



UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA

DISEÑO E IMPLEMENTACIÓN DEL PROTOTIPO DE UN
SISTEMA CIBER-FÍSICO PARA EL CONTROL INTELIGENTE DE
CARAVANAS DE CICLISTAS

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL ELÉCTRICO

MIKE HUGH HENDERSON CUEVAS

PROFESORA GUÍA:
SANDRA LORENA CÉSPEDES UMAÑA

MIEMBROS DE LA COMISIÓN:
JUAN MANUEL SALAMANCA GARCÍA
ALEXIS YÁÑEZ INOSTROZA

Este trabajo ha sido parcialmente financiado por el proyecto: “Inclusive Cooperative Intelligent Transportation Systems”, Silicon Valley Community Foundation, Cisco University Research Program Found.

SANTIAGO DE CHILE

2020

RESUMEN DE LA MEMORIA PARA OPTAR AL
TÍTULO DE INGENIERO CIVIL ELÉCTRICO
POR: MIKE HUGH HENDERSON CUEVAS
FECHA: 05/06/2020
PROF. GUÍA: SANDRA LORENA CÉSPEDES UMAÑA

DISEÑO E IMPLEMENTACIÓN DEL PROTOTIPO DE UN SISTEMA CIBER-FÍSICO PARA EL CONTROL INTELIGENTE DE CARAVANAS DE CICLISTAS

El presente trabajo de título consiste en la implementación y evaluación de una propuesta de diseño de una nueva versión del prototipo de un sistema ciber-físico para bicicletas inteligentes, en línea con los trabajos desarrollados por Céspedes et al. [25][26][27][28] sobre la coordinación explícita de caravanas de ciclistas, integrando un nuevo esquema de control propuesto por Lara en [29] y empleando la red celular en un esquema de comunicación basado en una base de datos en tiempo real de Firebase.

Así, este documento engloba la adaptación de los actuadores de la interfaz visual y háptica de prototipos presentes en la Universidad de Chile y su integración mediante Bluetooth de baja energía a una aplicación Android que ejecuta un nuevo módulo de control, orientado a coordinar la aceleración, velocidad y distancia entre los/as ciclistas de dichas caravanas. Lo último mediante recomendaciones de aceleración calculadas por un control predictivo robusto basado en modelos (o por sus siglas en inglés RMPC) con intervalos difusos, que son señalizadas a sus usuarios mediante la interfaz adaptada.

El trabajo culmina con un prototipo funcional del sistema, una estimación de los retardos de su esquema de comunicación y una evaluación de su funcionamiento integrado. Esto mediante campañas de medición de retardos de ida y vuelta en un entorno controlado, para 4 operadores de red, y un reporte de desempeño en pruebas de campo en la explanada del parque O'Higgins. De ese modo, se logran verificar limitaciones del esquema de comunicación propuesto, registrando retardos promedio superiores a los estudiados por Lara (factor de 2,46 en el mejor caso), y diferencias entre las tasas de adquisición y actualización efectiva de la ubicación de las bicicletas en el servicio de ubicación GPS empleado (servicios de Google), que vuelven inviable el funcionamiento del esquema de control en esta propuesta.

Finalmente, se complementan los resultados con propuestas de trabajo futuro para ajustar el esquema de control, limitar el uso del GPS, realizar una revisión más exhaustiva de tecnologías de comunicación local y evaluar alternativas para la estimación de distancias entre bicicletas.

TABLA DE CONTENIDO

1. MOTIVACIÓN Y ANTECEDENTES	1
2. DEFINICIÓN DEL PROBLEMA	2
3. OBJETIVOS	3
3.1. Objetivo general	3
3.2. Objetivos específicos	3
4. METODOLOGÍA	4
4.1. Caracterización de línea base	4
4.2. Revisión bibliográfica	4
4.3. Desarrollo del prototipo	4
4.4. Pruebas de funcionamiento	5
5. MARCO TEÓRICO	7
5.1. Velomovilidad	7
5.2. Sistema ciber-físico	7
5.3. Ciudad inteligente	7
5.4. CACC (Cooperative Adaptive Cruise Control)	8
6. ESTADO DEL ARTE	9
7. PROPUESTA DE DISEÑO	18
8. IMPLEMENTACIÓN	21
8.1. Implementación de dispositivos físicos	21
8.2.. Desarrollo de software	24
9. EVALUACIÓN DEL PROTOTIPO	29
9.1. Caracterización del retardo de transmisión de estado	29
9.2. Control de velocidad del líder	34

9.3. Pruebas del prototipo integrado	34
10. RESULTADOS OBTENIDOS	38
11. CONCLUSIONES	39
12. BIBLIOGRAFÍA	41
13. ANEXOS	43
13.1. Implementación del método <i>onLocationChanged</i> .	43
13.2. Implementación de los métodos <i>linkSetting</i> y <i>childAdded</i> .	43
13.3. Implementación del método <i>bleSend</i> .	44

LISTADO DE FIGURAS

1. Figura 1: Diagrama simplificado del prototipo actual.	2
2. Figura 2: Representación de la arquitectura de la propuesta.	18
3. Figura 3: Representación de los módulos funcionales de la propuesta de diseño.	19
4. Figura 4: Representación de los módulos funcionales del prototipo actual.	20
5. Figura 5: Dispositivos físicos.	21
6. Figura 6: Arduino Nano empleada en los prototipos.	22
7. Figura 7: Módulo Bluetooth 4.0 HC-05.	22
8. Figura 8: Controlador DC.	23
9. Figura 9: Step-down empleado en la implementación.	23
10. Figura 10: Flujo de actividades común.	27
11. Figura 11: Actividades diferenciadas por tipo de usuario.	28
12. Figura 12: Interfaz visual y sus modalidades.	28
13. Figura 13: Descripción del trayecto del paquete para medición de RTT.	29
14. Figura 14: Puntuaciones por operador del día 2 de las distintas semanas.	31
15. Figura 15: Puntuaciones promedio por operador (no comparables).	32
16. Figura 16: Velocidad del líder en el tiempo para 3 rapidezces distintas.	34
17. Figura 17: Distancia al líder en función del tiempo para 1 seguidor.	36
18. Figura 18: Aceleración sugerida v/s ejecutada.	36
19. Figura 19: Rapidez seguidor v/s líder.	37

LISTADO DE TABLAS

1. Tabla 1: Organización temática de los trabajos consultados según autor y tema.	9
2. Tabla 2: Especificaciones de periodos monitoreados y periodos analizados.	30
3. Tabla 3: Especificación de equipos celulares empleados para las pruebas.	30
4. Tabla 4: Retardo de envío promedio registrado para un día de cada operador.	33
5. Tabla 5: Retardo de envío promedio registrado para cada operador.	33
6. Tabla 6: Rapidez promedio del líder y desviaciones estándar.	34
7. Tabla 7: Intervalo real de adquisición de posición GPS.	37

1. MOTIVACIÓN Y ANTECEDENTES

El uso de bicicletas como medio de transporte y recreación presenta una diversidad de beneficios individuales y colectivos. La posibilidad de eludir las congestiones del tráfico de automóviles, la actividad física y su bajo costo monetario y ambiental en comparación con otros medios de transporte, son algunas de las razones por las cuales las personas se inclinan actualmente por su uso [1]. En ese mismo sentido, del uso de la bicicleta se derivan una serie de consecuencias valiosas para el desarrollo de la vida en la ciudad, como son la descongestión de la infraestructura automovilística, la reducción en las emisiones de carbono y la mejora de la salud pública.

Lo anterior ha dado origen a una comprensión de la relación entre el fomento del ciclismo y el desarrollo de la ciudad tal que, en determinadas ciudades europeas, la promoción de este medio de transporte ha formado parte de sus políticas gubernamentales por décadas [2]. Un ejemplo de ello es Copenhague, en que sus ciclovías a menudo se sobrecargan y experimentan congestiones en sus intersecciones en reflejo del éxito de dicho fomento. En contraposición, la situación de países como Chile, donde más recientemente se han comprendido los beneficios del fomento en determinados gobiernos comunales (e.g. Puerto Varas, Las Condes y Recoleta) [3][4][5], permite en cierta medida la apropiación de experiencias externas para la planificación de políticas que se anticipen a consecuencias conocidas.

El desarrollo de tales políticas de fomento e incentivo se ha visto acompañado de un conjunto de propuestas tecnológicas que buscan fomentar el ciclismo y reducir los riesgos del uso de bicicletas. Alertas de distancia entre autos y bicicletas [6], visualizaciones de conducta de ciclistas [7], aplicaciones de consulta y préstamo de bicicletas públicas [8] y controladores indirectos de respiración [9] son parte de esas propuestas.

Así, el presente trabajo tiene como antecedente un sistema experimental orientado a proveer un mecanismo de coordinación explícita a caravanas de ciclistas [28], pues se ha observado que estos tienden a realizar tales formaciones sin asistencia o alerta alguna [10], cuestión que significa un riesgo reducible. El sistema actual consiste en una interfaz háptica, un módulo de control cooperativo adaptativo alojado en un microcontrolador y un módulo de comunicación orientado a la coordinación de un ciclista líder y seguidores mediante enlaces ZigBee. Dicho sistema obtiene la ubicación del ciclista por GPS mediante una aplicación Android independiente, que se enlaza por Bluetooth con el microcontrolador, y la velocidad de un contador magnético de vueltas en las ruedas de la bicicleta. Solución restrictiva desde el punto de vista de la cantidad de ciclistas que admite en caravana, del tamaño de la electrónica que debe adherirse a la bicicleta al operar sobre una Romeo V2 y almacenar sus datos de manera local. Esto último implicando que no cualquier bicicleta puede cumplir el rol de líder de la caravana.

Por ello, el motivo de este trabajo es proponer un nuevo prototipo, el cual reúne el módulo de control y comunicación en una aplicación Android, reduce la señalización del enlace Bluetooth a la del control de la interfaz háptica y el tamaño de la electrónica necesaria, opera sobre la red celular 4G, considera un nuevo módulo de control distribuido y predictivo, almacena los datos de sus sesiones en un servicio de nube e introduce flexibilidad en la definición de roles de los participantes de la caravana.

2. DEFINICIÓN DEL PROBLEMA

Actualmente existen dos prototipos [28] en la Universidad de Chile compuestos por módulos de control, comunicación e interfaz humano-bicicleta (o HBI, del inglés *Human to Bicycle Interface*). El módulo de control consiste en una Romeo V2-All-in-one, que aloja la lógica del control y cálculo de la aceleración deseada, integrado a una Socket XBee y dos controladores de motores servo. El módulo de comunicación contempla una antena XBee Pro de 60mW, que permite (en principio) comunicación por la línea de vista (de ahora en adelante LOS, del inglés *Line Of Sight*) de hasta 1500 m para transmitir la identificación, ubicación geográfica y velocidad (contador de interrupciones) de cada bicicleta, y un módulo de comunicación Bluetooth para la adquisición de la posición obtenida mediante GPS desde el celular del usuario. Dichos datos y la aceleración calculada por el microcontrolador se transmiten a la bicicleta delantera y son almacenados en una tarjeta SD (esto en la bicicleta del líder). Finalmente, la interfaz humano-bicicleta está compuesta por un motor DC y piezas móviles, que rozan la palma del ciclista en coherencia con el sentido de la aceleración sugerida, y una interfaz visual, que indica si la aceleración está bajo (verde), sobre (rojo) o dentro (azul) de los márgenes deseados. Necesariamente, el prototipo líder contempla un *Shield* Micro SD para el almacenamiento de los datos experimentales.

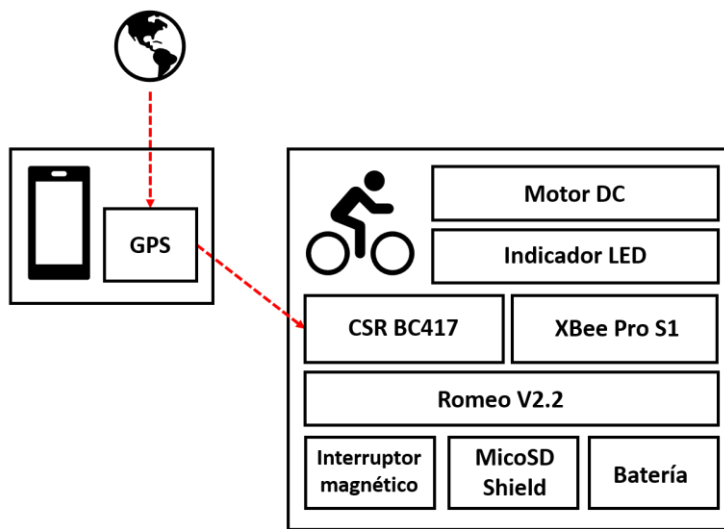


Figura 1: Diagrama simplificado del prototipo actual. Elaboración propia a partir de descripción disponible en [28].

Respecto a las limitaciones de la solución descrita, se identifican costes evitables de instrumentación, específicamente en el empleo de la plataforma V2-All-in-one, que dificultarían la eventual escalabilidad de una solución definitiva y pueden reducirse atendiendo a la tendencia en el estado del arte de emplear *smartphones* como integradores de las funciones de comunicación, medición y cómputo. Por otro lado, el hecho de que el sistema contemple solo al líder como almacenador de la información y lo distinga por la disposición de elementos físicos en su bicicleta resulta en una experiencia de usuario restrictiva, orientada a bicicletas seguidoras/líderes más que a usuarios seguidores/líderes.

En ese mismo sentido, específicamente respecto a los módulos de comunicación y control, hoy se desconoce si la actual red celular (4G) es capaz de soportar las exigencias que impone el esquema de control a las condiciones de comunicación entre los equipos. Factor determinante en la eventual migración de ambos módulos a los celulares de los usuarios o del módulo de control a una nube a la que estos accedan como clientes.

3. OBJETIVOS

3.1. Objetivo general

Diseñar e implementar un nuevo prototipo ciber-físico para el sistema de control inteligente de caravanas de ciclistas, que atienda a las limitaciones identificadas de costo y experiencia de usuario, integre la tendencia del estado del arte al desarrollo de aplicaciones para teléfonos inteligentes, permita evaluar el funcionamiento y desempeño del sistema sobre la red celular y atienda a los errores de interpretación de sus sugerencias de aceleración.

3.2. Objetivos específicos

Caracterizar las condiciones de conectividad bajo las cuales el nuevo esquema de control operará en la nueva configuración, mediante la identificación de sus retardos de transmisión. Integrando el módulo de comunicación en una aplicación móvil, que permita la transmisión de las variables relevantes del sistema mediante la red celular LTE/4G.

Evaluar la viabilidad del sistema de control bajo las modificaciones anteriores. Integrando un nuevo módulo de control en la aplicación que permita el cálculo de la acción de control en el celular de los usuarios o en un servicio de nube.

Adaptar la interfaz humano-bicicleta y el enlace a sus periféricos para admitir la comunicación de sus señales por medios inalámbricos, con previa evaluación de dispositivos electrónicos pertinentes y existentes en el mercado.

Evaluar el funcionamiento integrado de todos los módulos del prototipo y caracterizar sus condiciones de operación mediante un reporte que contemple el retardo introducido por el nuevo esquema de comunicación, la verificación de la efectividad de sus interfaces y una caracterización del funcionamiento del sistema de control.

3. METODOLOGÍA

A continuación, se presenta la metodología conforme a la cual se concretan los objetivos planteados.

4.1. Caracterización de línea base

El diseño e implementación de un nuevo prototipo se realiza en primera instancia identificando el estado actual de funcionamiento del prototipo existente. Para ello se enlistan los dispositivos instalados en las bicicletas y recopilan los documentos que especifican sus características y funcionamiento (*datasheets*). Se identifican de ese modo las condiciones de operación y limitaciones del prototipo actual. Esto para determinar criterios de diseño que permitan dimensionar el nuevo dispositivo a la luz de las condiciones actuales de funcionamiento. Lo mismo se realiza para la aplicación Android.

4.2. Revisión bibliográfica

La justificación de las decisiones de diseño se realiza a partir de fuentes verificables de información. Se distinguen los documentos destinados a indagar en el estado del arte y orientar el diseño, de aquellos para caracterizar la línea base y orientar la implementación.

4.2.1. Estado del arte

Se revisan trabajos recientes que atiendan a tareas relacionadas con el presente trabajo en la literatura especializada. Específicamente se seleccionan trabajos relacionados con la implementación de prototipos, desarrollo de aplicaciones, simulaciones de sistemas de comunicación y coordinación, entre otros. El énfasis está en aquellos considerados más similares al proyecto que orienta el presente trabajo, sus antecedentes y la implementación de prototipos en bicicletas y *smartphones*.

4.2.2. Documentos especializados de dispositivos

En esta categoría se ubica la documentación técnica de los dispositivos que componen actualmente los prototipos y los de aquellos a evaluar para la elaboración de la propuesta de diseño.

4.2.3. Librerías y manuales de desarrollo de aplicaciones

Se emplean librerías en el programa de la aplicación base que son tomadas en cuenta en la nueva aplicación. La integración de los módulos de comunicación y control se realiza conforme a los principios de programación Android, para lo cual existen manuales especializados (<https://developer.android.com/>). Estos describen la programación de aplicaciones que emplean las memorias y periféricos (e.g. antenas y pantallas) de los *smartphones*, para acceder a servicios de telefonía, red, contenido, ubicación, etc. Además, permiten aprovechar los recursos computacionales de los móviles para cálculos y operaciones.

4.3. Desarrollo del prototipo

4.3.1. Definición de la propuesta de diseño

Se realiza y describe un esquema de la propuesta de diseño, en conjunto con documentos que especifican los dispositivos a utilizar, las conexiones entre ellos, su propósito y la justificación técnica de su elección. Además, se realiza un esquema de la ubicación de cada dispositivo en la bicicleta.

4.3.2. Lista y orden de compra

Se enlistan aquellos dispositivos y materiales a utilizar que no se encuentran disponibles actualmente en los prototipos o dependencias respectivas de la facultad. Se consulta a las ofertas para incorporar el costo de dispositivos y materiales en un presupuesto. Una vez aprobado se ejecuta la respectiva compra.

4.3.3. Implementación

Las consideraciones de *software* como de *hardware* propuestas se desarrollan conforme a ciclos iterativos de implementaciones y validaciones de funcionamiento. Esto para cada módulo considerado, emulando el desarrollo incremental de prototipos de *software*.

4.3.4. Desarrollo de la aplicación

Se programa una aplicación Android que implementa comunicación inalámbrica con los periféricos del prototipo, el cómputo del esquema de control y permite la comunicación entre las bicicletas mediante la red celular.

4.3.5. Instrumentación e integración

Se ubican los dispositivos en las partes de la bicicleta detalladas en la propuesta de diseño y se prueba su funcionamiento por separado. Luego se procede a probar la operación de los dispositivos en conjunto.

4.4. Pruebas de funcionamiento

Se realizan pruebas de funcionamiento de los módulos de comunicación de los prototipos por separado de pruebas orientadas a evaluar su funcionamiento integrado. Las de los módulos de comunicación se realizan en la localidad de Linderos, ubicada en la comuna de Buin, Región Metropolitana. Mientras que la de funcionamiento integrado se realizan en la explanada del parque O'Higgins (Región Metropolitana). En todos los equipos participante se instala la última versión de la aplicación.

4.4.1. Pruebas módulo de comunicación

Para los módulos de comunicación se emparejan dos equipos celulares configurando su modo de operación en líder y seguidor respectivamente, procurando que a ambos el acceso a la red sea provisto por la misma compañía. Se procede a iniciar una sesión en que ambos dispositivos permanecen estáticos y se registra el tiempo que tarda la información del líder en ir y volver desde el equipo del seguidor. Esta prueba se realiza por periodos de al menos 8 horas durante días hábiles consecutivos, con los operadores WOM, Movistar, Entel y Claro, durante 5 días consecutivos o hasta que se agote la cantidad de datos asignados a las tarjetas disponibles. De este modo se extraen estimaciones del tiempo de ida y retorno de los mensajes en la caravana, para luego derivar estimaciones del tiempo de viaje entre bicicletas de los paquetes del prototipo y generar una segmentación de los porcentajes de paquetes con retardos de envío menores a 0,25 s, 0,5 s y 1 s. Las pruebas se realizan en Linderos, Chile (33°46'11.9"S 70°43'28.8"W).

4.4.2. Pruebas de funcionamiento integrado

Por otro lado, para las pruebas de funcionamiento integrado se realiza una evaluación del indicador de velocidad de líder y pruebas con un seguidor. La del indicador de velocidad es realizada una vez para cada una de las tres rapidezces de líder disponibles, esto por un voluntario designado como tal. La del control de espaciamiento es realizada con un voluntario adicional

durante dos días consecutivos. En ambas se realiza un trayecto de ida y vuelta por la totalidad de la explanada del parque O'Higgins, procurando obedecer las recomendaciones de aceleración de la aplicación y probando las distintas configuraciones de espaciamiento posibles, considerando la rapidez de líder intermedia entre las disponibles. De estas pruebas se adquieren tanto la velocidad real efectuada por el líder, la evolución temporal del espaciamiento entre ciclistas y las aceleraciones efectuadas por y sugeridas al ciclista seguidor.

5. MARCO TEÓRICO

A continuación, se presenta un conjunto de conceptos con cuya definición se persigue caracterizar el ámbito en que se inscribe el presente trabajo, especialmente en relación con el estado del arte al cual busca contribuir. Los términos escogidos obedecen a nociones que se presentan explícita e implícitamente en la revisión de los trabajos discutidos en la sección Estado del arte. Se aborda el concepto de velomovilidad por su estrecha relación con las motivaciones del presente trabajo. La noción de sistema ciber-físico es abordado por tratarse de la categoría más precisa conocida para referirse al objeto concreto de este trabajo. También se precisa lo que se entiende por ciudad inteligente, por tratarse del supuesto y horizonte tecnológico que las soluciones estudiadas tienen en común. Finalmente, se caracterizan de modo general los sistemas de control cooperativos y adaptativos, pues proveer una plataforma experimental que permita la implementación y evaluación de una alternativa a estos es el objetivo general del presente trabajo de título.

5.1. Velomovilidad

Se trata de un concepto que describe “una preocupación específica por las materialidades provenientes de las tecnologías del ciclismo, sus prácticas y de los sistemas que constituyen y son constituidos por esas materialidades y prácticas” [11]. Ha sido planteado como “una herramienta teórica para entender la complejidad del ciclismo y aquellas relaciones de poder conectadas a las políticas de planificación del transporte” [12]. En ese sentido, se trata de más que el uso o producción de unos dispositivos específicos con una orientación determinada, como son los sistemas ciber-físicos. En la sección Estado del Arte se verá qué diversidad de desarrollos tecnológicos se han visto inspirados por el ámbito de la velomovilidad.

En términos generales, se trata de un término relacionado a los estudios en torno al ciclismo y la movilidad que sucede mediante bicicletas [13].

5.2. Sistema ciber-físico

Es un término empleado para referirse a aquellos sistemas cuyas funciones son realizadas por una componente cibernética, esto es, un determinado arreglo de componentes electrónicos que implementa niveles de programación específicos, y componentes que interactúan directamente con su usuario. El presente trabajo consiste en el diseño y desarrollo del prototipo de uno de estos sistemas, orientado a la coordinación de caravanas de ciclistas mediante un algoritmo CACC, que sugiere aceleraciones a ser ejecutadas, en última instancia, por un/a ciclista.

En la sección Estado del Arte se presenta un recorrido por una serie de trabajos entre los cuales el desarrollo de este tipo de sistemas son el enfoque central. Se trata de sistemas que bien pueden estar localizados en un medio de transporte como la bicicleta, los automóviles o incluso distribuidos en la infraestructura de una ciudad inteligente.

5.3. Ciudad inteligente

Para efectos de este documento, se entiende por ciudades inteligentes aquellos “lugares en que las tecnologías de la información y la infraestructura, la arquitectura, los objetos cotidianos e incluso nuestros cuerpos se reúnen para abordar asuntos sociales, económicos y medioambientales” [14].

5.4. CACC (Cooperative Adaptive Cruise Control)

Las estrategias de control CACC son aquellas que, mediante módulos de control, comunicación y un modelo de la movilidad de sus vehículos, persiguen el control del manejo de un conjunto de móviles (subsistemas), conforme a un objetivo como, por ejemplo, velocidad constante y distancia óptima entre vehículos. Los sistemas que implementan estos algoritmos son denominados sistemas CACC, cuya estabilidad de interés en este contexto es la estabilidad de línea, esto es, la continuidad de la efectiva comunicación de los estados de sus subsistemas entre ellos [15].

6. ESTADO DEL ARTE

A continuación, se presenta un recorrido por una serie de estudios relacionados con el trabajo que motiva el presente informe. A modo de estado del arte, se abordan reportes e investigaciones en torno a la electro-velomovilidad, con énfasis en aquellos que abordan la instrumentación de bicicletas para medir variables de interés (posiciones, inclinaciones, velocidades, aceleraciones, etc.) y/o establecer redes de comunicación entre esta y otras entidades (otras bicicletas, automóviles, dispositivos de comunicación en la infraestructura vial, redes celulares, etc.). Cabe mencionar que se ha procurado abordar también desarrollos que involucren el uso de *smartphones* como parte de los sistemas implementados. Además, se decide explicitar aquellas tecnologías y estándares más propensos de ser considerados como alternativas para la propuesta de diseño de esta memoria. La Tabla 1 presenta una organización temática de los trabajos consultados, distinguiendo aquellos situados en el paradigma de ciudades inteligentes, en la implementación de sus desarrollos en *smartphones* y aquellos para los cuales se requirió de la instrumentación de una bicicleta.

	Ciudades inteligentes	Smartphones	Instrumentación	Año
I.Tal et al. [19]				2016
O.M.F. Abu-Shark et al. [14]				2016
U. Hernandez-Jayo et al. [4]				2015
Y. X. Zhao et al. [16]				2017
G. Liao et al. [6]				2014
J. Iwasaki et al. [17]				2016
S. Sweeney et al. [7]				2017
C. Portilla et al. [15]				2017
H. Kato et al. [5]				2017
J.A. Brizuela Mendoza et al. [18]				2016
W. Jeon y R. Rajamani [21]				2018
S. Smaldone et al. [22]				2011
L. W. Chen et al. [19]				2018
S. Cespedes et al. [28]				2018

**Tabla 1: Organización temática de los trabajos consultados según autor y tema.
Elaboración propia.**

Un ejemplo de instrumentación es el desarrollado por H. Kato *et al.* [7]. Trabajo que consistió en la simplificación de un sistema de sensores de una bicicleta, orientado a la generación de visualizaciones de la conducta de ciclistas. En una primera instancia, el sistema contemplaba un sensor de velocidad, un codificador rotacional para la inclinación del manubrio y un sensor de inclinación para el marco. Con este arreglo alimentaban un modelo dinámico del movimiento de una bicicleta con el que generaban visualizaciones orientadas a identificar diferencias en el manejo de ciclistas de distintos rangos etarios. Habiendo logrado un prototipo funcional, los autores identificaron que su implementación a mayor escala resultaba demasiado costosa.

Por ello, en su segunda iteración decidieron deshacerse de los sensores independientes, delegando dichas funciones de medición al acelerómetro incluido en un *smartphone* (Xperia), cuya medición del ángulo azimutal, dado por la inclinación del marco de la bicicleta, alimenta un modelo cinemático de trayectoria que genera visualizaciones prácticamente idénticas. El resto de las señales obtenidas mediante el acelerómetro resultaron demasiado ruidosas para ser consideradas

en su estudio. Finalmente, probando su prototipo con 6 ciclistas de distintas edades en 3 trayectorias *slalom* distintas, los autores validan el funcionamiento de su prototipo y logran establecer preliminarmente que ciclistas más viejos tienden a tardar más tiempo en ubicar y fijar el centro de rotación del viraje.

Por su lado, el trabajo de O. M. F. Abu-Sharkh *et al.* [16] se posiciona dentro de un ámbito que reúne el uso de *smartphones*, tecnología de comunicación inalámbrica ZigBee (especificación de protocolos de bajo consumo, basados en IEEE 802.15.4 y orientado a redes inalámbricas de área personal) y consideraciones infraestructurales del paradigma de ciudades inteligentes. Se trata del desarrollo de un sistema para recolección de datos y provisión de servicios orientados a potenciar la experiencia de ciclistas en dichas ciudades.

Mediante la integración de esos tres elementos, implementan sistemas de comunicación en múltiples lenguajes (sobre una red ZigBee ad-hoc de bicicletas), visualización sobre un mapa interactivo (a partir ubicaciones y velocidades determinadas mediante GPS), alerta frente a situaciones de emergencia y recolección de energía mediante un dínamo. Además, destaca en este trabajo la implementación de un manejo adaptativo, mediante software, de las consultas GPS para el uso eficiente de la batería del dispositivo móvil.

Para la implementación de las funcionalidades se desarrolla una aplicación Android. La comunicación en múltiples lenguajes se logra mediante herramientas de reconocimiento y procesamiento de voz de Google, en conjunto con Apertium para las funciones de traducción. La geolocalización, por su lado, emplea mapas obtenidos de OpenStreetMap para generar sus visualizaciones. Se trata de una implementación que contempla la posibilidad de funcionamiento online-oportunista y offline, además de la activación de sus distintas funcionalidades mediante comandos de voz.

Como conclusiones los autores identifican que entre las principales restricciones de su sistema se encuentra el consumo de potencia (todas las funciones se desarrollan en los *smartphones*) y las tasas de transmisión en los procesos de comunicación. La primera logran paliarla parcialmente mediante el dínamo recolector, la consulta de GPS adaptativa y el uso de ZigBee. Sin embargo, las bajas tasas de datos asociadas a esta última terminan limitando significativamente la provisión en tiempo real de los servicios considerados.

El desarrollo de U. Hernandez-Jayo *et al.* [6] es otro caso que aprovecha las bondades de los *smartphones* como herramientas para medición y su alta (si no, creciente) disponibilidad entre la población. Además, en línea con el enfoque de las investigaciones y desarrollos que orientan el presente trabajo, también adopta para la validación de su propuesta un enfoque de simulación y montaje de prototipo.

En este, los autores proponen un sistema de alerta (V-Alert) para usuarios vulnerables de la infraestructura vial, desde una preocupación por su seguridad y el aumento de su fatalidad de acuerdo con estudios de la Comisión Europea para la Movilidad y Transporte. Para ello desarrollan una aplicación que, mediante la comunicación, recolección y procesamiento (en un servidor) de la posición, velocidad y dirección de ciclistas y automóviles suscritos, difunde alertas entre ellos dados ciertos umbrales de distancia considerados seguros. Se trata de un diseño que reposa sobre el marco infraestructural del paradigma de ciudades inteligentes y en cuya aplicación Android el/la ciclista puede escoger entre un rol de seguidor/a o líder de un determinado grupo de ciclistas.

En cuanto a las simulaciones llevadas a cabo, los autores consideran una arquitectura centralizada heterogénea que integra enlaces IEEE 802.11p entre RSU (del inglés *Road Side Unit*, entidades para comunicación 802.11p al costado de la calle) y automóviles, redes WiFi (IEEE 802.11b) entre ciclista líder y seguidores, y LTE (del inglés *Long Term Evolution* o comunicaciones de evolución a largo plazo) entre servidor-líder y servidor-RSU. El objetivo de sus simulaciones es validar la concepción del sistema y poner a prueba su escalabilidad, mediante la medición y análisis de los retardos implicados.

Para las pruebas de su prototipo reducen la cantidad de automóviles, ciclistas y RSUs consideradas, no obstante, las especificaciones de la red empleada fueron prácticamente las mismas. Por ello realizaron su montaje experimental en una intersección de Bilbao (España) empleando una RSU, un automóvil y una bicicleta. Las métricas de interés fueron en este caso la tasa de paquetes recibidos (o *packet delivery ratio*) y el retardo entendido como el tiempo que tarda la notificación desde el automóvil al celular del ciclista.

Los autores validan sus simulaciones, obteniendo en ellas retardos similares a las pruebas empíricas. Empíricamente se miden retardos promedio de 1,0059 [s], sin comprometer la confiabilidad de la aplicación. Además, los autores identifican como principales desafíos la mejora de la interfaz humano-máquina y el sistema de GPS (error ~5 [m]), por tratarse de las principales debilidades del sistema. Concretar pruebas empíricas de mayor magnitud es otro de sus focos declarados.

Otro ejemplo de un sistema ciber-físico es el desarrollado por S. Sweeney *et al.* [9]. En esta publicación presentan un diseño que busca mitigar los efectos de la polución sobre la salud de ciclistas mediante el control indirecto del ritmo de su respiración en áreas en que la contaminación se identifica como crítica. Particularmente relevante para el desarrollo del presente trabajo resulta que, en última instancia, el funcionamiento adecuado del dispositivo implementado por los autores se valida en función de su efecto sobre la conducta del ciclista.

Como su dispositivo está pensado para bicicletas asistidas eléctricamente, el control indirecto de la respiración se logra modulando dicha asistencia mediante control de potencias en las áreas críticas. Para la verificación de su efectividad estudian la tasa de ventilación del usuario, métrica que definen como el volumen total de aire inhalado por minuto (medido con un COSMED Spiropal 6MWT, dispositivo especializado para la medición de tasas de ventilación y saturación de oxígeno en la sangre). En su desarrollo los autores enfatizan que las posibilidades de estudio y diseño del sistema propuesto están sujetas al adecuado modelamiento de las interfaces e interacciones entre el ciclista y los algoritmos que determinan la respuesta de la asistencia eléctrica del pedaleo, refiriéndose con esto a un aspecto común de los sistemas ciber-físicos.

Entre los resultados de su prueba de concepto, se verifica la capacidad del usuario de mantener una velocidad objetivo para distintos grados de asistencia eléctrica. Se identifica una clara correlación entre el grado de asistencia considerado y la tasa de ventilación del ciclista para una velocidad objetivo. Se estudian los tiempos de respuesta de los usuarios y se verifica el control indirecto de su respiración para dos grados distintos de asistencia. Los autores, además de validar el principio de control indirecto de la tasa de ventilación, se proponen como trabajos futuros la validación de la escalabilidad de su solución, la optimización energética del uso del motor eléctrico y el desarrollo de algoritmos orientados específicamente a la interacción entre el ciclista y el sistema de control.

En una línea algo distinta a las anteriores emergen propuestas como la de C. Portilla *et al.* [17]. El foco de su publicación es la optimización de la interacción dinámica entre bicicletas y vehículos en intersecciones multimodales, mediante un sistema que combinaría instrumentos de medición y algoritmos de control predictivo basado en modelos. Se trata de una solución que opera sobre el tiempo de señalización de semáforos, buscando regular el compromiso entre los tiempos de espera de ciclistas y automovilistas. Para ello trabajan en base a modelos de flujos, colas y número de vehículos (autos y bicicletas) en intersecciones. Es un trabajo exclusivamente teórico y sobre simulaciones.

Caracterizando y estudiando el compromiso entre los tiempos de espera de ambos tipos de vehículos mediante un análisis de Pareto en [17], ajustan la función objetivo del controlador MPC. Las simulaciones se realizan en un entorno que considera una red de tráfico con dos intersecciones y el uso de MPC es justificado por la posibilidad de plantear la estrategia de control desde una perspectiva multiobjetivo, aunque en este trabajo solo se considere el tiempo total gastado en cada intersección. Se trata de uno de los pocos trabajos que aborda el modelamiento de la interacción entre flujos de vehículos y ciclistas. En las simulaciones se abordan 3 escenarios. En el primero de ellos no se conoce la demanda de tráfico, en el segundo se realiza medición de la demanda para retroalimentar su medición y, en el tercero, la demanda se considera completamente conocida. Los autores enfatizan que, aunque este último no sea prácticamente implementable, permite establecer referencias de desempeño para análisis posteriores.

Desde una perspectiva crítica, los autores reconocen la variedad de desafíos que supone el adecuado modelamiento de la conducta de ciclistas, dada la multiplicidad de factores que influye su comportamiento (estado de infraestructura, disponibilidad de estacionamientos, aprendizaje de rutas óptimas, etc.). Ello explica que se lo propongan como trabajo futuro en conjunto con el levantamiento de datos de grandes cantidades de ciclistas. El resultado más potente de este trabajo es la constatación de que el desempeño del controlador MPC está sujeto a la calidad de la estimación de la demanda de los influjos tanto de ciclistas como automovilistas.

Un ejemplo de trabajo menos ambicioso, pero más concreto es el desarrollado por G. Liao *et al.* en [8]. En él los autores presentan un sistema de información basado en Android para la consulta de disponibilidad y préstamo de bicicletas públicas. Ello a partir del monitoreo de bicicletas totales y prestadas. Se trata de una aplicación orientada a resolver el servicio público de préstamos de bicicletas en Hangzhou. Esta trabaja en base a la identificación de las estaciones, su ubicación geográfica, la cantidad de bicicletas disponibles para ser rentada y los números de asientos de retorno. A partir de esta información la aplicación genera visualizaciones útiles para los usuarios.

El resultado de este trabajo es el funcionamiento de aplicación (despliegue de ubicación, consulta de estado de préstamos y actualización de información de las estaciones), que opera mediante la API Baidu, archivos XML para la información relativa a las estaciones de bicicleta y datos JSON para el intercambio de información en la actualización de las bases de datos pertinentes. La gran virtud identificada en este artículo es el nivel de detalle con que explicitan las herramientas utilizadas en el desarrollo de su aplicación.

En la misma línea de desarrollos que combinan el uso de sensores con *smartphones* se encuentra el trabajo de Y. X. Zhao *et al.* [18]. En este los autores combinan un sistema de recolección de datos basado en *smartphone* y un sistema de retroalimentación en tiempo real

embebido en la bicicleta, capaz de recolectar y proveer información relativa al ejercicio físico desarrollado y el camino recorrido por el ciclista.

El sistema de recolección de datos se compone de un módulo GPS, un acelerómetro, un giroscopio, una brújula eléctrica y módulos de integración. La ruta realizada, la cantidad de camino recorrido, aceleración, desaceleración y cambios de pendiente son algunos de los datos que el sistema es capaz de recolectar y analizar posterior a la sesión. Además, el sistema es capaz de compartir vía internet información sobre las rutas a otros ciclistas que no las han realizado anteriormente. Por otro lado, el sistema de retroalimentación en tiempo real se compone de un módulo interruptor de lámina (para estimación de velocidad a partir de vueltas de rueda), un display de información en tiempo real y un módulo de transmisión Bluetooth.

El propósito de los autores es la provisión de información que permita a ciclistas alcanzar los resultados que esperan de su actividad física, además de poder compartir información con otros ciclistas que puedan estar interesados en sus rutas. Concretamente, mediante un sistema que requiere de la adición de los módulos mencionados y cargar los programas respectivos al *smartphone* del usuario, los autores logran validar la funcionalidad de su propuesta.

Por su parte, J. Iwasaki *et al.* presentan en [19] un sistema de detección de estado de calles y simulan varios sistemas para la compartición de información entre ciclistas. Se trata de otro sistema basado en *smartphones*. Específicamente, para el sistema de detección de estado de calles aplican árboles de decisión evaluados mediante validación cruzada sobre los datos recolectados por el acelerómetro del celular y su sensor de velocidad angular. Por ejemplo, se asocian virajes suaves y amplios a caminos congestionados y cambios abruptos en el manubrio a malas/peligrosas condiciones de las calles. Además, para el sistema de comunicación se evalúan arquitecturas cliente-servidor, ad hoc (Bluetooth), nodos repetidores y una combinación de las dos últimas. Ello mediante la herramienta de simulación Artisoc en un mapa de parte de la ciudad de Kyoto obtenido de la GSI (del inglés *Geospatial Information Authority of Japan*).

En síntesis, los autores validan el funcionamiento de su sistema de detección obteniendo un 82% de precisión en validación cruzada y prueban que la mayor cantidad de compartición de información (~65%) se logra con una arquitectura cliente-servidor sobre Internet.

En [20], J. A. Brizuela Mendoza *et al.* proponen una bicicleta destinada a probar algoritmos de control automático para su manejo sin conductor. Este trabajo, si bien escapa del ámbito de la electro-velomovilidad por no considerar la participación de ciclistas en el funcionamiento de su diseño, se posiciona como un referente sobre todo por el grado de detalle alcanzado en el reporte de la instrumentación realizada. En ese sentido, también la necesidad de retroalimentar el controlador a partir del procesamiento de mediciones en tiempo real resulta pertinente al desarrollo del presente trabajo. Para la dinámica de la bicicleta en cuestión emplean un modelo dinámico lineal con parámetros variantes en el tiempo, que considera velocidad traslacional, posición angular del manubrio y velocidad del ángulo de balanceo como salidas. Es a partir de estas que establecen las necesidades de medición y actuación de su propuesta.

Concretamente, su sistema se encuentra seccionado en un módulo de procesamiento, que corre un algoritmo programable en *Simulink*, una unidad de comunicación para la interacción entre procesamiento y medición y un módulo de entradas y salidas, que contiene las interfaces requeridas para la medición y actuación. Adicionalmente, su implementación contempla módulos de acondicionamiento electrónico para las señales involucradas. El resultado de su trabajo es un

prototipo funcional que, en principio, permitiría la validación experimental de algoritmos de control automático de bicicletas. No obstante, solo se presentan resultados sobre el funcionamiento electrónico de los módulos considerados, dejando fuera de enfoque casos de control automático efectivo de la bicicleta instrumentada.

El trabajo de I. Tal, B. Ciubotaru *et al.* [21] aborda una de las principales desventajas de las bicicletas asistidas eléctricamente: la duración de su batería. Por ello proponen un sistema de comunicación vehicular entre una bicicleta y un semáforo, que mediante un algoritmo basado en GLOSA (del inglés *Green Light Optimized Speed Advisory System*) recomienda velocidades para evitar frenar completamente a causa de una luz roja. Además, implementan un controlador basado en lógica difusa (T&S de grado cero) para recomendar velocidades máximas que, determinadas en función del viento, permiten ahorrar aún más energía.

Específicamente, el sistema propuesto se compone del *smartphone* del ciclista, unidades de comunicación en semáforos y un medidor de potencia para la batería. El primero recopila y procesa información relativa a la posición, velocidad, dirección, pendiente de la calle y dos campos necesarios para las recomendaciones: tiempo para el cambio de color del semáforo y estado actual del semáforo. Ambos campos son obtenidos de unidades de comunicación presentes en los semáforos (IEEE 802.11p). Un módulo de cómputo y recomendación informa a partir de los datos los requerimientos de velocidad mediante una visualización en la pantalla del celular. El sistema también considera módulos de modelamiento de la dinámica y consumo energético de la bicicleta, que realimentan el controlador difuso y el módulo de cómputo y recomendación.

Las evaluaciones son realizadas a nivel de prototipo (un cruce de semáforo) y simulaciones (múltiples cruces). Esto para probar la validez de los modelos utilizados (al contrastarlos con datos reales) y tener una mejor idea del desempeño del sistema en otros escenarios posibles. En la ejecución de las pruebas alternan el funcionamiento del controlador difuso para comprender su contribución al ahorro total. Con la integración del prototipo los autores registran ahorros energéticos de hasta 46% para bicicletas con partida asistida y de hasta 32% en bicicletas sin dicha asistencia. Sobre las simulaciones, en la diversidad de escenarios considerados (e.g distintos largos de calles, cantidad de luces, topologías, etc.) lograron ahorros de hasta 18% y 15% respecto a versiones clásicas de GLOSA. Como trabajos futuros, proponen estudiar empíricamente su solución para un mayor número de bicicletas y complejizar los factores considerados en sus modelos (e.g. integrar la conducta de ciclistas y tasas de cumplimiento).

Otro trabajo relevante es el de L. Chen *et al.* [22], que centra su inquietud por la utilización de conexiones 3G/LTE en el contexto de múltiples redes ad-hoc móviles (MANETs, por su sigla en inglés). En este desarrollan un esquema de agrupación de ciclistas cuyos grupos emplean una red IEEE 802.11 para transferir datos de sensores entre sus participantes, mientras que la comunicación mediante la red 3G/LTE es utilizada fuera del rango de las MANETs específicas de cada grupo.

Se emplea un enfoque cooperativo para la recolección de datos y su distribución mediante un protocolo que evita colisiones en redes móviles lineales y finas (MLTNs, por sus siglas en inglés) formadas por *smartphones* en bicicletas, que reduce la cantidad de conexiones 3G/LTE necesarias para la agregación de la información de todos los grupos. Más específicamente, el protocolo de comunicación emplea un mensaje tipo canasta a la que se va agregando la información

de cada bicicleta en la red IEEE 802.11. Cuando la canasta ha pasado por todos los miembros del grupo, el líder de este envía dicha canasta mediante la red 3G/LTE.

Entre los resultados de los autores se encuentra una comparación de la cantidad de datos enviados por la red 3G/LTE para cantidades crecientes de ciclistas con el esquema propuesto, logrando reducciones significativas con respecto a esquemas LTE4V2X (del inglés LTE para redes vehiculares) y de uso exclusivo de 3G/LTE. Por otro lado, su propuesta es comparada en la misma métrica para intervalos de medición crecientes donde, nuevamente, demuestran un número de conexiones menor. Además, ponen a prueba un mecanismo de *beacons* (pequeños paquetes diseminados periódicamente) de reordenamiento de ciclistas con que logran reducir aún más las conexiones en función de los intervalos entre mediciones. Ambas evaluaciones son realizadas analíticamente y mediante simulaciones en Qualnet 5.0.

W. Jeon *et al.* [23] desarrollan otro trabajo en que se emplea un enfoque de simulación junto a un montaje experimental sobre una bicicleta instrumentada. En este los autores abordan el problema de la detección de automóviles cercanos mediante un sensor laser montado en una plataforma rotatoria. El desafío es controlar el ángulo del sensor para mantener su foco en un automóvil cuya trayectoria futura es desconocida. La principal ventaja de este sistema es el costo relativamente bajo de su sensor, en comparación a otros sistemas de mayor complejidad como los LIDARs (del inglés *Light Detection And Ranging*) que permiten un escaneo de área completa. Sin embargo, ello compromete la obtención de información clave para la estimación de la ubicación y orientación del vehículo, producto de mediciones puntuales a una baja tasa de muestreo (50 Hz). Para hacer frente a ello, los autores emplean un algoritmo inteligente que determina, en tiempo real, la orientación del sensor requerida para mantener un monitoreo efectivo.

Se emplean modelos predictivos para la estimación de las trayectorias de vehículos detectados mediante técnicas de *clustering* sobre valores de rango e inclinación del sensor. Y proceden mediante un enfoque de interacción entre múltiples modelos (o por su sigla en inglés, IMM) para abordar el monitoreo de movimiento en línea recta y en viraje. A esto se adiciona el empleo de filtros de Kalman y filtros de Kalman extendidos para la predicción de la trayectoria utilizada en la determinación de la inclinación del sensor en tiempo real.

En suma, los autores simulan y evalúan experimentalmente el desempeño de su propuesta obteniendo concordancias destacables, que le permiten proveer alertas a ambos actores satisfactoriamente (ciclista y automovilista) a una velocidad relativa máxima de 11.9 m/s con un rango de medición de 25m en línea recta. Mientras, factores como la inclinación de la bicicleta y viraje del manubrio se presentan como principales limitantes del rango de medición y la continuidad del seguimiento del automóvil por parte del sensor. Como trabajo futuro se plantean pruebas operacionales de 6 meses en 10 ciclistas diarios.

En el trabajo de S. Smaldone *et al.* [24] también se desarrolla un sistema ciber-físico orientado a atender el problema de la seguridad en el uso de bicicletas. Se trata de un sistema que emplea técnicas de visión computacional sobre un *streaming* de datos generados por una cámara en la parte trasera de la bicicleta, que permite detectar vehículos que se acercan a la bicicleta de manera riesgosa y alertar en tal caso. Así, las principales dificultades emergen de la escasa capacidad energética y de cómputo que es posible añadir a una bicicleta sin limitar su funcionamiento, en conjunto con la dificultad de distinguir acercamientos y alejamientos de vehículos, riesgosos y no riesgosos.

Para la detección se combinan análisis de flujos ópticos con modelos de mezcla de gaussianas en espacios de colores HSV (del inglés *Hue, Saturation, Value*, que en español significa matiz, saturación y valor/brillo). De ese modo se obtienen imágenes en que se distinguen aquellos autos que se aproximan de aquellos que se alejan. Una vez detectados, el seguimiento de los automóviles se logra mediante una versión adaptativa del análisis de componentes principales (PCA), resiliente a cambios de orientación, escala e iluminación, que afectan típicamente el seguimiento de objetos que se aproximan. Adicionalmente, el sistema define una zona de seguridad virtual y calcula constantemente la distancia entre el móvil que se acerca y la bicicleta, para distinguir acercamientos riesgosos de normales y dar las notificaciones acordes a dicho riesgo. Se trata de una implementación con la que logran un 73,1% de precisión.

Los autores logran validar su propuesta obteniendo un desempeño razonable en detección, incluso en modos de funcionamiento con baja cantidad de fotogramas (3.7 - 1.3 FPS). También reconocen que su implementación se trata de un peor caso en tanto requiere de adherir un netbook (HP Mini 311) a la estructura de la bicicleta. Esto implicaría un incremento de menos del 10% del peso de una bicicleta de conmutación promedio según declaran, no obstante, no especifican valor alguno (ni peso del computador ni peso de bicicleta considerada). Además, en términos de requerimientos energético, los autores estiman que su sistema tendría una duración aproximada de 5 horas.

Finalmente, se presenta un breve recorrido por el conjunto de publicaciones que, perteneciendo al estado del arte de la materia, orientan directamente este trabajo. El propósito es brindar un marco comprensivo del desarrollo de la problemática que se pretende abordar.

En [25-28] Céspedes *et al.* han abordado sistemáticamente el problema de la coordinación inteligente y explícita de caravanas de ciclistas. Motivados por los efectos de congestión en la infraestructura vial que han implicado las políticas tendientes al fomento del uso de bicicletas como medio de transporte en determinados países, han propuesto un sistema ciber-físico que persigue la formación de caravanas en base a una política de seguimiento de líder (velocidad objetivo) y distancia segura entre sus suscriptores. Esto en miras a optimizar el uso de la infraestructura disponible como alternativa a su ampliación.

En un inicio [25] los autores han concebido el sistema como una red mesh entre vehículos que comparten la información necesaria para el funcionamiento de un algoritmo CACC (del inglés *Cooperative Adaptive Cruise Control*), que determina recomendaciones de aceleración a ser transmitidas mediante una interfaz háptica. Tempranamente desarrollaron simulaciones para verificar el funcionamiento del algoritmo, que satisfactoriamente logró agrupaciones de ciclistas equiespaciados, y experimentaciones que pusieron de relieve la necesidad de complementar la interfaz háptica con una visual para reducir las malas interpretaciones de la señal de recomendación. El sistema como módulos de comunicación, control e interfaz humano-máquina han sido invariantes en la totalidad del desarrollo de su prototipo.

Los siguientes pasos [26] consistieron en pruebas controladas de la interpretación de la interfaz visual-háptica, la incorporación del error humano de aceleración registrado en la lógica interna del algoritmo CACC (en simulaciones) y la continuación del desarrollo del prototipo. En esta etapa la interfaz visual-háptica resultó ser razonablemente efectiva en su recomendación (70% de precisión en los rangos considerados). En cuanto a las simulaciones, el algoritmo demostró ser lo suficientemente robusto como para mantener su efectividad, a pesar de que la incorporación del error implicó aumentos en las varianzas entre las velocidades alcanzadas y las deseadas. Por su

lado, la bicicleta de prueba terminó incorporando un módulo Romeo V2 *All-in-one* como unidad de procesamiento, un sensor de interrupciones y *Shield* GPS como unidades de medición, una rueda dentada y motorizada como actuador y un módulo XBee Pro 60mW (IEEE 802.11.4) como módulo de comunicación.

Más recientemente [27, 28] se ha continuado con las pruebas de los prototipos y el desarrollo extensivo de simulaciones orientadas a determinar rangos recomendados de operación del sistema propuesto. Se han realizado pruebas en entornos urbanos controlados en que, si bien se han presentado peores interpretaciones de la interfaz visual-háptica, el sistema ha sido capaz de mantener las formaciones deseadas. En ese sentido, también se ha constatado que los módulos XBee empleados solo permiten operar en línea de vista y la baja confiabilidad del *Shield* GPS ha llevado a emplear una aplicación Android para el envío de la señal GPS obtenida desde un *smartphone*.

7. PROPUESTA DE DISEÑO

La propuesta consiste en un conjunto de dispositivos físicos y de software que se disponen en una bicicleta y en el celular de su usuario. Se trata de una aplicación Android que, mediante una base de datos común en la nube permite compartir posición, velocidad y aceleración de sus usuarios. En cada teléfono inteligente se determina una sugerencia orientada a influir la aceleración del usuario, de modo tal que permita mantener distancia y velocidad constante entre los miembros de una caravana de ciclistas. De esta se distinguen su arquitectura y sus bloques funcionales.

La Figura 2 es una representación gráfica de los enlaces y nodos de la aplicación. En esta se aprecia un enlace Bluetooth de baja energía (o BLE, del inglés *Bluetooth Low Energy*) entre la interfaz háptica y el teléfono del usuario, que presenta una interfaz visual como medio para la señalización de sugerencias. La comunicación entre usuarios de la aplicación es mediante la sincronización con una base de datos común (proveída por Firebase Cloud Services) en internet, a la cual se accede, en principio, a través de distintos operadores de la red LTE/4G. También puede apreciarse el enlace con el sistema de posicionamiento global (o GPS, por sus siglas en inglés) a partir del cual los usuarios adquieren su propia ubicación, velocidad y aceleración. A grandes rasgos, la aplicación emplea un algoritmo de control predictivo con intervalos difusos Takagi y Sugeno (T&S) para integrar el error histórico humano de aceleración al cálculo de la acción de control actual. Lo anterior en cada bicicleta salvo la del líder, que fija la velocidad de referencia para el conjunto de los suscritos a la caravana.

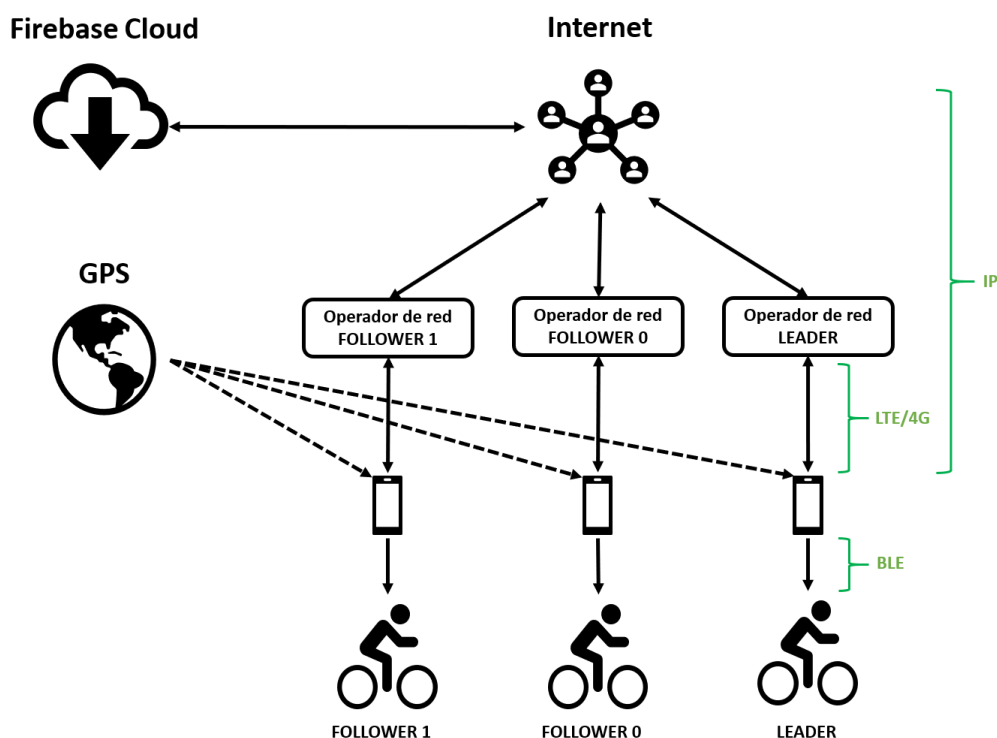


Figura 2: Representación de la arquitectura de la propuesta. Elaboración propia.

Al respecto del sistema de control, se emplea un esquema predictivo basado en modelos, que a partir de un modelo lineal acelerado del movimiento de traslación de la bicicleta y su antecesora permite estimar su distancia relativa en una ventana de tiempo de largo configurable [29]. Este es incorporado dentro de un funcional cuadrático que considera las distancias relativas en la ventana de tiempo considerada, las diferencias entre las velocidades de la bicicleta y el líder

y la variación de la aceleración de la bicicleta. Dicho funcional es optimizado mediante un algoritmo genético, para el cual las variables de optimización son las aceleraciones presentes y futuras. Además, el algoritmo integra elementos de predicción robusta mediante intervalos derivados de un modelo Takagi y Sugeno, elaborado a partir de datos históricos de aceleración de ciclistas, que son incorporados como restricciones en el espacio de soluciones factibles para el problema de optimización, es decir, restringiendo el espacio de búsqueda de las acciones de control presente y futuras. El detalle de su formalización se encuentra íntegramente documentado en [29]. En esta propuesta se traduce dicha implementación a JAVA y se generaliza el funcional para una cantidad de pasos configurable.

Por otro lado, en la Figura 3 se presenta con más detalle el conjunto de bloques funcionales que integran el prototipo. En conjunto de la izquierda, en orden, GPS es el módulo encargado de la adquisición de la ubicación GPS, DB es el módulo encargado del intercambio de información con una base de datos alojada en una nube, MPC es el módulo que implementa la lógica del cálculo de la acción de control, VIU es la interfaz visual del usuario y BLE el de comunicación con los periféricos de la bicicleta. A continuación, en el capítulo 8 se describe en mayor detalle cada uno de ellos en el sistema.

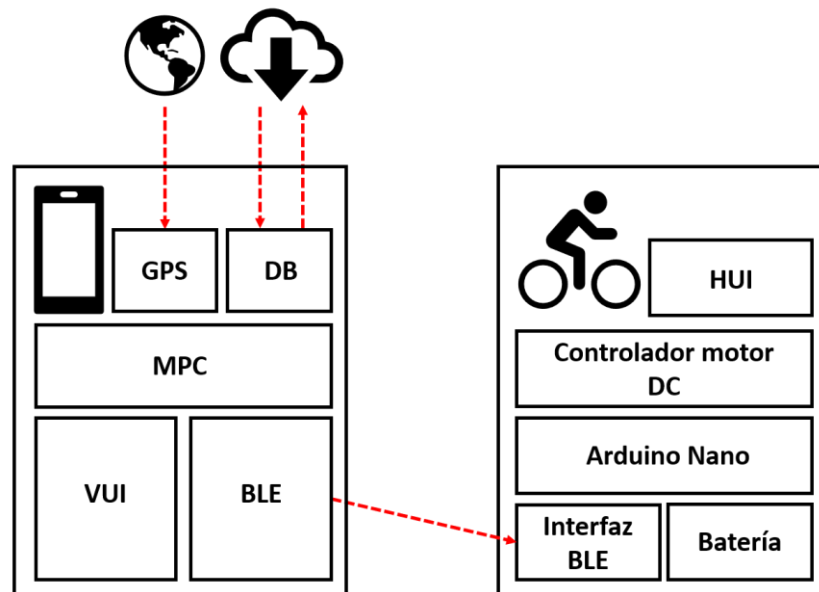


Figura 3: Representación de los módulos funcionales de la propuesta de diseño. GPS por módulo de Sistema de Posicionamiento Global (en inglés), DB por módulo de comunicación con servicio de Base de Datos, VUI por Interfáz Visual de Usuario, BLE por modulo de comunicación Bluetooth Low Energy y HUI por interfáz háptica de usuario. Elaboración propia.

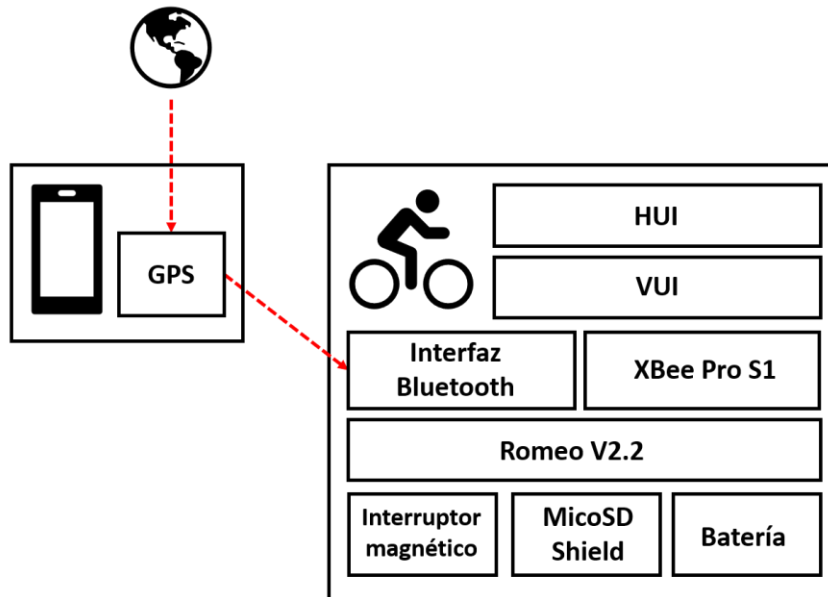


Figura 4: Representación de los módulos funcionales del prototipo actual. Elaboración propia.

La figura 4 presenta una representación del prototipo actual para poner en perspectiva los cambios introducidos en esta nueva versión. En comparación con la figura 3, se aprecia la sustitución de la tarjeta Romeo V2.2 por una Arduino Nano, el cambio del enlace Bluetooth a Bluetooth Low Energy, la extracción del *Shield* MicroSD y la migración de la lógica de control e interfaz visual de usuario al *smartphone* de este. También, se representa mediante la adición del símbolo de nube y la extracción del módulo XBee Pro S1, la migración de una arquitectura local de comunicación mediante enlaces ZigBee entre las bicicletas a una de área amplia mediante la red celular LTE/4G.

8. IMPLEMENTACIÓN

En este capítulo se presentan y explican los detalles de la implementación de la plataforma experimental orientada a la evaluación del sistema de control presentado en la propuesta de diseño. En este se abordan por separado la implementación de dispositivos físicos alojados en las bicicletas de los desarrollos realizados en la aplicación alojada en el celular del usuario. En la Figura 5 puede apreciarse una vista en detalle del resultado de lo primero, específicamente, del conjunto de dispositivos que permiten el funcionamiento de la interfaz háptica para dos de las tres bicicletas instrumentadas. Respecto de lo segundo, en su respectiva subsección se presentan oportunamente aquellas clases, actividades, servicios y fragmentos de código necesarios para la implementación de los módulos de control, comunicación e interfaz visual en la aplicación Android.



Figura 5: Dispositivos físicos. A la izquierda, dos de los tres prototipos equipados. A la derecha, en detalle uno de ellos. Elaboración propia.

8.1. Implementación de dispositivos físicos

8.1.1. Módulo microcontrolador

Para este módulo se emplea una placa Arduino Nano (basado en ATmega), principalmente por la disponibilidad de periféricos que admiten modulación por ancho de pulsos (o PWM, por sus siglas en inglés), para el funcionamiento del motor DC asociado a la interfaz háptica, y un puerto serial para comunicarse al módulo Bluetooth. Este último permite traducir la señalización proveniente de una aplicación Android en un modo de operación específico de la interfaz. La Figura 6 presenta una vista en detalle de este módulo.

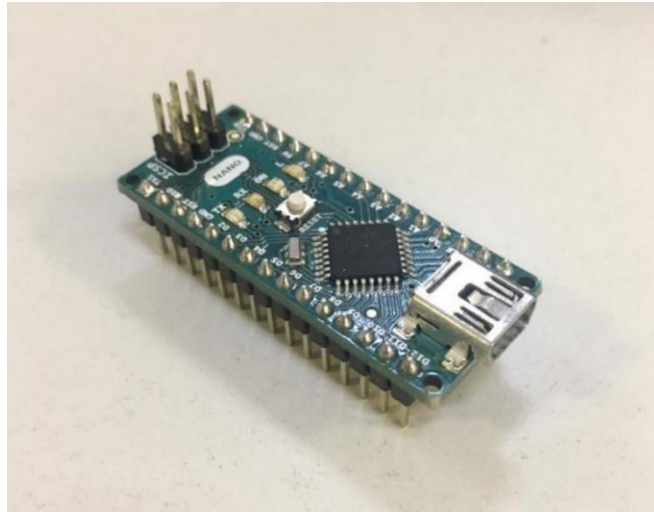


Figura 6: Arduino Nano empleada en los prototipos. Elaboración propia.

8.1.2. Interfaz serial BLE.

Esta consiste en un módulo maestro-esclavo HC-05 (ver Figura 7) que opera mediante la integración de una antena, un controlador con el estándar Bluetooth 4.0 y un puerto serial. Para su configuración y operación se emplea la librería SoftwareSerial, desarrollada y disponible íntegramente en las plataformas de desarrollo Arduino.

El manejo de la transmisión en esta implementación se programa específicamente para operar como esclavo frente a las señales enviadas por la aplicación Android, de modo tal que el envío de información es unidireccional. No obstante, cabe mencionar que la naturaleza del enlace permite el intercambio bidireccional de información entre los dispositivos alojados en la bicicleta y la aplicación, para la cual fue programado el método `dataAndSuggestionExchange`.

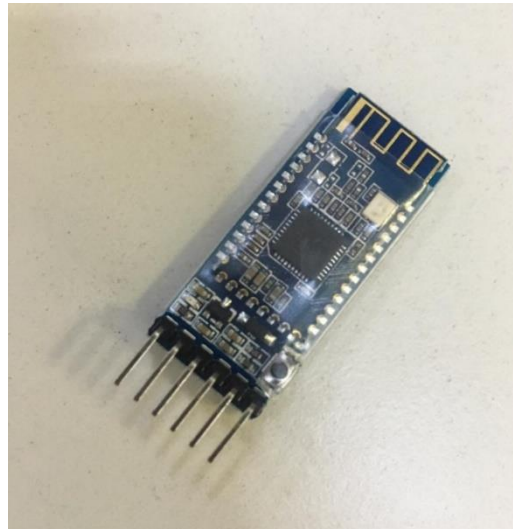


Figura 7: Módulo Bluetooth 4.0 HC-05. Elaboración propia.

8.1.3. Interfaz háptica (HUD).

Esta interfaz se implementa mediante dos bloques funcionales. Por un lado, se utiliza un controlador de motores DC (ver Figura 8) que integra un puente H y un control de carga para las bobinas del motor de la interfaz. Mientras, la interfaz propiamente tal consiste en un motor DC ubicado al interior del manubrio de la bicicleta, encargado de la rotación de una pieza que roza el dedo del usuario indicando acelerar o desacelerar dependiendo del sentido de giro.

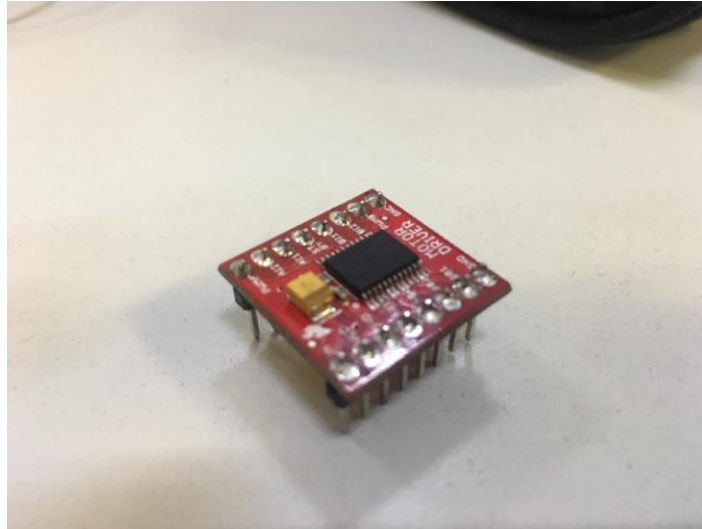


Figura 8: Controlador DC. Elaboración propia.

8.1.4. Módulo de alimentación.

Para la alimentación se emplea una batería de 9V y 600mAh conectada a un Stepdown,(ver Figura 9) calibrado para garantizar una salida de 6V. Suficiente para la operación tanto del motor DC como de la electrónica de control y comunicación.

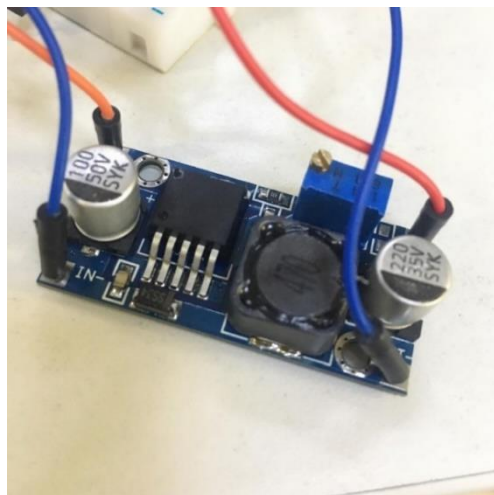


Figura 9: Step-down empleado en la implementación. Elaboración propia.

8.2. Desarrollo de software

8.2.1. Módulo de control

a. Entradas del modelo predictivo:

Las entradas del modelo predictivo son gestionadas mediante la clase `pairStatePackage`, que contempla la distancia con la bicicleta delantera, calculada mediante la diferencia en sus estampas GPS y estimaciones de su velocidad.

b. Modelo predictivo:

El modelo predictivo a partir del cual se calcula la acción de control se encuentra condensado en 4 clases relacionadas. Estas son reseñadas brevemente a continuación.

RMPC

Se trata de la clase en que se gestiona la entrada del paquete con la información del estado de la caravana y del ciclista delantero (`pairStatePackage`) para determinar la recomendación de aceleración a ejecutar mediante su método `getControlAction`. En esta se emplean en las clases `GeneticAlgorithm`, `errorBuffer`, `fuzzyInterval` y `TSmodel` para determinar las restricciones de un problema de optimización cuadrático, configurable para distintas cantidades de pasos y que considera distancia entre usuarios y velocidad del líder de la caravana.

errorBuffer

Es la clase que gestiona el almacenamiento y actualización de las últimas 5 diferencias entre la aceleración que ejecutó un usuario y aquella recomendada por el sistema de control para ese momento (lo que se entiende por “error humano”). Esta es empleada específicamente en el cálculo de los intervalos difusos que se integran en las restricciones del espacio de búsqueda del algoritmo genético empleado para la minimización del funcional cuadrático.

fuzzyInterval

Es la clase que determina cuantitativamente la compresión de las restricciones del espacio de búsqueda a partir de `errorBuffer` y un modelo T&S del error humano ajustado previamente en [29]. Este último fue construido a partir de data histórica de errores de aceleración en etapas previas del desarrollo del presente prototipo.

TSmodel

Consiste en el conjunto de reglas, funciones de pertenencia y parámetros de los modelos lineales que componen típicamente un modelo T&S. Esta clase se encuentra específicamente ajustada conforme a lo desarrollado en [29] a partir de los datos de error humano recolectados en la experiencia descrita en [28]. Cabe especificar que se trata de un conjunto de 5 reglas sobre los errores de aceleración, asociadas a modelos lineales que contemplan el primer y quinto regresor de dicho error. Estos últimos son proveídos por la clase `errorBuffer` para el cálculo de las salidas de cada regla, que luego son ponderadas dependiendo de los respectivos grados de activación para el cálculo de los intervalos que restringirán el espacio de búsqueda del optimizador.

c. Optimizador:

GeneticAlgorithm

Se trata de una clase que implementa los lineamientos de optimización de poblaciones mediante selección, cross-over y mutación de individuos propios del paradigma de los algoritmos genéticos. Es en esta que se especifica el funcional cuadrático a N-pasos a modo de función de fitness (*calcFitness*), además de aplicarse los intervalos difusos (*fuzzyInterval*) que integran el error humano en las restricciones del espacio de búsqueda sobre el cual el algoritmo itera en la búsqueda de los mejores individuos. En este contexto, son individuos las aceleraciones sugeridas a la bicicleta presente y precedente, mientras que la población es el conjunto de dichas acciones de control en una ventana de predicción a N-pasos.

8.2.2. Módulo de comunicación

a. Servicio de ubicación de Google:

Para adquirir la ubicación en cada una de las bicicletas se emplea el servicio de ubicación de Google. Se trata de una interfaz programable que permite acceder mediante la clase *Location* y *LocationRequest* (propias de Android) a la ubicación del dispositivo mediante su módulo GPS e información proveniente de las antenas del operador de red de ser necesario. A grandes rasgos, esta interfaz permite instanciar un *GoogleApiClient* para la gestión de permisos de acceso a los periféricos del equipo (GPS y antena LTE/4G) y, una vez concedidos, habilita un conjunto de métodos que permiten adquirir la ubicación de la manera que resulte más conveniente a la aplicación que se desarrolla. Uno de ellos es el método *getLastLocation*, que permite obtener la última ubicación registrada por el servicio una vez empleado.

En esta implementación se optó por emplear responsivamente los métodos *startLocationUpdates* y *onLocationChanged* (Anexo 13.1). Esto por la conveniencia de que permiten fijar un intervalo de actualización de la ubicación en el dispositivo y ejecutar rutinas contenidas en *onLocationChange* una vez que la actualización se vuelve efectiva. Así, se procura calcular la acción de control presente con la última actualización disponible del servicio GPS y se sube a la nube el estado actual de la bicicleta una vez que los dos procesos anteriores se han completado. Detalles sobre la implementación del método *onLocationChanged*, sobre escrito en la clase *RecordingActivity*, pueden encontrarse en Anexo 13.1.

b. Base de datos en tiempo real de Firebase:

Para la transmisión de la información entre bicicletas se empleó una base de datos común a la que cada usuario sube constantemente su ubicación, velocidad y aceleración. Además, es a dicha base de datos que cada usuario consulta por el estado del líder y del usuario que le precede en la caravana. Esto se logra mediante el servicio *Realtime Database* de *Firebase Cloud*, alojados en *Google Cloud*.

Se trata también de una interfaz programable que una vez implementada permite interactuar mediante métodos *set* y *get* con las direcciones y valores de una base de datos exportable en formato *JSON*. Específicamente, se elabora un nuevo hilo de datos cada vez que inicia una nueva sesión. Dentro de ella, cada usuario cuenta con su respectiva trama a la que sube su estado (*FOLLOWER1*, *FOLLOWER0* y *LEADER*) y consulta a la trama del usuario que se encuentra delante de él para ejecutar luego sus propias rutinas. Para este propósito se desarrolla la clase *Post*, que permite dar

orden y formato a estas acciones, además de facilitar modificaciones que resultaron necesarias tanto en las pruebas de laboratorio como de campo.

De especial relevancia es el método `onChildAdded` que puede ser empleado una vez configurado el listener correspondiente mediante el método `addChildEventListener`. A grandes rasgos, se trata de las funcionalidades de Firebase que permite desarrollar aplicaciones sensibles a modificaciones en la base de datos. Específicamente, a cada tipo de usuario se le adiciona un listener en la dirección del usuario que se encuentra delante de él, de modo tal que cada vez que el usuario delantero actualiza su ubicación en la base de datos, dicha información es transmitida de la base de datos al equipo de quien lo sigue. La configuración de las direcciones o referencias a la base de datos se encuentra implementada en el método `linkSetting`. Para una noción más acabada de la implementación de ambos métodos (*linkSetting* y *childAdded*) se sugiere revisar el Anexo 13.2.

c. Servicio de comunicación BLE:

El enlace de comunicación Bluetooth es establecido mediante las clases `SerialService`, `SerialSocket` y la interfaz `SerialListener`. En conjunto, estas permiten establecer un enlace accesible durante el tiempo de vida de la aplicación mediante los métodos `send` y `receive`. El primero opera al ser invocado y el segundo es invocado por defecto al recibirse un mensaje en el socket serial implementado por la clase `SerialSocket`. La virtud de esta implementación es que permite mantener el enlace como servicio, es decir, con un ciclo de vida paralelo, invisible y subordinado a la interfaz gráfica del usuario, previniendo bloqueos o tiempos de espera sensibles.

Las clases mencionadas se encuentran basadas en las librerías que Android contempla para la administración de permisos y funcionalidades de los periféricos disponibles en el teléfono que corre la aplicación (`android.bluetooth`). Si el equipo cuenta con la posibilidad de establecer enlaces BLE, entonces lo hace. En el caso contrario monta un socket mediante la tecnología Bluetooth que esté disponible en el equipo.

El método `bleSend`, implementado específicamente para la transmisión de la sugerencia de aceleración puede verse en Anexo 13.3. En este se puede distinguir un modo de señalización para el líder y uno para los seguidores. Además, se aprecia que la señalización de aceleración sugerida a los seguidores se encuentra sujeta a la comparación entre la aceleración calculada y un umbral establecido a partir del promedio y desviación de aceleración ejercida por el usuario en experiencias previas con el prototipo. Para el caso del líder se señala dependiendo de si lleva o no la velocidad de caravana escogida en su interacción con la interfaz gráfica (esto último se aborda en más detalle en la próxima sección).

Es relevante considerar que este es uno de los métodos sobre los cuales es conveniente iterar conforme se realizan pruebas experimentales. La evolución y resultados asociados a la implementación de esta versión se presenta en la sección Evaluación del prototipo.

8.2.3. Interfaz gráfica de usuario (VUI)

La interfaz gráfica de usuario es desplegada en la medida en que avanza la secuencia de actividades (android.activity) de la aplicación. De esta se distinguen fundamentalmente dos flujos: uno de seguidor y uno de líder de caravana. En la etapa de desarrollo actual de la aplicación existen modos de usos específicos que deben respetarse para poner a prueba el esquema de control que se implemente, es decir, existen secuencias posibles que originan conflictos fatales para esta. Un ejemplo de ello es que dos usuarios en celulares distintos se definan como líderes, o que se defina un seguidor 1 sin definir un seguidor 0. A continuación se detalla aquel flujo que permite poner a prueba el esquema de control implementado.

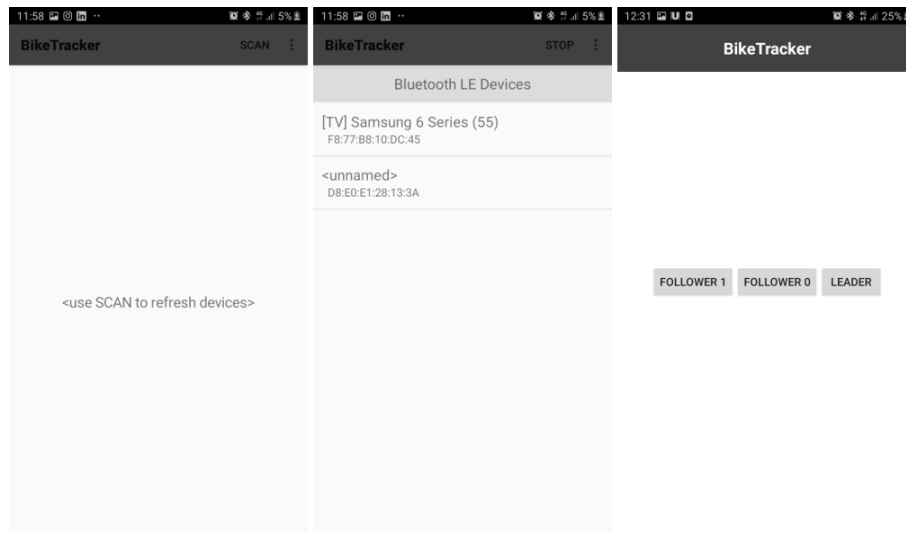


Figura 10: Flujo de actividades común. Se trata de aquella secuencia de actividades a la que todos los usuarios tienen acceso. Elaboración propia.

En la Figura 10 se muestra la primera sección del flujo de la aplicación. De izquierda a derecha se presentan las actividades asociadas al escaneo bluetooth que permite identificar los módulos de las bicicletas y establecer el emparejamiento al escogerlos de entre las opciones disponibles (primer y segundo *layout*). El tercer *layout* corresponde a la primera bifurcación posible en que corresponde la selección del tipo de usuario: líder, seguidor 0 y seguidor 1.

En la Figura 11 se presentan los *layouts* de la primera bifurcación posible, de las cuales se derivan 3 opciones para cada una de ellas. A la izquierda se aprecia la vista para el líder de la caravana, quien puede escoger entre tres velocidades establecidas en coherencia con los desarrollos de [28]: 2.77 m/s, 4.16 m/s y 5 m/s. A la derecha se aprecian las opciones de espaciamiento disponibles para los usuarios seguidores de la caravana: 2 m, 5 m y 8 m. Con independencia de lo escogido, todos los flujos convergen a la gráfica de señalización visual.

Finalmente, la Figura 12 presenta la opción de inicio de sesión y el objeto de señalización visual, que carga una flecha verde cuando la aceleración que debe ejercer el usuario es positiva y una flecha roja cuando la aceleración debe ser negativa. Además, se define un tercer estado blanco de inactividad para señalar aceleración cercana a cero con velocidad próxima a la del líder y espaciamiento deseado.

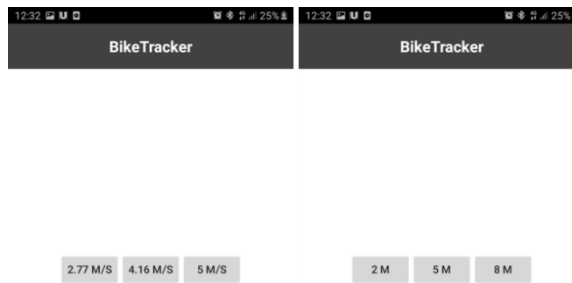


Figura 11: Actividades diferenciadas por tipo de usuario. A la izquierda la elección de velocidad que corresponde al líder. A la derecha, la elección de espaciamento respecto al líder o seguidor que precede en la caravana. Elaboración propia.

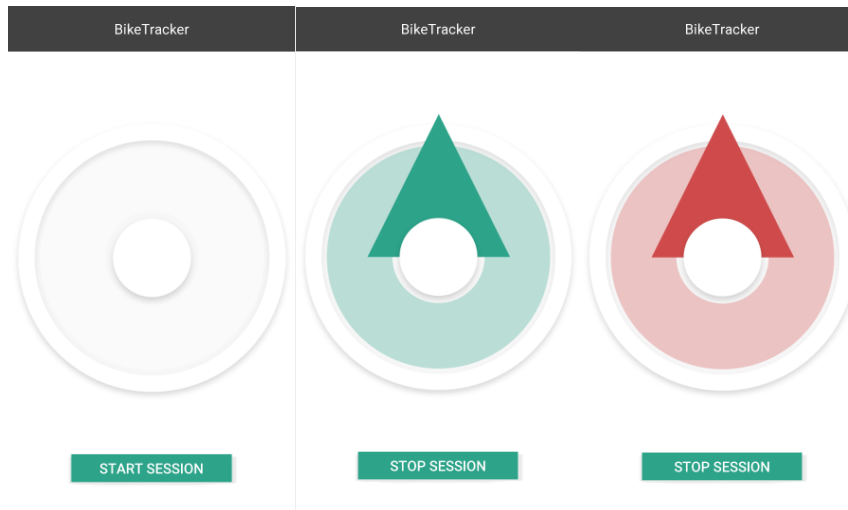


Figura 12: Interfaz visual y sus modalidades. Elaboración de Daniela Delgado.

9. EVALUACIÓN DEL PROTOTIPO

En esta sección se presentan los resultados, en conjunto con una breve discusión, del proceso de diseño e implementación del prototipo. Esta se divide entre aquellos relativos a la medición de los retardos de transmisión entre pares, la capacidad de control de velocidad del líder y las pruebas del sistema de control integrado. Las discusiones presentadas se plantean específicamente en relación con el cumplimiento de los objetivos planteados o los factores influyentes en su incumplimiento. Cabe mencionar que existen datos derivados de pruebas experimentales que por síntesis no son presentados en esta sección.

9.1. Caracterización del retardo de transmisión de estado

Para la determinación del intervalo de transmisión de estado de las bicicletas (posición, velocidad y aceleración) se realizan sesiones de transmisiones consecutivas del estado del líder y un seguidor en el laboratorio de comunicaciones avanzadas. Esto se logra programando el envío de una estampa de tiempo desde el líder a un seguidor, quien inmediatamente luego de recibirlo envía de vuelta la estampa. Esta luego es comparada con una nueva estampa obtenida en el origen tan pronto llega el mensaje desde el seguidor al líder. Dicho trayecto y metodología de cálculo se encuentra esquematizado en la Figura 13.

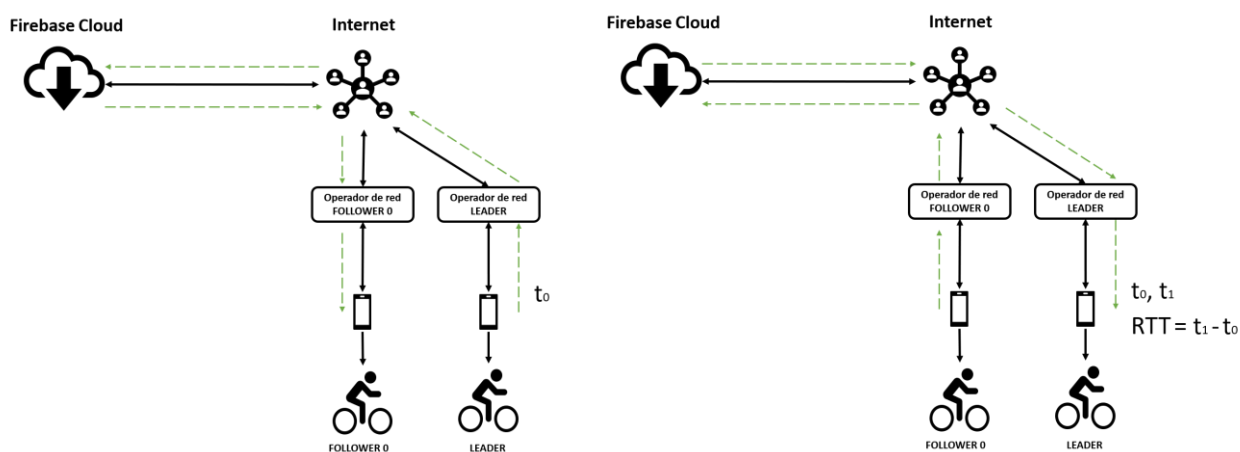


Figura 13: Descripción del trayecto del paquete para medición RTT. Elaboración propia.

Para esta prueba se modifican las rutinas asignando el valor “RTT_TEST” a la variable string MODE del método RTTPacket2DB, de modo tal que ni el cálculo de la acción de control, ni la adquisición de la ubicación GPS influyeran en el intervalo de tiempo registrado. Esta prueba fue realizada sobre los operadores de red WOM; Movistar, Claro y Entel. El detalle de los periodos monitoreados y los periodos analizados pueden encontrarse resumidos en la Tabla 2. Durante los primeros se realizan mediciones en todo el horario especificado, mientras en los segundos solo se consideran aquellos con muestreos para los cuales el operador fue capaz de proveer conexión 4G. Como se procuró monitorear durante 5 días hábiles de distintas semanas, se opta de ahora en adelante por referirse mediante números enteros a cada uno de ellos, de modo tal que al lunes corresponde el número 1, al martes el número 2 y así sucesivamente. Sobre ambos periodos se emplearon los celulares Lenovo K33b36 (Android 7.0, procesador MSM8937 1,4 GHz y 2GB de RAM) y Samsung SM-J400M (Android 9, procesador Exynos 7570 1.4GHz y 2GB de RAM, como se encuentra resumido en la Tabla 3.

Operador	Periodos monitoreados	Periodos analizados	Horario
WOM	[20-04-2020, 24-04-2020]	[20-04-2020, 21-04-2020] [23-04-2020, 24-04-2020]	10:00-18:00
Movistar	[04-05-2020, 08-05-2020]	[04-05-2020, 08-05-2020]	10:00-18:00
Claro	[14-05-2020, 20-05-2020]	[14-05-2020, 15-05-2020] [19-05-2020, 20-05-2020]	10:00-18:00
Entel	[11-05-2020, 12-05-2020]	[11-05-2020, 12-05-2020]	10:00-18:00

Tabla 2: Especificaciones de periodos monitoreados y periodos analizados. Elaboración propia.

Nombre	Modelo	Android	Procesador	RAM [GB]
Samsung J4	SM-J400M	9	Exynos 7570 1.4GHz	2
Lenovo vibe K6	Lenovo K33b36	7	MSM8937 1,4 GHz	2

Tabla 3: Especificación de equipos celulares empleados para las pruebas. Elaboración propia.

Las Figuras 14 y 15 presentan los porcentajes de paquetes que, durante el segmento del día especificado, tuvieron un retardo de envío de paquetes menor a 3 valores distintos del parámetro alpha (0,25 [s], 0,5 [s] y 1 [s]). Específicamente, en la primera se presentan dichos valores para el 2 día de los periodos monitoreados, mientras que en la segunda se presenta el promedio de dichos valores, en cada segmento, para la totalidad del periodo analizado de cada proveedor. De estos (Figura 14) se puede apreciar que virtualmente todos los paquetes tienen retardos menores a 1 segundo, salvo para WOM, que muestra el peor desempeño en las métricas consideradas. Al detenernos en Movistar, Claro y Entel es posible apreciar que sus desempeños son relativamente similares (todos superiores a 95%) para una exigencia de 0,5 [s]. No obstante, las diferencias emergen al considerar solo los paquetes enviados en menos de 0,25 [s], para lo cual Movistar exhibe mayores porcentajes y estabilidad a lo largo del día analizado.

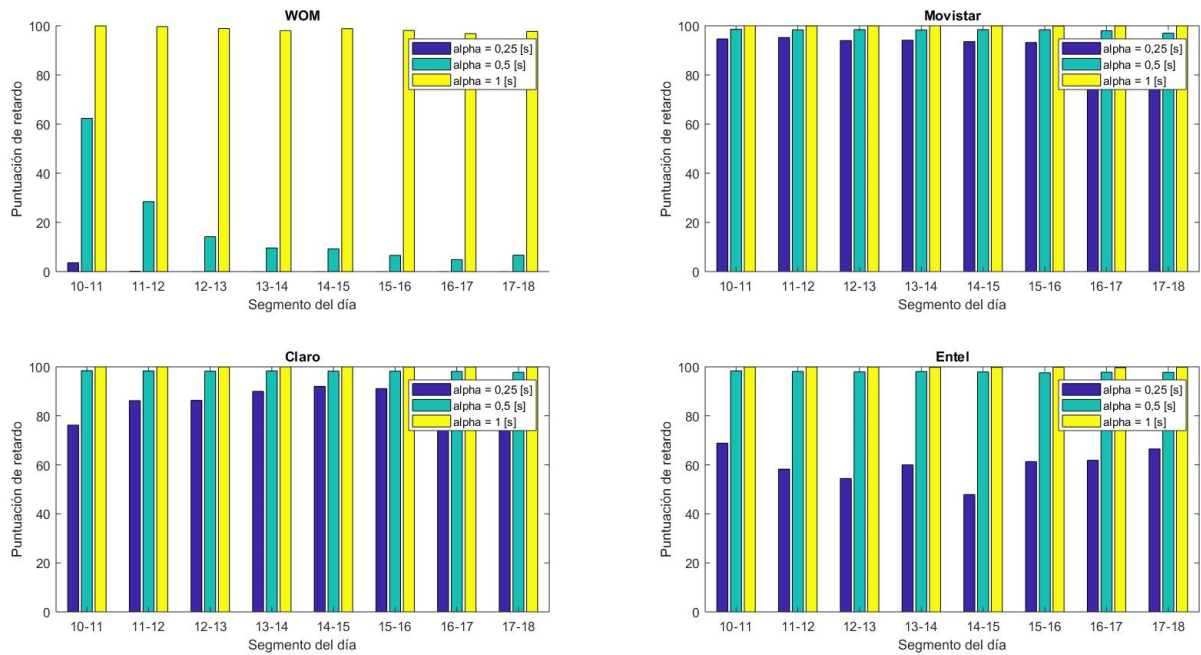


Figura 14: Puntuaciones por operador del día 2 de las distintas semanas de medición, para la misma ventana de tiempo. En azul el porcentaje de retardos registrados menores a 0,25 [s], en turquesa los menores a 0,5 [s] y en amarillo los menores a 1 [s]. Elaboración propia.

Sobre la Figura 15 la comparación entre proveedores pierde validez por no contarse con la misma cantidad de tiempo de monitoreo, pero sí pueden apreciarse para cada uno tendencias. Sobre WOM se aprecia un mejor desempeño en los primeros segmentos del día, mientras que Claro y Movistar sugieren un mejor comportamiento en torno al segmento de las 13:00 y 14:00. Entel pareciera seguir en una posición inferior a Movistar y Claro, no obstante, debe tenerse en cuenta que para dicho proveedor solo fue posible muestrear 2 días.

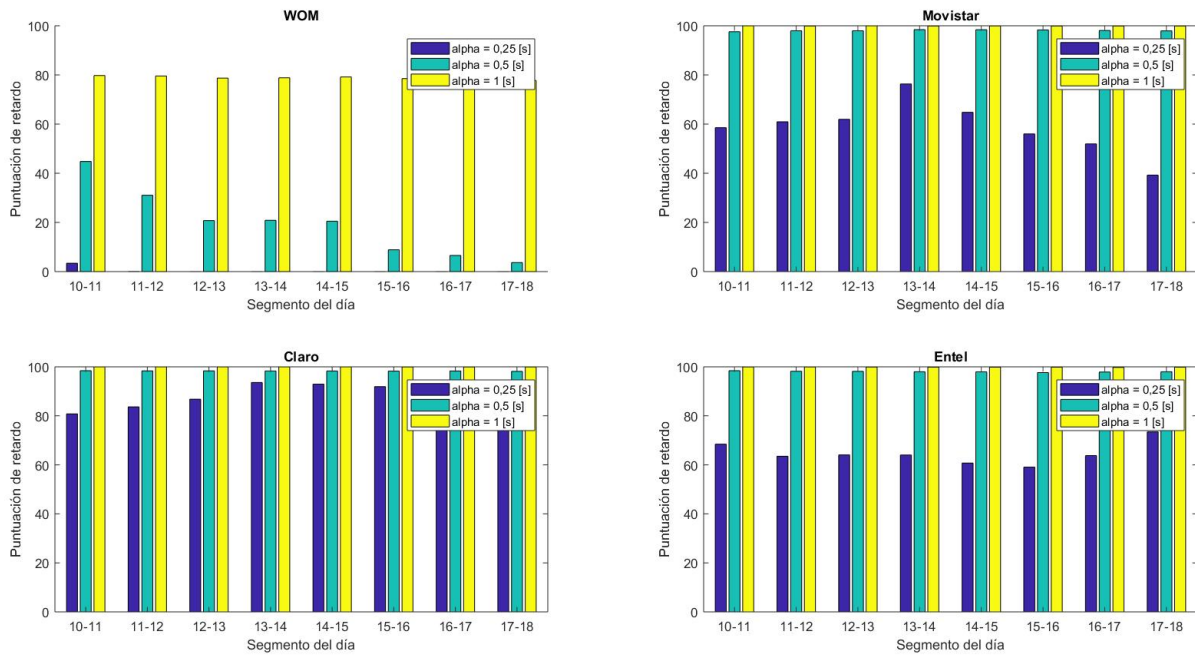


Figura 15: Puntuaciones promedio (no comparables) por operador de la totalidad de sus periodos analizados, para la ventana de tiempo considerada. En azul el porcentaje de retardos registrados menores a 0,25 s, en turquesa los menores a 0,5 s y en amarillo los menores a 1 s. Elaboración propia.

Finalmente, a modo de síntesis de los dos se presentan las tablas 3 y 4, cuya correspondencia con los periodos de análisis es idéntica a la de las Figuras 14 y 15. En estas se presentan los retardos de envío estimados a partir de la mitad de los RTT promedio registrados. Para el caso de la Tabla 4 la diferenciación entre WOM y el resto de las compañías se reproduce. Además, se puede observar que Claro cuenta un retardo promedio 8,88% más breve que Entel y sugiere tener, en promedio, el mejor desempeño.

Operador	Retardo de envío promedio [s]	Std [s]
WOM	0,536	0,288
Movistar	0,251	0,160
Claro	0,246	0,088
Entel	0,270	0,089

Tabla 4: Retardo de envío promedio registrado para un día de cada operador. Se trata del valor para el día 2 de las distintas semanas de medición, en idénticas ventanas de tiempo. Elaboración propia.

Operador	Retardo de envío promedio [s]	Std [s]
WOM	0,498	0,269
Movistar	0,265	0,149
Claro	0,249	0,152
Entel	0,266	0,124

Tabla 5: Retardo de envío promedio registrado para cada operador (no comparables). Se trata de la estimación más completa posible para la totalidad de mediciones durante los periodos analizados. Elaboración propia.

Los desempeños y valores promedios presentados anteriormente permiten esbozar una de las dificultades que enfrenta el esquema de control en esta propuesta de diseño, que refiere a la significativa diferencia entre los valores de retardo de enlace entre ciclistas validados teóricamente y aquellos registrados empíricamente. Para comprenderlo debe tenerse en cuenta que este trabajo considera una implementación en JAVA del algoritmo de cálculo de acciones de control evaluado en [29], cuyos escenarios teóricos de éxito fueron validados para retardos de enlace de 100 ms como máximo. Mientras que, si atendemos al que resulta ser empíricamente el mejor caso en términos de retardo de enlace promedio (para Claro, ver Tabla 4), notamos que el valor máximo teóricamente validado es superado en un factor de 2,46. En otras palabras, el retardo empírico esperado es significativamente superior al doble del teóricamente validado. Al respecto cabe señalar que ninguno de los paquetes monitoreados tuvo un retardo menor a 100 ms.

Las consideraciones anteriores permiten dilucidar la necesidad de evaluar esquemas de comunicación alternativos e iterar sobre el módulo de control en simulaciones con condiciones de retardo de enlace más restrictivas. Lo primero para aproximarse a condiciones más cercanas a las supuestas teóricamente, y lo segundo para buscar ajustes en los ponderadores del funcional cuadrático y en el horizonte de predicción, con miras a controlar caravanas en escenarios de mayor retardo.

9.2 Control de velocidad del líder

Para esta prueba se llevó uno de los prototipos a la explanada de parque O'Higgins y se ejecutó una sesión de líder sin seguidores. Esto para verificar el funcionamiento de la señalización visual y háptica, además de poner a prueba la posibilidad de control de la velocidad del líder.

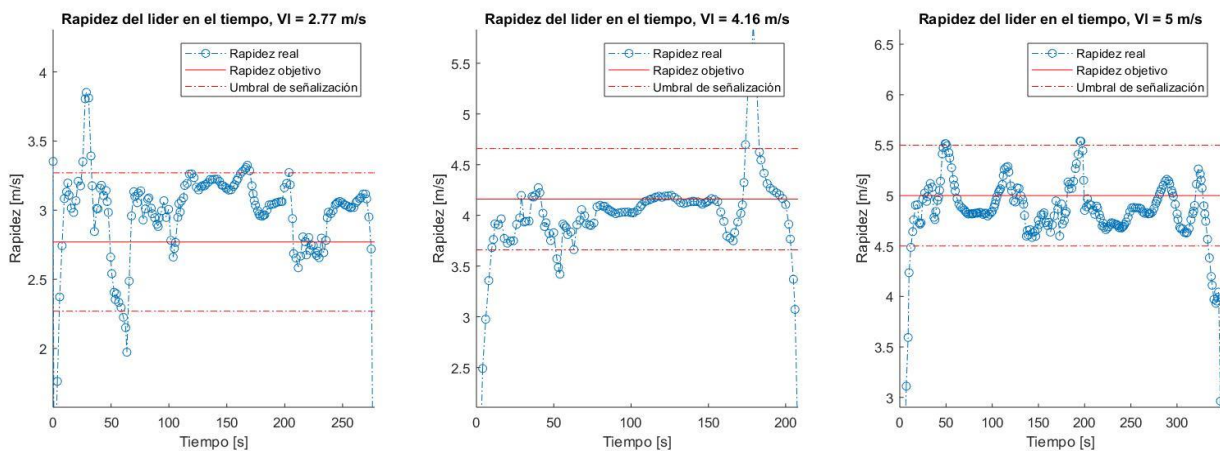


Figura 16: Velocidad del líder (VI) en el tiempo para 3 rapidezces distintas. Elaboración propia.

La Figura 16 exhibe los resultados de esta prueba para 3 rapidezces de líder distintas. En esta se puede apreciar tanto la evolución en el tiempo de la rapidez de la bicicleta, obtenida a partir del GPS, la rapidez objetivo y los umbrales de señalización. Estos últimos determinan si la señal enviada al usuario es de acelerar, no acelerar o desacelerar, considerando un margen de error admitido de 0,5 m/s. Estos resultados sugieren que el prototipo permite mantener al líder de la caravana razonablemente, en la Tabla 6 se ordenan las rapidezces promedio y su desviación estándar para cada caso. Cabe aclarar que los extremos del registro se explican por la partida en cero de la bicicleta y la puesta en cero de ella al final de la prueba.

Objetivo [m/s]	Rapidez promedio [m/s]	Std [m/s]
2,77	2,94	0,46
4,16	3,97	0,62
5	4,77	0,62

Tabla 6: Rapidez promedio del líder y desviaciones estándar para cada una de las rapidezces. Elaboración propia.

9.3. Pruebas del prototipo integrado

En la prueba del prototipo integrado se habilitan todas las funciones del prototipo para un líder y un seguidor. En esta, tanto al seguidor como al líder se les indica obedecer la señalización de la aplicación por sobre su intuición en la formación de la caravana. Solo se indica la excepción al seguidor de desobedecer la señalización en caso de que esta ponga en peligro su integridad y la del líder, esto para el caso en que la aplicación sugiera acelerar incluso cuando los ciclistas se encuentren riesgosamente cerca.

La razón de la cantidad restringida de seguidores se debe a la disponibilidad de voluntarios en el tiempo disponible para la realización de pruebas, no obstante, la aplicación permite el funcionamiento con hasta dos seguidores. Más pruebas son necesarias para comprobar el

funcionamiento del sistema de control en esta modalidad, dado que en las que fue posible realizar se pasó por alto un error de implementación en el registro de la posición en la base de datos que fue corregido posteriormente.

Sobre la configuración del esquema de control en este caso, se tiene que el horizonte de predicción se escoge de acuerdo con el caso de éxito de las pruebas en simulaciones, es decir, a 5 pasos. Los parámetros de sintonización del funcional cuadrático son tratados del mismo modo, en que se le da un ponderador unitario a posiciones, velocidades y aceleraciones. La única diferencia, en este sentido, radica en la posibilidad de variación del espaciado objetivo y en la cantidad de iteraciones consecutivas que deben tenerse dentro de los márgenes de tolerancia de la función objetivo para dar por terminado el cálculo de la acción de control (2000 iteraciones en lugar de 800). Además, debe tenerse en cuenta que el intervalo de tiempo que separa las predicciones se encuentra fijo en 0,5 s, puesto que su modificación implicaba la pérdida de control en el simulador.

Es relevante aclarar que no se realiza un estudio detallado del retardo introducido por el cálculo de la acción de control dado que no se registran tiempos de cómputo mayores a 5 milisegundos en ninguna de las pruebas.

En la Figura 17 se presenta la distancia entre el seguidor y el líder en función del tiempo para distintas distancias objetivo y velocidad de líder de 4,16 m/s. En verde se presentan aquellas muestras en que la señalización indica acelerar y en rojo las de desacelerar. Puede notarse que existe un tramo dentro del cual no existe señalización, que se explica por el intervalo en que el buffer de errores debe llenarse por primera vez. También es evidente una tendencia del prototipo a solicitar la aceleración del usuario, incluso en casos en que la distancia entre las bicicletas es menor a la deseada. Ahora bien, dicha señalización resulta ser más coherente para distancias mayores al objetivo. Una posible explicación de esto es que, al ponderar distancia, velocidad y aceleración de igual modo en el controlador, conforme la distancia se torna menos relevante, mayor es la competencia en términos del control con las demás variables consideradas en el funcional cuadrático (e.g. el controlador puede volverse mucho más sensible a diferencias respecto de la velocidad del líder). Una posible mitigación podría ser la sintonización de dichos parámetros adaptativamente dependiendo de la distancia relativa entre las bicicletas.

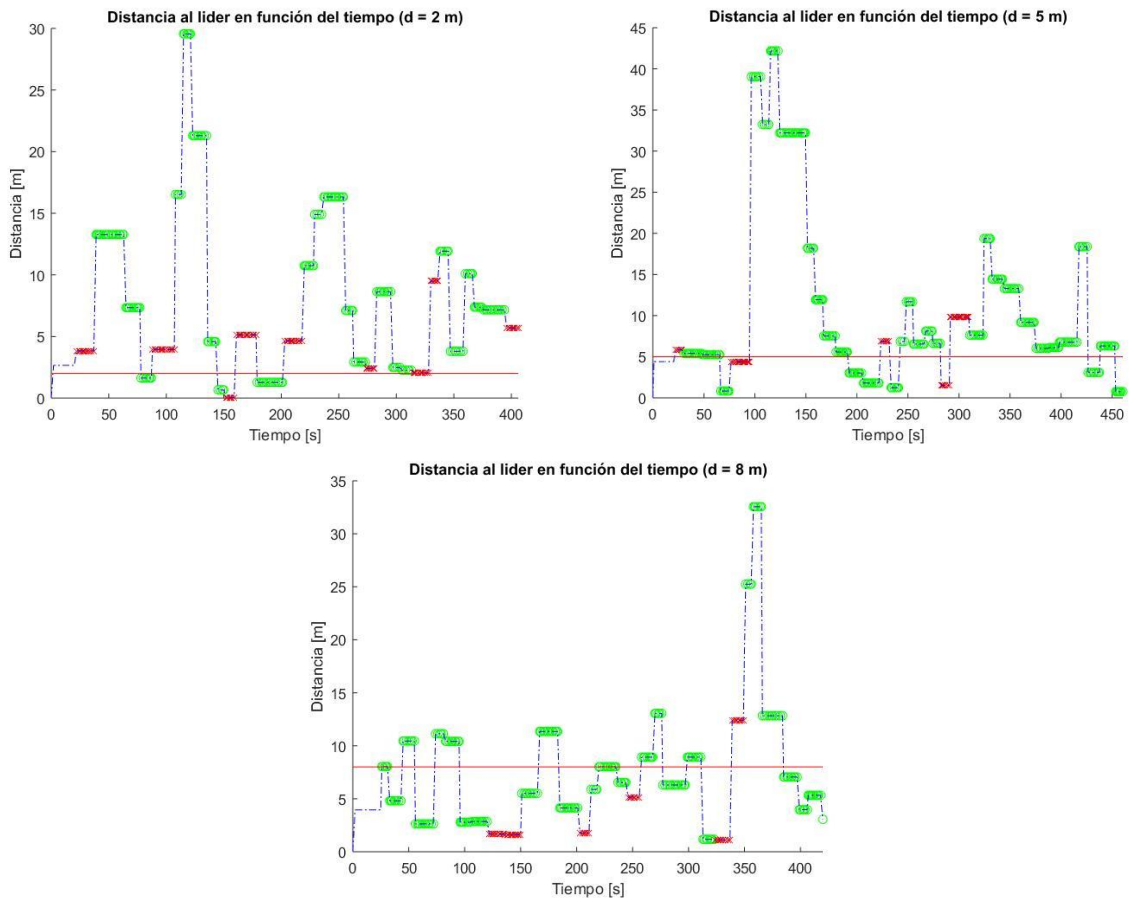


Figura 17: Distancia al líder en función del tiempo para 1 seguidor. Con círculos (o) verdes se muestran los instantes en que la señalización es de aceleración, en cruces (x) rojo de desaceleración. Elaboración propia.

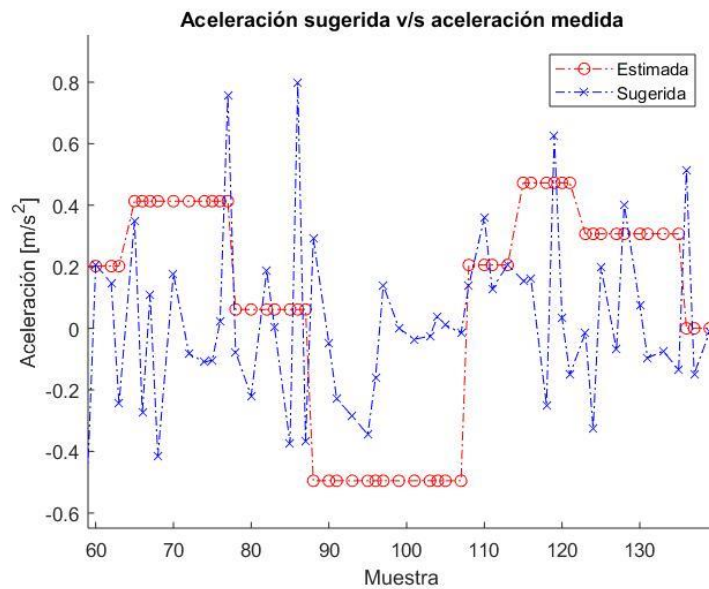


Figura 18: Aceleración sugerida v/s ejecutada. Se trata de un extracto de la sesión con distancia objetivo de 2 metros. Elaboración propia.

La Figura 18 consiste en un extracto de la comparación entre la aceleración recomendada por el control y la estimada por el GPS para un espaciamiento objetivo de 2 m. De esta se puede apreciar otra de las limitaciones de la implementación en relación con la obtención de las variables de estado exclusivamente mediante GPS. A primera vista se puede apreciar la ausencia de distinciones en la aceleración ejecutada por el usuario para varias muestras, cuestión que también se puede apreciar en los gráficos de espaciamiento. Cabe referirse a que, si bien la posición se consulta cada 1 segundo, la actualización de su cálculo por parte de la parte del sistema GPS no es tan veloz, lo que resulta en consultas consecutivas que adquieren un idéntico valor. Este resultado pone en evidencia otra de las dificultades de este diseño, que ya se ha registrado en instancias previas de prueba en la serie de trabajos de Céspedes y Salamanca, aunque en esta oportunidad se trata de una implementación diferente (Google Service Api).

Por completitud se presenta la Figura 19, en que se grafica a la vez la velocidad del líder y del seguidor para la misma prueba. En línea punteada se indica el objetivo establecido para el líder. Sobre esta, se aprecia una coherencia razonable en el control de la velocidad del líder respecto de las pruebas realizadas para dicha funcionalidad por separado.

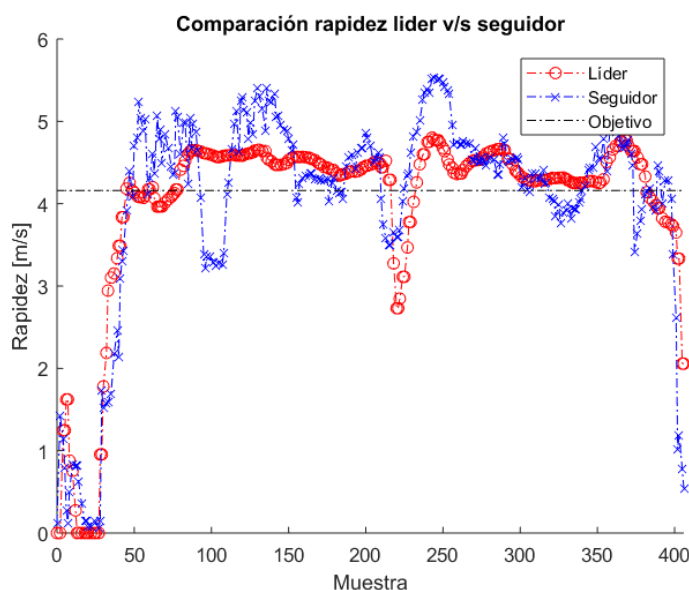


Figura 19: Rapidez seguidor v/s líder. Elaboración propia.

Finalmente, y para futuras referencias, se presentan intervalos de actualización de ubicación GPS medidos durante las pruebas en la Tabla 7. Específicamente, los valores presentados corresponden a los promedios de las diferencias entre estampas de tiempo GPS consecutivas. De ese modo puede verificarse claramente que no solo existen dificultades en la calidad de la estimación de la posición de las bicicletas, sino que también en la efectividad del intervalo de actualización escogido para esta implementación.

D [m]	Tiempo de adquisición [s]	Std [s]
2	1,52	0,50
5	1,49	0,50
8	1,53	0,50

Tabla 7: Intervalo real de adquisición de posición GPS. Elaboración propia.

10. RESULTADOS OBTENIDOS

En línea con el objetivo general y considerando los alcances planteados en los objetivos específicos, el resultado del presente trabajo consiste en un nuevo prototipo para el sistema ciberfísico de control de caravanas de bicicletas inteligentes que contempla:

- Una aplicación Android que integra los módulos de comunicación y control. Adquiere la ubicación GPS mediante la interfaz de aplicación programable de Google, calcula la acción de control para el usuario respectivo y transmite y almacena la información relativa a ciclistas y sesiones en los servicios de nube de Firebase. Realizando la comunicación empleando la red celular 4G.
- Integración de un nuevo módulo de control distribuido inteligente, que opera en base a los principios del modelamiento predictivo difuso y resuelve el correspondiente problema de optimización mediante un algoritmo genético. En un módulo de control que integra el error humano de aceleración en el cálculo de la acción de control.
- Implementación de un enlace inalámbrico Bluetooth Low Energy que permite la comunicación con periféricos situados en la bicicleta compatible con la aplicación desarrollada.
- Integración de una interfaz háptica de que involucra una instrumentación de menor tamaño y costo en la bicicleta, manteniendo a estas independientes de la condición de seguidor o líder de caravana del usuario que se le empareje.
- Elaboración de un reporte que contempla retardos promedio de transmisión y métricas que permiten constatar las condiciones de operación del prototipo para distintos operadores de red, dando cuenta de las limitaciones de la propuesta de diseño para permitir el efectivo funcionamiento del esquema de control implementado. Esto último debido a tiempos de envío significativamente superiores (factor 2,46) a los considerados en la validación teórica del módulo de control y limitaciones en la adquisición de la ubicación GPS mediante el servicio empleado.

11. CONCLUSIONES

A continuación, se presentan las conclusiones del trabajo desarrollado en relación con los objetivos específicos planteados y el grado de cumplimiento del objetivo general del cual fueron derivados.

En primer lugar, se logra una primera caracterización de las condiciones de conectividad bajo las cuales operaría un nuevo esquema de control. Esto mediante la estimación del RTT del prototipo en la transmisión de las variables de estado de los ciclistas, para un mismo par de equipos móviles y los operadores WOM, Movistar, Entel y Claro. Con el que se provee un marco general de tiempos de transmisión esperables en el uso de la aplicación sobre la red LTE/4G durante el transcurso del día. Esta sugiere tanto los mejores operadores como las mejores horas para desarrollar pruebas del prototipo integrado. En ese sentido, se constata que los mejores retardos promedio de enlace registrados resultan al menos 2,46 veces mayores que aquellos considerados en la validación teórica del esquema de control. Por lo que también se identifican con estas pruebas, una de las principales limitaciones de la propuesta.

En segundo lugar, se logra la traducción íntegra del módulo de control a un lenguaje compatible con la aplicación desarrollada, permitiendo la determinación de la sugerencia de la acción de control en el celular de sus usuarios dentro de márgenes de tiempo razonables (menores a 5 ms).

En tercer lugar, se adapta la interfaz humano-bicicleta y el enlace de sus periféricos para admitir la transmisión de sus señales por medios inalámbricos. Lo cual se verifica en las señalizaciones durante las pruebas de campo relativas el control de la velocidad del líder y en la puesta a prueba del prototipo integrado.

En cuarto lugar, se verifica la inviabilidad del esquema de control en el estado actual de implementación durante las pruebas del prototipo integrado. Evidenciado en una señalización excesiva de aceleración, incluso para distancias menores al objetivo. Lo que se explica, por un lado, por la existencia de retardos de red significativamente superiores a los considerados en la validación teórica y, por otro, debido al hecho de que la adquisición de la posición GPS, si bien se realiza en un periodo promedio menor a 1,6 s, el valor adquirido presenta un tiempo de actualización significativamente menor (hasta 5 veces mayor en ocasiones).

En suma, se logra diseñar e implementar un nuevo prototipo ciber-físico para el sistema de control inteligente de caravanas de ciclistas, integrando la tendencia del estado del arte respecto del uso de *smartphones* en el desarrollo de este tipo de aplicaciones, que permite probar el funcionamiento del sistema sobre la red celular y atiende a los errores de interpretación de la señalización por parte de sus usuarios. Lo anterior logrando una mayor flexibilidad sobre la asignación de roles a las bicicletas que la versión anterior.

Finalmente, se propone para trabajos futuros una revisión más profunda de esquemas de comunicación local y evaluar el esquema de control limitando el uso del GPS a la obtención de la ubicación de las bicicletas, derivando a partir de un contador magnético de vueltas su rapidez y aceleración instantánea (para dar mayor fidelidad a los valores de velocidad y aceleración, resuelto en la versión anterior del prototipo). La sintonización del esquema de control mediante la variación del número de pasos de predicción y los ponderadores del funcional cuadrático, buscando mitigar las falencias de su respuesta en espaciamientos próximos al objetivo. Y evaluar el efecto de una sintonización adaptativa de dichos ponderadores dependiendo de la distancia entre las bicicletas.

Además, se propone considerar tecnologías alternativas para la determinación de las distancias entre bicicletas, como aquellas basadas en Banda Ultra Ancha (o por sus siglas en inglés UWB).

12. BIBLIOGRAFÍA

- [1] K. C. Heesch and S. Sahlqvist, “Key influences on motivations for utility cycling (cycling for transport to and from places),” *Heal. Promot. J. Aust.*, vol. 24, no. 3, pp. 227–233, 2013.
- [2] K. C. Heesch and G. Turrell, “Key influences on cycling for transport,” *J. Australas. College Road Saf.*, vol. 25, no. 4, pp. 43–46, 2014.
- [3] Ptovaras.cl, “Vecinos y visitantes cuentan con servicio gratuito de bicicletas”, 2018. [Online]. Disponible en: <http://www.ptovaras.cl/noticias/vecinos-y-visitantes-cuentan-con-servicio-gratuito-de-bicicletas>. [Accedido: 2 – Jun - 2019].
- [4] Bicilascondes.cl, “¿Qué es Bici Las Condes?”. [Online]. Disponible en: <https://www.bicilascondes.cl/es/servicio/que-es-bici-las-condes>. [Accedido: 2 – Jun - 2019].
- [5] Recoleta.cl, “Recoleta da el primer paso para tener bicicletas públicas”, 2014. [Online]. Disponible en: <https://www.recoleta.cl/recoleta-da-el-primer-paso-para-tener-bicicletas-publicas/>. [Accedido: 2 – Jun - 2019].
- [6] U. Hernandez-Jayo, I. De-La-Iglesia, and J. Perez, “V-alert: Description and validation of a vulnerable road user alert system in the framework of a smart city,” *Sensors (Switzerland)*, vol. 15, no. 8, pp. 18480–18505, 2015.
- [7] H. Kato, R. I. Takahashi, Y. Sakajyo, and S. Kaneda, “Visualization Method for Bicycle Rider Behavior Analysis Using a Smartphone,” *Proc. - Int. Comput. Softw. Appl. Conf.*, vol. 2, pp. 354–359, 2017.
- [8] G. Liao and J. Zhang, “Information query for public bicycle service based on Andriod,” *Proc. 2014 IEEE Int. Conf. Secur. Pattern Anal. Cybern. SPAC 2014*, no. 61303143, pp. 126–129, 2014.
- [9] S. Sweeney, R. Ordonez-Hurtado, F. Pilla, G. Russo, D. Timoney, and R. Shorten, “Cyberphysics, pollution mitigation, and pedelecs,” no. August 2017, pp. 1–24, 2017.
- [10] H. Dambeck. (Aug. 8, 2013). Too Many Bikes? Challenges in Europe’s Cycling Boom. Accessed: Oct. 22, 2018. [Online]. Available: <http://www.spiegel.de/international/zeitgeist/bike-boom-popularity-of-cycling-leads-to-overcrowding-a-915598.html>
- [11] Horton, D., Rosen, P., Cox, P. (Eds.), 2007. *Cycling and Society*, Illustrate. Ed. Ashgate.
- [12] Koglin, T., 2015. Velomobility and the politics of transport planning. *GeoJournal* 80 (4), 569–586
- [13] F. Behrendt, “Why cycling matters for Smart Cities. Internet of Bicycles for Intelligent Transport,” *J. Transp. Geogr.*, vol. 56, pp. 157–164, 2016.
- [14] Townsend, A.M., 2013. *Smart Cities*. W. W. Norton & Company, New York.
- [15] D. Jia, K. Lu, J. Wang, X. Zhang, and X. Shen, “A survey on platoon-based vehicular cyber-physical systems,” *IEEE Commun. Surv. Tutorials*, vol. 18, no. 1, pp. 263–284, 2016.
- [16] O. M. F. Abu-Sharkh and Z. Dabain, “GreenBikeNet: an Intelligent Mobile Application with Green Wireless Networking for Cycling in Smart Cities,” *Mob. Networks Appl.*, vol. 21, no. 2, pp. 352–366, 2016.

- [17] C. Portilla, F. Valencia, J. Espinosa, A. Núñez, and B. De Schutter, "Model-based predictive control for bicycling in urban intersections," *Transp. Res. Part C Emerg. Technol.*, vol. 70, pp. 27–41, 2016.
- [18] Y. X. Zhao, Y. S. Su, and Y. C. Chang, "A Real-Time Bicycle Record System of Ground Conditions Based on Internet of Things," *IEEE Access*, vol. 5, pp. 17525–17533, 2017.
- [19] J. Iwasaki, A. Yamamoto, and S. Kaneda, "Road information-sharing system for bicycle users using smartphones," 2015 IEEE 4th Glob. Conf. Consum. Electron. GCCE 2015, pp. 674–678, 2016.
- [20] J. A. Brizuela Mendoza, C. M. Astorga Zaragoza, A. Zavala Rio, and F. Canales Abarca, "Instrumentation of a bicycle without driver for experimental testing of automatic control," *IEEE Lat. Am. Trans.*, vol. 14, no. 8, pp. 3540–3546, 2016.
- [21] I. Tal, B. Ciubotaru, and G. M. Muntean, "Vehicular-Communications-Based Speed Advisory System for Electric Bicycles," *IEEE Trans. Veh. Technol.*, vol. 65, no. 6, pp. 4129–4143, 2016.
- [22] L. W. Chen, Y. H. Peng, Y. C. Tseng, and M. F. Tsai, "Cooperative sensing data collection and distribution with packet collision avoidance in mobile long-thin networks," *Sensors (Switzerland)*, vol. 18, no. 10, pp. 1–15, 2018.
- [23] W. Jeon and R. Rajamani, "Rear vehicle tracking on a bicycle using active sensor orientation control," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 8, pp. 2638–2649, 2018.
- [24] S. Smaldone, C. Tonde, V. K. Ananthanarayanan, A. Elgammal, and L. Iftode, "The Cyber-Physical bike: A step towards safer green transportation," *HotMobile 2011 12th Work. Mob. Comput. Syst. Appl.*, pp. 56–61, 2011.
- [25] S. Cespedes, J. Salamanca, J. C. Sacanamboy, and C. Rivera, "Poster: Smart networked bicycles with platoon cooperation," *IEEE Veh. Netw. Conf. VNC*, vol. 2015-Janua, no. January, pp. 199–200, 2015.
- [26] S. Céspedes, J. Salamanca, A. Yañez, C. Rivera, and J. C. Sacanamboy, "Platoon-based cyclists cooperative system," *IEEE Veh. Netw. Conf. VNC*, vol. 2016-Janua, pp. 112–118, 2016.
- [27] J. Salamanca, S. Céspedes, D. Vinasco, and A. Yanez, "Demo: A prototype for a platoon-based cyclist cooperative system," *IEEE Veh. Netw. Conf. VNC*, pp. 0–1, 2017.
- [28] S. Cespedes, J. Salamanca, A. Yanez, and D. Vinasco, "Group Cycling Meets Technology: A Cooperative Cycling Cyber-Physical System," *IEEE Trans. Intell. Transp. Syst.*, pp. 1–11, 2018.
- [29] T. Lara, O. Cartagena, S. Céspedes and D. Sáez, "Robust Model-based Predictive Control for a Cooperative Cycling Cyber-physical System," 2019 IEEE CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON), Valparaiso, Chile, 2019, pp. 1-6.

13. ANEXOS

13.1. Implementación del método *onLocationChanged*.

// Este método es llamado cada vez que se vuelve efectiva una actualización de ubicación una vez transcurrido el intervalo de tiempo configurado.

```
public void onLocationChanged(Location location) {  
  
    if (location != null && sessionState) {  
        // Se actualiza la ubicación con la última adquirida  
        this.location = location;  
        // Se determina la aceleración efectuada desde la última iteración  
        MyAcc.updateAcc(location);  
        // Se calcula y envía a la interfaz la acción de control sugerida  
        suggestionUpdate();  
        // Se actualiza el estado de la bicicleta en la base de datos  
        toRTFirebase(location.getLatitude(), location.getLongitude(), location.getSpeed(),  
location.getTime());  
    }  
}
```

13.2. Implementación de los métodos *linkSetting* y *childAdded*.

```
public void linkSetting() {  
    // Referencia a la base de datos.  
    usersRef = database.getReference();  
    // Referencia a la sesión específica.  
    usersRef = usersRef.child(Integer.toString(sessionNumber));  
    // Ejemplo : si soy el seguidor 1, entonces permaneceré atento a las  
    // actualizaciones del seguidor 0.  
    if (USER_TYPE == "FOLLOWER1") {  
        // "UpLink" a la base de datos.  
        userType = usersRef.child("FOLLOWER1");  
        // "DownLink" desde la base de datos.  
        downLink = usersRef.child("FOLLOWER0");  
    }  
    else if (USER_TYPE == "FOLLOWER0") {  
        userType = usersRef.child("FOLLOWER0");  
        downLink = usersRef.child("LEADER");  
    }  
    else if (USER_TYPE == "LEADER") {  
        userType = usersRef.child("LEADER");  
        // Empleado en la medición del RTT.  
        //downLink = usersRef.child("FOLLOWER0");  
    }  
    if (USER_TYPE != "LEADER") {  
        // En general, solo los seguidores están atentos a actualizaciones.  
        childAdded();  
    } else {  
        // Empleado en la medición del RTT.  
        //childAdded();  
    }  
}  
}  
(...)  
public void childAdded() {  
    // Se incorpora el escuchador de eventos.  
    downLink.addChildEventListener(new ChildEventListener() {  
        @Override  
        public void onChildAdded(DataSnapshot dataSnapshot, String prevChildKey) {  
            previousPost = dataSnapshot.getValue(Post.class);  
            // Se establece a nivel de clase que hay una nueva  
            // actualización.  
            setNewChild(true);  
            // Empleado en la medición de RTT.  
            /*if (USER_TYPE != "LEADER") {  
            } else {  
            }  
            }  
        }  
    }  
}
```

```

        RTT = System.currentTimeMillis() -
previousPost.getSendTime();
    }*/
}
(...)
});.

```

13.3. Implementación del método *bleSend*.

```

public void bleSend(double u_k) {
    if (USER_TYPE != "LEADER") { //FOLLOWER CASE

        (...)

        if (u_k > mean_acc + 0.5 * std_acc) {
            // Notificación a la interfaz háptica
            send("{+}");
            // Notificación mediante interfaz visual
            compass.setImageResource(R.drawable.speed_up_compass);
        } else if (u_k < mean_acc - 0.5 * std_acc) {
            send("{-}");
            compass.setImageResource(R.drawable.slow_down_compass);
        } else {
            send("{0}");
            compass.setImageResource(R.drawable.inicial_compass);
        }
    } else { //LEADER CASE
        if (Math.abs(location.getSpeed() - desiredLeaderSpeed) > 0.5) {
            if (location.getSpeed() - desiredLeaderSpeed < 0) {
                send("{+}");
                compass.setImageResource(R.drawable.speed_up_compass);
            } else if (location.getSpeed() - desiredLeaderSpeed > 0) {
                send("{-}");
                compass.setImageResource(R.drawable.slow_down_compass);
            }
        } else {
            send("{0}");
            compass.setImageResource(R.drawable.inicial_compass);
        }
    }
}
}

```