecciones de calles, se repite un patrón de tres nodos en forma triangular.

La Figura 27 muestra en forma gráfica estas dos situaciones.



Fuente : Elaboración propia

Figura 27: Exceso de nodos en la ruta (Abajo), comportamiento erróneo (Arriba)

La primera situación descrita no se presenta como un problema mayor, sólo podría serlo para algoritmos cuyo desempeño dependa fuertemente del número de nodos de la ruta.

Por otra parte la segunda situación representa un error en las estimaciones de distancias de la ruta, en otras palabras, generará una ruta más larga y errática que la esperada.

Para solucionar estas dos situaciones se propuso inicialmente eliminar los nodos donde el ángulo generado por los segmentos formados con ambos nodos vecinos fuera lo suficientemente obtuso, donde esto último estaría definido por un umbral por testear. Para esto se creó un programa⁹ que calcula esta nueva ruta simplificada y la imprime para visualizarla en Google Earth, los resultados mostraron que esta metodología no es adecuada para este caso. Las curvaturas largas no son bien resueltas generando una muy mala aproximación de la ruta ideal.

Después de eliminar el método anterior se optó por utilizar el algoritmo de simplificación de polígonos Douglas-Peucker¹⁰ [1] [2] [3], la particularidad de este algoritmo es generar una ruta simplificada compuesta por un subconjunto de los nodos de la ruta original y, además, poder definir que ningún nodo a una distancia d quede fuera de la ruta, esto permite que sea el procedimiento adecuado para solucionar los problemas ya mencionados.

⁹ El nombre del software es getRoute. Más adelante se describirá su funcionamiento en detalle.

¹⁰ http://en.wikipedia.org/wiki/Ramer-Douglas-Peucker_algorithm

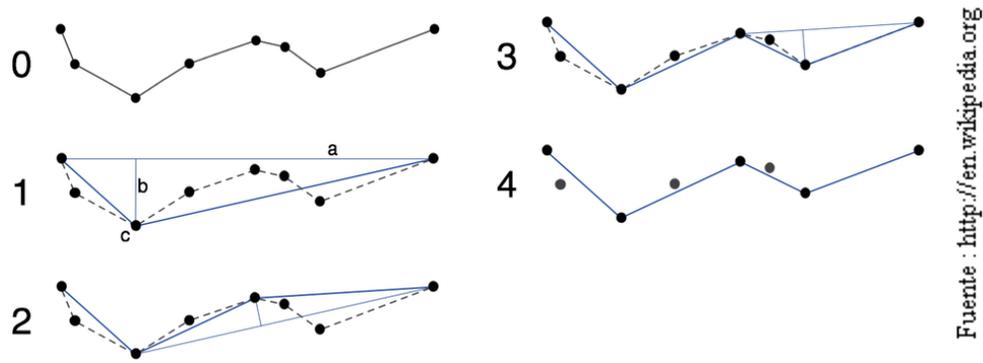


Figura 28: Ejemplo método de simplificación de polígonos.

Fuente : <http://en.wikipedia.org>

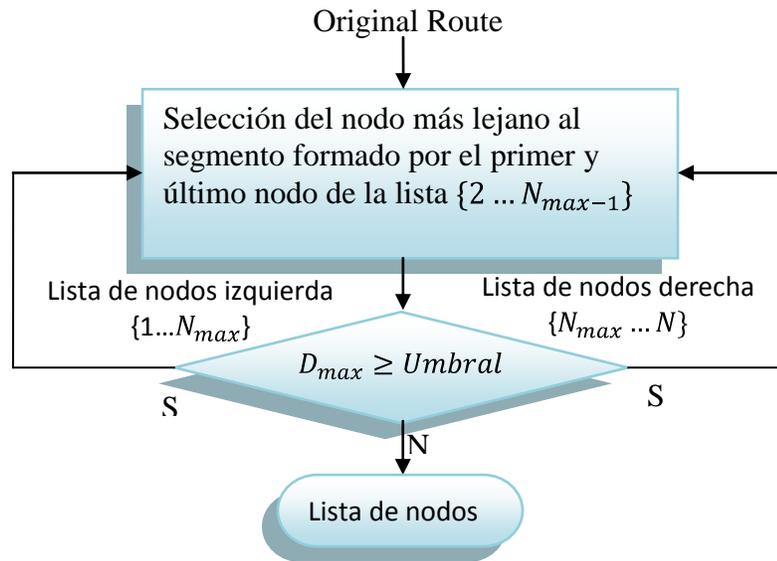


Diagrama 14: Descripción algoritmo recursivo de simplificación de polígonos.

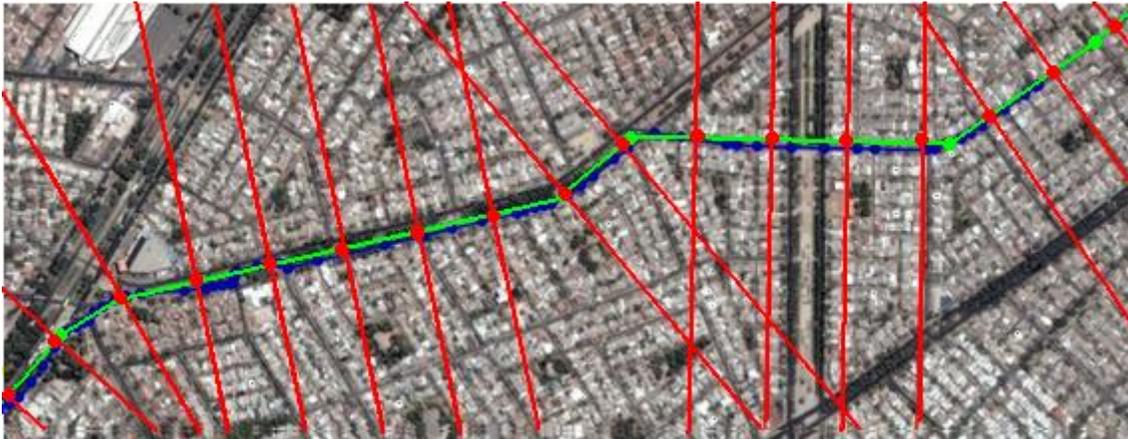
Los resultados mostrados por el algoritmo son aceptables, pero es necesario calibrar con mayor precisión el umbral de distancia de selección de nodos simplificados, actualmente fue definido en 4 mts, pero se han visto casos donde esto debe ser corregido para mejorar la calidad de la aproximación. La Figura 29 muestra cómo la ruta simplificada por el algoritmo Douglas-Peucker eliminó los dos problemas descritos.



Fuente : Elaboración propia

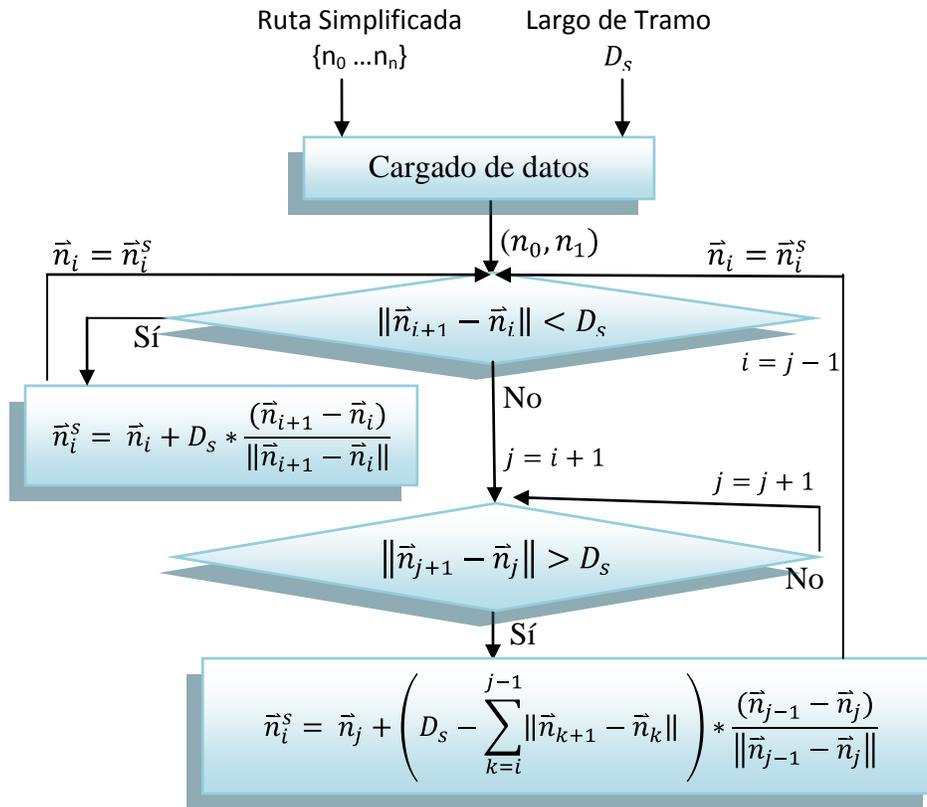
Figura 29: Ejemplo ruta original (azul) y ruta simplificada (verde)

Una vez definida la ruta, sólo falta la última etapa del proceso que corresponde a dividirla en tramos. Para efectos de esta investigación, esta generación de tramos se construirá en forma automática, dependiendo de un único parámetro que será el largo de estos tramos. En el futuro se ha discutido que estos tramos serán definidos manualmente con el objetivo de estudiar ciertos sectores específicos, por ejemplo una avenida entre dos calles. Para este proceso, que llamaremos tramificación, se desarrolló un algoritmo que toma como entrada la ruta simplificada y el parámetro de largo de tramo d_t y entrega un conjunto de puntos sobre la ruta que están distanciados d_t entre ellos, esta distancia es calculada sobre la ruta. La Figura 30, un ejemplo de una ruta tramada y el Diagrama 15 describe el funcionamiento de este algoritmo.



Fuente : Elaboración propia

Figura 30: Ejemplo de aplicación de algoritmo de tramificación (ruta 506).



\vec{n}_i^s = Nodo que representa el inicio del tramo i
 \vec{n}_i = Nodo i de la ruta simplificada

Diagrama 15: Diagrama de flujo para algoritmo de tramificación de una ruta.

Finalmente, la ruta es compuesta por dos conjuntos de nodos, los nodos de la ruta simplificada más los nodos del tramado. Un proceso final marca todos los segmentos de la ruta según al tramo que pertenecen. Esto último es realizado para facilitar las implementaciones posteriores.

La clase Ruta contiene dos métodos relevantes más, el cálculo de la distancia entre un punto en el plano y la ruta, y el cálculo de la distancia entre un punto cualquiera sobre la ruta y el inicio de la ruta.

El Diagrama 16 muestra el algoritmo utilizado para calcular la distancia a la ruta, básicamente se establece la distancia del punto contra todos los segmentos de la ruta y como resultado se entrega la distancia más corta.

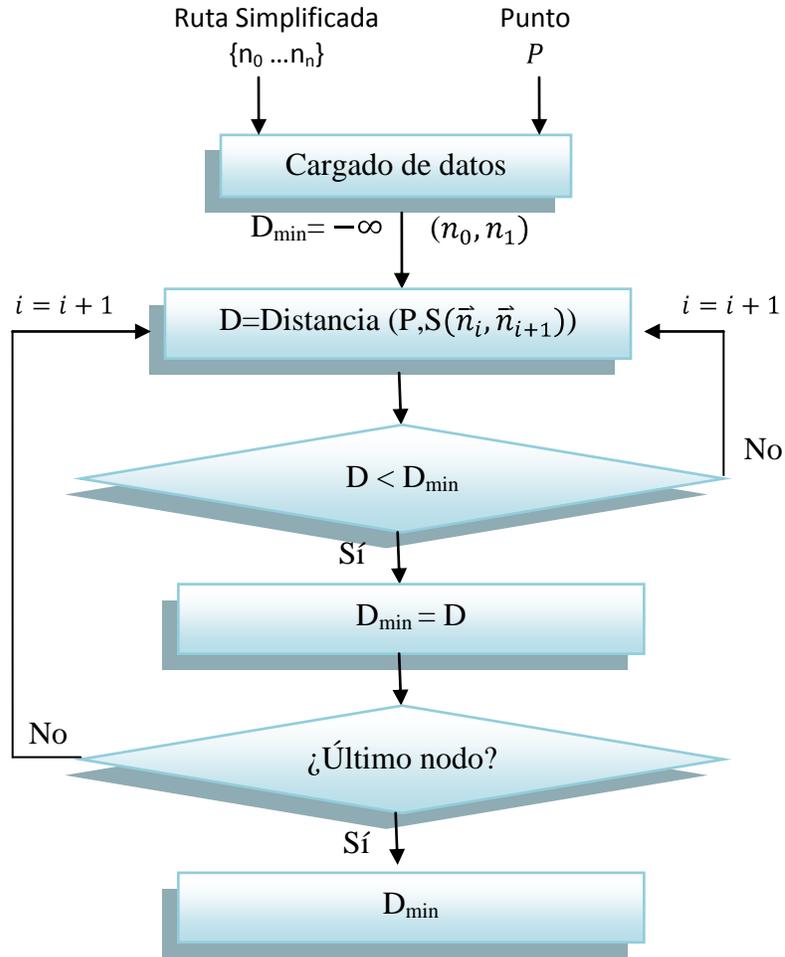


Diagrama 16: Algoritmo para encontrar la distancia a la ruta.

Usando el mismo algoritmo se calcula la distancia recorrida sobre la ruta, la única excepción es que a medida que se recorren los segmentos se guarda la distancia recorrida sólo cuando la condición $D < D_{\min}$ es cierta, de tal manera que al terminar de recorrer todos los segmentos no sólo se calcula la mínima distancia a la ruta sino que también la distancia recorrida sobre la ruta.

Se desarrolló una clase para manejar lo descrito en el manejo de rutas, la clase Ruta almacena la lista de nodos que componen la ruta original, la lista de nodos simplificados, la lista de nodos cada 500 metros (tramado) y contienen los métodos para obtenerlas. Esta clase utiliza la clase GeometryUtils que contiene variados métodos estáticos de geometría computacional utilizados por los métodos de la clase Ruta, también contiene una lista de paraderos utilizada en otros estudios y también utiliza la clase para manejar cambios de coordenadas en los casos que la ruta original solo venga en coordenadas latitud-longitud. El Diagrama 17 detalla lo explicado visualmente.

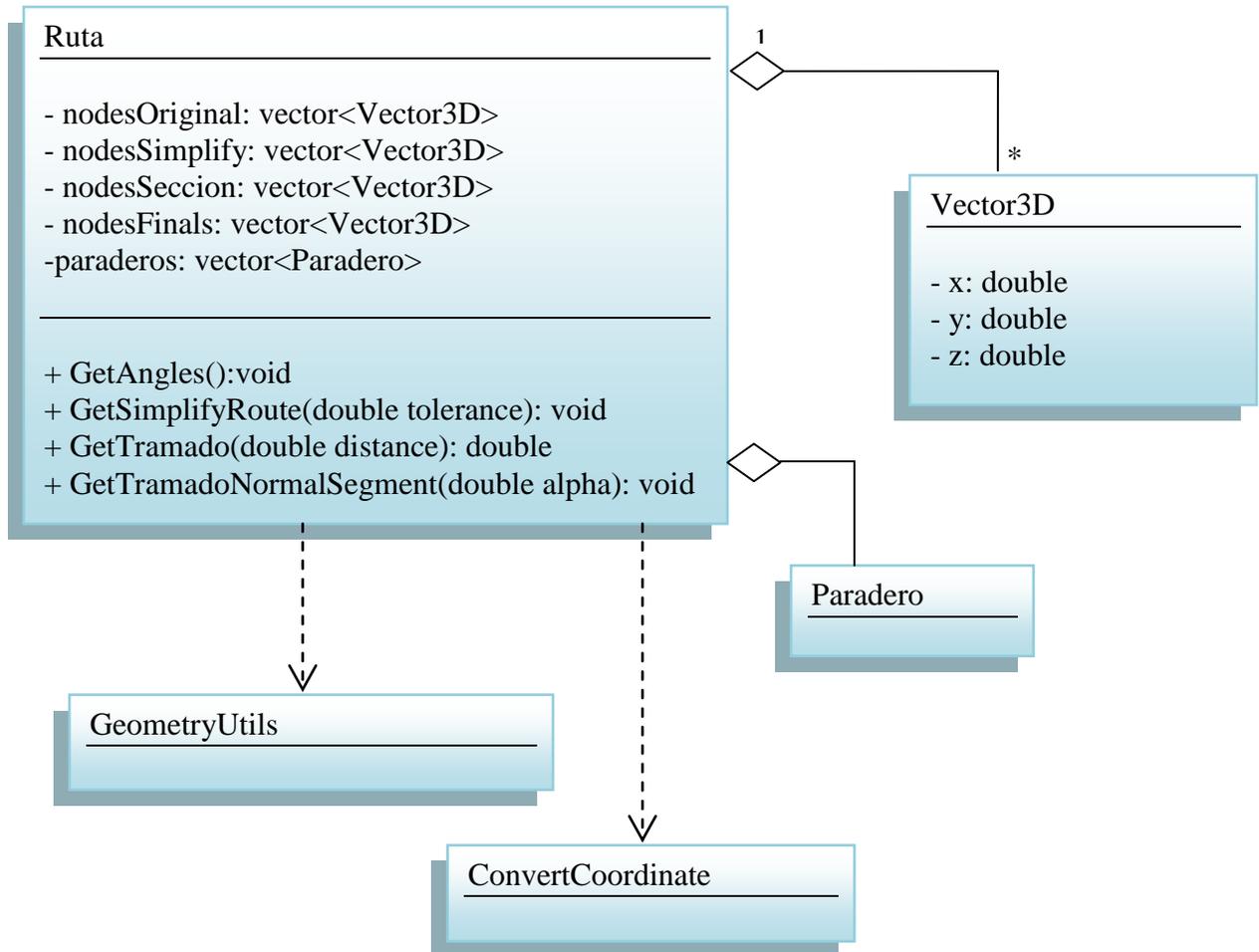


Diagrama 17: Descripción clase Ruta y sus dependencias.

4.3.2.3 Comunicación base de datos

La comunicación con la base de datos se implementó utilizando la librería open-source libpqxx¹¹ disponible para Linux. Esta librería fue instalada utilizando el administrador de paquetes (synaptic) de Ubuntu 8.10 y agregada al proyecto en Kdevelop.

La librería provee de métodos y objetos para manejar conexiones y consultas con la base de datos. A continuación se muestra un ejemplo de código para conectarse a la base de datos y recorrer una consulta:

Conexión a la base de datos:

```
Connection C("dbname=transantiago user='postgres' hostaddr='127.0.0.1' port='5432'
password='database'");
```

Instanciamiento de objeto manejador de la conexión:

```
work W(C);
```

Creación de consulta SQL:

```
String query= string("SELECT * FROM TABLE");
```

Ejecución de consulta:

```
result Resultado = W.exec(query);
```

Recorrido sobre el resultado de la consulta:

```
for (result::const_iterator q = Resultado.begin(); q != Resultado.end(); ++q)
{
    cout < q[0]; //primera columna
    cout < q[1]; //segunda columna
    //....
}
```

El programa contiene algunas funciones que reciben como parámetro un puntero a un objeto de tipo *work* para utilizar la misma conexión para hacer otras consultas.

¹¹ <http://pqxx.org/development/libpqxx/>

4.3.2.4 Cambios de coordenadas

Uno de los primeros desafíos en el inicio de este trabajo fue que muchos datos provistos por Transantiago venían con dos distintos formatos de coordenadas geográficas. Se utilizaban Latitud/Longitud (LL) o X/Y (UTM). Ambos tipos de coordenadas son necesarios según sea el caso. Por ejemplo, para poder visualizar sobre algún mapa georeferenciados son necesarias las coordenadas LL, pero si se necesita calcular distancias entre puntos esto se hace con coordenadas (UTM) que están en metros. La necesidad de tener ambos formatos según sea el caso hizo necesario construir o conseguir un módulo que sea capaz de pasar de un formato a otro.

Es importante entender que esta conversión no es trivial, dado que latitud y longitud son coordenadas sobre una superficie elipsoidal y UTM es una proyección de tales coordenadas sobre un plano, la Figura 31 muestra la idea global de esta conversión.

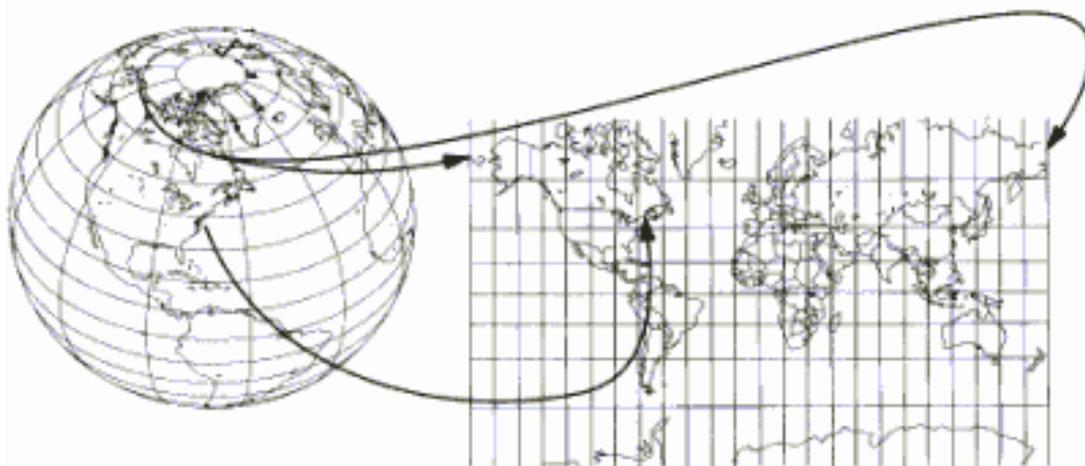


Figura 31 : Conversión de coordenadas LL a UTM

Se realizó una investigación en internet para buscar código abierto que realizara esta conversión y coincidiera con la usada por Transantiago, se buscó que coincidiera porque UTM tiene varios estándares. Para esto se seleccionó una muestra de datos de la tabla de posicionamiento ya que esta contiene los dos tipos de coordenadas entregadas originalmente por Transantiago.

Con esta investigación se calculó que el estándar utilizado por Transantiago es WGS84¹² con zona 19H, este estándar se ha utilizado para todos los datos involucrados en este trabajo, hay que tener en consideración que datos como la posición de los nodos de las rutas, las posiciones de los paraderos pudieron haber estado en otros estándares.

El primer código fuente encontrado fue GPSY¹³. Este se integró al programa como una clase (ConvertCoordinate) con los métodos de conversión como estáticos. El Diagrama 18 muestra el contenido de este módulo.

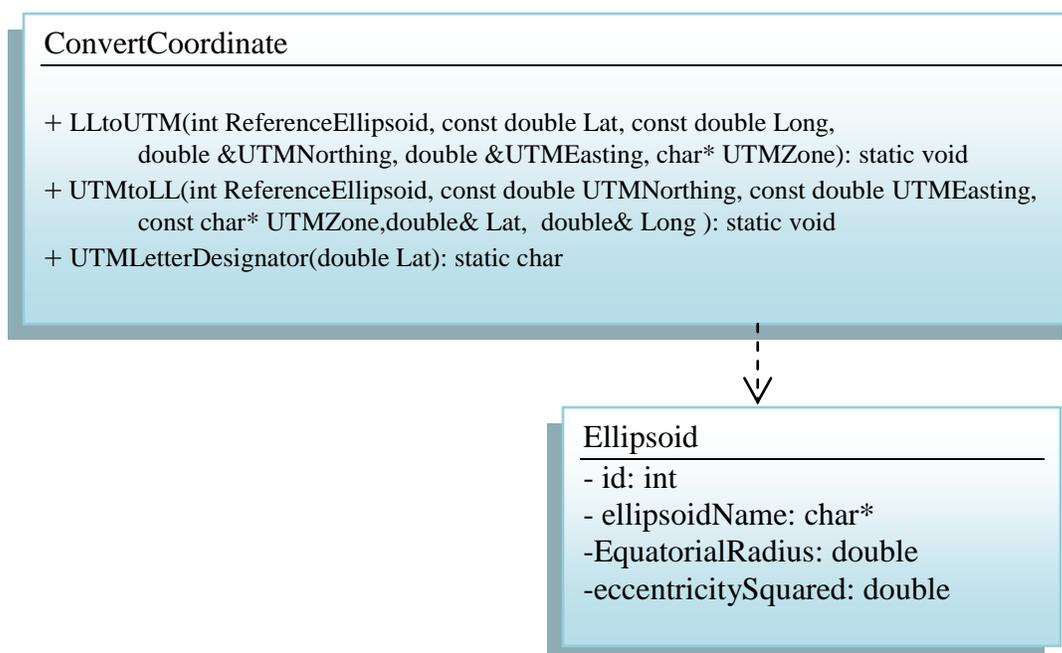


Diagrama 18: Descripción clase CovertCoordinate

Una vez pasado el test de igualdad entre las coordenadas de Transantiago se decidió no seguir buscando otros códigos fuentes que resolvieran este problema, sólo en el caso que más adelante se determinara que este módulo es un cuello de botella en rendimiento se buscará uno nuevo o se intentará mejorarlo.

¹² <http://es.wikipedia.org/wiki/WGS84>

¹³ <http://www.gpsy.com/gpsinfo/geotoutm/>

4.3.2.7 Filtrado de datos

A medida que se desarrollaron las distintas metodologías se identificaron datos que no debían participar en los cálculos. Entre estos se encuentran buses que estaban fuera de ruta y errores en las observaciones GPS.

Datos fuera de ruta

La Figura 32 muestra varios ejemplos de buses fuera de ruta, la ruta es la curva que atraviesa la imagen de izquierda a derecha por el centro. El primero encerrado en un círculo blanco representa al reiterativo caso donde los buses han tenido que tomar un desvío por razones de construcción de nuevas vías o algún otro elemento desconocido. El segundo caso encerrado en círculos negros representa un bus en particular que está realizando otro servicio o por alguna razón el chofer no está realizando del servicio correcto.

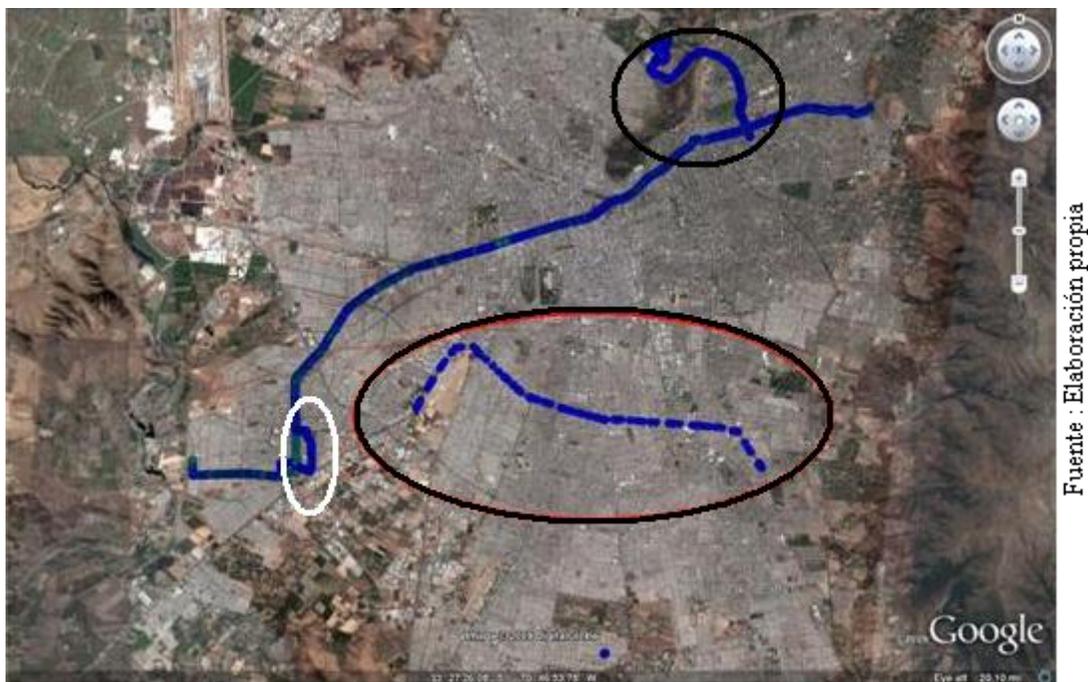
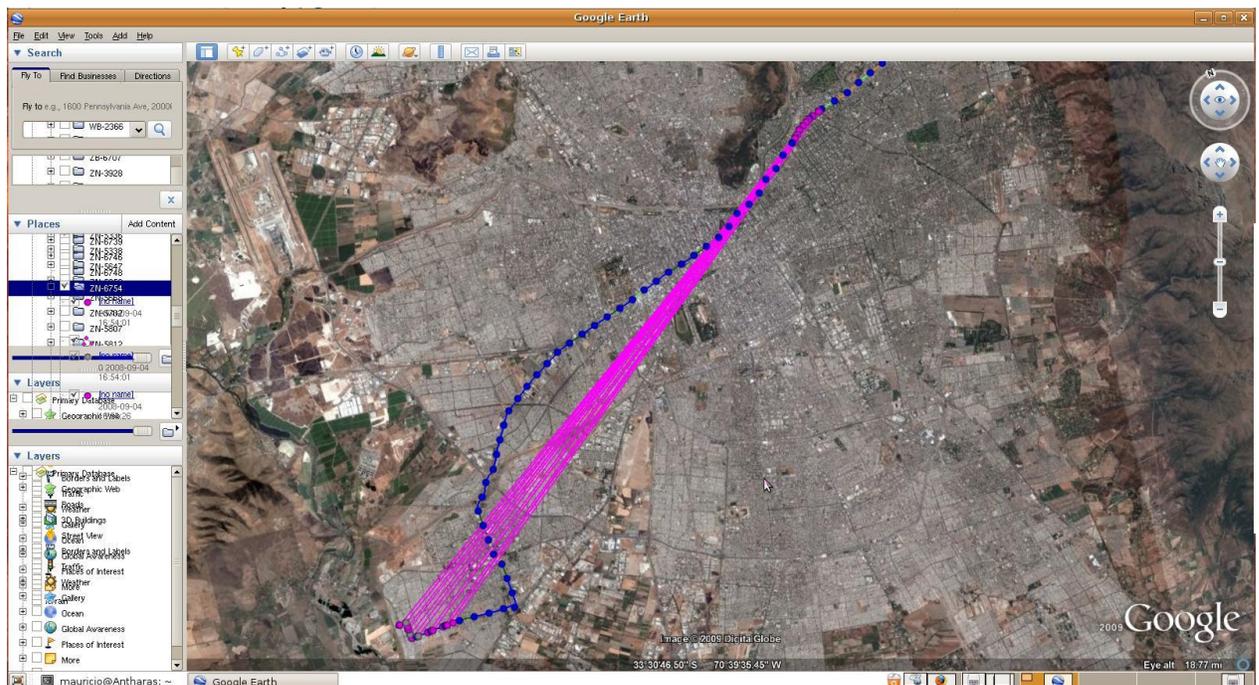


Figura 32: Buses fuera de ruta

Dado estos problemas encontrados se tomó la decisión de construir un procedimiento que permitiera detectar estos elementos para filtrarlos de los cálculos. Se construyó un algoritmo que reutiliza la proyección de nodos GPS a la ruta y entrega la distancia a esta, y se agregó al software de pre-procesamiento de los datos de posicionamiento, con esto se insertó un nuevo campo que contiene la distancia a la ruta. Una vez aplicado, los expertos en el área definieron una distancia umbral de 20 metros para determinar que dato es válido para los cálculos de velocidad.

Errores GPS

Por otra parte se encontraron distintos problemas relacionados con errores de las emisiones GPS. Por ejemplo emisiones GPS cada 30 segundos pero separadas por kilómetros de distancia. La Figura 33 muestra un ejemplo de esto último, los nodos rosados son las emisiones GPS y los azules representan la ruta, las líneas representan un segmento formado por dos nodos que están cada 30 segundos, las líneas que parecen atravesar Santiago muestran la existencia de saltos del bus cada 30 segundos.



Fuente : Elaboración propia

Figura 33: Emisiones GPS que saltan espacialmente kilómetros por segundo.

Para poder detectar este tipo de datos, se optó por agregar cuatro campos más a la base de datos. Estos son las diferencias de tiempo y distancia con los registros anteriores y posteriores de posicionamiento. De esta manera cuando se encuentra algún tiempo mayor a 5 minutos se considera un corte de trayectoria y si se encuentra una relación distancia/tiempo mayor a 200 km/hr se considera un dato inválido.

4.3.2.8 Geometría-Matemáticas

Como se mostró en la implementación de los anteriores módulos, parte importante del software consta de una clase que contiene un conjunto de métodos estáticos de geometría computacional. Este paquete cuenta con los métodos básicos para manejo de vectores como por ejemplo producto cruz, magnitud, normalizado, producto punto como también cuenta con métodos de nivel intermedio como distancia entre puntos, distancia de un punto a un segmento, proyección de un punto sobre un segmento, intersección de segmentos y por último métodos más complejos como por ejemplo la simplificación de polígonos de Douglas-Peucker o tramado regular de una polilínea.

Otro módulo que contiene el software es un paquete de métodos estadísticos. En la actualidad solo contiene máximo, mínimo y medias, pero más adelante deben integrarse métodos para la construcción de histogramas, y cálculo de estadísticos especiales, entre otros.

GeometryUtils

```
+ Magnitude(Vector3D v): double
+ Normal(Vector3D v1,Vector3D v2, Vector3D v3): Vector3D
+ Normalize(Vector3D v1): Vector3D
+ Dot(Vector3D v1,Vector3D v2): double
+ Cross(Vector3D v1,Vector3D v2): Vector3D
+ DistPointSegment(Vector3D p, Vector3D v1, Vector3D v2): double
+ DistPoint2Point(Vector3D p1,Vector3D p2): double
+ GetAngle(Vector3D v1,Vector3D v2): double
+ PolySimplify( double tol, vector<Vector3D> V, vector<Vector3D> *sV ): int
+ SimplifyDP( double tol, vector<Vector3D> *v, int j, int k, vector<int> *mk ): void
+ PolyTramado( vector<Vector3D> V,double distance,vector<Vector3D> *tV): double
+ NormalSegment(Vector3D v1,Vector3D v2): Vector3D
+ Intersect2DSegments( Segmento S1, Segmento S2, Vector3D *I0, Vector3D *I1 ): int
+ InSegment( Vector3D P, Segmento S): int
+ Perp2D(Vector3D u, Vector3D v): double
```

Estadísticas

```
+ Máximo(vector<double> in): double
+ Mínimo(vector<double > in): double
+Media(vector<double> in): double
+ Histograma(vector<double> in, double ancho): vector<int>
```

Diagrama 19: Paquetes matemáticos

4.3.2.9 Documentación

Para documentar el desarrollo del software, aparte de este informe, se utilizó la herramienta Doxygen¹⁴ configurada para su uso a través del IDE KDevelop3. Esta herramienta, en pocas palabras, genera documentación automática a partir de la documentación escrita en el código más la arquitectura de clases. Esta documentación puede ser generada en variados formatos como por ejemplo HTML y/o Latex.

4.3.3.0 Estandarización de código fuente

Un elemento importante en el desarrollo de software consiste en crear código fuente ordenado y de fácil entendimiento para otras personas, de tal manera que una vez terminado pueda ser modificado con mediana facilidad por otros desarrolladores. Con esta finalidad existen variados estándar de codificación. Para este trabajo se utilizó el estándar GNU¹⁵.

¹⁴ <http://www.stack.nl/~dimitri/doxygen/>

¹⁵ <http://www.gnu.org/prep/standards/standards.html>

4.4 Interfaces

Una importante característica de este trabajo consiste en el despliegue gráfico de la información, este fue diseñado y construido en varios ciclos iterativos hasta obtener los resultados mostrados en el presente informe.

En términos generales el objetivo de este módulo consiste en visualizar e interactuar con los resultados de la estimación de las velocidades comerciales de tal manera que permita diagnosticar las razones de cualquier comportamiento. Desde un principio esto no fue un problema trivial al tratar de ver las trayectorias de los buses sin un mapa de Santiago de fondo o simplemente contrastar una ruta de un servicio con las trayectorias de los buses que deberían estar realizando este servicio.

Un requerimiento importante de este trabajo consiste en que Transantiago pueda revisar la información obtenida de los cálculos, para esto se especificaron ciertos ítems importantes por servicio, nombrados a continuación:

1. Ruta sobre un plano de Santiago
2. Cortes de tramos sobre la ruta con algún color que identifique el rango de velocidad
3. Una ruta coloreada por cada periodo de tiempo (30 minutos)
4. Información numérica de las velocidades representativas por tramo y periodo
5. Los buses que participaron en los cálculos por tramo y periodo con sus respectivas velocidades individuales
6. Las trayectorias de los buses
7. Gráficos que resumen la información para ver patrones sobre espacio-distancia

Inicialmente se investigaron algunos GIS¹⁶ (Sistemas de Información Geográfica) de código abierto que permitieran dibujar cualquier tipo de elemento 2D o 3D sobre el mapa de Santiago. El principal problema con ellos es que necesitaban de la orthofoto de Santiago, y los tiempos para gestionar esta información se estimaron más altos que los necesarios para tener resultados. La orthofoto es una imagen de gran escala que contiene una vista satelital de Santiago.

Por otra parte un punto importante de esta etapa es construir un sistema de visualización que pueda ser utilizado tanto por la universidad como también para Transantiago, lo que significa que utilizar algún software licenciado podría implicar alguna barrera para cualquiera de estas entidades

¹⁶ http://en.wikipedia.org/wiki/Geographic_information_system

Otro punto a considerar es la portabilidad del sistema de despliegue entre sistemas operativos distintos dado que la plataforma utilizada por la universidad es Linux y la de Transantiago es Windows. Por tanto es deseable que el sistema de despliegue sea multiplataforma asumiendo que cambiar la plataforma de cálculo y desarrollo sea un problema aun más complejo.

Considerando las restricciones y requerimientos ya descritos se investigó Google-Earth como posible solución al problema de visualización, para esto se estudio su API de desarrollo y se testearon distintas características que se explicarán con más detalle en las siguientes secciones de este capítulo.

Finalmente al pasar todos los tests se optó por Google-Earth como plataforma de visualización y algunas librerías extras para generar gráficos bidimensionales que resumen algunos datos específicos. Este capítulo se divide en dos módulos desarrollados para el despliegue gráfico.

1. Desarrollo sobre Google-Earth
2. Creación de gráficos bidimensionales

El diseño utilizado en la actualidad fue construido a partir de varias iteraciones con los profesores y Transantiago, aún está en estado de cambio a medida que se integran o eliminan requerimientos.



Diagrama 20: Ciclo de diseño y re diseño según uso.

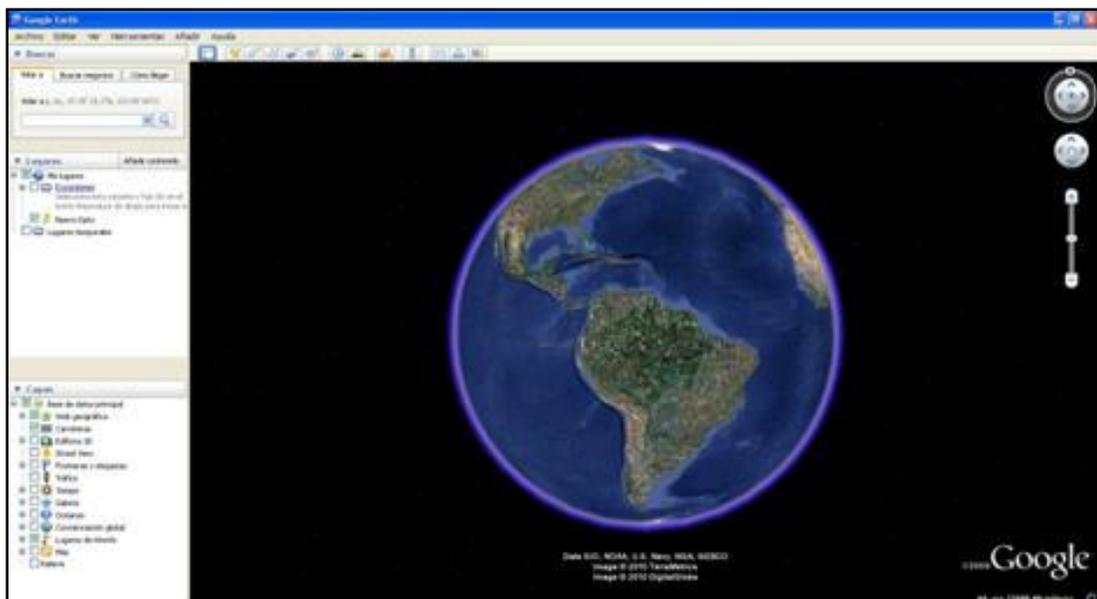
Tal como muestra el Diagrama 20 el diseño es modificado a medida que los investigadores de la universidad y Transantiago entregan retroalimentación al utilizar el sistema de despliegue.

4.4.1 Visualización en Google-Earth

Google-Earth es una herramienta gratuita, sólo la versión no profesional, que tiene como objetivo permitir al usuario navegar sobre un mapa del globo terráqueo formado por imágenes satelitales, además provee variadas herramientas para marcar lugares georeferenciados, organizar distintas capas de información, dibujar elementos básicos en tres dimensiones, sobre poner imágenes tanto en la superficie terrestre como en la misma interfaz del programa, entre otros.

Con todas estas características podemos utilizar esta herramienta como una plataforma GIS, de la misma manera que Travis M. Smith [8] la utiliza para una aplicación de análisis climático.

La Figura 34 muestra una imagen de la interfaz de este software, en términos generales está compuesta por un menú en la parte superior, selector de información a mostrar a la izquierda, la pantalla de visualización donde se muestra el planeta desde una perspectiva muy alejada, y botones de navegación en la parte superior derecha.



Fuente : Elaboración propia

Figura 34 : Interfaz Google-Earth

Otra característica importante de mencionar de este software es que el usuario puede dibujar puntos, caminos, integrar imágenes o información en formato HTML¹⁷ a algún punto que él defina y toda esta información es guardada en un archivo con un formato creado por Google llamado KML [9], este formato es una versión modificada de XML, lo que permite que una persona o software cree estos archivos en forma externa y puedan ser leídos visualizados sobre Google-Earth.

¹⁷ <http://en.wikipedia.org/wiki/Html>

De esta manera el módulo presentando en este capítulo muestra el desarrollo de un paquete de funciones que permiten crear estos archivos KML con la información que se quiera mostrar y la forma de mostrarla.

El primer paso para desarrollar esta plataforma de despliegue gráfico, fue construir y testear varios métodos básicos que utilizaban la API¹⁸ de desarrollo. Para ello se implementaron inicialmente los dibujos de elementos básicos como puntos y segmentos para luego construir métodos capaces de construir caminos compuestos por una lista de segmentos, inserción de imágenes sobre la pantalla de la interfaz de Google-Earth. A continuación se listan y describen las funciones de dibujo que componen este módulo.

Dibuja punto: Dibuja un punto sobre la superficie terrestre con coordenadas Latitud-Longitud, la forma y el color del elemento que representa al punto también se definen en el método.

Dibuja segmento: Dibuja una línea entre dos puntos georeferenciados sobre la superficie terrestre, de la misma forma que el punto su color es un parámetro.

Dibuja camino: Dibuja un listado de segmentos unidos que conforman un camino.

Integra imágenes viewport: Dibuja una imagen de formato jpg, gif, png sobre la pantalla que muestra la superficie terrestre, la posición donde se ubica esta imagen en pantalla es manejada por parámetros dentro de este método.

Manejo de colores de los elementos: Este método depende de un módulo extra construido para manejar escalas de colores en forma automática, este módulo será descrito con más detalle más adelante en este capítulo.

Despliegue de información sobre los elementos: Al hacer click con el mouse sobre algún elemento (punto o segmento) se despliega una sub-ventana que muestra información en formato HTML (página web).

Dibujo con animación: Manejando un campo adicional de tiempo se pueden mover los elementos sobre la superficie terrestre.

Para la implementación de estos métodos se creó una clase llamada CreaKML. Esta clase permite construir un objeto que dibuja sobre un archivo propio los distintos elementos que se requieran utilizando el estándar kml. Además contiene un conjunto de métodos para el manejo del resto de elementos necesarios para construir un archivo KML como son:

¹⁸ <http://code.google.com/apis/kml/documentation/>

Capas de información: Dentro de la información que se despliega es necesario seguir una jerarquía de capas, por ejemplo, si dibujo 10 veces un camino pero con distintos colores cada vez se debe especificar que son caminos distintos para poder definir cual ver dentro de la interfaz.

Estilos: Para poder definir formas y colores en KML, se deben definir estilos, estos son un conjunto de colores y parámetros de forma, que se definen en el encabezado del archivo y que son codificados para su posterior uso en los elementos como puntos o segmentos.

Manejo archivo: Esto refiere al manejo de creación y escritura de un archivo kml.

El constructor de la clase utiliza como parámetro un nombre de archivo, con el cual se abre el archivo y se escribe a través de los distintos métodos de la clase, a continuación se muestra un ejemplo donde se dibujan un punto y un segmento.

```
//Constructor de la clase  
KML kml("ejemplo.kml");
```

```
//Se abre un documento  
kml.OpenDocument();
```

```
//Se crean dos estilos (azul y rojo)  
kml.InsertStyleColor("#FFFF0000", "Azul");  
kml.InsertStyleColor("#FF0000FF", "Rojo");
```

```
//Se abre un directorio donde estará dibujado lo que este dentro de él (layer)  
kml.OpenFolder("Punto");
```

```
//Se dibuja un punto con el estilo Azul ya definido  
kml.DrawPoint("NombrePunto", "", -33.123456, -70.123456, "Azul", 1);
```

```
//Se cierra el directorio que lo contiene  
kml.CloseFolder();
```

```
//Se crea un nuevo directorio para separar el punto del segment, esto para el ejemplo, ya que  
//tambien pueden estar agrupados en el mismo directorio  
kml.OpenFolder("Segmento");
```

```
//Se dibuja el segmento utilizando el estilo con color Rojo ya definido  
kml.DrawSegment("NombreSegmento", "",  
                -33.123456, -70.123456,  
                -33.123456, -70.123456, "Rojo", 1);
```

```
//Se cierra el directorio que contiene el segmento  
kml.CloseFolder();
```

```
//Se cierra el documento  
kml.CloseDocument();
```

```
//Se cierra el archivo  
kml.CloseFile();
```

Finalmente la clase KML queda definida tal como muestra el Diagrama 21 a continuación.

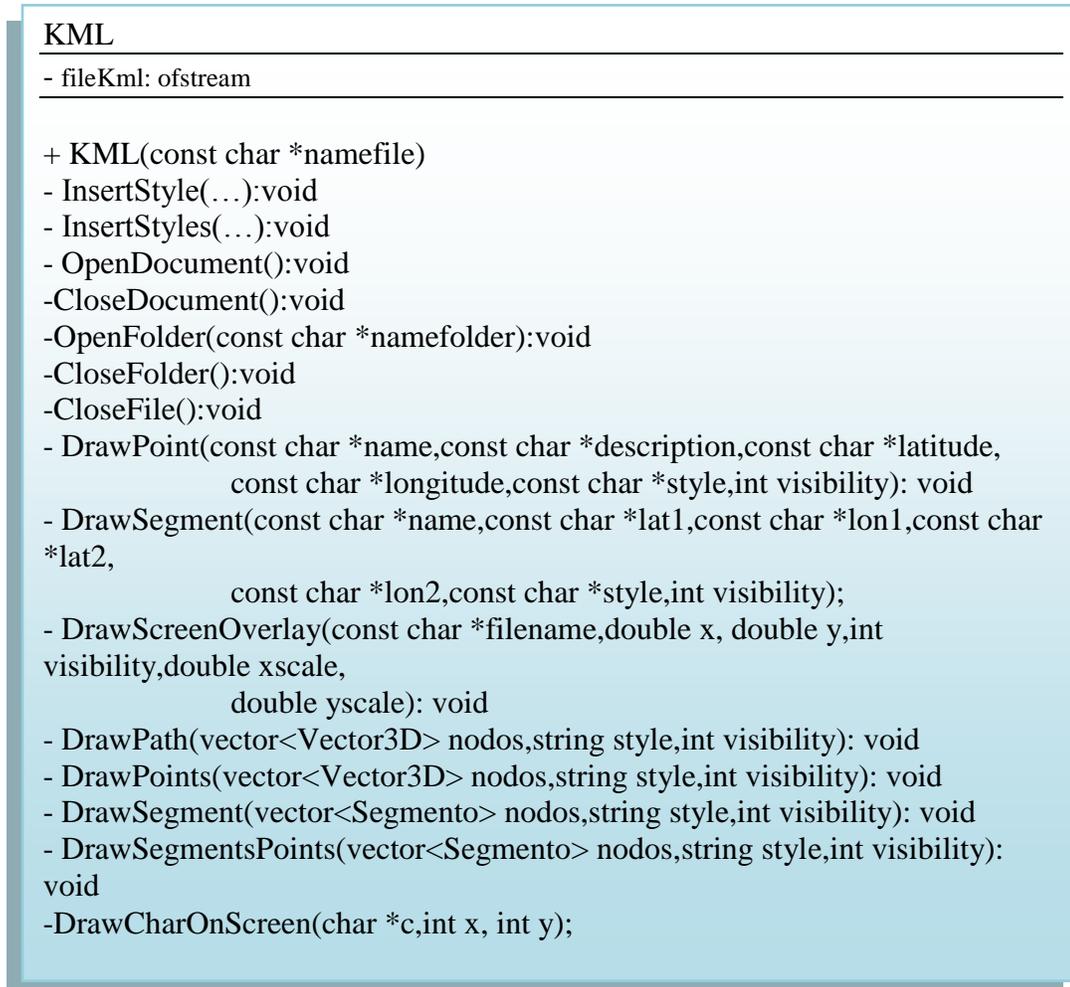


Diagrama 21: Clase kml.

Con esta clase es posible crear todos los elementos que serán mostrados más adelante en el presente informe, pero es fácil observar que es posible extenderla a una librería mucho más completa en lo que respecta a dibujos sobre Google-Earth.

La Figura 35 muestra un ejemplo construido por el módulo de dibujo en formato KML, en la imagen se puede apreciar una ruta coloreada en cada tramo, y cada tramo limitado por dos puntos azules sobre la ruta. Tanto la escala de colores como los demás elementos desplegados en la imagen serán explicados en detalle en la sección de resultados del presente informe.

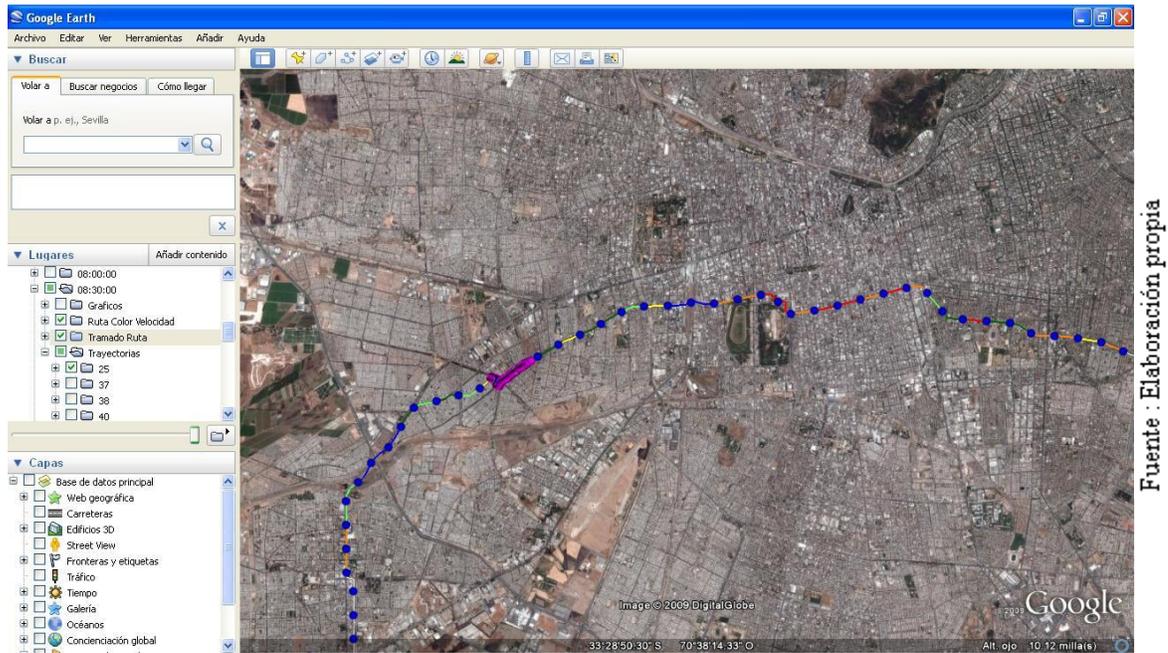


Figura 35: Ruta coloreada según magnitud de velocidades comerciales.

4.4.2 Gráficos automáticos

Una necesidad que apareció durante el desarrollo fue la de visualizar en forma rápida y automática algunos resultados numéricos, como por ejemplo la curva de velocidades por tramo en un periodo. Para esto, se tomó la decisión de buscar una librería en C++ para gráficos, capaz de dibujar en algún formato de imagen (.bmp, .jpg, .png..etc) de tal forma de poder visualizarlas sobre la interfaz de Google-Earth.

ChartDirector

Después de una búsqueda sobre la web se encontraron variadas librerías que cumplían estos requerimientos, pero se optó por tomar la librería ChartDirector¹⁹ por proveer una interfaz muy sencilla para realizar los gráficos que se necesitaban. La Figura 36 muestra un ejemplo de un grafico generado por esta librería. Es importante notar que ChartDirector no es una librería gratuita y se utilizó su versión trial que agrega a los gráficos una leyenda en la parte inferior. Para un futuro software profesional se recomienda comprar esta librería o utilizar otra gratuita.

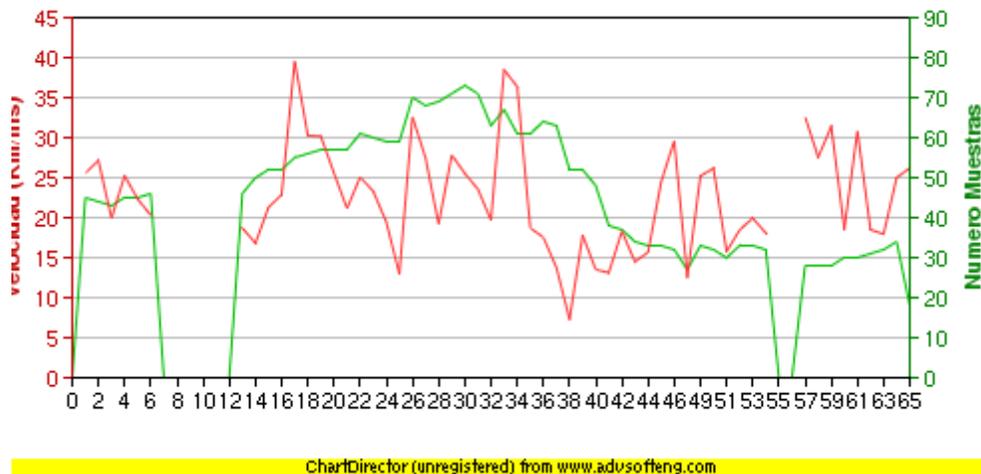


Figura 36: Ejemplo de gráfico creado con la librería ChartDirector.

¹⁹ <http://www.advsofteng.com/>

Gnuplot

También durante el periodo de desarrollo apareció la necesidad de construir otro tipo de gráficos, uno capaz de apoyar la visualización de las velocidades globalmente por servicio. Este gráfico consiste en pintar la matriz formada por las velocidades por tramo por periodo, con una escala de colores que depende de la velocidad.

La Figura 37 muestra un ejemplo de este último gráfico. La interpretación y significado del gráfico será explicado en detalle en el capítulo de resultados

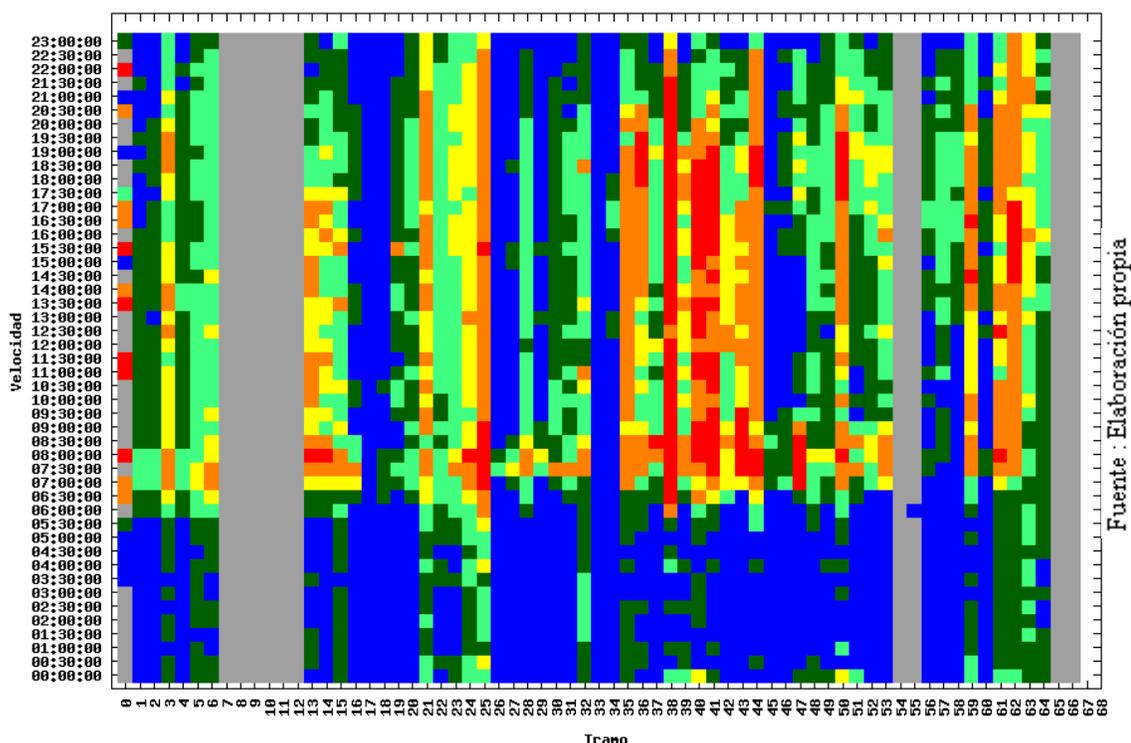


Figura 37: Representación de las velocidades medias por tramo (abscisa) y por periodo (ordenada) creada utilizando gnuplot desde el software en C++.

El problema de este tipo de gráficos fue que no se logró generar a través de la librería `chartDirector`, para lo cual se optó por utilizar otra herramienta, `Gnuplot`²⁰, esta herramienta permite generar casi cualquier tipo de gráfico desde la línea de comandos de Linux. Y para su utilización se creó una clase llamada `toGnuplot` que contiene métodos para crear cada uno de estos gráficos.

El Diagrama 22 muestra los métodos que componen la clase `toGnuplot` en la actualidad, de la misma manera que la clase `Kml`, se puede extender a medida que se requiera algún tipo de gráfico especial anexándolo como en un nuevo método.

²⁰ <http://www.gnuplot.info/>

toGnuplot

+ PaintDiscretScaleMatrix(...):void
+ PaintContinuosScaleMatrix(...):void

Diagrama 22: Clase toGnuplot.

La temática de cada método consiste en crear un archivo de datos y otro con los comandos para graficarlos y ejecutar desde la consola (utilizando la función de C++ system) el comando gnuplot, una vez ejecutado se eliminan los archivos y se obtiene un archivo en formato imagen, de preferencia .png con el gráfico resultante.

A continuación se muestra un ejemplo de estos gráficos insertados automáticamente en un archivo .kml a través del método InsertScreenOverlay de la clase KML. Tal como muestra el Diagrama 23 en la esquina superior derecha tenemos el gráfico creado usando Gnuplot y al lado izquierdo el gráfico creado con ChartDirector.

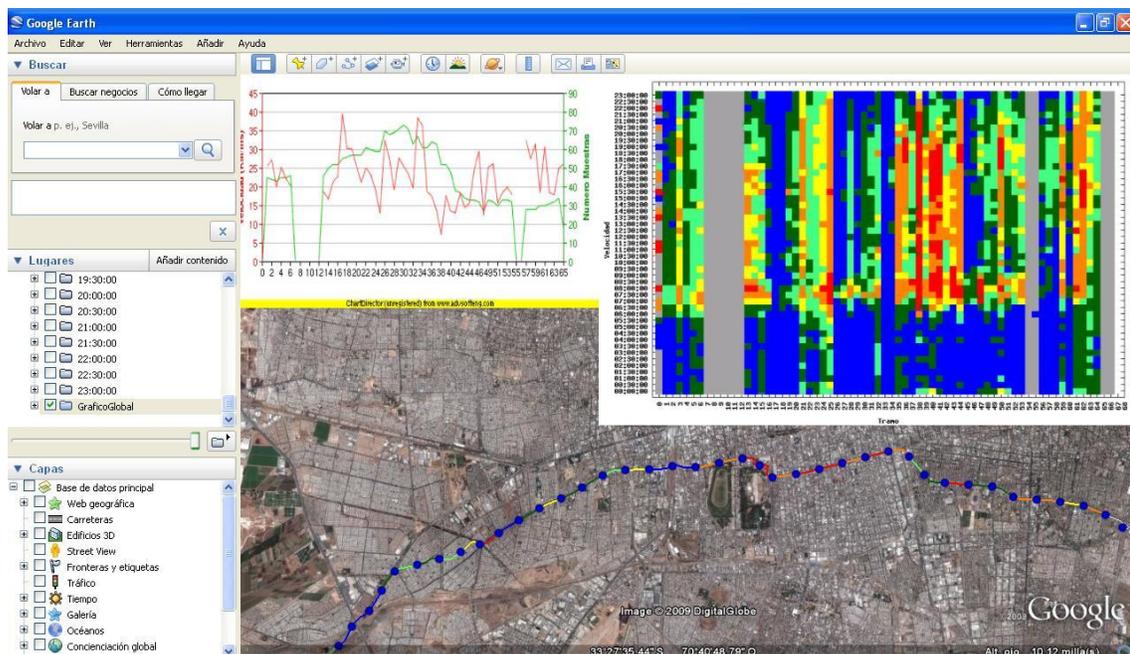


Diagrama 23: Ejemplo de inserción de gráficos automáticos.

4.4.3 Manejo de colores

Un importante elemento en las interfaces de este trabajo son los colores, a través de ellos es posible entender mejor y más fácil los distintos resultados. Uno de los desarrollos hechos en este trabajo consiste en manejar distintas escalas de colores en distintos formatos que los representan. La razón por la cual fue necesario implementar esta clase es porque Google-Earth en su formato kml no provee este tipo de características y Gnuplot al manejar esto en forma automática no entrega la flexibilidad suficiente para algunos problemas, a continuación se detalla la implementación de cada uno de estos desarrollos.

Escala continua

En un principio en este trabajo para identificar mejor las magnitudes de los resultados se tomó la decisión de utilizar escalas de colores para pintar los puntos o segmentos dibujados en Google-Earth.

Las escalas continuas son utilizadas para representar valores reales, por ejemplo, si se tiene una velocidad de 20 km/hr donde la muestra tiene un mínimo de 0 km/hr y un máximo de 40 km/hr el color que representa a los 20 km/hr debería ser la mitad en la escala de color elegida y así para cualquier valor el color que corresponda a la proporción en la escala.

Para entender un poco mejor las escalas a continuación la Tabla 3 muestra las escalas más utilizadas en gráficos.

Método	Colores
RGB	Mínimo=rojo, medio=verde, máximo=azul
BGR	Mínimo=azul, medio=verde, máximo=rojo
RWB	Mínimo=rojo, medio=blanco, máximo=azul
BWR	Mínimo=azul, medio=blanco, máximo=rojo
RWG	Mínimo=rojo, medio=blanco, máximo=verde
GWR	Mínimo=verde, medio=blanco, máximo=rojo
GWB	Mínimo=azul, medio=blanco, máximo=azul
BWG	Mínimo=azul, medio=blanco, máximo=verde
BlkW	Mínimo=negro, máximo=blanco
WBlk	Mínimo=blanco, máximo=negro

Tabla 3: Métodos de escala de colores continuas

El Diagrama 24 muestra un ejemplo con el caso RGB, donde se aprecia como disminuyen o aumentan las tres componentes a medida que aumenta el valor.

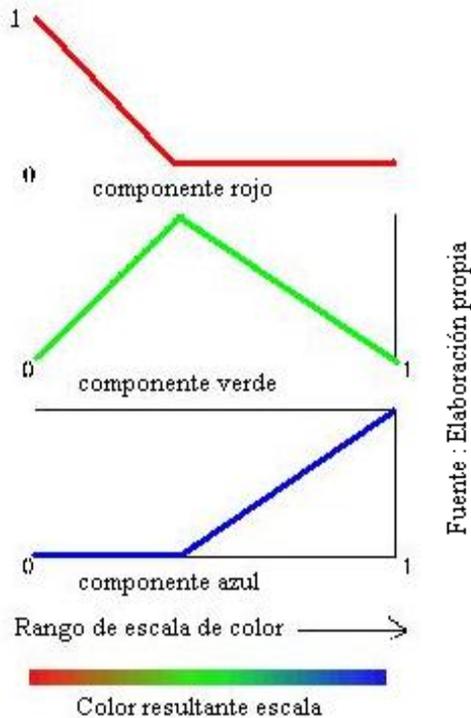


Diagrama 24: Método RGB

Para el manejo de estas escalas se creó una nueva clase llamada ColorScale, la cual contiene métodos que proveen la obtención del color correspondiente según una escala definida por un mínimo y máximo, en otras palabras el método recibe como parámetros los rangos mínimo-máximo y el valor buscado para entrega el color resultante

Escala discreta

Después de un primer periodo de uso de la interfaz los investigadores consideraron no utilizar escalas de colores, sino que asignar colores fijos para ciertos rangos de valores, por ejemplo para las magnitudes de velocidades entre de 0 km/hr a 10 km/hr el color es rojo y de esta manera para distintos índices de resultados se definieron estas tablas de asignación de colores.

La implementación se hizo agregando métodos a la clase ColorScale por cada una de estas tablas de asignación de colores, es decir, para los colores de las velocidades se entrega el valor de la velocidad como parámetro luego el método busca a que rango corresponde y entrega el color del rango al que pertenece.

Formatos

La clase ColorScale también contiene métodos para manejar, hasta el momento, dos tipos de formatos para codificar colores, estos son el formato utilizado por Gnuplot que es una combinación lineal de las componentes RGB (Rojo, verde y azul) y el formato utilizado por Google-Earth es hexadecimal

El formato usado por Gnuplot representa los colores a través de un único número entero formado por la siguiente combinación lineal:

$$\text{COLOR} = \text{ROJO} * 65536 + \text{VERDE} * 256 + \text{AZUL}$$

Y por otra parte Google-Earth utiliza un formato muy similar al estándar de HTML pero no igual, el Diagrama 25 a continuación detalla el formato:

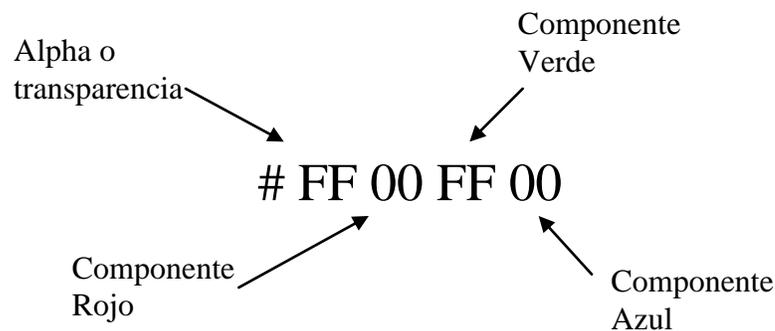


Diagrama 25: Codificación de colores Google-Earth

De esta manera esta clase permite manejar cualquier escala de colores para utilizarlo tanto en los dibujos en Google-Earth como también para gráficos usando Gnuplot. El Diagrama 26 muestra el contenido de esta clase. Hasta el momento sólo se ha implementado la escala BGR.

ColorScale

```
+ BGRScale(double min, double max, double value) static string  
+ SpeedScale(double value) static string  
+ RGB2Hex(int r, int g, int b): static string  
+ Hex2RGB(string value) static string
```

Diagrama 26: Clase ColorScale

5. Resultados

Este capítulo se dividirá en varias secciones las cuales representan diferentes módulos que pueden ser considerados como resultantes de este estudio, para ejemplificar tenemos el módulo de visualización que contiene tanto los resultados visibles sobre Google-Earth como también los gráficos necesarios para poder encontrar rápidamente casos de velocidades muy bajas relacionadas con alguna sección de la ruta o un periodo de tiempo (diagnóstico).

5.1 Visualización

El módulo de visualización está compuesto por tres secciones:

1. Ruta coloreada por magnitud de velocidad
2. Trayectorias de los buses sobre el mapa
3. Trayectorias de los buses sobre grafico distancia-tiempo

5.1.1 Ruta coloreada por magnitud de velocidad

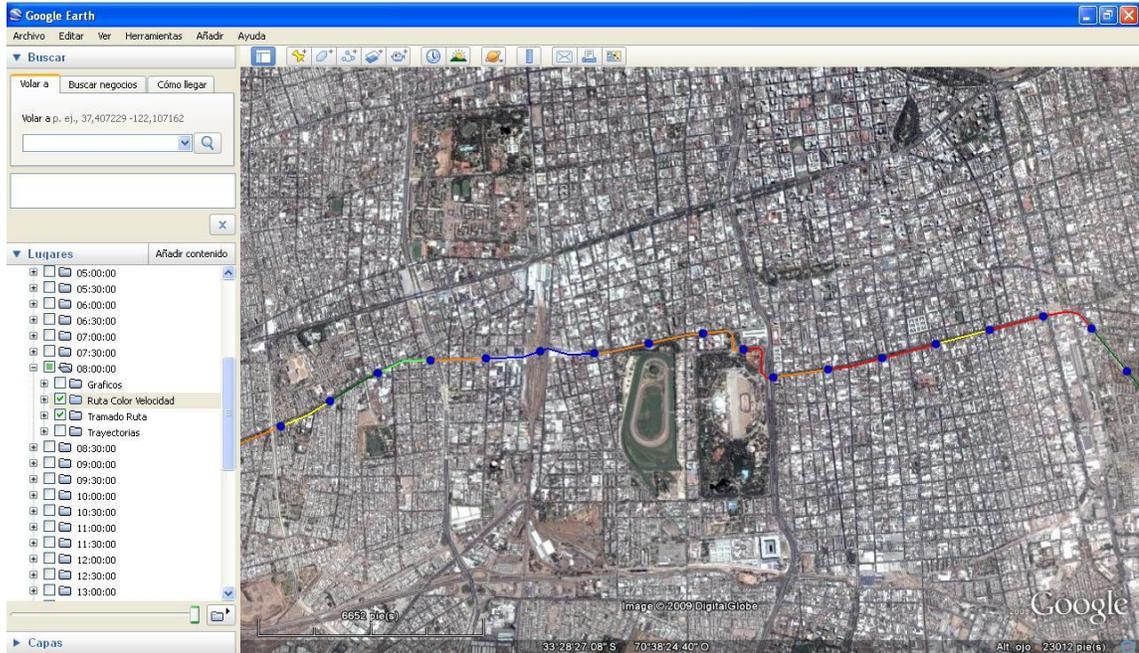
El primer elemento importante que se puede visualizar a través de Google-Earth es una ruta por cada periodo de tiempo del día (cada 30 minutos) dividida en los tramos definidos (cada 500 metros) donde cada uno de estos tramos tiene un color que representa la magnitud de la velocidad de ese tramo en ese periodo.

Para entender la notación de colores la Tabla 4 muestra la equivalencia entre los colores y las velocidades que representan.

Rango Velocidad (Km/hr)	Color
≤ 15	ROJO
>15 y ≤ 19	NARANJO
>19 y ≤ 21	AMARILLO
>21 y ≤ 25	VERDE CLARO
>25 y ≤ 30	VERDE OSCURO
>30	AZUL
Sin Observaciones (no se puede calcular)	GRIS

Tabla 4: Escala de colores usada para las velocidades

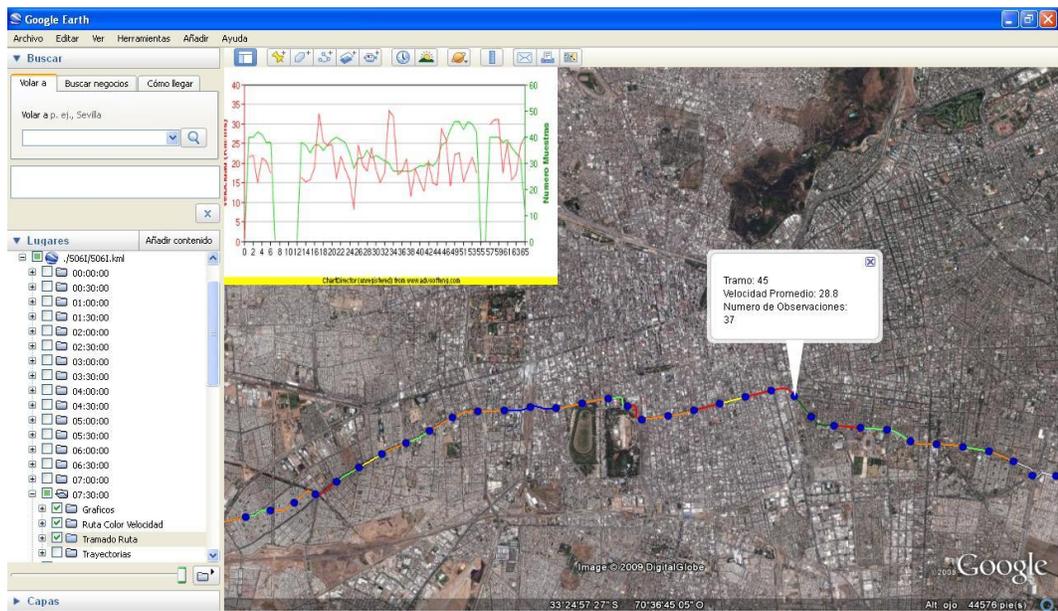
La Figura 38 muestra un ejemplo de estas rutas, donde los puntos azules definen el inicio y fin de cada tramo y la curva que los une representa la ruta entre esos dos puntos con el color que corresponde según la magnitud de la velocidad en ese tramo.



Fuente : Elaboración propia

Figura 38: Ruta del servicio 506, del periodo 8:00 a 8:30.

Utilizando este tipo de visualización se hace sencillo poder revisar y detectar los sectores con mayores problemas en términos de velocidad. Además se construyó un gráfico de apoyo que muestra la curva de velocidades de esta ruta en conjunto con la curva que representa el número de observaciones, es decir, la cantidad de buses que pasaron por ese tramo en ese periodo de tiempo.



Fuente : Elaboración propia

Figura 39: Ruta coloreada, gráfico de velocidad-observaciones, información adicional al hacer click con el mouse en un nodo de la ruta.

Como se puede apreciar en la Figura 39 tenemos este gráfico que en su abscisa tiene el número de cada tramo ordenados de inicio a fin y en su ordenada tiene por un lado las velocidades en cada tramo (curva de color rojo) y por el otro una curva que representa el número de observaciones (curva en verde). Con este último gráfico se apoya mucho mejor el comportamiento de las velocidades a lo largo de la ruta en un periodo de tiempo.

Con todo lo descrito anteriormente se vuelve muy sencillo hacer un análisis de un servicio en particular en cierto periodo de tiempo, pero en términos globales aparece el problema de poder analizar el fenómeno durante todo el día sobre todos los tramos de la ruta. Dado este último problema se construyó un gráfico en dos dimensiones que representa una matriz coloreada con la misma notación ya explicada de las velocidades, esta matriz tiene en la horizontal los números de cada tramo de la ruta, y en la vertical cada periodo de tiempo, de tal forma que cada voxel representa la velocidad comercial en un tramo de ruta en un periodo de tiempo.

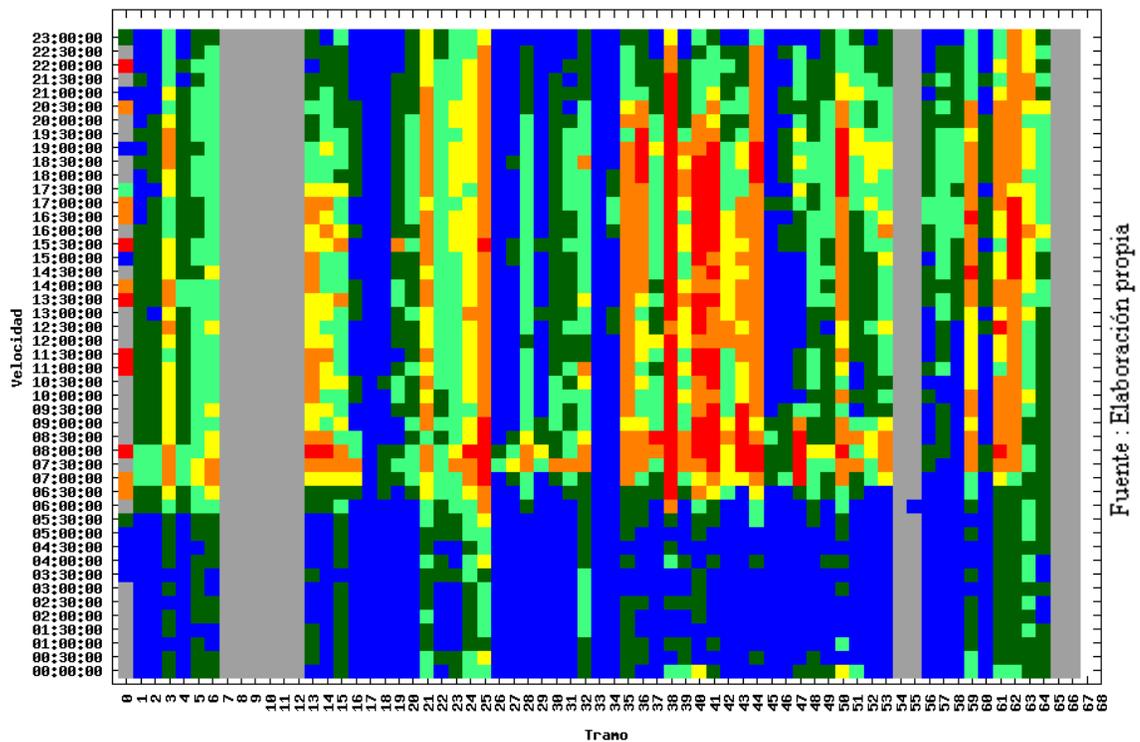
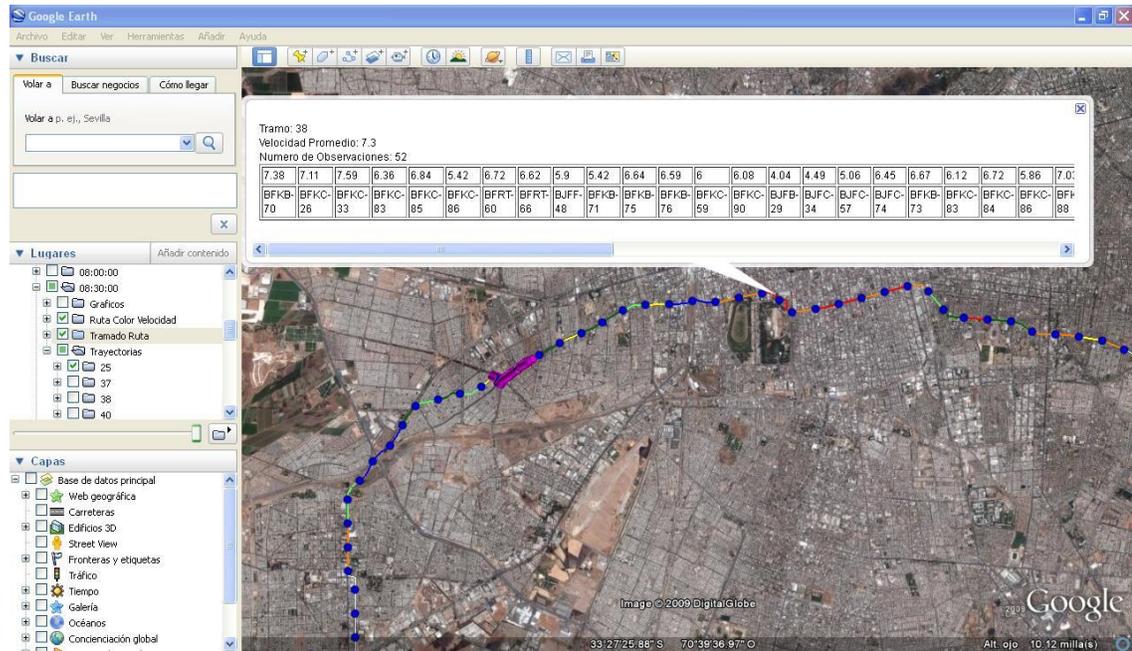


Figura 40: La Gráfico para el servicio 506.

Como se aprecia en la Figura 40 este tipo de gráficos resume muy bien el fenómeno completo, es decir, se puede visualizar en forma rápida las características de cada tramo, por ejemplo, el tramo 38 contiene un gran porcentaje de periodos en rojo indicando que es un tramo de velocidades lentas durante todo el día. También se pueden apreciar los tramos que no tienen observaciones durante todo el día lo que indica que esa ruta sufrió de un desvío en esos tramos durante ese día (en la figura los tramos 7 al 12). Otra cosa que se puede apreciar son los periodos de tiempo; en la figura se ve claramente que en la madrugada las velocidades son muy altas, y en los periodos de punta mañana (7:30 a 9:00) las velocidades decaen considerablemente prácticamente por toda la ruta.

Para apoyar aún más la detección y análisis de problemas en las velocidades comerciales por servicio, se integró información detallada de los buses y sus trayectorias. Se incorporaron dos nuevos elementos al archivo de salida para Google-Earth. El primero consiste en agregar la información de las patentes que participaron en un cierto periodo-tramo y sus respectivas velocidades. La Figura 41 muestra un ejemplo del despliegue de esta información.



Fuente : Elaboración propia

Figura 41: Despliegue de la información de cada patente para el tramo 38.

5.1.2 Trayectorias de los buses sobre el mapa

El segundo elemento que apoya la información en detalle es el dibujo de las trayectorias de los buses, esto permite revisar e identificar si un bus está detenido o donde existen aglomeraciones de emisiones GPS (muchos buses detenidos). Con esto se puede diagnosticar mejor si la razón de que un tramo tenga una baja velocidad comercial es por un semáforo, un paradero de pasajeros o alguna otra razón. La Figura 42 muestra esta opción de visualización, en la imagen se encerró con un círculo en rojo un sector con una gran aglomeración de emisiones GPS y que para ese caso están relacionadas con el semáforo de la intersección de calles.

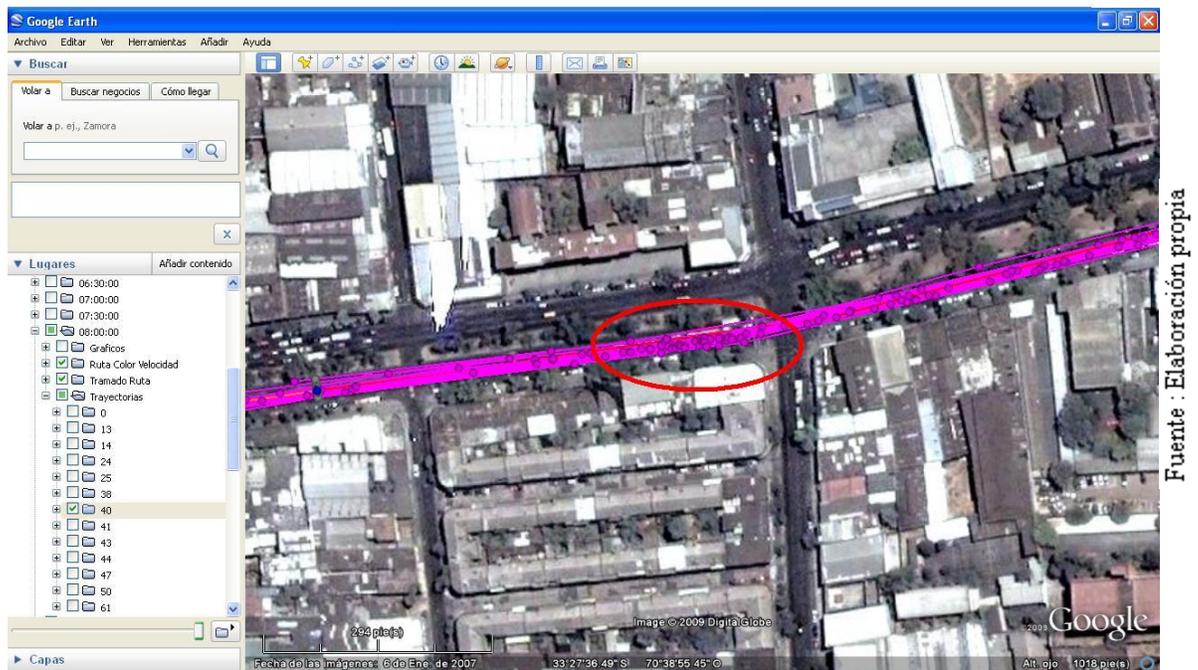


Figura 42: Dibujo de todas las trayectorias de los buses involucradas con algún periodo-sección de la ruta.

5.1.3 Trayectorias de los buses sobre grafico distancia-tiempo

Un tercer y hasta ahora último elemento perteneciente al módulo de visualización corresponde a un gráfico de las trayectorias de los buses en un eje coordenado distancia-tiempo, donde la distancia refiere a la distancia avanzada sobre la ruta, utilizando la proyección de las observaciones GPS sobre la ruta. La Figura 43 muestra un gráfico ejemplo creado por el software, en el se aprecia cada trayectoria de bus que participó en los cálculos de un día. Como se observa, la imagen permite detectar varios elementos muy importantes tales como:

- **Regularidad del servicio:** Esto significa que entre las trayectorias de los buses exista un espacio de tiempo regular.
- **Desvíos:** Tramos en la ruta donde no existen emisiones GPS válidas, las emisiones GPS no válidas han sido marcadas con color azul.
- **Velocidad:** A través de la inclinación de las trayectorias se pueden identificar sectores con velocidades más bajas o más altas que lo normal.
- **Bunching:** Nombre con el que se le conoce al fenómeno donde se aglomeran los buses de un mismo servicio en la ruta.

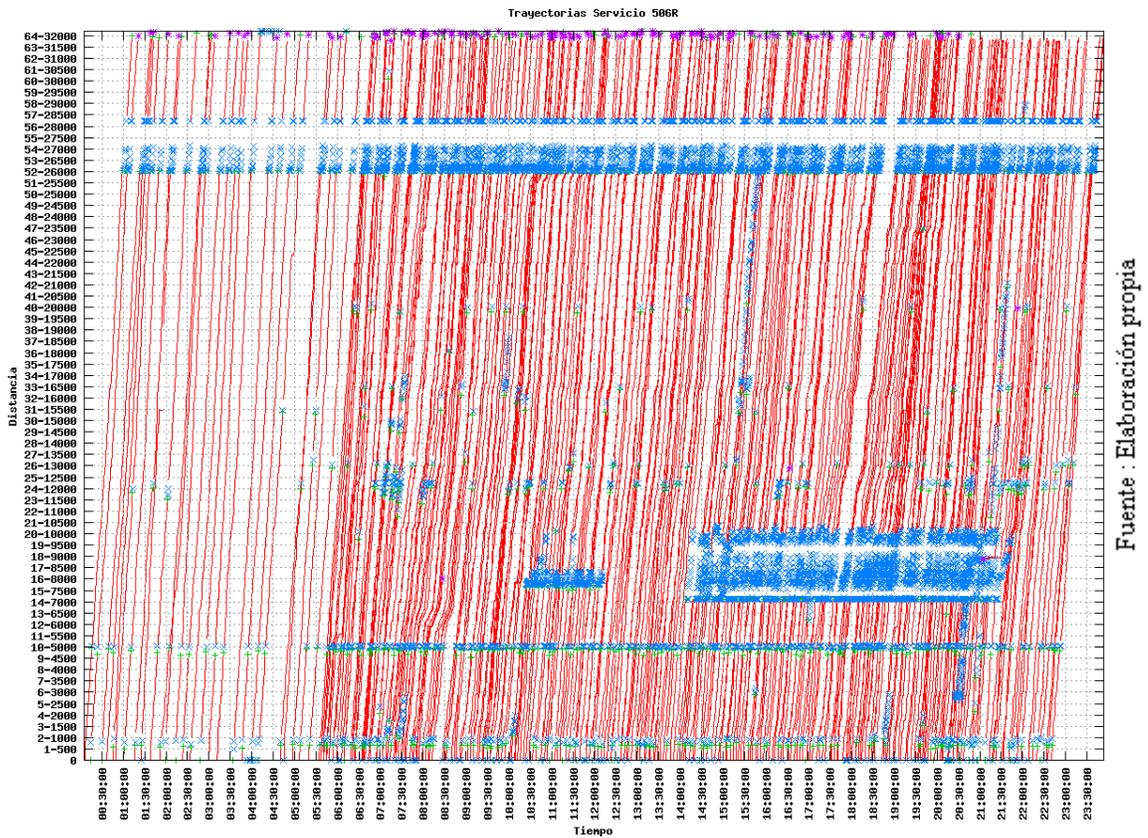


Figura 43: Trayectorias de los buses proyectadas a la ruta (servicio 506).

Cada línea que se aprecia representa la trayectoria de un bus en la ruta 506, los sectores azules representan emisiones GPS consideradas inválidas ya sea por estar muy lejos de la ruta o por tener algún patrón fuera de rango. Los espacios en blanco y azul en la horizontal representan sectores de la ruta con algún tipo de problema, por ejemplo, algún desvío o accidente de tráfico que por consecuencia hicieron que los buses no pasar por ahí durante cierto periodo de tiempo.

También es fácil apreciar sectores donde no hay trayectorias de buses en periodos largos de tiempo, esto muestra periodos de tiempo con baja frecuencia de buses.

5.2 Base de datos

Parte importante de este estudio está orientado al uso real de esta información por lo que se tornó imprescindible poder calcular las velocidades para todos los servicios. Para esto se tomó la decisión de llevar los resultados del software a un formato compatible con su posterior manejo a través de una base de datos o planillas de cálculo.

Cumpliendo con lo anterior el software además de construir un archivo para la visualización en Google-Earth construye un archivo en formato CSV²¹ con los siguientes campos:

1. ID del tramo
2. Distancia en la ruta
3. Servicio
4. Hora
5. Velocidad
6. Número de observaciones

Dado que el software calcula la información sólo para un servicio a la vez se creó un script en Linux que ejecuta el programa para todos los servicios y una vez terminados procesa cada archivo CSV fusionándolos en uno, este último es posteriormente llevado a una tabla en una base de datos para facilitar las consultas posteriores. La Figura 44 ejemplifica el resultado de este proceso.

	id_tramo integer	distonrou double pri	servicio character	hora time withc	velocidad double pri	nobservac integer
17	1	500	101cl	07:30:00	12.9116	28
18	1	500	101cl	08:00:00	13.8597	21
19	1	500	101cl	08:30:00	17.3657	18
20	1	500	101cl	17:00:00	19.9446	12
21	1	500	101cl	17:30:00	17.6759	30
22	1	500	101cl	18:00:00	17.1972	17
23	1	500	101cl	18:30:00	12.4436	21
24	1	500	101cl	19:00:00	13.1942	18
25	1	500	101cl	19:30:00	16.8813	25
26	1	500	101cl	20:00:00	17.561	18
27	1	500	101cl	20:30:00	24	2
28	1	500	101cl	22:30:00	23.3766	1
29	2	1000	101cl	06:30:00	17.7778	12
30	2	1000	101cl	07:00:00	23.0758	18
31	2	1000	101cl	07:30:00	18.2613	30
32	2	1000	101cl	08:00:00	18.9259	22

Figura 44: Base de datos con velocidades para todos los servicios-tramo-periodo.

²¹ <http://es.wikipedia.org/wiki/CSV>

5.3 Evaluación de servicio

Después de revisar los primeros resultados se observó que era necesaria una manera de representar en alguna forma más resumida la velocidad por servicio, de tal forma de poder comparar entre servicios o poder decidir qué servicio presenta mayores problemas para analizarlo primero. Con este objetivo los expertos construyeron un índice que se puede interpretar como una velocidad representativa del servicio en un periodo de tiempo.

$$I_{jk} = \frac{S_R \sum_i \frac{1}{s_{ijk}}}{N_{jk}} \text{ For } s_{ijk} \neq 0$$

i= segmento

j= periodo

k=servicio

Este índice fue integrado al software y permite construir gráficos como los de la Figura 45, donde se observan las velocidades representativas de un conjunto de servicios (eje ordenado) en cada periodo del día (eje abscisa).

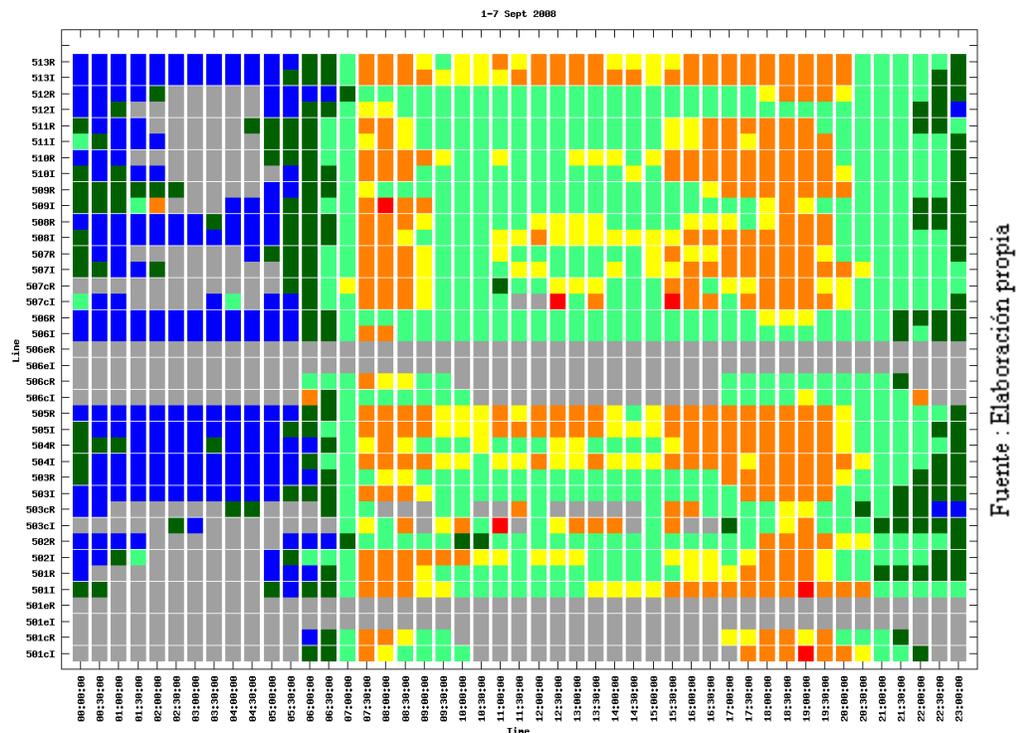
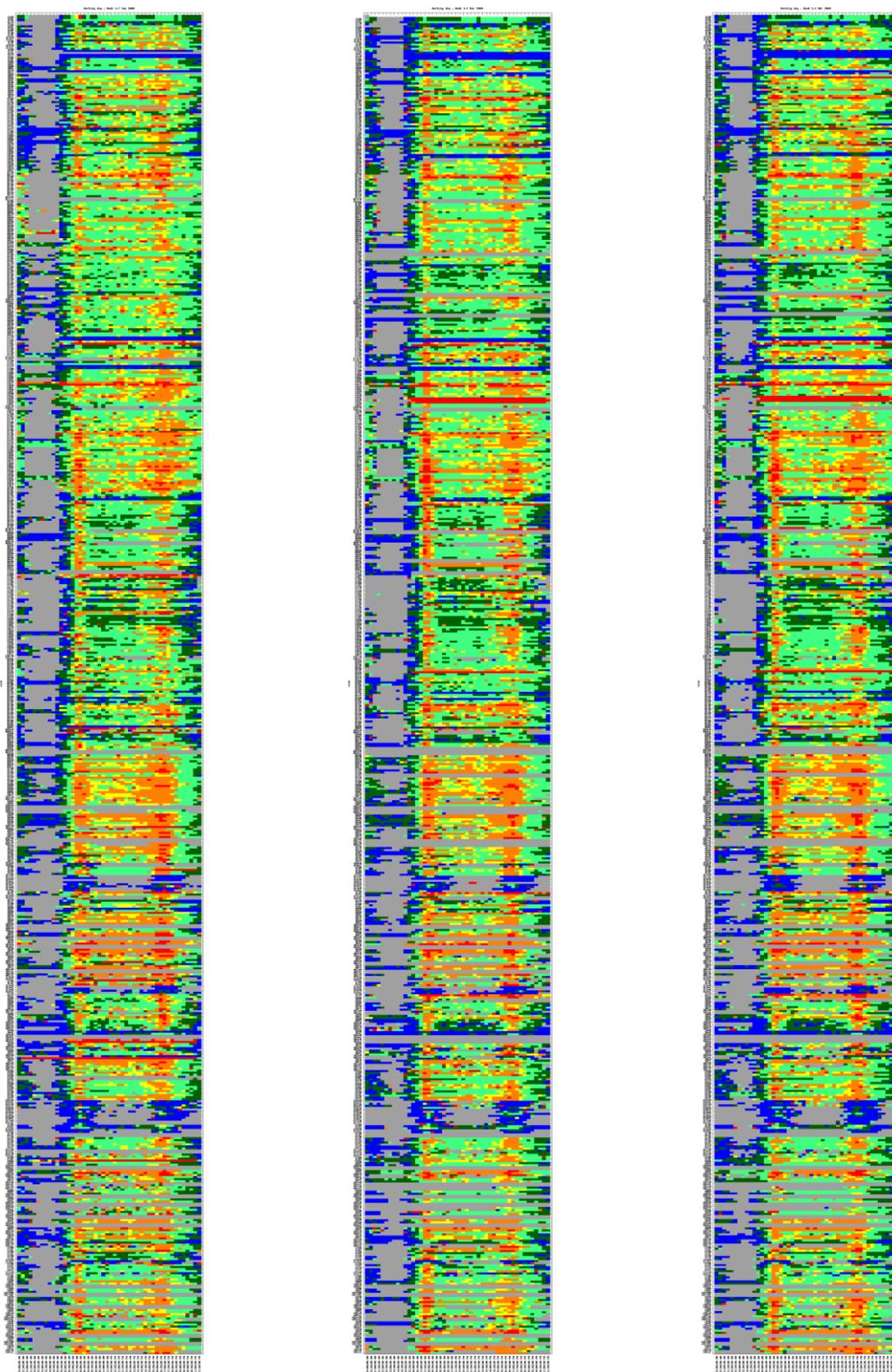


Figura 45: Grafico de velocidades representativas (índice) por cada servicio en cada periodo del día.

El siguiente paso fue el cálculo de todos los servicios, tal como muestra la Figura 46, donde se pueden todos los servicios en tres semanas distintas.



Fuente : Elaboración propia

Figura 46 : Las figuras representan todos los servicios, la imagen de la izquierda corresponde a Septiembre 2008, la del centro a Marzo 2009 y la de la derecha a Abril 2009.

Utilizando este tipo de gráficos rápidamente se puede establecer un diagnóstico general, por ejemplo, se observa que en las primeras columnas, recorriendo de izquierda a derecha, prepondera el color gris lo que significa la ausencia de buses en la madrugada (1:00AM-4:30). De la misma manera como se detectan periodos sin información también podemos observar otros tipos de comportamientos globales, como por ejemplo:

Velocidades comerciales más altas en la madrugada: Junto con el color gris en la madrugada también es fácil apreciar color azul, lo que indica altas velocidades en este periodo de tiempo cuando existe información. Este comportamiento es obvio dado que en ese horario casi no existen inconvenientes de tráfico.

Existencia clara de una punta mañana con su respectiva baja de velocidad: Verticalmente se observan varios patrones, uno de ellos corresponde a una franja roja entre los horarios 7:00AM > 9:00AM. Este muestra las señales claras de la punta mañana donde existen variados fenómenos que bajan la velocidad comercial del sistema en general.

Existencia de una punta tarde que es un poco más marcada que la punta mañana: De la misma manera como se describe la punta mañana, en este gráfico también se aprecia otra franja vertical roja la cual corresponde al periodo de tiempo punta tarde, donde la población vuelve a sus hogares desde sus lugares de trabajo. Un detalle que también se aprecia es que la franja de punta tarde se muestra un poco más gruesa que la de punta mañana.

Una preponderancia de velocidad en el segmento verde-regular: En términos globales se aprecia que el segmento verde-regular es el segmento preponderante en el sistema a excepción de los periodos puntas.

Existencia de servicios sin información: Es importante notar que algunos servicios no tienen información, es decir, se mantienen grises durante todo el día. Esto es debido en su mayoría por el proceso de asignación de buses de Transantiago, el cual considera algunos servicios que recorren la misma ruta como iguales.

Existencia de servicios con una mala velocidad comercial durante todo el día: La principal misión de este método de visualización es la detección de problemas en servicios, en la actualidad es usado para detectar los servicios con mayores problemas del sistema.

Cada uno de los comportamientos descritos anteriormente es posible de diagnosticar sólo observando las características de este gráfico.

5.4 Comparaciones entre servicios

Como se explicó en el capítulo anterior, ahora existe un índice que es capaz de entregar un diagnóstico de un servicio en cierto periodo de tiempo. En este capítulo se presenta una propuesta inicial de comparación visual entre esta información entre distintas semanas.

El primer gráfico que permite dar una idea inicial de cómo se distribuyen las velocidades entre estos segmentos de calidad es un sencillo histograma, tal como muestra la figura a continuación.

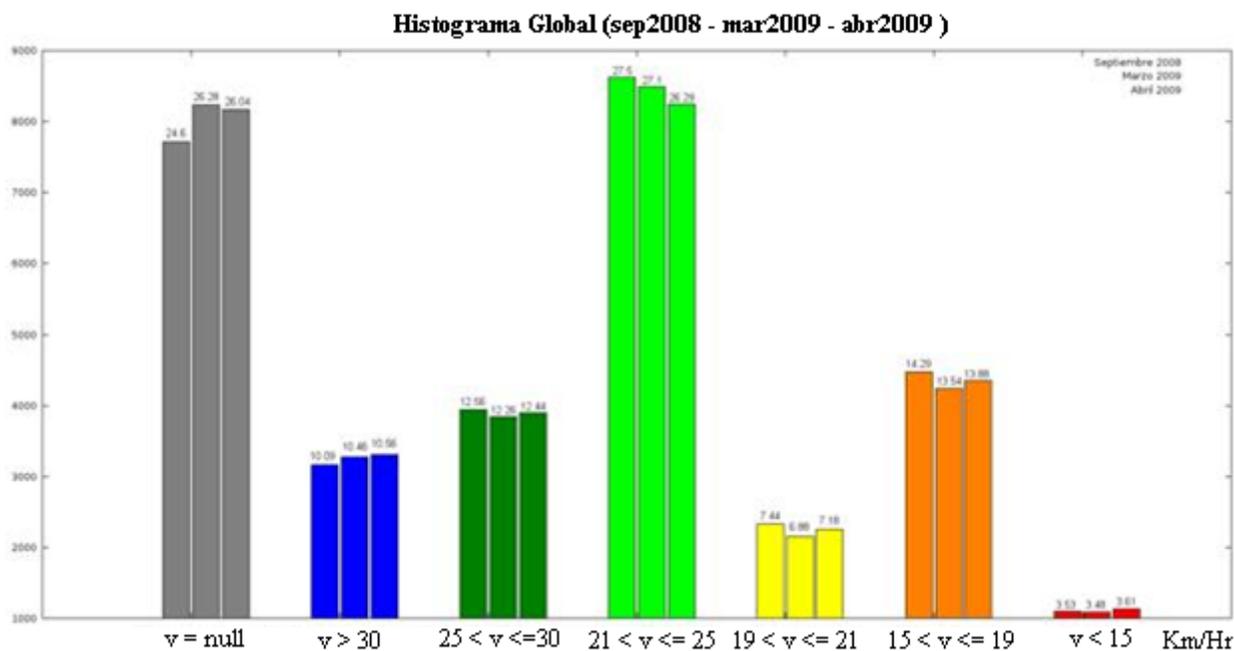


Figura 47: Histograma de las velocidades representativa para las tres semanas procesadas.

De la Figura 47 se puede observar que el segmento preponderante es el verde, lo que significa que en la mayoría del tiempo los servicios se comportan con una velocidad regular, por otra parte se advierte una caída de este segmento a medida que pasa el tiempo y un aumento en el segmento azul, que corresponde a velocidades comerciales excelentes.

Para una identificación más ágil de las diferencias entre un mismo servicio entre distintas semanas de información se propone inicialmente un gráfico de dispersión donde se comparen la medias de los índices entre dos semanas distintas, la Figura 48 y la Figura 49 muestran este tipo de gráficos.

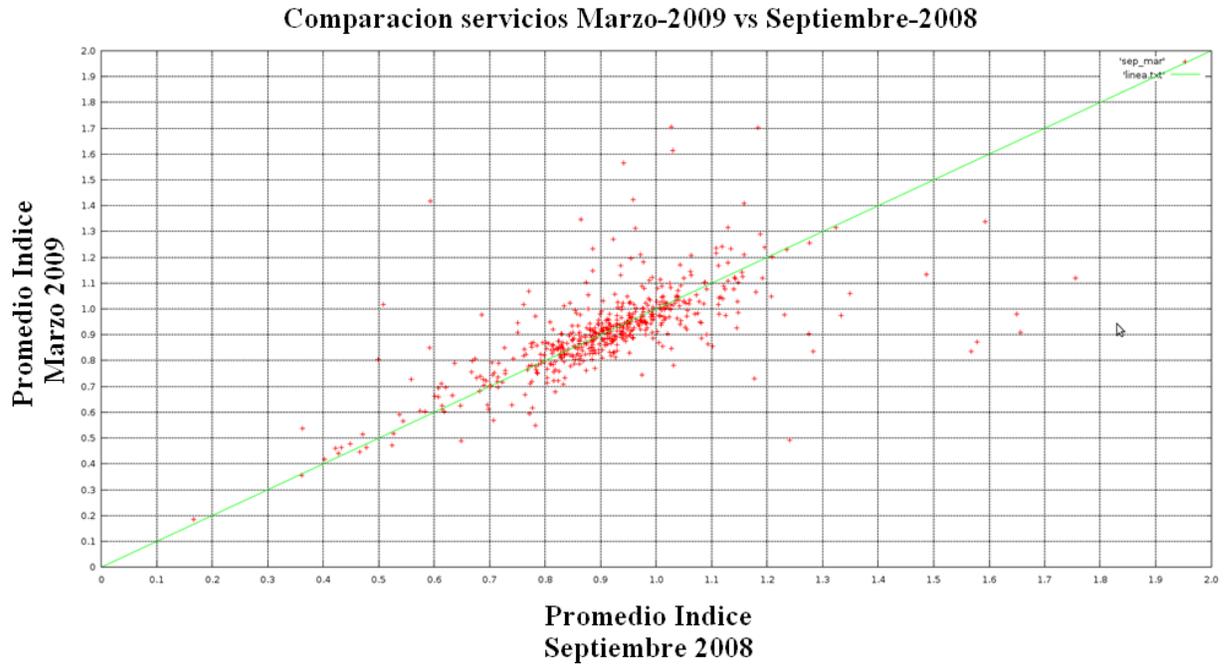


Figura 48: Comparación entre las medias de todos los servicios entre Marzo y Septiembre.

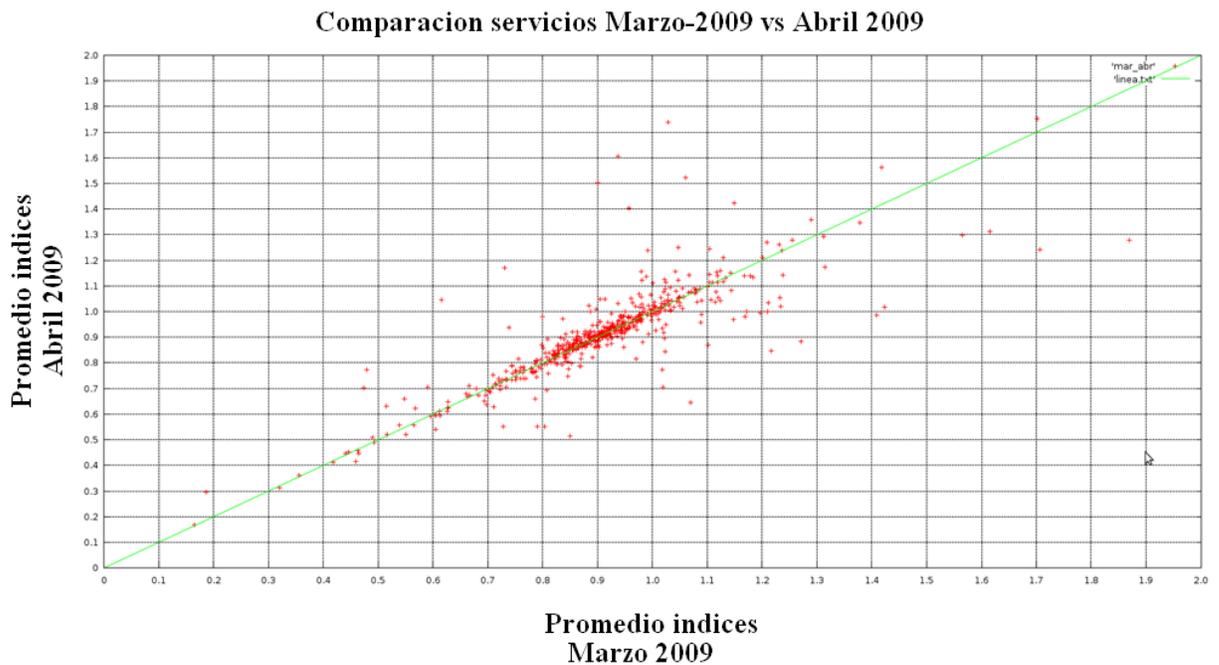


Figura 49: Comparación de las medias de los índices entre semanas similares, Marzo y Abril del 2009.

Cada punto en estos gráficos está compuesto por las medias de los índices de cada semana como coordenadas, es decir, la coordenada del eje X es la media de la semana de Marzo y la coordenada Y la media de la otra semana, de tal manera que un punto que se encuentre justo en la diagonal del grafico significa que tiene el mismo valor para ambas semanas y un punto que esta fuera de la diagonal indica un diferencia entre las semanas.

Con las figuras anteriores se pueden identificar variados tipos de comportamientos tanto locales como globales, como por ejemplo:

Grandes variaciones de velocidades en un servicio a lo largo del tiempo: Cuando se observa un punto muy alejado de la diagonal estamos en presencia de un servicio con una gran variación de velocidad comercial entre estas semanas.

Nivel de dispersión de las velocidades de todos los servicios: Con este tipo de gráficos podemos fácilmente ver donde se concentran los datos.

Sesgos globales en las velocidades entre distintas semanas: Si se observase que la línea de concentración de los datos esta por sobre o por debajo de la diagonal significa que las velocidades en su conjunto muestran un sesgo ya sea más lentos o más rápidos en una semana respecto a otra.

Información fuera de rango: En la construcción de estos gráficos se encontraron valores fuera de rango de las medias, por lo que también es un gran aporte para identificar este tipo de problemas.

Servicios con problemas: De la misma manera que se identifica muy rápido información fuera de rango, también se observan servicios que están en el límite de lo normal, como por ejemplo índices muy bajos y otros demasiados altos.

El gráfico que se muestra en este informe no es el final, dado que en la actualidad está en construcción un gráfico que además muestre el nombre de los servicios sobre los puntos, para los casos que tienen valores poco normales, con el objetivo de poder identificar aun más rápido los servicios que deben ser revisados con más detalle.

6. Conclusiones

En lo que respecta a los objetivos propuestos, todos fueron logrados. Lo más importante de este trabajo es que a partir de la información GPS de los buses se logró construir una herramienta que a través de indicadores sintéticos facilitan la labor de los planificadores de Transantiago.

Uno de los aportes de este trabajo consistió en proveer nuevas metodologías de visualización de datos para el área, como por ejemplo la imagen de la matriz periodo-tramo de velocidades con colores. Esto permite muy rápidamente diagnosticar problemas e identificar si contienen algún patrón temporal o geográfico.

El diagnóstico de situaciones o problemas en la velocidad comercial de algún servicio puede realizarse desde lo más agregado hasta llegar a lo más desagregado, esto permite identificar problemas o comportamientos indeseados ágilmente, y lo más importante permite rankear o clasificar las situaciones desde lo peor a lo mejor, lo que permite ir directamente a las situaciones más problemáticas.

La construcción de la herramienta no solo logró cumplir con los objetivos propuestos sino que también proveyó de una nueva plataforma compuesta de pequeños módulos útiles para otras investigaciones en el área, como por ejemplo, herramientas de visualización, transformación de coordenadas, manejo de información GPS, manejo de base de datos...entre otros.

Una característica importante que fue lograda es la flexibilidad de la herramienta lo que permitió un desarrollo ágil a pesar de los cambiantes requerimientos y lo que permite la fácil integración de nuevos módulos en un trabajo futuro.

Finalmente, la herramienta ha permitido obtener resultados positivos en las dos líneas trabajo, por una parte a permitido a la universidad realizar investigación y por otra ya está apoyando a los planificadores del transporte público a tomar decisiones con mayor conocimiento de los fenómenos en forma ágil y de bajo costo.

6.1 Trabajo futuro

A la fecha la herramienta construida en este trabajo está en una etapa final de desarrollo en lo que refiere a la metodología de estimación de la velocidad, pero a medida que más se trabaja sobre ella siguen apareciendo otros temas que pueden integrarse para mejorar el diagnóstico de problemas en el sistema. A continuación se menciona y describen algunos de ellos.

6.1.1 Velocidades por tramo

Uno de los temas pendientes más relevantes para Transantiago es utilizar esta metodología para estimar la velocidad representativa de cualquier tramo de ruta de Santiago independizándose del análisis por servicio, en otras palabras, estimar la velocidad representativa de un tramo con todos los servicios que por ahí fluyen. Para esto es necesario realizar tres procesos:

1. Calcular que servicios pasan por un tramo cualquiera.
2. Diseñar y desarrollar un nuevo modulo en el software.
3. Diseñar una nueva interfaz para analizar los resultados.

El primer proceso consta de una componente manual, donde se definen los tramos en los cuales que se quiere medir la velocidad y una componente automática donde se encuentran geoméricamente que rutas o servicios pasan por ese tramo.

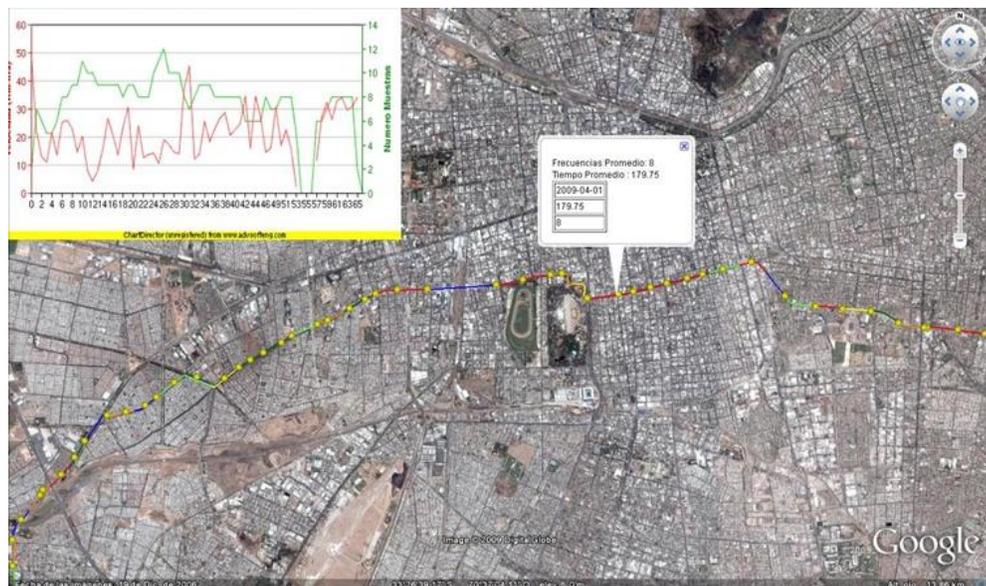
El segundo proceso consiste en crear el software que toma como entrada el tramo y los servicios desde una base de datos y estima las velocidades.

Y finalmente el tercer proceso es para diseñar una nueva interfaz que sea capaz de entregar la información resultante de una manera orientada al análisis que necesita realizar Transantiago.

6.1.1 Paraderos y pasajeros

Un dato hasta ahora no utilizado es la información de las posiciones de los paraderos en la ruta. Se ha discutido la utilidad de integrar estos a la herramienta para poder distinguir mejor las razones de por qué un tramo podría tener una velocidad baja. En otras palabras si se identifica un tramo de baja velocidad pero existe un paradero de mucha afluencia, lo más probable es que la razón de la baja velocidad sea por la subida de pasajeros a los buses y no por alguna otra razón del tráfico vehicular. La Figura 50 muestra como en la actualidad ya está integrado un módulo que es capaz de obtener y relacionar los paraderos con las rutas para un futuro análisis.

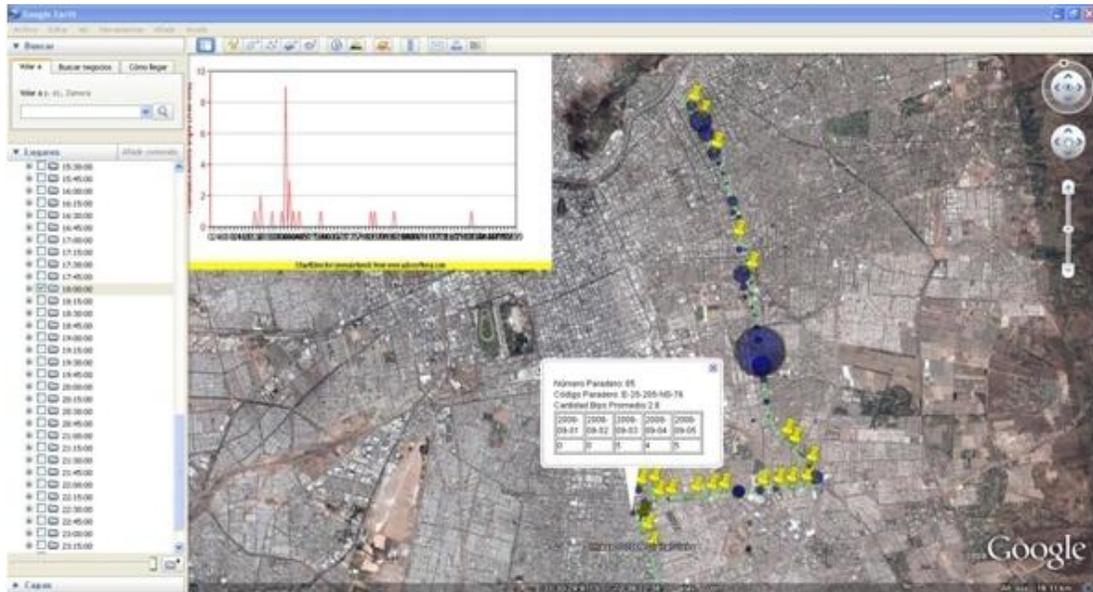
Por otra parte también existe ya una herramienta capaz de estimar las subidas de pasajeros por paradero. Esta información también puede ser muy útil en el futuro para apoyar el diagnóstico de situaciones conflictivas. La Figura 51 muestra un ejemplo de esta otra herramienta que utiliza los mismos módulos que la del presente trabajo, sólo faltaría una integración entre ambas.



Fuente : Elaboración propia

Figura 50: Los puntos amarillos representan la existencia de un paradero en esa posición.

En la Figura 51 los círculos representan los paraderos con subidas en ese periodo de tiempo y su tamaño esta en proporción a la cantidad de subidas.

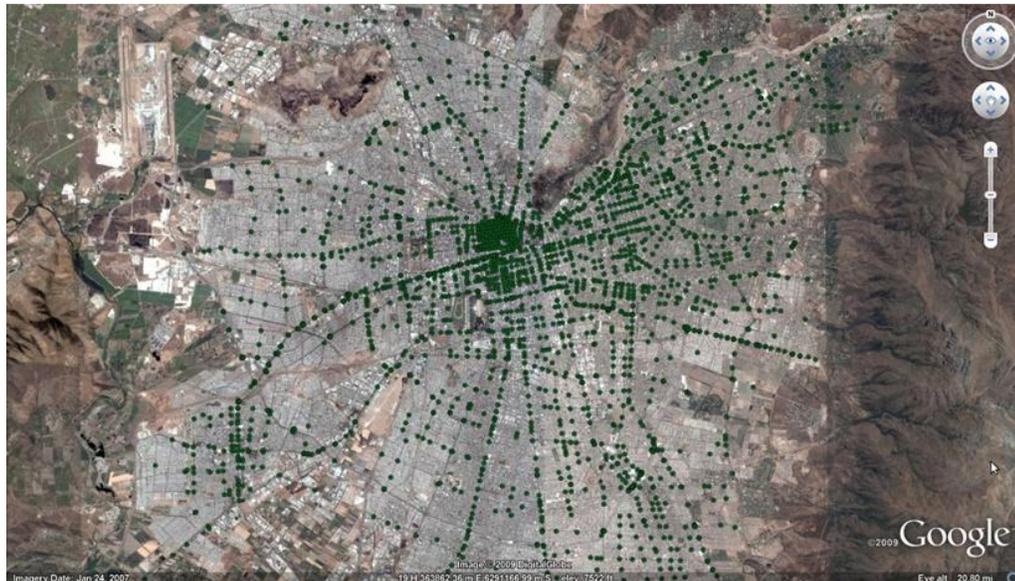


Fuente : Elaboración propia

Figura 51: Subidas de los pasajeros en los paraderos donde fueron realizadas.

6.1.2 Semáforos

La unidad operativa de control de tránsito (UOCT) también proporcionó información para la investigación. Esta consiste en la posición de todos los semáforos que están instalados en Santiago. Esta información aún no es integrada a la herramienta de velocidades, pero cuando lo esté, será otra información adicional para discriminar y diagnosticar mejor las razones del por qué algunos tramos tienen bajas velocidades comerciales. La Figura 52 muestra esta información ya pre-procesada.

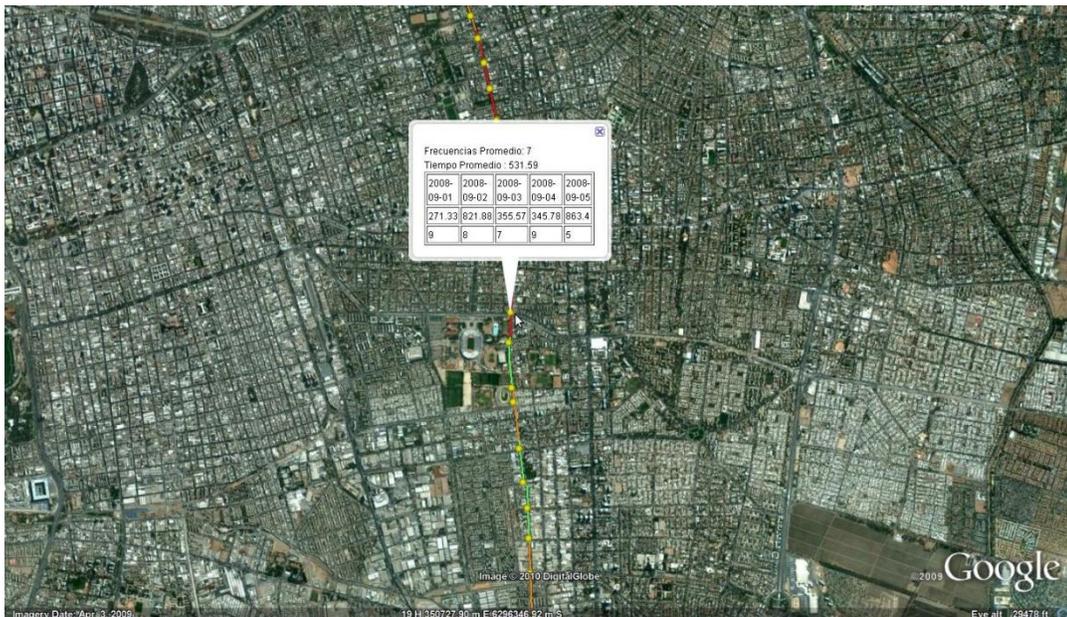


Fuente : UOCT

Figura 52: Posiciones de los semáforos. Cada punto en el mapa es un semáforo

6.1.3 Frecuencias

Un cálculo adicional que fue integrado a la herramienta de este trabajo es la obtención de las frecuencias de los buses y los intervalos de tiempo entre un bus y otro en cada paradero de la ruta. Este módulo fue creado para satisfacer las necesidades de otra investigación, pero en la actualidad existe la opción de que sea utilizada para ayudar o apoyar un poco más el diagnóstico de problemas en las velocidades comerciales, para esto es necesario diseñar y desarrollar un modo de mezclar esta información.



Fuente: Elaboración propia

Figura 53: Frecuencias e intervalos de tiempo de buses por paradero.

7. Bibliografía

- [1] Urs Ramer, "An iterative procedure for the polygonal approximation of plane curves", *Computer Graphics and Image Processing*, 1(2), 244-256 (1972)
- [2] David Douglas & Thomas Peucker, "Algorithms for the reduction of the number of points required to represent a digitized line or its caricature", *The Canadian Cartographer* 10(2), 112-122 (1973)
- [3] John Hershberger & Jack Snoeyink, "Speeding Up the Douglas-Peucker Line-Simplification Algorithm", *Proc 5th Symp on Data Handling*, 134-143 (1992).
- [4] Greaves, S.P. and M.A. Figliozzi. Collecting Commercial Vehicle Tour Data with Passive Global Positioning System Technology Issues and Potential Applications. *Transportation Research Record: Journal of the Transportation Research Board*, No. 2049, Transportation Research Board of the National Academies, Washington, D.C., 2003, pp. 158–166.
- [5] Hudson, M. and G. Rhys-Tyler. Using GPS data to calculate the length and variability of freight vehicle journey times on motorways. 12th IEEE International Conference on Road Transport, 2004, pp. 49-55.
- [6] Jiang, J. and Li, S. (2001) Measuring and analyzing vehicle position and speed data at work zones using global position system. Presented at the 80th Annual Meeting of the Transportation Research Board, Washington DC.
- [7] Muñoz, J. C. and A. Gschwender. Transantiago: A tale of two cities. *Research in Transportation Economics*, Vol. 22, 2008, pp. 45-53.
- [8] Travis M. Smith, Utilizing Google Earth as a GIS platform for weather applications, 22nd International Conference on Interactive Information Processing Systems for Meteorology, Oceanography, and Hydrology
- [9] Ricket,D. Google Maps and Google Earth Integration using KML, *American Geophysical Union, Fall Meeting 2006*.
- [10] Gurushinghe, G.S., T. Nakatsuji, Y. Azuta, P. Ranjiktar and Y. Tanaboriboon. Multiple car-following data with real time kinematic global positioning system. *Transportation Research Record: Journal of the Transportation Research Board*, No.1802,Transportation Research Board of the National Academies, Washington, D.C., 2002, pp. 166-180.
- [11] Quiroga, C.A. and D. Bullock. Travel time studies with global positioning and geographic information systems: an integrated methodology. *Transportation Research Part C*, Vol. 6, 1998, pp. 101-127.
- [12] Storey, B. and R. Holtom. The use of historic GPS data in transport and traffic monitoring. *Traffic Engineering and Control*, Vol. 44, 2003, pp. 376-379.

Links utilizados en el desarrollo

<http://pqxx.org/development/libpqxx/>

<http://www.postgresql.org/docs/>

<http://www.advsofteng.com/>

<http://www.gnuplot.info/>

http://softsurfer.com/Archive/algorithm_0205/algorithm_0205.htm

<http://www.cs.sunysb.edu/~algorithm/implement/cgal/implement.shtml>

<http://code.google.com/apis/kml/documentation/>

<http://tldp.org/LDP/abs/html/>

<http://www.gnu.org/software/sed/manual/sed.html>

<http://www.gnu.org/software/gawk/manual/gawk.html>

<http://www.kdevelop.org/>

8. Anexos

8.1 Definición de requerimientos

Como se señaló en la sección de metodología de trabajo este estudio siguió la línea de desarrollo de un software sin requerimientos iniciales completamente definidos, sólo estaba establecido el objetivo general que es el de obtener información acerca de las velocidades del Transantiago. A medida que se concretaron las reuniones de trabajo se fueron estableciendo ciertos alineamientos para lograr este objetivo, de tal manera que los requerimientos fueron creados, modificados o eliminados en conjunto se analizaban los distintos prototipos.

De esta forma en el presente informe se describen

ID	1
Descripción	Calcular las velocidades medias por tramos y periodos en una ruta
Módulo	Global
Prioridad	1

ID	2
Descripción	Las velocidades deben estar por servicio
Módulo	Global
Prioridad	1

ID	3
Descripción	Se debe seleccionar el mejor método para calcular velocidades de los ya descritos
Módulo	Global
Prioridad	1

ID	4
Descripción	Los tiempos de cálculo no deben exceder un tiempo que no permitan llevar este software a una modalidad on-line
Módulo	Base de Datos,Cálculo
Prioridad	2

ID	5
Descripción	La herramienta debe explicar sus resultados visualmente
Módulo	Interfaz
Prioridad	1

ID	6
Descripción	El software debe realizar los cálculos y generar resultados sin intervención humana.

Módulo	Global
Prioridad	2

ID	7
Descripción	El software debe ser modulable, para reutilizar estos módulos en otros desarrollos involucrados con los datos de Transantiago.
Módulo	Global
Prioridad	2

ID	8
Descripción	El código fuente debe seguir un estándar de codificación para facilitar su mantenimiento y reutilización.
Módulo	Global
Prioridad	2

ID	9
Descripción	El código fuente debe ser documentado.
Módulo	Global
Prioridad	2

ID	10
Descripción	Utilización de herramientas open-source
Módulo	Global
Prioridad	2

8.2 Script SQL

crea_posicionamiento_sep_2008.sql:

```
CREATE TABLE posicionamiento
( id int primary key,
concesionaria varchar(200),
patente varchar(7),
mtc int,
gps boolean,
ignicion boolean,
tiempo_sys timestamp,
tiempo_gps timestamp,
latitud double precision,
longitud double precision,
velocidad double precision);
```

```
COPY posicionamiento FROM 'PATH/posicionamiento.dat' DELIMITERS '|';
```

crea_as_sep_2008.sql :

```
CREATE TABLE ab_sep_2008
(
tipo_bus character varying(100),
un character varying(50),
operador character varying(200),
servicio character varying(50),
patente character varying(10),
sentido character varying(10),
inicio time without time zone,
intervalo_salida time without time zone,
control time without time zone,
fin time without time zone,
distancia double precision,
tiempo time without time zone,
velocidad double precision,
capacidad smallint,
estado character varying(10),
fecha date,
dia smallint,
tipo_dia character varying(100),
periodo character varying(200),
cada_5_min time without time zone,
cada_30_min time without time zone,
fecha_inicio timestamp without time zone,
fecha_fin timestamp without time zone
)
WITH (OIDS=FALSE);
ALTER TABLE ab_sep_2008 OWNER TO postgres;

CREATE INDEX asig_buses_patente ON ab_sep_2008 USING btree (patente);
CREATE INDEX asig_buses_ptt ON ab_sep_2008 USING btree(patente, fecha_fin, fecha_inicio);
CREATE INDEX asig_buses_tfin ON ab_sep_2008 USING btree (fecha_fin);
CREATE INDEX asig_buses_tini ON ab_sep_2008 USING btree (fecha_inicio);
```

Crea_ruta.sql

Creación Tabla Ruta:

```
CREATE TABLE Rutas (  
    tipo VARCHAR(100),  
    id INTEGER,  
    x DOUBLE PRECISION,  
    y DOUBLE PRECISION,  
    servicio VARCHAR(10)  
    etapa INTEGER);
```

```
copy Rutas from 'PATH/rutas.csv';
```