



UNIVERSIDAD NACIONAL DEL SUR

Tesis Doctor en Ingeniería

**Desarrollo de Sistemas Computacionales para Sustento  
de Tareas Agropecuarias en el Sudoeste de la  
Provincia de Buenos Aires**

Matías Javier Micheletto

BAHIA BLANCA

ARGENTINA

2020





UNIVERSIDAD NACIONAL DEL SUR

Tesis Doctor en Ingeniería

**Desarrollo de Sistemas Computacionales para Sustento  
de Tareas Agropecuarias en el Sudoeste de la  
Provincia de Buenos Aires**

Matías Javier Micheletto

BAHIA BLANCA

ARGENTINA

2020





## Prefacio

Esta tesis se presenta como parte de los requisitos para optar al grado Académico de Doctor en Ingeniería, de la Universidad Nacional del Sur y no ha sido presentada previamente para la obtención de otro título en esta Universidad u otra. La misma contiene los resultados obtenidos en investigaciones llevadas a cabo en el Laboratorio de Sistemas Digitales dependiente del Departamento de Ingeniería Eléctrica y de Computadoras, y el Laboratorio de Suelos Plantas y Atmósfera dependiente del Departamento de Agronomía durante el período comprendido entre el 01 de Abril de 2016 y el 28 de Septiembre de 2020, bajo la dirección de los profesores Dr. Rodrigo M. Santos, profesor adjunto de Computadoras Digitales, Dr. Juan A. Galantini, investigador independiente CIC.

---

Ing. Matías J. Micheletto



UNIVERSIDAD NACIONAL DEL SUR  
Secretaría General de Posgrado y Educación Continua

La presente tesis ha sido aprobada el ....../...../..... ,  
mereciendo la calificación de .....(.....)



## Agradecimientos

A la Universidad Nacional del Sur y al Departamento de Ingeniería Eléctrica y Computadoras, por brindarme el espacio y los medios que me permitieron desenvolverse con comodidad durante este periodo doctoral.

A mis directores, Dr. Rodrigo Santos y Dr. Juan Galantini por la confianza, la paciencia y las incontables horas de su tiempo que han dedicado a guiarme durante el desarrollo de esta tesis.

A quienes forman parte del Laboratorio de Sistemas Digitales, del Laboratorio de Suelos Plantas y Ambiente y de la Estación Experimental INTA de Hilario Ascasubi, por lo mucho que me han ayudado en tantas cosas.

Finalmente, y no menos importante, a mis familiares y amigos, por el acompañamiento y por estar siempre presentes.



## Resumen

El desarrollo de tecnología agropecuaria involucra actores de todas las disciplinas científicas, principalmente de las ciencias naturales, seguido por las ciencias formales y en menor medida, las ciencias humanas o sociales. Es a partir de la colaboración interdisciplinaria de distintos grupos de investigación que permitió la elaboración del presente trabajo, en el cual se abordan diversas cuestiones respecto de la incorporación de sistemas computacionales con distintas funcionalidades y características en pos de brindar soporte a las actividades agropecuarias, particularmente aquellas que predominan en el sector del Sudoeste de la Provincia de Buenos Aires. Se intenta cubrir los aspectos más relevantes referidos al diseño e implementación de dichos sistemas, desde un punto de vista técnico y con un especial enfoque hacia el empleo de metodologías de diseño con cierto grado de optimalidad.

El eje central de la tesis se divide en tres partes con la finalidad de separar las problemáticas relacionadas a la adquisición, transmisión y el análisis o procesamiento de los datos. En la cuarta y anteúltima parte de este trabajo, se integran los conceptos anteriormente expuestos y se exploran las arquitecturas donde interactúan sistemas más complejos del que forman parte tanto las máquinas como las personas. Dentro de este último grupo se engloban los sistemas colaborativos, para los cuales se analizan dos propuestas diferentes, con un enfoque hacia los aspectos de comunicación por un lado y por el otro, el diseño de un sistema que tiene el objetivo de mejorar cuestiones de logística en el sector agropecuario, especialmente en zona rural.

## Abstract

The development of agricultural technology involves actors from all scientific disciplines, mainly from the natural sciences, followed by the formal sciences and to a lesser extent, the human or social sciences. It is from the interdisciplinary collaboration of different research groups that allowed the accomplishment of this work, in which certain issues are addressed regarding the incorporation of computer systems with different functionalities and characteristics in order to provide support to agricultural activities, particularly those that predominate in the Southwest sector of the Buenos Aires Province. It is intended to cover all technical aspects related to the design and implementation of said systems, from a technical point of view and with a special focus on the use of design methodologies with a certain degree of optimality.

The central axis of the current thesis is divided into three parts in order to separate the problems related to the data acquisition, transmission and analysis or processing. In the fourth and penultimate part of this work, the previously exposed concepts are integrated and the architectures where more complex systems interacts, of which both machines and people are part, are explored. Collaborative systems are included in this last group, for which two different proposals are analyzed, with a focus on communication aspects on the one hand and on the other, the design of a system that aims to improve logistics issues in the agricultural sector, especially in rural areas.

# Índice general

|           |                                                             |           |
|-----------|-------------------------------------------------------------|-----------|
| <b>I</b>  | <b>Introducción</b>                                         | <b>2</b>  |
| <b>1.</b> | <b>Marco de referencia</b>                                  | <b>3</b>  |
| 1.1.      | Motivación . . . . .                                        | 3         |
| 1.2.      | Estado del arte . . . . .                                   | 6         |
| 1.3.      | Objetivos . . . . .                                         | 7         |
| 1.4.      | Contribuciones . . . . .                                    | 8         |
| 1.4.1.    | Artículos de revista de alcance internacional . . . . .     | 8         |
| 1.4.2.    | Artículos en conferencias . . . . .                         | 9         |
| 1.4.3.    | Resumen de detalles . . . . .                               | 9         |
| <b>2.</b> | <b>Modelado de sistemas</b>                                 | <b>11</b> |
| 2.1.      | Motivación . . . . .                                        | 11        |
| 2.2.      | Casos de uso . . . . .                                      | 12        |
| 2.3.      | Modelado de requerimientos . . . . .                        | 15        |
| 2.4.      | Diseño de software basado en requerimientos . . . . .       | 16        |
| 2.4.1.    | Diagrama de actividades: enfoque básico . . . . .           | 16        |
| 2.4.2.    | Método de diseño . . . . .                                  | 16        |
| 2.5.      | Organización arquitectural . . . . .                        | 18        |
| 2.6.      | Consideraciones finales . . . . .                           | 18        |
| <b>II</b> | <b>Adquisición de datos</b>                                 | <b>20</b> |
| <b>3.</b> | <b>Sistemas embebidos</b>                                   | <b>21</b> |
| 3.1.      | Motivación . . . . .                                        | 21        |
| 3.2.      | Registrador de esfuerzos para maquinaria agrícola . . . . . | 22        |
| 3.2.1.    | Requerimientos . . . . .                                    | 23        |
| 3.2.2.    | Hardware . . . . .                                          | 24        |
| 3.2.2.1.  | Medición de RPM . . . . .                                   | 25        |
| 3.2.2.2.  | Medición de esfuerzos de tracción . . . . .                 | 25        |
| 3.2.2.3.  | Registro de datos . . . . .                                 | 25        |
| 3.2.2.4.  | Fuente de alimentación . . . . .                            | 26        |
| 3.2.2.5.  | Montaje . . . . .                                           | 26        |
| 3.2.3.    | Firmware . . . . .                                          | 26        |
| 3.2.4.    | Software . . . . .                                          | 27        |

|            |                                                                |           |
|------------|----------------------------------------------------------------|-----------|
| 3.2.5.     | Resultados y discusión . . . . .                               | 28        |
| 3.3.       | Escaner CNC LiDAR para análisis de cobertura vegetal . . . . . | 28        |
| 3.3.1.     | Hardware: sistema mecánico . . . . .                           | 29        |
| 3.3.2.     | Hardware: sistema electrónico . . . . .                        | 30        |
| 3.3.3.     | Firmware . . . . .                                             | 31        |
| 3.3.4.     | Software . . . . .                                             | 31        |
| 3.3.5.     | Pruebas experimentales . . . . .                               | 32        |
| 3.3.6.     | Resultados y discusión . . . . .                               | 34        |
| 3.4.       | Consideraciones finales . . . . .                              | 35        |
| <b>4.</b>  | <b>Aplicaciones móviles</b>                                    | <b>36</b> |
| 4.1.       | Motivación . . . . .                                           | 36        |
| 4.2.       | Soporte para tareas de pulverización agrícola . . . . .        | 37        |
| 4.2.1.     | Requerimientos generales . . . . .                             | 38        |
| 4.2.2.     | Parámetros de aplicación . . . . .                             | 38        |
| 4.2.3.     | Verificación de picos . . . . .                                | 39        |
| 4.2.4.     | Calculadora de mezclas . . . . .                               | 40        |
| 4.2.4.1.   | Cantidad de cargas . . . . .                                   | 40        |
| 4.2.4.2.   | Cantidad de producto . . . . .                                 | 41        |
| 4.3.       | Soporte para cálculos de siembra . . . . .                     | 41        |
| 4.3.1.     | Requerimientos . . . . .                                       | 42        |
| 4.3.1.1.   | Requerimientos generales . . . . .                             | 42        |
| 4.3.2.     | Arquitectura . . . . .                                         | 43        |
| 4.3.3.     | Parámetros de labor y verificación de siembra . . . . .        | 45        |
| 4.3.4.     | Calculador de insumos y generación de reportes . . . . .       | 47        |
| 4.4.       | Consideraciones finales . . . . .                              | 48        |
| <b>5.</b>  | <b>Sistemas de tiempo real</b>                                 | <b>49</b> |
| 5.1.       | Motivación . . . . .                                           | 49        |
| 5.2.       | Modelo del sistema . . . . .                                   | 50        |
| 5.3.       | Eficiencia de Pareto . . . . .                                 | 52        |
| 5.4.       | Métodos de optimización . . . . .                              | 53        |
| 5.4.1.     | Búsqueda aleatoria . . . . .                                   | 53        |
| 5.4.2.     | Enjambre de partículas . . . . .                               | 54        |
| 5.4.3.     | Algoritmos genéticos . . . . .                                 | 55        |
| 5.5.       | Evaluaciones experimentales . . . . .                          | 55        |
| 5.6.       | Consideraciones finales . . . . .                              | 57        |
| <b>III</b> | <b>Transmisión de datos</b>                                    | <b>59</b> |
| <b>6.</b>  | <b>Redes de comunicación de alta latencia</b>                  | <b>60</b> |
| 6.1.       | Motivación . . . . .                                           | 60        |
| 6.2.       | Modelo del sistema . . . . .                                   | 61        |
| 6.3.       | Planificación y ruteo de mensajes . . . . .                    | 62        |

|            |                                                               |           |
|------------|---------------------------------------------------------------|-----------|
| 6.4.       | Cálculo de la trama óptima . . . . .                          | 64        |
| 6.5.       | Validación . . . . .                                          | 66        |
| 6.6.       | Consideraciones finales . . . . .                             | 68        |
| <b>7.</b>  | <b>Redes de comunicación con topología configurable</b>       | <b>70</b> |
| 7.1.       | Modelo del sistema . . . . .                                  | 71        |
| 7.1.1.     | Topología adaptativa basada en ahorro de energía . . . . .    | 72        |
| 7.2.       | Evaluación de rendimiento . . . . .                           | 73        |
| 7.3.       | Consideraciones finales . . . . .                             | 74        |
| <b>IV</b>  | <b>Procesamiento de datos</b>                                 | <b>76</b> |
| <b>8.</b>  | <b>Análisis predictivo</b>                                    | <b>77</b> |
| 8.1.       | Motivación . . . . .                                          | 77        |
| 8.2.       | Objetivos . . . . .                                           | 78        |
| 8.3.       | Materiales y Métodos . . . . .                                | 78        |
| 8.3.1.     | Esquema de medición . . . . .                                 | 78        |
| 8.3.2.     | Análisis de los datos . . . . .                               | 79        |
| 8.3.3.     | Modelo del sistema . . . . .                                  | 80        |
| 8.3.4.     | Simulación . . . . .                                          | 81        |
| 8.3.5.     | Análisis de los resultados . . . . .                          | 83        |
| 8.4.       | Consideraciones finales . . . . .                             | 84        |
| <b>9.</b>  | <b>Análisis descriptivo</b>                                   | <b>85</b> |
| 9.1.       | Motivación . . . . .                                          | 85        |
| 9.2.       | Adquisición de los datos . . . . .                            | 86        |
| 9.3.       | Características de los tratamientos . . . . .                 | 87        |
| 9.4.       | Estimación de materia seca . . . . .                          | 89        |
| 9.5.       | Clasificación automática de tratamientos . . . . .            | 90        |
| 9.6.       | Consideraciones finales . . . . .                             | 92        |
| <b>V</b>   | <b>Sistemas colaborativos</b>                                 | <b>94</b> |
| <b>10.</b> | <b>Soporte LPWAN en entornos colaborativos de tiempo real</b> | <b>95</b> |
| 10.1.      | Motivación . . . . .                                          | 95        |
| 10.2.      | Modelo del sistema . . . . .                                  | 96        |
| 10.2.1.    | Interacciones . . . . .                                       | 97        |
| 10.3.      | Implementación . . . . .                                      | 98        |
| 10.3.1.    | Unidades de asistencia . . . . .                              | 98        |
| 10.3.2.    | Dispositivos móviles . . . . .                                | 99        |
| 10.3.3.    | Refugios . . . . .                                            | 100       |
| 10.3.4.    | Centro de operaciones . . . . .                               | 100       |
| 10.4.      | Despliegue de la red . . . . .                                | 101       |
| 10.5.      | Pruebas experimentales . . . . .                              | 102       |



|                                                                              |                |
|------------------------------------------------------------------------------|----------------|
| 10.6. Consideraciones finales . . . . .                                      | 103            |
| <b>11.Sistema colaborativo para el mantenimiento de redes viales rurales</b> | <b>104</b>     |
| 11.1. Motivación . . . . .                                                   | 104            |
| 11.2. Modelo del Sistema: Casos de usos . . . . .                            | 106            |
| 11.2.1. Arquitectura . . . . .                                               | 106            |
| 11.2.2. Actores . . . . .                                                    | 107            |
| 11.3. Composición del Sistema . . . . .                                      | 108            |
| 11.3.1. Dataloggers de vehículos . . . . .                                   | 108            |
| 11.3.2. Estaciones meteorológicas . . . . .                                  | 109            |
| 11.3.3. Dataloggers de maquinaria vial . . . . .                             | 110            |
| 11.3.4. Sistema de información geográfica . . . . .                          | 110            |
| 11.3.5. Modelo de datos y algoritmos . . . . .                               | 111            |
| 11.3.6. Central de procesamiento . . . . .                                   | 112            |
| 11.4. Discusión . . . . .                                                    | 115            |
| 11.5. Consideraciones finales . . . . .                                      | 116            |
| <br><b>VI Conclusión</b>                                                     | <br><b>117</b> |
| <b>12.Conclusiones y trabajo futuro</b>                                      | <b>118</b>     |
| <b>A. Algoritmos de optimización</b>                                         | <b>127</b>     |
| A.1. Optimización por enjambre de partículas . . . . .                       | 127            |
| A.2. Algoritmos genéticos . . . . .                                          | 128            |

# Parte I

## Introducción

# Capítulo 1

## Marco de referencia

### 1.1. Motivación

Los registros históricos de economía mundial de los últimos 70 años muestran una tendencia constante de reducción de costos de producción y de participación del empleo total en el sector agrícola [1, 2]. Para el caso de Argentina, el porcentaje total de empleo dedicado al campo era del 27.5 % en 1950, reduciéndose a tan solo el 6.5 % en 2011. Esta tendencia se repite en prácticamente todos los países del mundo, como puede apreciarse en la Figura 1.1, donde se evidencia la evolución de los índices de empleo agrícola de los distintos países vs. el PBI per cápita de cada Nación.

El valor de tierra arable necesaria para producir una cantidad constante de cultivos se calcula como la superficie de tierra cultivable dividida entre el índice de producción de cultivos (PIN, por sus siglas en inglés). El PIN es la suma de toda la materia prima o commodities que se obtienen de los cultivos, descontando aquellos que son utilizados para alimentación animal, ponderado por los valores de sus precios. Este indicador también presenta un ritmo de decrecimiento casi constante a nivel mundial y en Argentina no ha sido diferente, como puede observarse en la Figura 1.2. Hoy en día, el mundo puede producir tres veces más de volumen de cultivos que hace 60 años, empleando la misma superficie de tierra y la tendencia indica que estos valores continuarán incrementándose.

La disminución de la proporción de mano de obra en el sector agropecuario al mismo tiempo del incremento en los índices de producción, no es más que una consecuencia de la constante incorporación de tecnología a la producción agrícola. En cualquier contexto, el desarrollo tecnológico tiene como objetivo incrementar el volumen de bienes o productos obtenidos a partir de algún proceso disminuyendo el uso de insumos y de energía. Esta relación entre costos y beneficios también puede ser descripta por el concepto de eficiencia productiva o económica.

La tecnología agropecuaria se compone de todo el conocimiento aplicado en función de incrementar esta eficiencia con la que se extrae la materia prima de la tierra para la producción de alimentos y abarca prácticamente todas las disciplinas científicas, principalmente las ciencias naturales, seguido por las ciencias formales y finalmente las ciencias sociales.

Las ciencias naturales aportan el conocimiento que permite entender las necesidades de las especies animales y vegetales, que son el principal producto de las actividades agropecua-

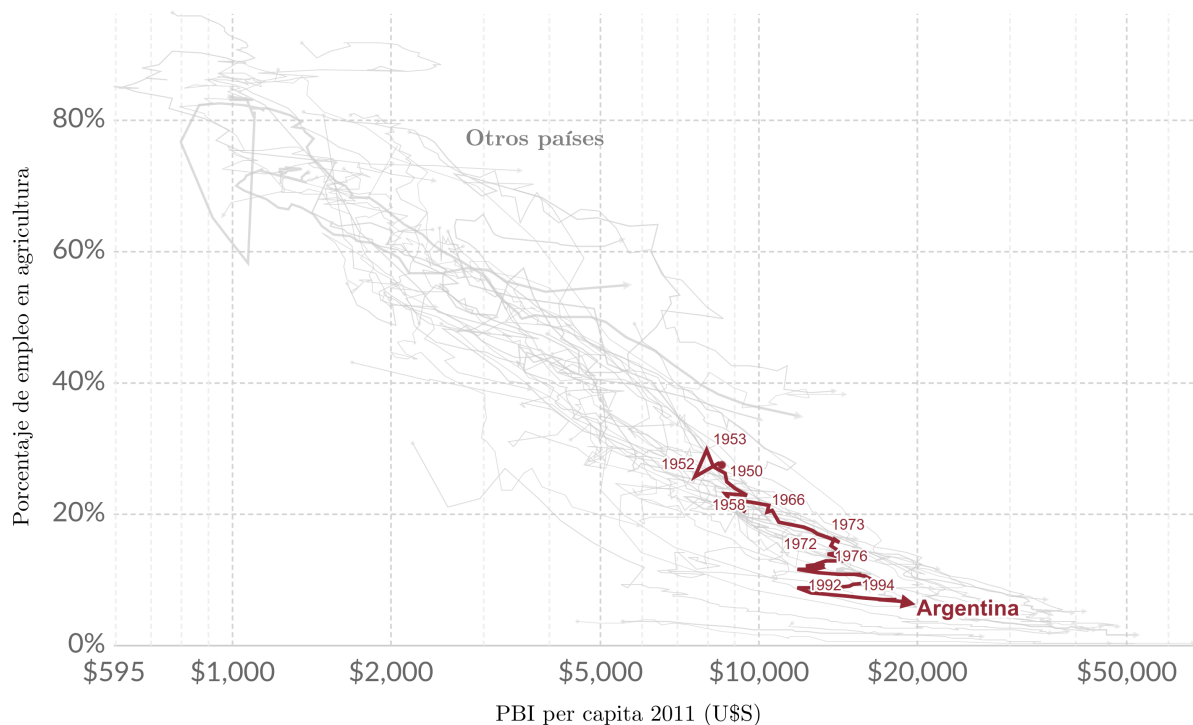


Figura 1.1: Porcentaje de empleo agrícola respecto del total vs. PBI per cápita en dólares de cada país, resaltando la posición de Argentina desde 1950 a 2011 [1].

rias. Las ciencias formales por su parte, colaboran con el desarrollo de materiales, técnicas, mecanismos, máquinas y sistemas computacionales, entre otros. Por último, las ciencias sociales estudian la organización de las empresas e instituciones, la coordinación del trabajo colaborativo y todo lo inherente a la educación y a la forma en que se relacionan las personas del ámbito agropecuario.

En términos generales, los sistemas computacionales manipulan datos en formato digital, esto es, representados mediante un código binario, en pos de resolver tareas y obtener información del medio como resultado de ciertos procedimientos lógicos. Particularmente los sistemas computacionales modernos aplicados a la producción agrícola se enfocan en tres aspectos principales que involucran la adquisición, transmisión y procesamiento de los datos. En este sentido, la adquisición de los datos consiste en extraer información del medio ambiente y cuantificarla en variables de formato digital. La transmisión de los datos, que significa transportarlos de un lugar a otro para contar con su disponibilidad en lugares físicamente separados y en lapsos de tiempos acotados, involucra muchos aspectos técnicos, pero que de manera general se basan en diseñar una infraestructura y protocolos de comunicación que maximicen la tasa de transmisión de los datos minimizando costos, principalmente de energía. Finalmente, el procesamiento de estos datos consiste en extraer información relevante a partir de lo adquirido del medio ambiente y para lograr este objetivo se cuenta con un amplio espectro de métodos y algoritmos.

De manera transversal a estas tres etapas por las que viaja la información dentro de los

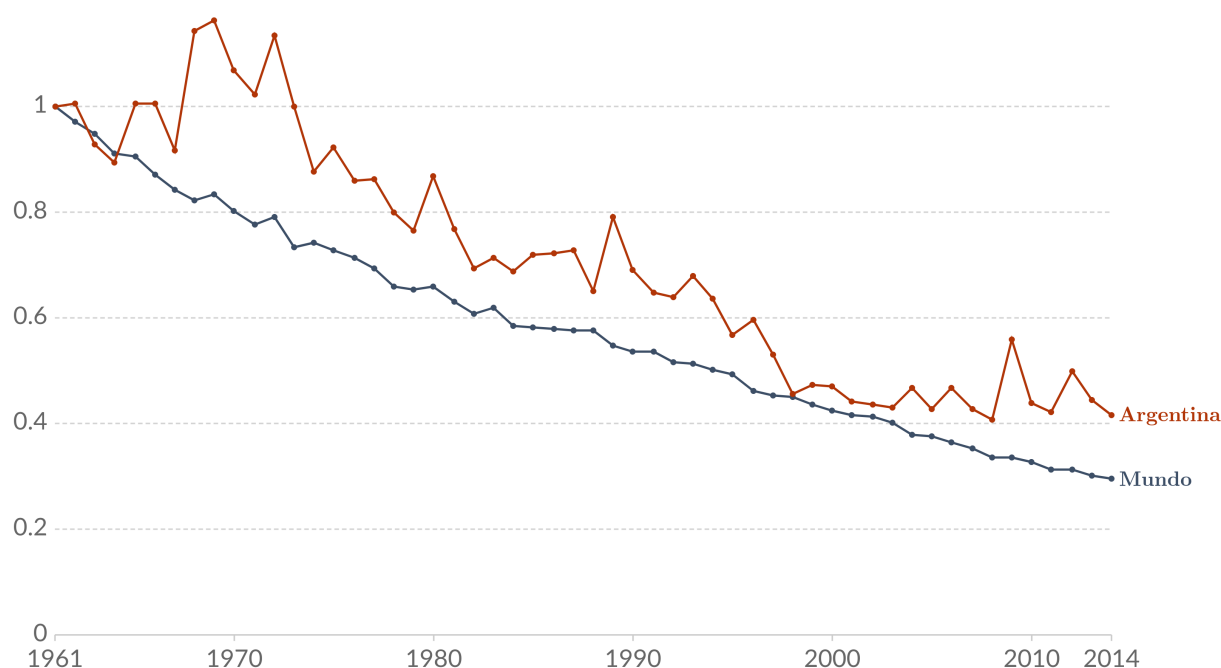


Figura 1.2: Tierra arable necesaria para producir una cantidad constante de cultivos, relativa a los valores de 1961 [2].

sistemas computacionales, existen varios factores comunes y uno de ellos es la optimización. La optimización es un concepto muy amplio que se aplica constantemente en cualquier ámbito de la ingeniería, que en un sentido muy abarcador, consiste en buscar la combinación de valores, dentro de todas las variables presentes en el sistema, que optimizan el resultado, es decir, que logran un balance ideal entre los costos y los beneficios.

La optimización puede aplicarse al diseño y formar parte del funcionamiento de cada uno de los sistemas de adquisición, transmisión y procesamiento de la información. En el marco de esta tesis, se presentan distintas problemáticas y soluciones para cada una de éstas etapas con el fin de resolver distintas cuestiones de los sistemas computacionales mediante un enfoque de diseño óptimo y con orientación a soluciones agrícolas como principal caso de estudio.

Finalmente, se aborda un tema integrador que permite la incorporación de todas las metodologías y herramientas descriptas a lo largo de este trabajo, que son los sistemas colaborativos y son aquellos que permiten la interacción y coordinación de un grupo de personas que buscan alcanzar una meta común. Las tecnologías de uso cotidiano como las redes sociales y las plataformas de mensajería han provocado su impacto también en la forma en la que interactúan productores, profesionales y demás trabajadores del sector agrícola. Para favorecer el desarrollo de un espacio colaborativo, se requiere de una infraestructura de adquisición, comunicación y de procesamiento de datos, así como considerar algunos aspectos que involucren al tamaño del grupo de usuarios con sus roles, los tipos de interfaces que utilizan los miembros, el esquema de coordinación y comunicación entre ellos, etc. Estas herramientas exceden a las aplicaciones de computación y se pueden utilizar en diversas áreas de trabajo utilizando a la computación como un medio y no como un fin. En este trabajo se

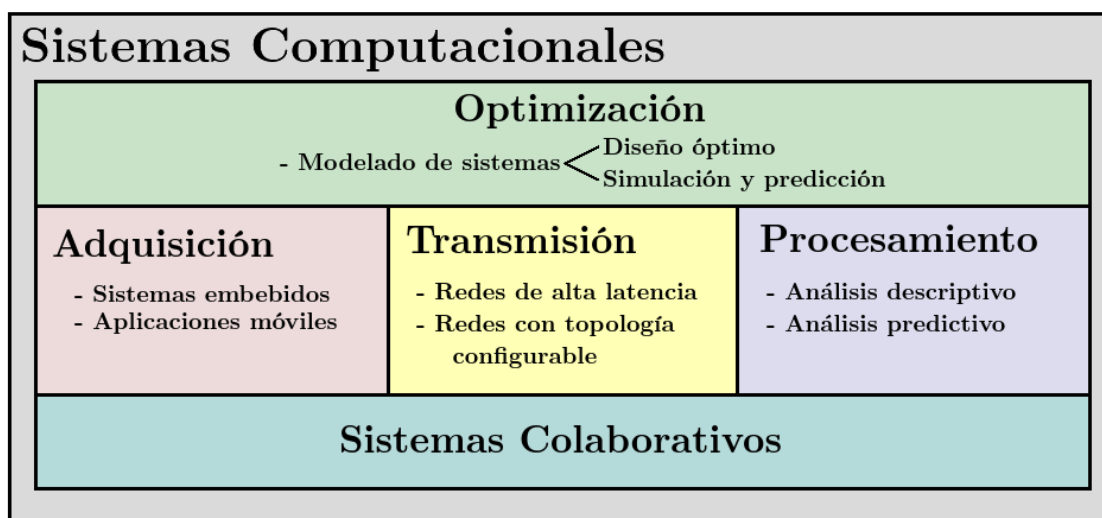


Figura 1.3: Los temas desarrollados en el marco de esta tesis organizados en el contexto de las principales áreas de los sistemas computacionales.

presenta el análisis de dos propuestas que se enmarcan dentro de este tipo de sistemas.

Por un lado, se presenta un esquema colaborativo con principal enfoque en los mecanismos de adquisición y procesamiento de la información, dejando de lado y asumiendo resueltas, a las cuestiones vinculadas a la transmisión de los datos. En este caso, se trata de un sistema que apunta a mejorar la eficiencia en el mantenimiento de la red vial rural. Por otro lado, se introduce un sistema basado en inteligencia de ambiente, donde los principales aspectos en los que se hace énfasis es en los mecanismos de adquisición y transmisión de la información, teniendo menor grado de relevancia la etapa de procesamiento de la información. El sistema propuesto en este caso, permite vincular usuarios con un centro de logística dada la no disponibilidad de redes de comunicación habituales, situación que puede darse ante un desastre natural u otro tipo de catástrofe.

En el resto de este primer capítulo se enmarcan las publicaciones realizadas en el desarrollo de esta tesis y la manera en que se relacionan con el objeto de la misma. Se da el marco teórico general sobre el que se planteó el trabajo al inicio configurando una colaboración interdisciplinaria que involucró a los institutos de doble dependencia del CONICET y de la Universidad Nacional del Sur, IIIE y CERZOS, además de una participación activa de la Estación Experimental INTA de Hilario Ascasubi.

## 1.2. Estado del arte

El Sudoeste de la Provincia de Buenos Aires abarca cerca del 20 % del territorio de la provincia de Buenos Aires y está integrado por los partidos de Adolfo Alsina, Saavedra, Puán, Tornquist, Coronel Rosales, Coronel Dorrego, Bahía Blanca, Villarino, Patagones y parte de los partidos de Guaminí, Coronel Suárez y Coronel Pringles. Esta región se caracteriza por la gran variabilidad climática y la escasez de precipitaciones, factores que restringen la diversidad de cultivos y determinan un menor potencial de producción, respecto a la región

pampeana central [3].

La siembra directa (SD) significó un cambio importante en el modo de trabajar la tierra. Estimaciones recientes dan cuenta que un total de 27 millones de hectáreas se encuentran bajo siembra directa en Argentina, lo que representa la mayor parte de la superficie agrícola del país. El crecimiento del área en SD fue muy rápido, desde unas pocas hectáreas en los años 80 hasta el 78,5 % de la superficie agrícola actualmente.

Fue a partir de la década del 90 cuando se comenzó a difundir un paquete tecnológico donde se incorporaba la soja resistente a herbicida, con amplias ventajas. Esta aparición impulsó el reemplazo del viejo paradigma del laboreo del suelo agrícola, base de la producción durante miles de años, por una agricultura sin remoción del suelo. El cambio es muy importante, porque significa modificar muchos de los aspectos productivos. El uso de la labranza favorece una mayor porosidad artificial, dinámica e inestable, mientras que en la SD predominan los poros biológicos. La cobertura en SD modifica la temperatura, la dinámica y la conservación del agua en el suelo, así como el desarrollo de las plagas, malezas y enfermedades. Estas diferencias, entre otras, generan un nuevo ambiente en el que el cultivo se implanta, crece y tiene que producir. Este cambio implica comenzar a generar nueva información sobre un sistema de producción con características totalmente diferentes y donde los resultados de largo plazo aún no se conocen. Esta falta de información de los cambios que se producen en la planta y en el suelo a lo largo del tiempo es más evidente en aquellas áreas marginales en donde la soja no llegó para acompañar la expansión de la SD. El SO bonaerense es uno de esos casos. Los campos del Sur y Sudoeste de la provincia de Buenos Aires se consideran frágiles en virtud de las características edáficas y climáticas, principalmente temperatura, humedad y viento que hacen un ambiente particularmente adverso para la producción de granos.

En el campo de la ingeniería electrónica y de computación, esta tesis contempla el desarrollo de diversos tipos de sensores así como también, el desarrollo de software para el control de sistemas, la captura de datos y el análisis de los mismos. La información puede ser obtenida de manera directa sobre el suelo o indirecta midiendo la materia orgánica (MO) relevada del campo o a través del esfuerzo de la maquinaria en la labranza.

### 1.3. Objetivos

El Plan Argentina Innovadora 2020 del Ministerio de Ciencia, Tecnología e Innovación Productiva de la Nación establece como área estratégica a la Agroindustria y entre sus consideraciones generales indica “Los avances en ciencia y tecnología, en especial en biotecnología, tecnologías de la información y las comunicaciones (TIC) y nanotecnología, abren nuevas oportunidades para transformar la actividad agropecuaria de Argentina”. En este marco, el presente trabajo doctoral inicia un camino de colaboración entre dos institutos de doble dependencia UNS-CONICET dentro del CCT Bahía Blanca como son el IIIE y el CERZOS.

La utilización de redes de sensores en el campo y en maquinaria agrícola permiten relevar una importante cantidad de datos e información que brindan conocimiento sobre el estado del suelo, su composición, materia orgánica, humedad y niveles de erosión entre otros. Esta información puede ser apropiadamente procesada por medio de modelos matemáticos y algo-

ritmos que permiten obtener conclusiones sobre el manejo más adecuado a realizar del suelo, los rindes y la productividad en general, además de brindar información sobre cuestiones medioambientales referidas a la sustentabilidad de la producción [4].

El objetivo general es dar tratamiento a esta problemática transversal denominada Agro-TICS como una disciplina superadora de la predecesora agroinformática en el marco de los temas estratégicos definidos por el CONICET en el marco del plan Argentina Innovadora 2020. En particular en el subtema de Agricultura familiar destaca: “Diseño y mejoramiento de maquinaria agrícola para agricultura familiar, nuevas tecnologías y aplicación de TICs para la gestión de procesos productivos y sistemas de comercialización de pequeña escala”. Las actividades desarrolladas en el marco de esta tesis entonces, se orientaron a formar recursos humanos en un área donde convergen las disciplinas de la ingeniería electrónica y de computación, y la ingeniería agronómica, para el análisis, modelado, diseño y realización de sistemas computacionales desde una visión integrada del problema que permita un uso eficiente y óptimo del suelo y de la maquinaria agrícola.

## 1.4. Contribuciones

En el periodo doctoral, el trabajo se enmarcó en el uso de técnicas de optimización basadas en metaheurísticas aplicadas, por un lado, al diseño con ciertos criterios de optimalidad y por el otro al modelado de sistemas para un análisis descriptivo y predictivo que colaboran en el desarrollo de la agricultura de precisión.

A continuación se listan las publicaciones científicas que forman parte de los resultados parciales obtenidos a lo largo del desarrollo de esta tesis y en lo que sigue se describen los trabajos realizados y el modo en que se van encadenando los resultados.

### 1.4.1. Artículos de revista de alcance internacional

- Opt01 Rodrigo Santos, Javier Orozco, Matías Micheletto, Sergio Ochoa, Roc Meseguer, Pere Milan, Carlos Molina. *Real-Time Communication Support for Underwater Acoustic Sensor Networks*. Sensors, 17(7), 1629. Año 2017.
- Opt02 Finochietto, M. and Eggly, G. and Micheletto, M. and Pueyo Centelles, R. and Santos, R. and Meseguer, R. and Ochoa, S. and Orozco, J., *An IoT-based Infrastructure to Enhance Self-Evacuations in Natural Hazardous Events*, Personal and Ubiquitous Computing, en revisión.
- Opt03 Matías Micheletto, Vinicius Petrucci, Rodrigo Santos, Javier Orozco, Daniel Mosse, Sergio Ochoa, Roc Meseguer. *Flying Real-Time Network to Coordinate Disaster Relief Activities in Urban Areas*. Sensors, 18(5), 1662. Año 2018.
- Opt04 Matías Micheletto, Rodrigo Santos, Javier Orozco. *Scheduling Mandatory-Optional-Time Tasks in Homogeneous Multi-Core Systems with Energy Constraints Using Bio-Inspired Meta-Heuristics*. Journal of Universal Computer Science, 25(4), 390-417. Año 2019.



- AP01 Matías Micheletto, Luciano Zubiaga, Rodrigo Santos, Juan Galantini, Miguel Cantamutto, Javier Orozco. *Development and Validation of a LiDAR Scanner for 3D Evaluation of Soil Vegetal Coverage*. Electronics, 9(1), 109. Año 2020.
- Mis01 Leonardo Ordínez, Rodrigo Santos, Gabriel Eggly, Matías Micheletto. *Utilizando UML para el aprendizaje del modelado y diseño de sistemas ciber-físicos*. IEEE-RITA, 15(1), 50-60. Año 2020.

### 1.4.2. Artículos en conferencias

- AP02 Matías J. Micheletto, Gabriel M. Eggly, Rodrigo Santos. *45<sup>o</sup> Jornadas Argentinas de Informática* 8<sup>o</sup> Congreso de AgroInformática (CAI 2016) “*Diseño e Implementación de un Registrador de Esfuerzos para Maquinaria Agrícola*”. Año 2016.
- AP03 Gabriel M. Eggly, Matías J. Micheletto, Juan P. D’Amico, Santiago J. Crocioni. *46<sup>o</sup> Jornadas Argentinas de Informática* 9<sup>o</sup> Congreso de AgroInformática (CAI 2017) “*Desarrollo de una Aplicación Móvil para Cálculos de Pulverizaciones Agrícolas*”. Año 2017.
- Mis01 Gabriel Eggly, Mariano Finochietto, Matías J. Micheletto, Rodrigo Santos. *48<sup>o</sup> Jornadas Argentinas de Informática* 1<sup>o</sup> Taller Argentino de Internet de las Cosas (TAIC) “*Internet de las Cosas como Bien Social*”. Año 2019.

### 1.4.3. Resumen de detalles

En [Opt01] se utiliza un algoritmo evolutivo para calcular el camino óptimo en redes subacuáticas con múltiples reflexiones. En el caso de las redes subacuáticas la conexión es inalámbrica y se realiza utilizando señales acústicas que son las únicas que tienen una propagación aceptable en el medio. Por este motivo, la transmisión es lenta y por lo general, una vez que el transmisor termina de modular el mensaje todavía no fue recibido por el receptor. La detección de colisiones entonces sólo es posible en el nodo receptor y el transmisor no se entera del error excepto por la falta de reconocimiento del mensaje. Al problema anterior se le suma el de las reflexiones múltiples que generan copias de los mensajes que llegan de manera desordenada. La técnica aplicada para determinar los caminos más adecuados es aplicable en otro tipo de redes, por ejemplo aquellas basadas en las redes de baja potencia y largo alcance LPWAN (por sus siglas en inglés, Low Power Wide Area Network) como LoRa [Opt02]. En [Opt03] se trabajó sobre la distribución de nodos que sirven de enlace en redes basadas en nodos aerotransportados por medio de vehículos no tripulados. El enfoque estaba orientado a brindar conectividad a los equipos de búsqueda y rescate en contingencias de desastres naturales cuando las redes habituales de comunicación se encuentran destruidas o colapsadas. El despliegue rápido de una red alternativa que permita mantener a los rescatistas comunicados de manera digital facilitando y asegurando el intercambio de información es muy importante. Para eso nuevamente se planteó una metaheurística que, asociada a una primera aproximación basada en el alcance de comunicación, permita brindar la cobertura necesaria. A diferencia del caso anterior no se establecían caminos óptimos sino cuestiones relacionadas a la cobertura. Las soluciones alcanzadas tienen impacto directo sobre las áreas

de la agricultura de precisión. Las grandes extensiones rurales de la Argentina conllevan la falta de conectividad. Para poder determinar el estado de cierta manada como rebaños o rodeos, las condiciones meteorológicas (milímetros de precipitaciones, humedad, temperatura, viento), la calidad del suelo, incluso el estado del cultivo es necesario el monitoreo permanente de los lotes. Esto puede ser hecho por los encargados de la producción o eventualmente, de forma remota. Pero para ello es preciso construir una red alternativa ya que la falta de cobertura y el costo de la telefonía celular hacen muy difícil la instrumentación de una solución general basada en las mismas. De igual modo, el uso de conexiones satelitales puede ser incluido pero nuevamente el costo resulta una limitación importante. La utilización de redes basadas en LoRa son por lo tanto una alternativa que resuelve algunos de estos inconvenientes pero que requiere de un adecuado sembrado de estaciones que recepcionen y retransmitan la información hasta alcanzar un enlace a la nube. El problema subyacente está relacionado con la planificación con restricciones temporales y de energía. En este sentido en [Opt04] se presentó una solución al problema de la asignación de tareas en sistemas de varios procesadores con restricciones de energía utilizando algoritmos evolutivos.

En [AP01] se describe el diseño, implementación y validación de un instrumento capaz de medir la cobertura del suelo. Este parámetro es muy importante porque repercute de manera directa sobre la calidad del mismo en cuanto un bajo porcentaje de cobertura significa una exposición mayor a los factores de erosión (lluvias y viento) que degradan la calidad del mismo. La determinación del tipo de cobertura es otro factor importante ya que ayuda al productor a definir el tipo de tratamiento que deberá brindar al terreno previo a la siembra, periodo que se denomina barbecho. El procesamiento inicial de los datos se realizó mediante técnicas de regresión lineal y se continúa trabajando en colaboración con investigadores del INTA sobre nuevas técnicas de análisis que permitan mejorar y generalizar los resultados para múltiples tipos de cobertura. Sobre este aspecto también se trabajó sobre la medición de los esfuerzos de tracción y deslizamiento a los que se ven sometidos los distintos mecanismos que forman parte de la maquinaria de labor [AP02]. La agricultura de precisión también requiere la incorporación de sistemas que permitan definir el modo en que se aplicarán los productos tales como fertilizantes y agroquímicos a los cultivos de tal modo que cumpliendo su objetivo no produzcan una contaminación indeseada y un gasto innecesario al productor. Para esto en conjunto con la estación experimental del INTA Ascasubi se realizó una aplicación que es utilizada en todo el país denominada Criollo [AP03]. Actualmente se trabaja en el desarrollo y mejoras de una segunda familia de aplicaciones relacionadas a la siembra denominadas Campero.

En el desarrollo de la tesis se han realizado trabajos complementarios que si bien no son parte del núcleo de la misma contribuyen aportando miradas y visiones diferentes sobre algunos elementos que conforman la misma. Entre ellas podemos mencionar en [MIS01] la definición de una técnica de enseñanza de modelado y diseño de sistemas embebidos que fue aplicada por ejemplo durante los desarrollos [AP01, AP02, AP03, MIS02].

# Capítulo 2

## Modelado de sistemas

### 2.1. Motivación

En el comienzo del siglo XXI, la generalización de los microcontroladores y el desarrollo del paradigma de hardware y software abierto sumado a la cada vez mayor capacidad de comunicación inalámbrica de los dispositivos dieron lugar a los sistemas ciber-físicos (CPS, por sus siglas en inglés) [5, 6, 7]. Este paradigma se plantea como una etapa superadora de aquel de los sistemas embebidos. La definición aceptada en este momento refiere a la integración de sistemas de computación, comunicación y control con procesos físicos. Es importante destacar que el análisis, estudio, modelado e implementación de esta clase de sistemas se realiza incluyendo todos los aspectos involucrados como las señales sensadas, los actuadores, el procesamiento de la información y obviamente el proceso físico con el que se interactúa. Esto involucra un proceso de desarrollo que atraviesa áreas de la Ingeniería de Software, el Control Automático y las Comunicaciones.

Los CPS son de importancia en muchas áreas de la ingeniería e impactan sobre la vida cotidiana de las personas aún en espacios en los cuales no se los ve. De hecho los mismos son parte de la Internet de las Cosas (IoT por su sigla en inglés) y contribuyen a la realización de ambientes inteligentes y cooperativos hasta en aplicaciones militares y espaciales [8]. Particularmente en el contexto de la tecnología agropecuaria, los CPS han adquirido gran relevancia por su amplio espectro de aplicaciones. La robustez de esta clase de sistemas implica un desafío considerable, ya que las condiciones de trabajo a las que son expuestos estos dispositivos electrónicos de aplicación agrícola pueden resultar extremas, ya que deben permanecer mucho tiempo a la intemperie lo que significa, en el caso de la región del SOB, soportar temperaturas que a lo largo del año oscilan entre los  $-10^{\circ}\text{C}$  y  $45^{\circ}\text{C}$ .

En este capítulo se presentan diferentes metodologías utilizadas para el diseño y modelado de aplicaciones de software y sistemas embebidos con especial enfoque a aquellos que pertenecen al grupo más genérico del que forman parte los CPS. Se emplea un subconjunto de herramientas de UML de una forma intuitiva y ordenada, comenzando con una descripción informal del sistema hasta obtener los detalles de implementación. Sin embargo, la parte de implementación del sistema y lo que significa la escritura de código, es dejada de lado ya que el lenguaje de programación a emplear es dependiente de la plataforma de hardware o software utilizada. Las metodologías que se resumen en este trabajo fueron aplicadas en las

primeras instancias de los diseños propuestos durante el desarrollo de esta tesis para describir los distintos tipos de requerimientos, actores y funciones de cada sistema.

## 2.2. Casos de uso

En los CPS existe una estrecha relación entre componentes mecánicos, elementos de hardware y software que interactúan con el medio físico. Por eso, es necesaria una visión global que integre todos los aspectos involucrados y las interacciones entre cada parte. Los requisitos pueden obtenerse a partir del propósito que debe cumplir el sistema, es decir, lo que se espera que haga. Adicionalmente, estos requisitos pueden evaluarse desde el punto de vista del entorno en el que funcionará el sistema, es decir, cómo debe responder ante determinada situación. En una primera instancia, se determinan las variables de entrada y salida o señales del sistema que se necesitan para lograr la funcionalidad deseada. Para esta primer etapa, los casos de uso son una herramienta simple que ayuda al diseñador a discriminar rápidamente cómo el sistema interactúa con el entorno físico. Los casos de uso tienen dos partes: una gráfica y una textual. La primera facilita la visualización de las interacciones entre los elementos involucrados en las diferentes situaciones. La segunda, permite una descripción de las interacciones con detalles tales como los aspectos computacionales y las interfaces entre los sistemas y el mundo real.

En el estándar UML [9] se determina que los diagramas de Casos de Uso cuentan con cuatro constructores: sistema, actores, el Caso de Uso y las relaciones de comunicación que expresan las asociaciones entre los constructores. En el caso de los CPS, la relación entre el medio físico y el mundo computacional es muy estrecha y debe ser considerada de manera especial al modelar los requerimientos. Para incorporar esta característica se propone un nuevo “constructor” que se representa por una línea punteada que encierra al *sistema*. Mientras que en los Casos de Uso tradicionales sólo se contemplan los elementos computacionales, en este caso se incluyen todas las entidades o elementos que conforman el sistema. Este constructor no tiene una interpretación semántica o sintáctica, sino que simplemente sirve para visualizar la real composición del sistema a modelar. En lo que sigue algunos constructores tradicionales de UML son replanteados en el contexto de los CPS.

1. **Actor:** Se extiende el concepto de “actor” para incorporar a los componentes que interactúan con el exterior. Un actor es una parte interna del sistema. Puede ser un rol desarrollado por el sistema, un sub-sistema sobre el cual no se tiene control directo (sistema operativo, controlador de red, dispositivo de comunicación), pero que brinda funciones. También puede ser una parte del proceso físico que el sistema debe supervisar. Al mirar a los actores en función de su rol dentro del sistema, es imposible distinguir desde afuera si uno en particular se refiere al espacio computacional o físico.
2. **Caso de Uso:** Las funcionalidades del sistema se definen por diferentes Casos de Uso que representan cada uno un flujo de acciones o actividades específicas. Cada uno representa entonces una actividad identificable en el sistema que puede interactuar con otras. Al mismo tiempo, esta actividad puede intercambiar información con el mundo físico. A su vez los Casos de Uso representan el propósito del sistema y de esta manera el conjunto de todos ellos constituye la descripción computacional completa del sistema.

3. **Requerimientos:** Este artefacto se utiliza para rodear los requerimientos computacionales del sistema. Todo lo que se encuentra dentro se relaciona con la computación integrada: software, interfaces, etc.
4. **CPS:** Dado que los sistemas CPS integran los procesos de computación con los físicos, es mejor analizar los dos aspectos simultáneamente. Dentro de este constructor todos los aspectos relacionados son incluidos, los roles, los dispositivos y las interfaces. Este constructor define el *alcance* del sistema.
5. **Asociación:** Una asociación entre un actor y un Caso de Uso indica que el primero provee o requiere una funcionalidad expresada en el Caso de Uso. Respecto de los Casos de Uso, una asociación entre ellos muestra que hay funcionalidades que dependen de otras para poder implementarse, se dice que entre Casos de Uso, la asociación es *directa*. Un caso especial de asociación es la *generalización*. Este tipo de relación se permite únicamente entre actores y su semántica es similar al concepto de herencia en la programación orientada a objetos. Además, las asociaciones pueden ser calificadas por un *estereotipo* que describe la clase de relación entre los constructores.
6. **Estereotipo:** Es un mecanismo que extiende a UML y puede ser utilizado para ajustar el modelo al dominio particular de la aplicación [10]. En el contexto propuesto, determina el tipo de relación entre dos Casos de Uso por medio del propósito de la misma. Es decir, que capturan el espíritu o la necesidad de la relación. Es importante notar que el estereotipo es una asociación que está direccionada de uno a otro Caso de Uso indicando la orientación de la misma.
  - **<<modeling>>** Representa a las leyes físicas o naturales que rodean al sistema. En forma genérica este estereotipo reúne a todas aquellas relaciones y elementos que no constituyen en sí la parte computacional y mecánica del sistema.
  - **<<communication>>** Indica que la asociación refiere a un aspecto de comunicación, por ejemplo un servicio de red o acceso a un puerto serie.
  - **<<call>>** Indica el llamado a un Caso de Uso auxiliar que realiza una tarea específica.
  - **<<syscall>>** Se utiliza para indicar la llamada a un servicio provisto por un software de otra jerarquía como un sistema operativo o monitor.
  - **<<sync-syscall>>** Se usa en el caso de llamadas al sistema que requieren de eventos de sincronización como el acceso a memoria compartida, los semáforos o los mailboxes.

Los siguientes estereotipos se refieren a asociaciones entre Casos de Uso y actores. Aquí no requieren orientación en el establecimiento de la relación, aunque de todos modos identifican el propósito de la misma.

- **<<requirer>>** El actor *requiere* una funcionalidad expresada en el Caso de Uso. Describe la relación por medio de un Caso de Uso entre dos actores, uno que requiere y otro que provee. Es asimilable a los conceptos de cliente/servidor o *request/response*.

|                     |                                                                                                                                                                                                                                                                                                                                                        |
|---------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                     | NOMBRE DEL CASO DE USO A DESCRIBIR                                                                                                                                                                                                                                                                                                                     |
| NOMBRE              | El nombre del Caso de Uso.                                                                                                                                                                                                                                                                                                                             |
| ACTORES             | Los actores asociados directamente al Caso de Uso, sino hubiera, queda en blanco.                                                                                                                                                                                                                                                                      |
| DESCRIPCIÓN         | Se describe de la manera más detallada posible lo que hace el Caso de Uso.                                                                                                                                                                                                                                                                             |
| FLUJO DE EJECUCIÓN  | Es la secuencia normal de ejecución.                                                                                                                                                                                                                                                                                                                   |
| FLUJO DE EXCEPCIÓN  | Situaciones anormales que requieran de procedimientos excepcionales.                                                                                                                                                                                                                                                                                   |
| VBLES MONITOR       | En el caso de que el Caso de Uso tenga una asociación de entrada del tipo <<sensing>>, se describen las variables monitoreadas, si hubiera un estereotipo <<analysis>>, las variables derivadas para el procesamiento que son utilizadas de manera indirecta. La descripción debe incluir rangos, precisión, unidades además de información relevante. |
| VBLES CONTROL       | Similar al caso anterior pero sobre el estereotipo <<actuating>>.                                                                                                                                                                                                                                                                                      |
| REQ. NO FUNCIONALES | Refiere a las restricciones que tenga el sistema. Pueden ser de memoria, de temporizado, de condiciones de uso (temperatura, humedad), consumo de energía, tamaño, peso, mecánicas, tipos de componentes, entre otros.                                                                                                                                 |

Cuadro 2.1: Plantilla Casos de Uso

- <<provider>> Es el anverso del anterior.

Estos dos estereotipos son importantes ya que ayudan a definir la arquitectura, cuando correspondiera, de cliente/servidor por medio de la identificación de servicios provistos y requeridos.

- <<h-mi>> Indica que el Caso de Uso asociado representa una funcionalidad de interacción con una persona (interfaz hombre-máquina). Es indistinto del tipo de interfaz: teclado, pantalla táctil, sonido, etc.
- <<sensing>> Expresa que el Caso de Uso destino involucra una acción de sentido para lograr su objetivo. Es uno de los estereotipos que representan la interacción entre el mundo físico y el computacional.
- <<actuating>> Expresa que el Caso de Uso destino involucra la ejecución de una acción por medio de un actuador. Como en el estereotipo anterior, sirve para establecer la interacción con el medio físico.
- <<analysis>> Expresa la funcionalidad del Caso de Uso que refiere específicamente al procesamiento y análisis de los datos. No hay interacción directa con el medio.

Los Casos de Uso gráficos permiten una rápida identificación de los actores y las relaciones y funcionalidades de cada elemento del sistema, pero no brindan los detalles necesarios para realizar un análisis de requerimientos. Para esto se incorpora la descripción textual de los mismos en la cual se agregan cuestiones como el tamaño de la memoria necesaria, restricciones temporales, precisión de los sensores o consumo de energía. Un ejemplo de esto se ilustra en el Cuadro 2.1. Estos aspectos se detallan en mayor profundidad en [11, 12].

## 2.3. Modelado de requerimientos

En esta sección se desarrollan los pasos a seguir para el modelado. Es una guía que en cada paso permite construir un modelo del sistema con todos los requerimientos. Busca ordenar el proceso de manera sencilla.

1. **Definir el sistema:** El primer paso consiste en una adecuada definición del sistema a realizar. Las descripciones suelen ser abiertas y plenas de ambigüedades. Es necesario escribir claramente los propósitos del sistema, es decir qué debe hacer y una descripción que complete lo anterior con cómo lo debería hacer.
2. **Determinar los actores:** En el contexto de este trabajo, los actores representan sistemas externos, entidades externas o internas con las que se intercambian acciones o datos. De este modo se incorpora al medio físico en la definición del sistema ya que se transforma en un actor con el cual se interactúa de manera tanto directa como indirecta.

|                      |                                                                 |
|----------------------|-----------------------------------------------------------------|
| NOMBRE DEL SISTEMA   |                                                                 |
| NOMBRE DEL ACTOR     | El nombre debe ser breve y representativo.                      |
| FUNCIONES PROVISTAS  | Principales características ofrecidas por el actor al sistema.  |
| FUNCIONES REQUERIDAS | Principales características requeridas por el actor al sistema. |
| DESCRIPCIÓN          | Describir el rol del actor en el sistema.                       |

Cuadro 2.2: Plantilla Actor

3. **Determinar los Casos de Uso:** Cada funcionalidad del CPS, es decir las acciones que hace tanto interna como externamente están asociadas a un Caso de Uso. El proceso para llegar a la determinación de los Casos de Uso es iterativo y depende de los interesados en el sistema o *stakeholders* y son quienes finalmente imponen los requerimientos. Los nombres de los Casos de Uso deben reflejar a los actores involucrados y la funcionalidad que se desarrolla en el mismo. Para Casos de Uso que no tienen asociación directa con actores no hay reglas explícitas para su nominación.
4. **Determinar entradas (*variables a monitorear*) y salidas (*variables controladas*) para cada Caso de Uso:** En general, las variables monitoreadas se asocian a sensores o información externa provista al Caso de Uso. Las variables controladas en cambio se asocian a actuadores o información interna. En algunos casos, una variable puede ser simultáneamente monitoreada y controlada (posición del rotor de un motor). En este punto, no es necesario describir completamente las variables, alcanza con enumerarlas. Más tarde cuando se realiza la descripción textual del Caso de Uso se deben incorporar todos los detalles.
5. **Resolver los Casos de Uso superpuestos:** En muchas ocasiones, dos o más actores pueden requerir el mismo Caso de Uso o varios Casos de Uso pueden contenerse parcialmente entre sí. En ambas situaciones, la estrategia de *refactorización* de los Casos

de Uso [13, 14, 15], resulta una alternativa adecuada y promovida desde diferentes enfoques de Ingeniería de Software.

6. **Redactar la especificación de los Casos de Uso:** Los Casos de Uso de UML son un método ágil para describir requerimientos de un sistema. Sin embargo, no pueden capturar algunos aspectos importantes que deben incorporarse para completarlos [16]. Con la descripción textual se puede incorporar todo aquello que no puede ser representado en los gráficos como son los requerimientos no funcionales referidos a características temporales, consumo de energía o disponibilidad de memoria.

Los pasos mencionados deben repetirse iterativamente hasta alcanzar una descripción satisfactoria.

## 2.4. Diseño de software basado en requerimientos

Una vez que los objetivos fueron identificados y los requerimientos modelados, el siguiente paso es describir las funcionalidades de manera precisa. Para esto se usan los Diagramas de Actividades de UML.

### 2.4.1. Diagrama de actividades: enfoque básico

El Diagrama de Actividades se compone de diversos constructores y describe una unidad lógica de trabajo [9]. Se puede dividir en *acciones*, entre otros elementos. Una acción es la unidad de trabajo atómica ya que no puede ser descompuesta en otras. La secuencia de acciones es controlada por aristas direccionadas que habilitan o no la transición marcada. Las acciones entonces son enlazadas por las aristas que representan el flujo del proceso o los eventos que se suceden. Se pueden transmitir y recibir señales para realizar llamadas al sistema. Cuando hay procesos auxiliares, se propone el uso de estados de *sub-actividades*.

### 2.4.2. Método de diseño

El proceso de describir Casos de Uso se divide en dos partes. En la primera se mapean las situaciones que surgen en los diagramas de los Casos de Uso en relación con sus correspondientes en los Diagramas de Actividades. Esto se realiza mostrando cómo las relaciones de los estereotipos se reflejan en los Diagramas de Actividades. En la segunda parte, se presenta un método para mapear los Casos de Uso en los Diagramas de Actividades. El resultado de este método es una descripción de cómo el Caso de Uso realiza su objetivo.

La descripción del Caso de Uso en la forma de un Diagrama de Actividades es directa si se consideran las representaciones textuales de cada uno de ellos. En particular, el flujo normal de ejecución y el excepcional constituyen el conjunto de acciones que determina que el Caso de Uso alcanzó su objetivo o no. Como paso intermedio, entre los Casos de Uso y los Diagramas de Actividades, se pueden utilizar Diagramas de Secuencia [17], que representan interacciones entre entidades (actores, clases, etc.), mediante el pasaje de mensajes. Esta estrategia se recomienda aplicarla ante la dificultad de realizar el pasaje en forma natural. A continuación se describen los estereotipos mencionados.



La Figura 2.1 a) muestra como el estereotipo «communication» se mapea en un Diagrama de Actividades. Se incluye un objeto “flujo” que es simplemente un camino por el cual se mueven datos.

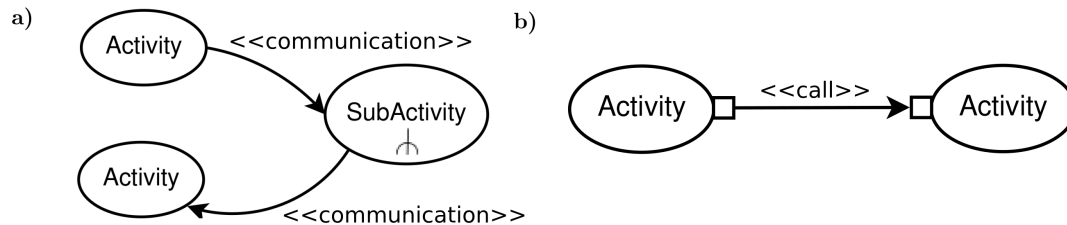


Figura 2.1: Estereotipos en Diagramas de Actividades.

Una situación común en un software es la llamada a una función auxiliar. La Figura 2.1 b) muestra un esquema para mapear el estereotipo «call». Se puede ver que hay una conexión entre dos nodos de actividades.

Los estereotipos «syscall» y «sync-syscall» son similares al anterior, pero la sub-actividad la realiza una entidad diferente por lo que se ejecuta en un carril distinto (*swim-lane*).

En este capítulo se extiende la clasificación que realiza [18]. Si bien en el trabajo mencionado se utiliza el concepto de *objetos*, aquí se extiende al de *tareas* que resulta más abarcativo en el contexto de los CPS. Las *tareas de control* corresponden a los Casos de Uso con estereotipos «syscall», «sync-syscall» y a aquéllos que brindan funcionalidades a otros Casos de Uso en sub-actividades. Las *tareas de interfaz* se refieren a los que brindan servicios de hardware y comunicación. Finalmente, las *tareas de entidades* reúnen a los que desarrollan el resto de las funcionalidades.

El proceso por el cual se describen las tareas es iterativo. Los Casos de Uso textuales se revisan y corrigen, se aclara y mejora continuamente hasta lograr una descripción completa, sin ambigüedades y ordenada de lo que se pretende del sistema. Basado en el método propuesto por Kimour y Meslati [19] se siguen los siguientes pasos:

1. **Diagrama de Actividades:** Cada Caso de Uso textual se transforma en un Diagrama de Actividades. Es importante realizar tanto el flujo normal como el excepcional de acciones. En este paso hay que verificar si hay puntos de sincronización y establecer las condiciones de guarda necesarias. Esto se puede complementar con el uso de Diagramas de Secuencia.
2. **Acciones:** Se especifican las acciones a realizar. Se deben considerar las variables de entrada y salida involucradas y las condiciones que las habilitan.
3. **Consistencia:** Se revisa la consistencia de las acciones y el flujo de las mismas en el Diagrama de Actividades a fin de que sea factible.

## 2.5. Organización arquitectural

Finalmente, resta mencionar brevemente que para la descripción de la arquitectura del software se utilizan Diagramas de Clases y de Paquetes de UML, los cuales exponen los datos y funciones asociados a cada módulo de software. Estos aspectos son comunes a la práctica de Ingeniería de Software y dado que los desarrollos presentados en el marco de esta tesis no involucran demasiados aspectos en tal sentido, este tema no se profundizará en detalle. En la Figura 2.2 se muestra un ejemplo genérico de componentes de clases y paquetes.

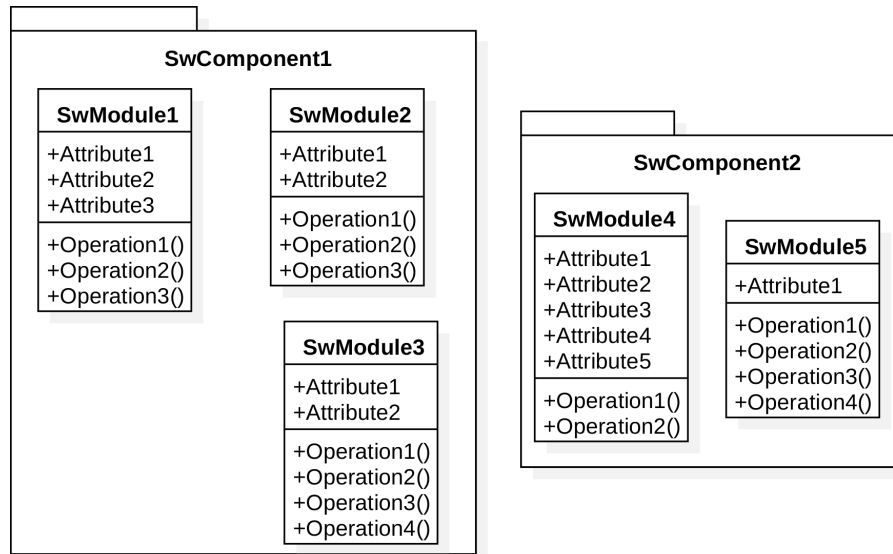


Figura 2.2: Diagrama de Clases y Paquetes

## 2.6. Consideraciones finales

En este capítulo introductorio se presentaron conceptos basados en elementos de UML para brindar un método concreto, sencillo, ágil y seguro para el modelado y diseño tanto de CPS como de aplicaciones de software.

La propuesta entonces presenta desde la descripción en lenguaje natural de los actores, sus relaciones, requerimientos y funcionalidades hasta los detalles de implementación por medio de Diagrama de Actividades. Lo expuesto en este capítulo son cuestiones que han sido puestas en práctica a lo largo del diseño de los distintos sistemas e implementaciones presentados en esta tesis y se puede asegurar que el resultado es bueno en cuanto al tiempo invertido en la etapa de definición del sistema y sus requerimientos. En el diseño inicial, se evita muchas iteraciones y ajustes que se usualmente se realizan en los casos en los cuales no se adopta la metodología. Con esto, la eficiencia mejora pero también, y más importante aún, la comprensión global del problema.

En los siguientes capítulos se presenta el diseño, prototipado e implementación de sistemas embebidos y de aplicaciones móviles para brindar soporte a tareas agrícolas, contextualizadas dentro de los mecanismos de adquisición de información. El planteo de requisitos

de estos sistemas fue llevado a cabo siguiendo los lineamientos propuestos en este capítulo, definiendo los requisitos funcionales y no funcionales junto a investigadores de áreas afines a la producción agrícola.

# Parte II

## Adquisición de datos

# Capítulo 3

## Sistemas embebidos

### 3.1. Motivación

Los sistemas electrónicos inteligentes o sistemas embebidos (SE) son aquellos basados en un procesador que cumplen una función dentro de un sistema más grande. Los SE interactúan con el medio ambiente que los rodea o con las personas por medio de unidades de entrada/salida. Entre lo más habitual encontramos displays y teclados, un puerto de comunicación para acceder a datos almacenados y distintos tipos de sensores y actuadores digitales y/o analógicos. Esta clase de sistemas se generalizó a partir de la reducción en los costos de los microcontroladores y la miniaturización de dispositivos.

En los últimos años el uso de las plataformas de hardware abierto como Arduino, Raspberry o Beagleboard entre otras [20, 21, 22] permitió el desarrollo de SE de bajo costo que permiten resolver distintos problemas sin necesidad de realizar el diseño completo. Estas plataformas presentan interfaces abiertas en las cuales los desarrolladores de sistemas y aplicaciones pueden conectar sensores y actuadores varios. Habitualmente disponen de una interface de programación sencilla en lenguaje C y una amplia comunidad que publica sus desarrollos tanto en hardware como en software. Este nuevo modo de desarrollar sistemas ha potenciado la realización de proyectos bajo el paradigma commercial-off-the-shelf (COTS, por sus siglas) que se basa en la implementación de sistemas en base a componentes discretos existentes en el mercado.

En este capítulo se detallan las cuestiones técnicas referidas al diseño e implementación de dos sistemas embebidos para brindar soporte a tareas de investigación agrícola. Por un lado, un dispositivo registrador que por medio de celdas de carga y sensores inductivos, genera registros de datos que permiten medir los esfuerzos de tracción que realiza una maquinaria agrícola y el deslizamiento de mecanismos rotativos como ruedas o poleas. Por otra parte, se presenta un sistema CNC (por sus siglas, Control Numérico Computarizado) para escanear, mediante tecnología LiDAR, la cobertura vegetal del suelo a fin de obtener un modelo tridimensional de la misma y analizar las características estructurales de dicha cobertura y su influencia en la erosión del suelo o su velocidad de descomposición, entre otras aplicaciones. Debido a la débil interacción que existe entre esos dispositivos y el entorno físico en el cual operan, éstos no se clasifican como CPS.

## 3.2. Registrador de esfuerzos para maquinaria agrícola

La agricultura de precisión requiere de herramientas tecnológicas para la adquisición, registro y procesamiento de la información. La incorporación de los sistemas embebidos a la maquinaria agrícola permite realizar el control y monitoreo de forma constante y precisa de las variables que influyen en el rendimiento y afectan a la correcta utilización de la maquinaria. La eficiencia del tractor agrícola puede determinarse como la relación entre el consumo de combustible y la energía mecánica producida que es aprovechable por la maquinaria que realiza la labor. Esta última puede medirse utilizando transductores de fuerza como las celdas de carga o de rotación de partes móviles utilizando sensores inductivos. En esta sección se presenta el desarrollo de un sistema electrónico para medir y registrar los datos obtenidos de celdas de carga y sensores inductivos a fines de generar la información requerida para realizar posteriores análisis de la eficiencia de la maquinaria agrícola sometida a pruebas de rendimiento. Además, se detallan los aspectos principales del proceso de desarrollo, diseño de hardware y software, cuestiones de implementación y resultados experimentales obtenidos en las pruebas de laboratorio.

El costo de producción es uno de los factores principales en cualquier tipo de actividad económica. Reducir los costos de producción es uno de los principales mecanismos para mejorar la eficiencia y por lo tanto la productividad. En el caso de actividades agrícolas, especialmente aquellas relacionadas a la producción de cultivos, los principales costos están asociados al uso de maquinaria agrícola, ya sean para labranza, pulverización, siembra o cosecha. Los costos operativos dependen de varios factores, entre los cuales el combustible es el principal. En el caso de labranza y siembra, el consumo de combustible depende, entre otras, de los esfuerzos de tracción y la dureza del suelo. Mediante la medición de estas variables puede estimarse la eficiencia de la maquinaria [23].

Al igual que en las demás actividades económicas, el productor agropecuario debe decidir varias cuestiones al inicio de cada ciclo productivo. La correcta elección de la maquinaria es un punto fundamental de la estructura de costos ya que se relaciona directamente con la eficiencia y la productividad. El productor debe además evaluar la forma en que se utiliza la maquinaria y tener en cuenta los costos de reparación en caso de fallas [24].

A pesar de que la maquinaria agrícola se ha tecnificado mucho en los últimos años, el uso de sistemas embebidos de medición de esfuerzos en este tipo de tareas no es generalizado y la incorporación de este tipo de instrumental en máquinas que no son de última generación no es simple y se debe acondicionar. La opción de realizar un registrador universal de esfuerzos para evaluar el desempeño de una máquina agrícola de siembra o arado resulta entonces de interés para una economía basada en la producción agropecuaria como la de Argentina. El tema fue abordado en [25] y en [26]. En el primero se plantea un dispositivo similar para celdas de carga pero que no incluye el deslizamiento de las ruedas del tractor. En este caso además el dispositivo no se amura a una máquina como el tractor sino a dispositivos de labranza más elementales utilizados en el ámbito de la agricultura familiar. En el segundo caso, se trata de una tesis de maestría en la cual realizan un sistema electrónico para medir el consumo de combustible en un tractor agrícola. En este caso, el dispositivo relaciona el esfuerzo medido en la toma de fuerza de la máquina con el consumo de combustible.

En este trabajo se presenta el diseño, implementación y evaluación de un SE registrador que puede ser utilizado sobre cualquier maquinaria agrícola, de arrastre o suspendida, para

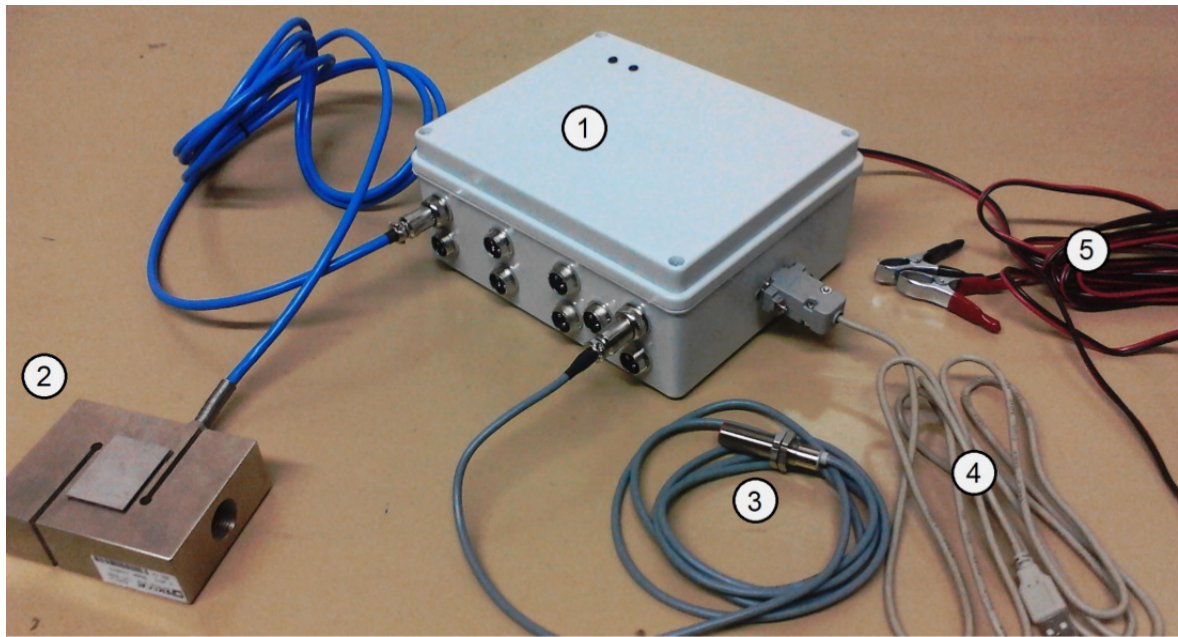


Figura 3.1: Referencias: 1-Registrador. 2-Celda de carga. 3-Sensor inductivo. 4-Conector USB. 5-Cable de alimentación.

medir los esfuerzos de tracción en distintos puntos de la misma, determinar el coeficiente de deslizamiento de la maquinaria propulsora y la velocidad del motor.

### 3.2.1. Requerimientos

El sistema debe ser capaz de muestrear y registrar todas las variables con una frecuencia mínima de dos muestras por segundo. Se requiere medir la velocidad de la toma de potencia en revoluciones por minuto, tomado este valor en la estría del eje y también la velocidad de rotación de las cuatro ruedas del tractor. Finalmente se requiere la medición de fuerza entregada por medio del uso de las denominadas celdas de carga que permiten determinar los esfuerzos a los que está sometida la maquinaria y mediante la velocidad de avance de la misma es posible determinar la potencia. El registrador debe poder tomar lecturas de cuatro celdas en paralelo. El sistema debe ser capaz de registrar los datos durante intervalos de al menos 15 minutos, que es el tiempo promedio que le lleva recorrer un sector de lote de labranza. Desde el punto de vista estructural, el sistema debe ser portátil y adaptable a diferentes maquinarias, tolerar un rango de temperatura variable de entre 10°C bajo cero y 40°C y tener tolerancia a altos valores de humedad o lluvias. A pesar de que el registrador será utilizado por técnicos especializados, debe tener una interfaz hombre-máquina sencilla para facilitar su configuración en el campo. El operario puede iniciar o detener el muestreo mediante un pulsador instalado en la carcasa del registrador o mediante un control cableado, que debe informar con un indicador visual el estado del sistema, ya sea detenido o grabando muestras. Además de estos requerimientos sobre el sistema, un software sencillo para instalar en una computadora personal estándar debe ser desarrollado a fin de poder obtener los datos medidos para su posterior procesamiento de manera práctica.

### 3.2.2. Hardware

En la Fig. 3.2 se muestra un diagrama simplificado de los componentes del sistema, donde se destacan principalmente, en el interior del gabinete, tres bloques: el microcontrolador, el módulo de almacenamiento y el reloj de tiempo real. Por fuera del gabinete se dispone de cinco sensores inductivos, cuatro celdas de carga y un pulsador para iniciar o detener el registro de los datos. La conexión a la PC puede realizarse mediante un cable USB para la descarga de los registros de medición.

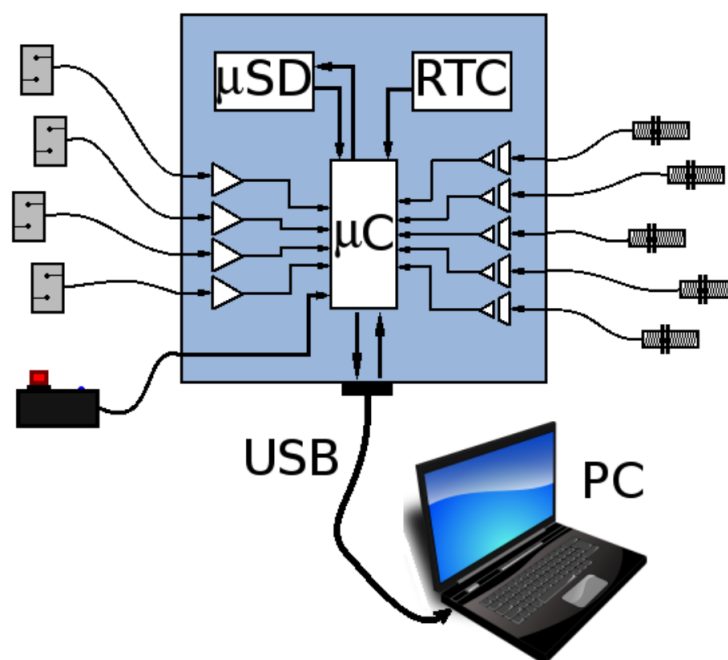


Figura 3.2: Diagrama de los componentes del sistema.  $\mu C$ : Microcontrolador,  $RTC$ : Módulo de reloj de tiempo real,  $\mu SD$ : módulo para lectoescritura de memorias micro-SD.

En la elección de la plataforma de hardware se tuvieron en cuenta varios aspectos. En primer lugar, el hardware debe estar disponible en Argentina, lo cual es una limitación importante dadas las restricciones económicas del sector. Segundo, los componentes elegidos deben tener asegurados varios años de producción en adelante. Por último, la cantidad de puertos y dispositivos debe ser tal que pueda manejarse todo el sistema desde un único microcontrolador. Bajo estas restricciones se ha elegido un microcontrolador Atmel, modelo ATmega 328, que satisface todos los requerimientos mencionados.

El ATmega 328 cuenta con seis conversores A/D de 10 bit, 10 interrupciones por cambio de estado de los pines, tres timers, SPI para controlar un módulo de lectura/escritura de memorias SD y cuenta con una gran comunidad de desarrollo. La plataforma abierta Arduino provee un excelente kit de desarrollo de sistemas COTS, una interfaz de desarrollo simple y una gran librería de repositorios.



### **3.2.2.1. Medición de RPM**

Para el cálculo de la potencia que desarrolla el motor se emplea la curva característica de potencia propia de cada motor y que puede determinarse previamente de manera experimental. En el procedimiento se aplica un freno dinamométrico o freno Prony en la toma de fuerza (PTO, por sus siglas en inglés) y se mide la potencia de salida en función de las revoluciones por minuto (rpm) de la PTO que a la vez se relaciona linealmente con las rpm del motor. Por lo tanto, conociendo la velocidad de rotación del eje de la PTO se puede estimar, para los fines prácticos, la potencia disponible del motor para realizar trabajo de tracción, con lo cual, las rpm de la PTO es una variable a medir y registrar con el instrumento.

La velocidad de avance del tractor se puede determinar según la velocidad de rotación de las ruedas multiplicando este valor por el factor de deslizamiento. Si bien pueden emplearse técnicas velocimétricas más precisas como las basadas en GPS, conocer la velocidad de rotación de las ruedas es relevante en la determinación de la eficiencia de la transmisión y el deslizamiento relativo del tractor en vacío o con carga. El deslizamiento indica en parte la característica del suelo sobre la que traccionan las ruedas y también la cantidad de potencia que se está perdiendo sin ser utilizada plenamente por la labor de la máquina.

Para el caso de tractores con tracción en ambos ejes, doble tracción o tracción asistida se desea medir rpm de las cuatro ruedas, con lo cual, el registrador debe disponer de cinco canales para medición de rpm, una para cada rueda y otra para la PTO.

Los sensores inductivos son una buena opción para instalar en la maquinaria porque no requieren de la colocación de piezas adicionales como los sensores ópticos o de Efecto Hall, aunque resultan mucho más costosos. El sensor utilizado es marca SAIA modelo XVJ B12G21/L y requiere una alimentación de 12V. Para adaptar la señal a salidas TTL se utilizaron optoacopladores TLP521.

### **3.2.2.2. Medición de esfuerzos de tracción**

Las maquinarias con las que se va a realizar las mediciones de eficiencia tendrán instalado como máximo cuatro celdas de carga en diferentes puntos de manera de medir los esfuerzos de tracción a los que están sometidas durante su operación. Las celdas de carga disponibles son de marca Flexar, modelo S-Type CZC, de distintas capacidades y una sensibilidad de 2mV/V. La alimentación utilizada para el puente es de 5V regulado. Para amplificar la señal diferencial de las celdas de carga se emplearon amplificadores de instrumentación INA122AP, de alimentación simple y con un valor de ganancia de voltaje de 518.

### **3.2.2.3. Registro de datos**

Todos los datos medidos son registrados en archivos de texto plano en una memoria micro SD que se comunica con el microcontrolador por medio de una interface SPI a través de un módulo uSD Card Adapter Catalex 1.0. Para acceder a la fecha y hora actuales se utilizó el circuito integrado DS1307 que posee una interface I<sup>2</sup>C y se mantiene actualizado gracias a la alimentación de una pila tipo botón CR2032 de 3V. Cada vez que se inicia una nueva grabación se crea un nuevo archivo de registro que contiene una primera línea con una estampa de tiempo y a continuación se escribe una línea por medición que contiene el número de muestra, los valores de los cuatro canales analógicos a los que están conectados

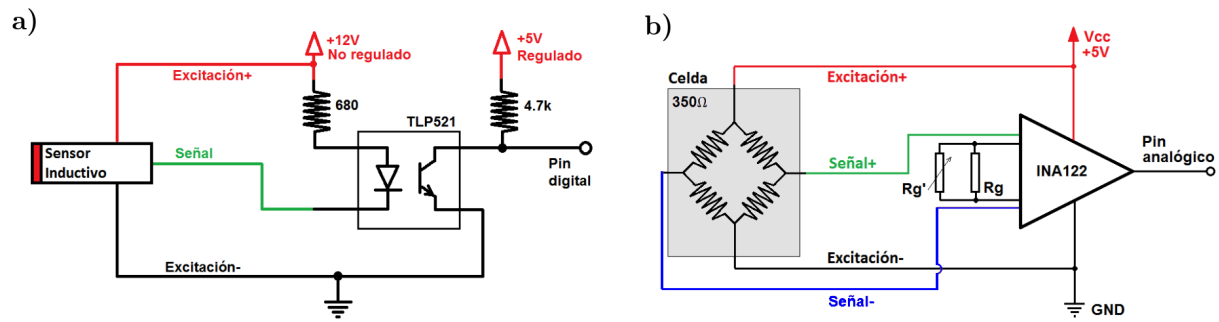


Figura 3.3: Esquema de conexión de los sensores inductivos (a) y celdas de carga (b).

las celdas de carga y los valores de los cinco contadores de pulso de los sensores inductivos. Los archivos podrán ser luego descargados a una PC mediante el software correspondiente que se detallará en la Sección 3.2.4.

#### 3.2.2.4. Fuente de alimentación

La fuente de alimentación disponible puede ser una o dos baterías de automóvil de 12V y su tensión puede variar entre 10V y 15V durante la operación del tractor dependiendo del consumo o la carga del alternador. Debido a que el microcontrolador, los amplificadores de instrumentación y el módulo micro SD requiere una alimentación de 5V, se diseñó una fuente conmutada de 5V regulada basada en el circuito integrado LM2575. Esta fuente provee una corriente máxima de 1A, aunque el máximo consumo de todo el sistema no supera los 500mA. La fuente cuenta con una protección por exceso de corriente que es limitada con un fusible polyswitch de 1A, una protección por exceso de voltaje implementada con un varistor de 25V y mediante un puente de diodos se evita que haya alimentación con polaridad invertida.

#### 3.2.2.5. Montaje

Se diseñó una plaqueta de circuito impreso doble faz de 93 x 130 mm. y se instaló en una carcasa de polivinil de dimensiones 85 x 195 x 160 cm. que permite mantener los componentes electrónicos al resguardo de las condiciones adversas en las que el dispositivo debe operar. Todos los sensores, los cables de alimentación y el pulsador se acoplan a la carcasa mediante conectores tipo SVP que se ajustan a rosca. Para la conexión USB se empleó un conector DB9 ya que permite un acople más firme que los utilizados en la norma correspondiente. Se dispone de un led brillante azul que indica que el registrador está encendido y otro de color rojo para conocer el estado de la grabación o la ocurrencia de errores. La Figura 3.1 muestra el dispositivo con los sensores, periféricos y cables de alimentación.

### 3.2.3. Firmware

El firmware se compiló utilizando la IDE de la plataforma Arduino donde se incluyeron las librerías SD.h para controlar del módulo lector de memoria micro SD y Wire.h y RTCLib.h para el manejo del circuito RTC.



Figura 3.4: Empleo del registrador desarrollado para medir potencia de tiro y deslizamiento de correas de un microtractor multipropósito.

El sistema tiene un bajo factor de utilización y ejecuta un conjunto de tareas síncronas con un periodo de 500ms, las cuales consisten en leer el valor del conversor A/D para cada uno de los cuatro pines analógicos con los que se mide los esfuerzos de las celdas de carga y luego registrar los valores sin procesar junto con los de los contadores de pulsos en el archivo de registro actual.

Mediante los comandos del protocolo de comunicación del registrador se puede manejar la información almacenada en la memoria SD. Se admiten tres operaciones fundamentales: listado, descarga y eliminación de registros. En la primera se enumera la lista de registros con su número de identificación, fecha y hora del inicio del mismo y su tamaño en bytes. En caso de solicitar la descarga de un registro particular, se transfieren los datos en forma secuencial y en caso de eliminación se borra la información correspondiente de la memoria.

### 3.2.4. Software

La aplicación para PC permite gestionar los registros guardados en la memoria interna del aparato, realizar operaciones de calibración de celdas y sensores midiendo en tiempo real y también ajustar fecha y hora del registrador en caso de que se haya agotado la pila del reloj interno.

Inicialmente, la aplicación se programó en C++ usando la librería Qt5.4 para la interfaz gráfica de usuario y QExtSerialPort 1.2 para la comunicación serie. Como la librería Qt admite la compilación de software para las plataformas Windows, Linux y Mac, el software desarrollado se puede compilar para cualquiera los sistemas operativos mencionados. Por motivos de licencia de esta librería, no es sencillo distribuir aplicaciones *standalone* para Windows, con lo cual se desarrolló una alternativa idéntica para este sistema operativo en lenguaje Visual Basic.

### 3.2.5. Resultados y discusión

El equipo se ensambló y construyó en una caja estanca resistente a los golpes. Sus componentes operan en el rango de temperaturas previsto para su uso en el campo. Tiene dimensiones y peso que permiten su traslado sin dificultades. El software desarrollado para la PC es muy simple, se utilizan ventanas con menus autocontenidos y su uso es intuitivo.

El sistema embebido diseñado integra en un dispositivo la posibilidad de sensar simultáneamente la carga de la máquina y el deslizamiento de sus ruedas. Esta información es almacenada para su posterior procesamiento. El sistema fue construido en base a componentes presentes en el mercado local de fácil acceso y económicos. Los datos deben ser utilizados por el Ingeniero Agrónomo a fin de evaluar la performance de la máquina y de manera indirecta parámetros del tipo de suelo. A saber, las celdas de carga fueron evaluadas con cargas conocidas y la respuesta fue correcta; los sensores inductivos detectan el paso de tornillos de una rueda de manera correcta. Los requerimientos demandados fueron satisfechos en cuanto a la frecuencia de medición y rango de fuerzas, así como la cantidad de datos a almacenar (dimensión de la memoria) y duración de la medición. En la Figura 3.4 se muestra el arreglo experimental realizado por investigadores de INTA para medir la potencia de tiro y el deslizamiento de correas del microtractor “Chango”, desarrollado en la Estación Experimental de Hilario Ascasubi [27].

## 3.3. Escaner CNC LiDAR para análisis de cobertura vegetal

La calidad del suelo tanto desde el punto de vista de la fertilidad química, física y biológica depende en gran medida de su cobertura vegetal y para poder comprender cómo se relacionan las distintas variables es necesario contar con métodos de cuantificación para las mismas. Si bien existen distintas técnicas para medir la cobertura vegetal, en general se emplean valores porcentuales aproximados referidos al área cubierta y no suele tenerse en cuenta la dimensión vertical, ni el volumen o la densidad espacial de la misma. En algunos casos se suele medir el peso de la materia vegetal, en seco o con humedad, pero esta práctica requiere de extraer una porción de la cobertura del suelo de la muestra para ser analizada en laboratorio y, en general, los resultados no se obtienen inmediatamente.

La arquitectura o disposición de los residuos, entendidos como la disposición de pie y distribuidos horizontalmente, resultan muy relevantes ya que puede alterar el almacenamiento de agua de superficie y el impacto en el suelo [28]. Las características físicas más importantes que afectan al transporte de vapor son el espesor y la densidad de la capa de residuo, por lo tanto, un residuo espeso aumenta la capa de aire relativamente no turbulento por encima de la superficie del suelo, disminuyendo el transporte de vapor desde el suelo. Los residuos en pie también tienen importantes efectos sobre las características micro-climáticas de la superficie del suelo, mediante la reducción de la velocidad del viento hasta en un 70 % y la pérdida de agua por evaporación.

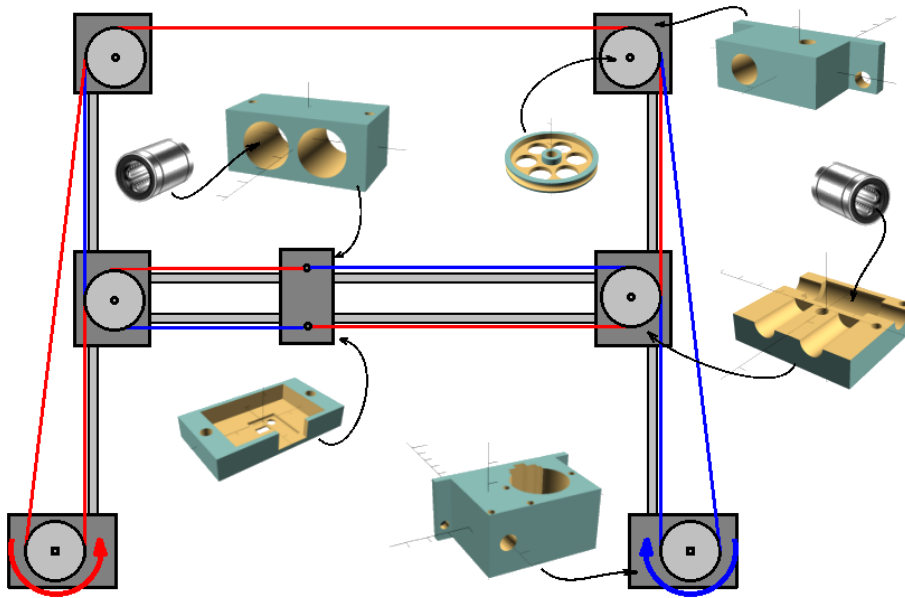


Figura 3.5: Esquema general del mecanismo de desplazamiento del sensor de distancia.

### 3.3.1. Hardware: sistema mecánico

Los sensores de ultrasonido son económicos y tienen alta disponibilidad en el mercado, pero debido a que tienen baja precisión, baja frecuencia de muestreo y, dada la naturaleza de la onda sonora, la alta dispersión del tren de pulsos hace que no sea posible realizar una medición de las variables de interés con la precisión deseada. Por este motivo se optó por el empleo de sensores de distancia láser. El modelo más accesible disponible en el mercado local es el módulo basado en el chip VL53L0X que posee un sensor ToF mediante VCSEL. Es posible manejar el dispositivo a través de una interface I2C lo cual reduce en gran medida el tiempo de diseño e implementación requerido para su uso.

El desplazamiento del sensor sobre el área rectangular se realiza con un mecanismo CNC. Dada las pequeñas dimensiones y bajo peso del sensor VL53L0X, el mecanismo necesario requiere de muy bajo torque, lo cual permite emplear motores livianos y de menor costo. En este caso, la opción elegida son los motores paso a paso BYJ48. Estos motores logran una buena precisión mediante una caja reductora con relación 1:64 y un ángulo de paso de  $5,625^\circ$ . Si bien son motores cuyo eje de salida tiene una rotación lenta, con una alimentación de 9 a 12V es posible que el sensor de distancia se desplace a una velocidad razonable sin requerir de un mecanismo adicional.

El mecanismo elegido para lograr el movimiento del sensor de distancia sobre un plano horizontal es el denominado CoreXY utilizado en impresoras 3D del proyecto RepRap [29]. La ventaja de este mecanismo radica en que requiere sólo dos motores que se disponen en lugares fijos lo cual reduce el peso del carro de desplazamiento y el torque necesario.

Teniendo en cuenta que el área a sensor es de unos  $2500 \text{ cm}^2$  y la densidad de muestras debe ser lo suficientemente alta para detectar estructuras finas como cañas de rastreo u hojas alargadas, se debe maximizar la velocidad de desplazamiento del sensor, para agilizar

la duración total de la medición.

La estructura se construye a partir de cuatro soportes fijos, dos de los cuales permiten fijar los motores paso a paso y los dos restantes sólo las poleas del mecanismo. Se utilizaron cuatro varillas de acero rectificadas de 8mm por 540mm de longitud, dos de las cuales permiten el desplazamiento del sensor en dirección horizontal y sobre las otras dos se monta el carro que se desplaza en dirección vertical. Todas las piezas se diseñaron con el software OpenSCAD y se imprimieron en ácido poliláctico (PLA) con una impresora 3D marca Prusa A8.

El mecanismo CoreXY posee dos sistemas de poleas independientes cuyo arreglo permite un movimiento en sentido ascendente/descendente del carro cuando los motores se mueven en dirección opuesta y un movimiento hacia la izquierda/derecha cuando los motores se mueven en el mismo sentido de giro.

El carro de desplazamiento vertical tiene dos soportes en sus extremos que se ajustan a las varillas de acero mediante rodamientos lineales y en la parte superior se dispone de las poleas correspondientes. El carro de desplazamiento horizontal, por otro lado, también se monta con rodamientos lineales y posee un soporte para ajustar el sensor de distancia en la parte inferior del mismo.

Para la detección del origen se utilizaron dos sensores de fin de carrera, uno para limitar el desplazamiento vertical y otro para el desplazamiento horizontal. La Fig. 3.5 muestra el esquema general simplificado del mecanismo CoreXY y las principales piezas del sistema.

### **3.3.2. Hardware: sistema electrónico**

El microcontrolador elegido está basado en un módulo para prototipos Arduino Nano que cuenta con las características de ser pequeño, de bajo consumo, económico y con una potencia de cómputo adecuada para las características del proyecto.

Todos los componentes del sistema son modulares y compatibles con el proyecto Arduino. Los motores BYJ48 emplean amplificadores de corriente integrados en el chip ULN2003 que consiste en un arreglo de siete pares Darlington NPN con salidas de cátodo común y también dispone de diodos flyback para cargas inductivas.

Para la comunicación inalámbrica se utiliza un módulo Bluetooth 4.0 AT-09 de bajo consumo (BLE, por sus siglas en inglés). Este módulo se controla mediante interface serie.

La regulación de voltaje de 5V requerida por el microcontrolador, el módulo Bluetooth, el sensor de distancia y los interruptores de fin de carrera se realiza con una fuente conmutada basada en el integrado LM2576. Se introduce un termistor en serie y un varistor en paralelo como protecciones de sobre voltaje y sobrecorriente y también un puente de diodos para evitar errores en la polaridad de la alimentación.

La alimentación de las fases de los motores paso a paso se controla con los arreglos Darlington y se alimentan con corriente continua sin regular, conectando directamente a la salida del puente de diodos. Esto permite también que la caída de voltaje de 1.4V del puente reduzca el voltaje de alimentación de los motores que no están preparados para soportar tensiones superiores a los 12V.

Se diseñó el PCB sobre el cual se montaron todos los componentes, exceptuando los módulos con interruptores de fin de carrera, que se ajustan sobre los soportes de plástico. La Fig. 3.6 muestra el diagrama de bloques de conexionado de los principales componentes del sistema.

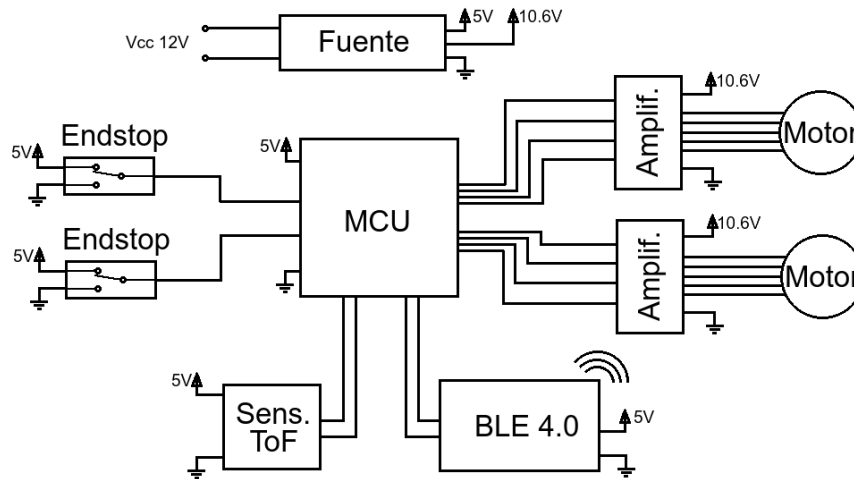


Figura 3.6: Esquema general del sistema electrónico.

### 3.3.3. Firmware

El microcontrolador se encarga de controlar el movimiento de los motores paso a paso, adquirir los valores de distancia a través del sensor láser y enviar los datos en forma de vector de tres dimensiones por puerto serie o bluetooth según el medio que se utilice.

Para determinar el movimiento de los motores, se dispone de una cola de setpoints donde cada elemento consiste en un vector que contiene las coordenadas (X,Y) hacia donde debe moverse el carro deslizante con el sensor. Cada vez que se alcanza un setpoint, se quita de la cola y se continúa con el siguiente. Cuando ya no quedan setpoints, se desenergizan los motores y el sistema queda a la espera de órdenes. En todo momento es posible pausar el movimiento, continuarlo o volver al origen.

Cuando el sistema inicia no se conoce la posición de los carros de desplazamiento, ya que no se dispone de sensores para estas variables, por lo tanto se accionan los motores de manera que el carro del sensor se desplace en dirección decreciente tanto en sentido horizontal como vertical hasta que se activan los topes de fin de carrera. Una vez que se alcanza el origen, el sistema está listo para iniciar el escaneo.

La adquisición de datos del sensor se puede activar o desactivar en todo momento. Mientras se realiza una medición el microcontrolador envía por puerto serie las coordenadas (X,Y) actuales y la componente Z que equivale a la altura que está midiendo el sensor de distancia.

El programa permite configurar la frecuencia de muestreo del sensor, la velocidad de avance de los motores paso a paso y modificar el recorrido del escaneo desde la aplicación del usuario.

### 3.3.4. Software

El usuario puede comandar el sistema desde una computadora personal o desde un smartphone según se disponga, para lo cual se diseñaron aplicaciones para las plataformas Windows, Linux o Mac y Android.

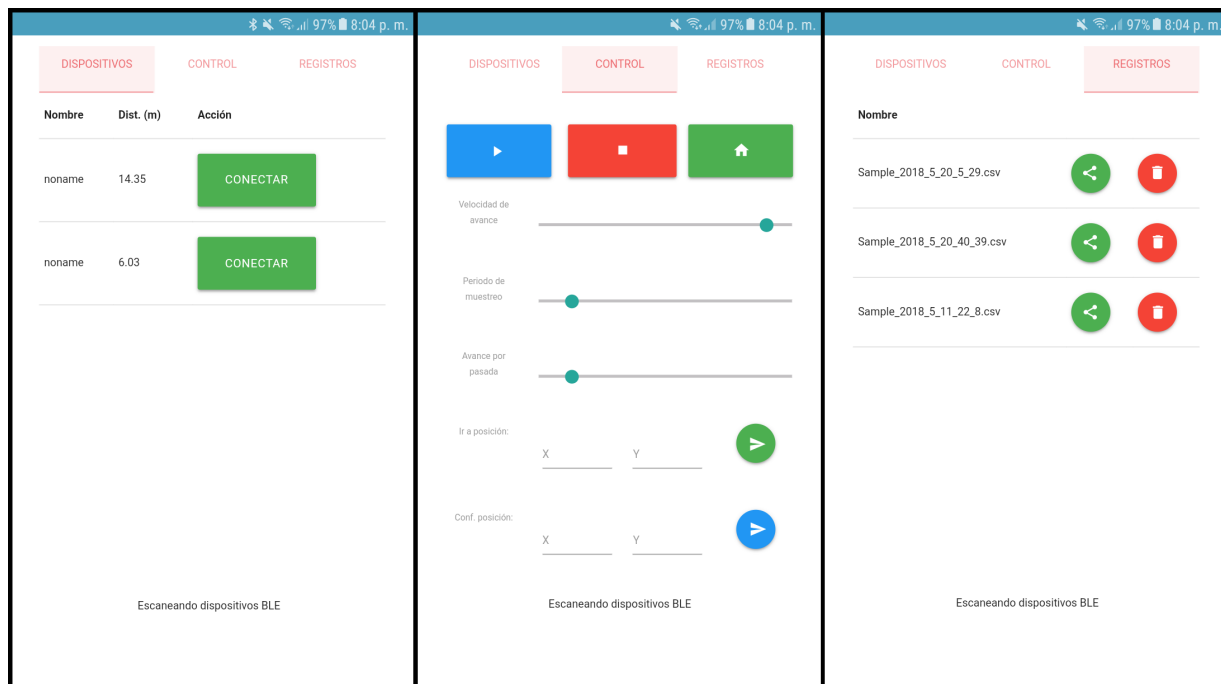


Figura 3.7: Capturas de pantalla de la aplicación Android.

Como se mencionó anteriormente, el resultado de la medición consiste en un archivo de texto plano en formato CSV.

Tanto la aplicación para PC como para smartphone tienen interfaces sencillas con los comandos básicos de control del escáner: Iniciar, pausar o cancelar medición y retornar carro al origen y también se cuenta con un espacio para configurar la velocidad de avance de los carros, el recorrido del sensor durante la medición y la frecuencia de muestreo del sensor. La única finalidad de la aplicación es la de adquirir los datos de la medición con lo cual la implementación es muy básica. La Figura 3.7 muestra unas capturas de pantalla de la aplicación.

Durante el proceso de medición, la aplicación debe mantenerse conectada al microcontrolador ya sea a través del puerto serie en el caso de la PC o por Bluetooth en el caso del smartphone y recibir cada una de las muestras y registrarlas en el archivo de salida. Una vez finalizado el ensayo, el usuario asigna un nombre al archivo y queda guardado en el medio de almacenamiento correspondiente y a disposición del sistema operativo en cuestión, ya sea para abrir, modificar o compartirlo.

### 3.3.5. Pruebas experimentales

Previo a la utilización del escáner en análisis de cobertura vegetal de suelos, se realizaron una serie de pruebas para validar el funcionamiento del mismo. Se construyó para este propósito, una maqueta empleando una base de telgopor sobre la que se insertaron varias plantas secas de trigo cosechado, para simular un rastrojo de este cultivo, como se muestra en la Figura 3.8.

En primer lugar se intentó determinar cómo influye la resolución del modelo capturado



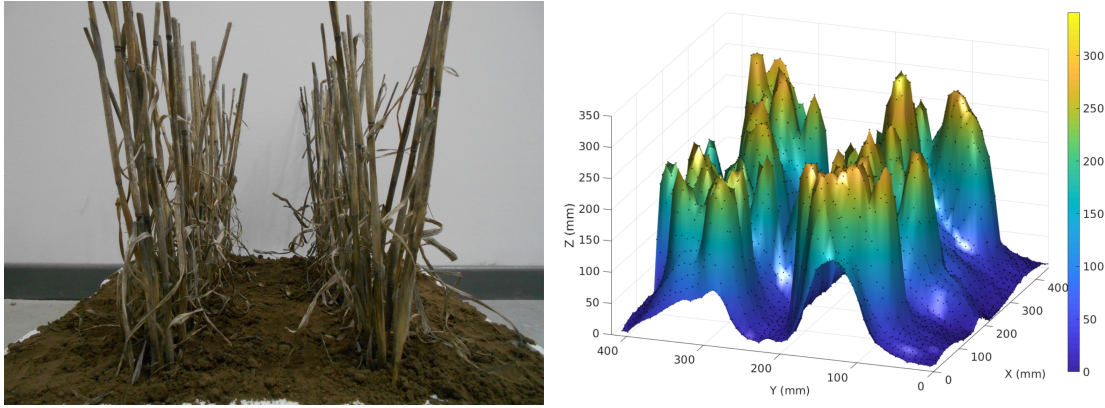


Figura 3.8: Maqueta de rastrojo de trigo empleada para la validación del instrumento (izq.) y modelo generado por computadora a partir de los datos adquiridos por el instrumento (der.).

en la estimación de las variables que describen el modelo. El error en la medición de la altura máxima ronda el 3 % para un modelo de 5.000 muestras o píxeles y se reduce al 1 % en mediciones de 10.000 muestras o píxeles. En el caso de la determinación de la altura promedio del modelo, el error es del 5 % y se reduce al 3 % en las imágenes de 5.000 y 10.000 muestras respectivamente.

La altura del sensor láser respecto de la superficie que se está escaneando también influye en la precisión de la estimación. El sensor tiene un área de visión (FOV, por sus siglas en inglés) de  $25^\circ$ . El área  $A$  de la superficie sobre la que refleja la luz al medir distancia, se puede expresar como función de la altura  $h$  del sensor respecto de la superficie que se está midiendo como,

$$A = \frac{\pi \cdot h^2}{k - 1}, \quad (3.1)$$

donde la constante  $k$  equivale a  $1/\sin(25^\circ/2)^2 \sim 5,59898$ . Esto significa que el área de reflexión se incrementa proporcionalmente al cuadrado de la altura a la que se encuentra el sensor, lo cual representa un inconveniente al momento de medir superficies con geometrías puntiagudas, porque el área de reflexión es muy pequeña comparada al campo de visión del sensor y por lo tanto no se alcanza a detectar correctamente. A medida que se eleva el escaner respecto de la superficie a medir, si ésta contiene una geometría alargada en la dirección vertical, la envolvente del modelo generado se aplasta cada vez más. Para evitar esto, se debe posicionar el instrumento lo más cercano posible a las partes más elevadas del cuerpo a escanear. La altura promedio medida no se ve afectada en gran medida y lo mismo ocurre con el volumen del modelo, ya que este valor es directamente proporcional a la altura promedio.

Un aspecto a tener en cuenta tiene que ver con la medición de la altura de la cobertura que se escanea. Al realizar la medición de altura de cierta cobertura vegetal desde una posición más elevada, es necesario calibrar la altura a la que se encuentra el sensor respecto del suelo, ya que este parámetro puede variar durante los ajustes previos a la medición que consisten en posicionar la plataforma de deslizamiento lo más cercana posible a la cobertura que se va a escanear. Además, no siempre se puede lograr que la estructura del instrumento asiente de

forma nivelada, dada la irregularidad que posee el terreno donde normalmente será utilizado.

### 3.3.6. Resultados y discusión

Mediante este sistema es posible realizar un escaneo de una porción de suelo obteniendo una medida cuantitativa de la distribución vertical de la cobertura vegetal con el fin de estudiar cómo afecta esta estructura o arquitectura de los residuos de la cosecha a la calidad del suelo obtenida durante el barbecho, aportando a la generación de información que permita promover la adopción e implementación de prácticas agropecuarias para la conservación del recurso suelo sobre todo en aquellas áreas limitadas tanto por la rigurosidad climática como por la fragilidad del ambiente.

El sistema permite medir y calcular la altura y densidad de la cobertura vegetal en una porción de 2500cm<sup>2</sup> de suelo por ensayo. El diseño se realizó con un muy bajo presupuesto ya que se pensó como un prototipo para realizar pruebas experimentales iniciales y, en caso de establecer esta como nueva metodología para analizar cobertura vegetal de suelo, evaluar la posibilidad de construir un instrumento de medición para uso científico y de mayor robustez.

El modelo capturado con el escáner puede ser representado mediante una envolvente tridimensional como se muestra en la Figura 3.8, lo cual permite apreciar la profundidad del modelo y comparar con la estructura real. Sin embargo, los datos pueden convertirse en un arreglo bidimensional como ser una imagen en escala de grises donde se codifica en la intensidad de gris, la altura del modelo en cada punto. Si se emplea suficiente resolución, esta conversión no implica la pérdida de información del modelo y permite aplicar algoritmos de filtrado morfológico.

La principal limitación radica en la escasa velocidad que pueden alcanzar los motores paso a paso utilizados, ya que obliga a reducir la resolución del modelo para acelerar el proceso de medición. Disponiendo de motores de mayor velocidad, sería posible agilizar el mecanismo e incrementar la frecuencia de muestreo del sensor logrando generar modelos de medición más precisos y con mayor densidad de datos.

El manejo del dispositivo puede ser realizado por personal con mínima capacitación ya que el funcionamiento es sencillo y las aplicaciones de usuario son intuitivas. El equipo puede ser transportado a mano fácilmente por una persona y también es posible cargarlo en vehículos chicos.

Respecto del consumo de energía, se alcanza un valor que ronda los 9 W durante el escaneo y se encuentra por debajo de los 0.5 W en reposo por lo que empleando una batería pequeña se logra autonomía suficiente para trabajar durante todo un día.

Una vez estandarizado el análisis de los datos obtenidos, estos cálculos pueden implementarse directamente en la aplicación móvil o para PC de manera que los resultados del muestreo puedan visualizarse en el momento o en tiempo real, a medida que el sistema realiza la medición.

En el Capítulo 9 se detallan los resultados obtenidos a partir de una serie de mediciones realizadas a campo.

### 3.4. Consideraciones finales

Las plataformas de hardware abierto como Arduino, Raspberry o BeagleBoard cuentan con modelos de propósito general con distintas prestaciones para diferentes aplicaciones y permiten reducir los tiempos de desarrollo en gran medida, evitando preocuparse por la elección de una gran parte de los componentes de un proyecto.

Los sistemas embebidos propuestos en este capítulo fueron implementados con herramientas de prototipado con el fin de mantener un bajo costo de desarrollo y contar con dispositivos funcionales para iniciar pruebas a campo en el menor plazo posible. La mayor parte de la implementación física de estos dispositivos consistió en el diseño de shields o plaquetas tipo zócalo que funcionan como soporte para los periféricos que se conectan a la placa principal que contiene el microcontrolador con sus interfaces de conexión.

A partir de los desarrollos planteados en este capítulo y luego de las evaluaciones a campo realizadas, se cuenta con resultados experimentales que abren la posibilidad, en caso de ser necesario, de desarrollar instrumentos para las mismas aplicaciones o similares pero con mayores prestaciones, para lo cual hará falta invertir en componentes de mejor calidad y contar con un proceso de fabricación que permita lograr un sistema robusto. Sin embargo, un prototipo de bajo costo no deja de ser el primer paso en la exploración de cualquier nueva tecnología.

Una de las principales dificultades que se presentaron durante las pruebas experimentales de ambos prototipos tiene que ver con la interacción del usuario y el instrumento a través de las interfaces hombre-máquina. Disponer de una interface intuitiva y sencilla de utilizar es un desafío importante y más aún es facilitarle al operario guías y protocolos a seguir en caso de falla o mal funcionamiento.

En el caso del prototipo para escanear cobertura vegetal, se provee junto con el instrumento una aplicación móvil que se vincula con el dispositivo por medio de un enlace bluetooth. Las aplicaciones móviles poseen la ventaja de poder brindarle al usuario mejoras en el sistema a través de actualizaciones de software. Por este motivo, en sistemas donde sea posible controlar un instrumento desde una aplicación móvil y en aquellos en los que la robustez no sea una característica importante, es recomendable distribuir la lógica del sistema de modo que la mayor parte se ejecute del lado del dispositivo móvil. Continuando con el eje temático de los sistemas de adquisición, en el siguiente capítulo se detallan dos desarrollos relacionados con las aplicaciones móviles.

# Capítulo 4

## Aplicaciones móviles

### 4.1. Motivación

Las herramientas ubicuas se caracterizan por *enmascarar* su presencia volviéndose transparentes al usuario que las emplea. Un ejemplo tradicional describe a los anteojos como una de las primeras tecnologías ubicuas. Al poco tiempo de usarlos, el portador deja de notar su presencia que se vuelve de este modo invisible hasta el momento en que el usuario se los quita y deja de ver con nitidez. Entre los sistemas computacionales hay una gran variedad de herramientas ubicuas que se han multiplicado con la generalización de las redes inalámbricas. Los smartphones o teléfonos móviles inteligentes son un ejemplo de tecnologías ubicuas ya que están siempre presentes y se vuelven prácticamente imperceptibles en la vida cotidiana de las personas. En el trabajo agrícola, esto también significa una ventaja importante dado que en todo momento se cuenta con una herramienta de cómputo y comunicación muy potente.

Encuestas realizadas a productores de todo el país por investigadores de INTA, arroja que un 78,2 % dice utilizar aplicaciones en las actividades que realiza a diario, mientras el 21,8 % no las incorpora como herramienta de decisión. Del total de encuestados, el 66,8 % indica que las herramientas fueron adoptadas porque mejoran la eficiencia en sus actividades incrementando la rentabilidad de las mismas [30]. A pesar de que aún existen regiones productivas que se encuentran sin cobertura de telefonía o internet, esto es cada vez menos frecuente ya que nuevas tecnologías como las redes de largo alcance y baja potencia (LPWAN, por sus siglas en inglés) se están generalizando.

En este capítulo se presentan dos aplicaciones móviles desarrolladas en conjunto con personal de la Estación Experimental INTA de Hilario Ascasubi, institución que se encarga de la promoción, distribución y mantenimiento de las mismas. Para el desarrollo de estas aplicaciones se utilizaron distintas tecnologías, en un principio se comenzó trabajando en un entorno de programación Java para compilar aplicaciones nativas para sistema operativo Android y en la segunda aplicación se optó por un modelo híbrido, que combina componentes nativos con tecnologías de desarrollo web, lo que permite que el código fuente de los componentes de la GUI y de las librerías de cálculo implementadas puedan reutilizarse para un amplio espectro de sistemas operativos e incluso permite a la aplicación ser alojada en sitios web para que sea accesible vía online.

## 4.2. Soporte para tareas de pulverización agrícola

La aplicación de agroquímicos destinada a la protección de los cultivos es una labor crítica ya que implica el manejo y liberación de productos con cierta toxicidad al ambiente. A nivel global, existe un cuestionamiento creciente por el impacto ambiental que tienen estas prácticas, que se traduce en barreras comerciales y mayor control sobre los estándares de producción.

Existe una gran brecha entre las prácticas de aplicación de agroquímicos usualmente llevadas adelante y las prácticas correctas, también denominadas buenas prácticas agrícolas. Las acciones que contribuyan a difundir y factibilizar la ejecución de las buenas prácticas agrícolas propenderán a cerrar esa brecha, reduciendo el impacto ambiental de la actividad [31]. En este marco, resulta estratégico el desarrollo de herramientas basadas en la disponibilidad tecnológica que brindan los dispositivos móviles, masivamente distribuidos en la población objetivo [32, 33].

La labor de pulverización agrícola requiere la realización previa de diferentes procedimientos, de los cuales, los de mayor relevancia son la calibración del equipo, la verificación del funcionamiento del mismo y el cálculo de las mezclas. La calibración del equipo debe realizarse para identificar el punto de operación de los cinco parámetros de aplicación: distancia entre picos, capacidad del pico, velocidad de avance, presión de trabajo y el volumen pulverizado.

En los equipos que no cuentan con regulación automática de estos parámetros, el operario debe realizar los cálculos manualmente. En los equipos con sistemas de regulación, los cálculos se realizan con cierta periodicidad para comprobar el funcionamiento del sistema. En ambos casos también se realizan para simular condiciones de operación, al momento de tomar decisiones sobre cambios en estos parámetros.

Los picos de pulverización cuentan con un orificio calibrado que permite el flujo de un determinado caudal de líquido al tiempo que lo pulverizan dando una población de gotas. Esto define el espectro de pulverización. En este sentido, existe una normalización de los picos en cuanto al caudal que erogan a una presión nominal de 3 bares [34]. De acuerdo a esta norma, la identificación de los picos queda definida por un color y el valor de este caudal nominal es expresado en galones por minuto.

Es necesario verificar que los picos entreguen el caudal correspondiente con cierta regularidad dependiendo de las horas de trabajo realizadas. Es de destacar que la normalidad en el caudal erogado por el pico tiene una alta correlación con el espectro de pulverización y puede tomarse como un indicador indirecto de la ocurrencia de anomalías sobre este aspecto. Para ello se realiza la medición del volumen de líquido descargado por cada pico en un determinado tiempo, mediante el uso de un recipiente graduado. Una vez realizada esta operación se calcula el caudal y se compara con el que debería entregar. Diferencias mayores al 10 % respecto de este valor indica cierta anomalía que se debe atender.

En el tanque de la máquina se debe incorporar el agua y los productos que van a componer el caldo de pulverización. Las cantidades de cada componente están en relación con la capacidad del tanque de la máquina, la superficie a tratar, el volumen pulverizado por hectárea y la dosis de cada producto. Es preciso incorporar las cantidades para mantener la dosis prescripta y evitar los sobrantes de caldo que luego deben ser descartados.

En condiciones de campo, la ejecución de estos procedimientos resulta compleja, la falta

de herramientas informáticas que permitan realizar los cálculos y gestionar la información correspondiente demanda una ardua tarea que atenta contra la precisión de los cálculos y la fidelidad de los registros.

#### 4.2.1. Requerimientos generales

El objetivo es desarrollar una herramienta que, en condiciones de campo, permita resolver los cálculos involucrados en los procedimientos y gestionar la información emergente de ellos. Para el desarrollo se establecieron las siguientes premisas:

- El uso de la aplicación implica el seguimiento de un protocolo de trabajo, definido implícitamente por la estructuración del entorno del usuario, y el flujo de datos.
- El entorno del usuario queda definido por una pantalla por cada herramienta de cálculo.
- El diseño permite una operación intuitiva de la aplicación.
- La transcripción de datos se realiza de manera automática entre las herramientas de cálculo, restringiendo la carga manual de información a lo estrictamente necesario.
- La funcionalidad de la aplicación y sus principales herramientas de cálculo prescinden de conectividad.
- La información generada puede guardarse y compartirse por diferentes medios en función de la disponibilidad de conectividad.

El desarrollo de los cálculos y el diseño de la estructura de la aplicación se realizaron en base a los procedimientos recomendados por el Instituto Nacional de Tecnología Agropecuaria [35, 36]. La programación se realizó utilizando el asistente de desarrollo Android Studio [37].

#### 4.2.2. Parámetros de aplicación

Considerando que, una vez realizado el alistamiento del equipo, la distancia entre picos y la capacidad del pico son parámetros fijos, sus valores constituyen datos de entrada. Con éstos, la herramienta permite calcular el punto de trabajo ingresando dos cualesquiera de los tres parámetros,  $Q$ ,  $V$  o  $P$  de la ecuación 4.1 y determinando la variable restante por medio del despeje correspondiente.

$$Q[L/ha] = \frac{\sqrt{P[bar]} \cdot k}{V[Km/h] \cdot d[m]}, \quad (4.1)$$

donde  $Q$  es el volumen de pulverizado expresado en L/ha.,  $P$  es la presión de trabajo,  $V$  la velocidad de avance de la maquinaria en Km/h,  $d$  es la separación entre picos en metros y la constante  $k$  depende de los valores nominales del pico, tal como se describe en la siguiente ecuación,

$$k = \frac{600 \cdot q_{nom}[L/min]}{\sqrt{P_{nom}[bar]}}, \quad (4.2)$$

donde  $q_{nom}$ , indicado en L/min y  $P_{nom}$  expresado en bares, son el caudal y presión nominales del pico respectivamente.

Inicialmente el usuario ingresa el valor de la distancia entre picos, para lo cual cuenta con tres opciones de valores estándar. En caso de necesitar ingresar un valor diferente, se provee una cuarta opción para acceder a un campo de datos y establecer esta variable.

Luego se requiere seleccionar el caudal de pico de una lista desplegable con la codificación de colores normalizada. Si el tipo de pico utilizado no se ajusta a la norma, se puede ingresar los valores de presión y caudal nominal provistos por el fabricante.

Por último, el usuario debe completar los valores conocidos de velocidad de avance, presión de trabajo o volumen pulverizado. Oprimiendo el botón correspondiente a la incógnita se ejecuta el cálculo de dicha variable.

Si la velocidad de avance es desconocida pero necesaria como dato de entrada para el cálculo de los otros dos parámetros, se cuenta con una herramienta complementaria para poder determinarla en base a la medición del tiempo insumido para recorrer una distancia conocida. El menú permite ingresar el valor de la distancia que va a recorrer y cronometrar el tiempo insumido. Luego de que el usuario detenga el cronómetro, se realiza el cálculo de la velocidad. Es posible realizar la determinación en tres repeticiones y el promedio de los valores calculados es exportado de manera automática al campo correspondiente de parámetros de aplicación.

### 4.2.3. Verificación de picos

En base a los parámetros de aplicación establecidos previamente, esta herramienta calcula el caudal del pico  $q$  de acuerdo a la ecuación 4.3.

$$q[L/min] = \sqrt{\frac{P[bar]}{P_{nom}[bar]}} \cdot q_{nom}[L/min]. \quad (4.3)$$

Además se permite ingresar la cantidad de picos que van a ser verificados y cronometrar el tiempo de muestreo  $t$ , para lo cual se cuenta con un cronómetro que funciona como temporizador de cuenta regresiva y permite fijar el valor de  $t$  en 30, 60 ó 90 segundos. Cuando el temporizador finaliza se emite una alerta sonora, para que el usuario interrumpa el muestreo retirando el recipiente de la zona de aspersión. Ingresando el valor del volumen recolectado  $V_r$ , la aplicación calcula el caudal del pico  $q_p$  de acuerdo a la ecuación 4.4.

$$q_p[L/min] = \frac{V_r[L]}{t[seg]} \cdot 60. \quad (4.4)$$

Simultáneamente calcula el porcentaje de desvío que el valor de  $q_p$  tiene respecto a  $q$ . En caso de que este valor se ubique por encima o debajo del 10 %, la fila de la tabla correspondiente a esa medición se marca con una cruz para indicar que es conveniente atender tal anomalía.

La herramienta genera una tabla de resultados. Para cada pico, la tabla contiene una línea de datos con la siguiente información: número de pico, caudal del pico, desvío porcentual y un símbolo indicador del resultado de la verificación.

La imagen central de la Fig. 4.1 muestra una captura de pantalla de la actividad de la aplicación que permite realizar la verificación de los picos.

The screenshot displays the application interface with three main panels:

- Parámetros de aplicación:**
  - Distancia entre picos: 0,35 m, **0,70 m**, 0,525 m, Otro:
  - Caudal de pico: **Con código de color**, Sin código de color
  - Caudal nominal: 0.80 l/min
  - Presión nominal: 3.00 bar
  - Velocidad de avance: **10.0 km/h**
  - Presión de trabajo: **2.70 bar**
  - Caudal de trabajo: 0.76 l/min
  - Volumen pulverizado: **65.0 l/ha**
- Verificación de picos:**
  - Tiempo de muestreo: **30 s**, 60 s, 90 s
  - Cantidad de picos: **10**
  - Caudal de trabajo: **0.76 l/min**
  - 00:30:00 (Timer)
  - Volumen recolectado: 0.380 l
  - Agregar
  - Generar reporte

| Pico # | Caudal ef. l/min | Desvío % | Correcto |
|--------|------------------|----------|----------|
| 1      | 0.71             | -6.37    | ✓        |
| 2      | 0.80             | 5.49     | ✓        |
| 3      | 0.77             | 1.54     | ✓        |
| 4      | 0.68             | -10.33   | ✗        |
| 5      | 0.76             | -0.04    | ✓        |
| 6      | 0.78             | 2.86     | ✓        |
| 7      | 0.82             | 8.13     | ✓        |
| 8      | 0.84             | 10.77    | ✗        |
| 9      | 0.75             | -0.57    | ✓        |
| 10     | 0.76             | 0.22     | ✓        |
- Calculador de mezcla:**
  - Lote: **XV**
  - Superficie: **40.0 ha**
  - Volumen: **65.0 l/ha**
  - Capacidad: **1500 l**
  - Cantidad de cargas: 1 carga(s) completa(s) de: 1500 l, 1 fracción de carga de: 1100 l
  - Producto, Dosis, Agregar
  - Calcular Mezcla

| N° | Producto | Dosis       |
|----|----------|-------------|
| 1  | A        | 50 ml/ha    |
| 2  | B        | 75 ml/100l  |
| 3  | C        | 100 gr/ha   |
| 4  | D        | 125 gr/100l |

Figura 4.1: Capturas de pantalla de la aplicación desarrollada.

## 4.2.4. Calculadora de mezclas

### 4.2.4.1. Cantidad de cargas

Dados los parámetros de aplicación establecidos en la sección correspondiente, esta utilidad toma como valor de entrada el volumen pulverizado  $Q$ . Ingresando la superficie a tratar  $S$  y la capacidad del tanque de la máquina  $T$ , se calcula el número de cargas  $N_c$  completas y el volumen de la fracción de carga  $V_f$ , que representa el remanente luego de restar las cargas completas, de acuerdo a las ecuaciones 4.5 y 4.6 respectivamente.

$$N_c = \left\lfloor \frac{S[ha] \cdot Q[L/ha]}{T[L]} \right\rfloor, \quad (4.5)$$

$$V_f[L] = \left( \frac{S[ha] \cdot Q[L/ha]}{T[L]} - N_c \right) \cdot T[L]. \quad (4.6)$$

En caso de que el valor de  $V_f$  resultante sea menor al 20 % de  $T$ , la herramienta emite un mensaje sugiriendo la conveniencia de equilibrar los volúmenes de carga, reduciendo el valor de  $T$ .



#### 4.2.4.2. Cantidad de producto

Partiendo de los cálculos anteriores y tomando como datos de entrada los valores de  $Q$ ,  $T$  y la dosis deseada  $V_d$ , la herramienta determina la cantidad de producto  $C_{pp}$  para cada carga completa y para la fracción de carga. Es posible especificar la dosis eligiendo alguna de las cuatro unidades que usualmente se emplean: mL/Ha, gr/Ha, mL/100L y gr/100L. Para dosis expresadas en mL/Ha o gr/Ha, la cantidad de producto por carga está determinada por la ecuación 4.7.

$$C_{pp} = \frac{T[L]}{Q[L/Ha]} V_d[mL/Ha \text{ ó } gr/Ha]. \quad (4.7)$$

En cambio, para dosis expresadas en mL/100L o gr/100L la cantidad de producto por carga viene determinado por la ecuación 4.8.

$$C_{pp} = \frac{T[L]}{100[L]} V_d[mL/100L \text{ ó } gr/100L]. \quad (4.8)$$

Luego, la sumatoria de estas cantidades, afectadas por el número de cargas arrojará el total de cada producto.

La herramienta genera una tabla de resultados. Para cada producto, la tabla contiene una línea de datos con el nombre del producto, la cantidad para cada carga completa, para la fracción de carga y la cantidad total.

### 4.3. Soporte para cálculos de siembra

La siembra es una labor agrícola delicada cuya técnica afecta a las condiciones que permiten la germinación de las semillas de cualquier cultivo. En general se emplean maquinarias que facilitan la labor de siembra y existen distintos sistemas de distribución de semilla ya que la densidad de siembra puede variar mucho entre distintas especies y variedades de cultivos.

Los parámetros de siembra involucran la densidad de siembra, que puede ser expresada en semillas o kilogramos por hectárea, semillas por metro, distancia entre semillas o población objetivo, dependiendo del sistema de siembra y tipo de distribución. Además se debe tener en cuenta la eficiencia de labor, que indica la efectividad de la maquinaria, el peso de mil semillas, la pureza y el poder germinativo de las mismas. Todos estos cálculos deben realizarse durante la planificación de la labor de siembra y también es necesario realizar verificaciones del funcionamiento de la maquinaria periódicamente.

Se diseñó e implementó una aplicación móvil basada en tecnología web que permite, además de ser instalada en dispositivos móviles con cualquier sistema operativo, ser alojada en un servidor como sitio web de manera que la herramienta es también accesible por medio de cualquier navegador a través de internet. Esta aplicación contiene herramientas para el cálculo de todos los parámetros de siembra así como también para la verificación del funcionamiento de la maquinaria y el cálculo de insumos de siembra como ser fertilizantes, insecticidas, entre otros.

### 4.3.1. Requerimientos

Siguiendo los lineamientos propuestos en el Capítulo 2, se planteó el diagrama de casos de uso que se ilustra en la Figura 4.2. La aplicación posee un modelo de negocio simple ya que, al igual que la aplicación presentada en la sección anterior, el objetivo es lograr una calculadora que asista al operario o productor encargado de la siembra en los cálculos de insumos como semilla, fertilizantes, insecticidas, inoculantes, etc., verificación de densidad de semillas y demás insumos y en la generación de reportes de labores.

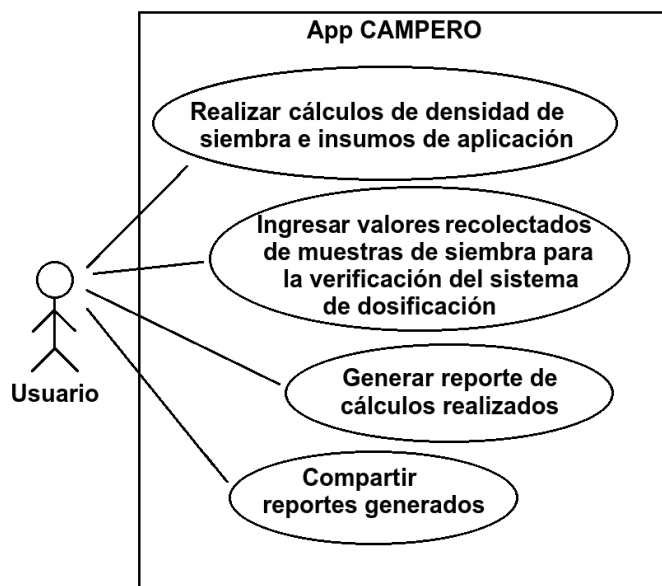


Figura 4.2: Diagrama de casos de uso de la aplicación.

A continuación se listan los requerimientos inicialmente planteados previo al comienzo del desarrollo.

#### 4.3.1.1. Requerimientos generales

- La aplicación debe contar con una herramienta de cálculo para la determinación de densidad de siembra a partir de una serie de parámetros que incluyen por un lado las variables de labor, como sistema de siembra, tipo de distribución, distancia entre líneas o surcos y cantidad de líneas por surco y por otro lado las propiedades de la semilla, como poder germinativo, peso de 1000 semillas, pureza y eficiencia de implantación.
- Se debe contar con una sección adjunta a la de ingreso de parámetros para indicar nombres de productos y densidades de aplicación de fertilizantes y otros insumos.
- Se debe contar con un apartado que permita el control a modo de verificación del funcionamiento de los cuerpos de siembra y fertilización. Para ello se debe contar con una tabla donde sea posible indicar las cantidades recolectadas como muestra de cierto número de cuerpos de siembra que el usuario debe indicar y a lo largo de un trayecto

en metros también ingresado por el usuario. En esta sección se determinarán las condiciones de trabajo de la maquinaria y se alertará sobre posibles fallas o desviaciones atípicas de los cuerpos de siembra, tanto para semilla como para fertilizantes y otros insumos.

- La sección de control debe contener un espacio especial para el caso de que el usuario indique sistema de siembra tipo “monograno” que permita realizar el control de distribución de semilla. Para ello se debe contar con cinco (5) pulsadores tipo contadores que permitan registrar las distancias entre semillas que arroja el alimentador en funcionamiento.
- Se debe contar con una sección que permita realizar cálculos de insumos de siembra, ya sea de semilla como fertilizantes u otros insumos. El formulario de esta sección debe requerir nombre de lote y superficie así como también la presentación de cada insumo de modo que sea posible calcular las cantidades de compra de cada producto previo a la realización de la labor.
- Todos los cálculos y resultados generados en cada sección debe ser posible adjuntarlos a un reporte que consiste en un resumen de toda la información ingresada y calculada. Se debe poder registrar un número de reportes que sólo esté limitado por la capacidad de almacenamiento del dispositivo.
- Los reportes guardados deben poder listarse en una sección dedicada que permita editar los nombres, borrar o visualizar los reportes guardados. Debe ser posible exportar estos reportes en formato PDF tanto para ser almacenados en el dispositivo móvil como para ser compartido externamente por correo electrónico, redes sociales, etc.
- Se debe compilar una versión móvil compatible con sistemas operativos Android y una versión tipo PWA para acceso online.
- La versión PWA debe estar disponible en un sitio web público con cifrado web SSL.
- La versión híbrida debe estar disponible para su descarga e instalación gratuita desde la tienda de aplicaciones Google Play.
- La versión híbrida de la aplicación no debe ocupar más de 10Mb de espacio de instalación en el dispositivo móvil.
- La versión híbrida de la aplicación debe requerir como único permiso, el acceso al sistema de almacenamiento de archivos. No se debe hacer uso de ningún otro periférico disponible.
- La aplicación no debe requerir datos de autenticación ni personales del usuario.

### **4.3.2. Arquitectura**

Se denomina aplicación web a aquellos programas computacionales que se ejecutan en el entorno de un navegador web. Las aplicaciones web son páginas o sitios web muy dinámicos

en cuanto a su contenido y requieren de dos actores principales que son el cliente y el servidor. El servidor suele ser un sistema computacional con acceso a internet que dispone de los recursos multimediales, por ejemplo, documentos, hojas de estilo, scripts, imágenes, audio, etc. que componen uno o más sitios web y a los que el cliente, usualmente empleando otro dispositivo, también con conexión a internet, puede acceder mediante una solicitud HTTP. Con la incorporación del lenguaje de programación javascript a los sitios web, se logra una interacción más dinámica e inteligente con el usuario y dado las mejoras en el rendimiento de los navegadores para ejecutar estas aplicaciones, han surgido una gran cantidad de herramientas denominadas *frameworks*, que facilitan la tarea de crear aplicaciones con interfaces gráficas (GUI, por sus siglas en inglés) complejas en base a los documentos HTML.

Una aplicación web progresiva (PWA, por sus siglas en inglés) es la extensión de una aplicación web que admite, mediante los navegadores que soportan esta característica, la instalación de la misma en un dispositivo como smartphone o PC, con cierto nivel de acceso al hardware y también siendo posible utilizarla en modo *offline*, es decir, sin estar conectado al servidor donde se encuentra alojada la web.

Una técnica de desarrollo de aplicaciones móviles que ha crecido rápidamente es la del desarrollo de aplicaciones híbridas. Este paradigma consiste en empaquetar una aplicación web junto a un servidor local y un *webview* en un ejecutable que es compilado para distintas plataformas. A diferencia del *webview* de un navegador web típico, el utilizado para crear aplicaciones híbridas brinda el acceso desde la aplicación web a interfaces del sistema operativo (APIs, por sus siglas en inglés) por medio de *plugins*. De esta manera es posible, por ejemplo, que una aplicación híbrida pueda leer archivos locales, utilizar la cámara integrada del dispositivo, controlar una comunicación bluetooth o puertos USB, etc., etc. La ventaja es que el desarrollador programa una gran parte de la aplicación, principalmente la GUI, una sola vez y luego, esta misma puede compilarse y distribuirse, no sólo vía web, sino como aplicación instalable para múltiples plataformas como Windows, Android, Linux, Mac o iOS, por mencionar las más conocidas.

La Figura 4.3 muestra los principales componentes de la aplicación web. Como puede observarse, la arquitectura se divide en tres secciones, modelo de negocio, *data-binding* y GUI. El modelo de negocio está compuesto por una librería con todas las fórmulas, procedimientos y ecuaciones que se detallan posteriormente, en la Sección 4.3.3. El *data-binding* es un mecanismo de acople y sincronización entre datos del modelo de negocios y la GUI, lo cual permite que cuando el usuario ingresa datos a los formularios de la GUI, éstos se cargan directamente en las ecuaciones y reglas definidas por el modelo de negocio y a la inversa, cuando las fórmulas del modelo producen resultados, éstos se ven reflejados inmediatamente en la GUI. Esto se implementó con la librería AngularJS [38]. Finalmente se dispone de las librerías que permiten lograr un aspecto visual de la GUI con estilos enriquecidos compuestos por muchos elementos gráficos como ser ventanas, botones, pestañas, íconos, sombras, transparencias, menús, etc. La librería elegida para este propósito es Framework7 [39], por su simplicidad de uso, eficiencia y aspecto visual, que se puede observar en la Figura 4.4 donde se muestra tres capturas de pantalla de la aplicación.

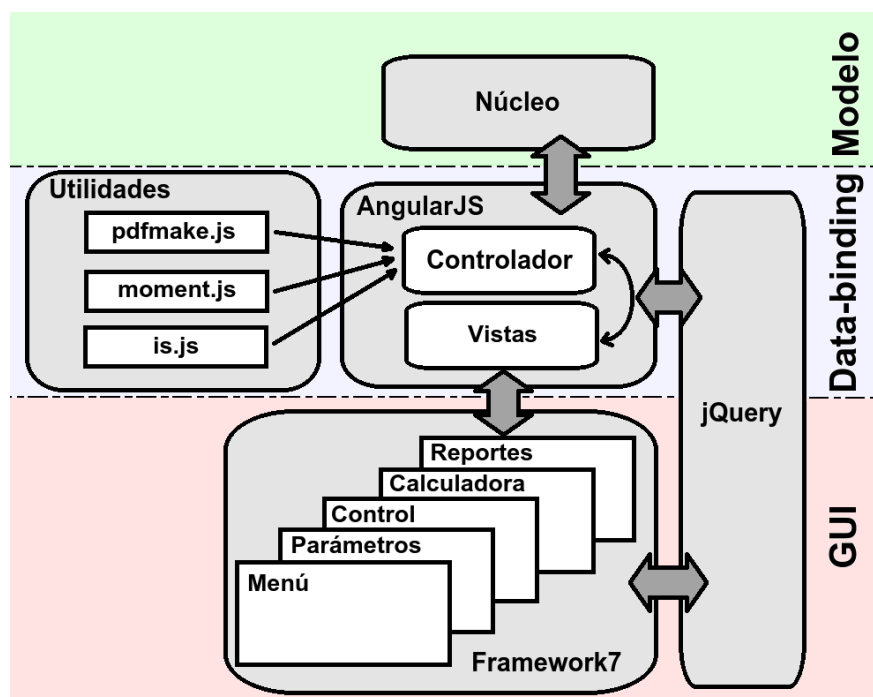


Figura 4.3: Modelo de arquitectura de la aplicación web.

### 4.3.3. Parámetros de labor y verificación de siembra

La aplicación cuenta con una sección de parámetros que permite ingresar las principales características de la labor de siembra. En primer lugar, el usuario indica si el tipo de siembra es monograno o chorrillo, a veces también denominados siembra fina o siembra gruesa respectivamente. Mediante otro selector, se ingresa el tipo de siembra, que puede ser plano o por surcos.

El primer campo que debe completar el usuario es el de distancia entre líneas, si se trata de un sistema de siembra plano y distancia entre surcos, para el caso de siembra en surcos o tabloncillos. En este último también debe indicar la cantidad de líneas de siembra por surco.

Luego es necesario completar cierta información sobre el tipo de semilla. La aplicación cuenta con varios datos por defecto en caso de que el usuario no disponga de la información requerida. Estos datos son, el peso de 1000 semillas, el porcentaje de pureza y el poder germinativo. La estimación de logro es también una variable que el usuario puede ingresar, es un parámetro que depende de varios factores y es un porcentaje que permite estimar la eficiencia de implantación.

Sobre el final de la vista de parámetros, el usuario debe ingresar la densidad de siembra deseada y lo puede expresar en seis unidades diferentes que son, kilogramos por hectárea (kg/ha), semillas por hectárea (sem/ha), semillas por metro (sem/m), distancia en centímetros entre semillas (dist/sem), semillas viables por hectárea o población objetivo (pl/ha).

Con esta información es posible calcular la densidad de siembra expresada en el resto de las unidades. Por ejemplo, si el usuario expresa la densidad de siembra ( $S_m$ ) en semillas por metro, es posible calcular los valores de la densidad según las siguientes ecuaciones:

Figura 4.4: Capturas de pantalla de la aplicación para cálculos de parámetros de siembra.

$$\begin{aligned}
 d_{\frac{kg}{ha}} &= \frac{S_m \cdot k \cdot pms}{pz \cdot 10000}, \\
 d_{\frac{sem.}{ha}} &= S_m \cdot k, \\
 d_{\frac{sem.}{m}} &= S_m, \\
 d_{\frac{dist.}{sem.}} &= \frac{100}{S_m}, \\
 d_{\frac{viab.}{m}} &= S_m \cdot k \cdot pg, \\
 pob_{obj} &= \frac{S_m \cdot k \cdot pg \cdot ef}{10000}, \\
 rect_{teo} &= S_m \cdot r
 \end{aligned} \tag{4.9}$$

Donde  $pms$  indica el peso de 1000 semillas,  $pz$  es el porcentaje de pureza,  $pg$  es el poder germinativo,  $ef$  es la eficiencia de implantación,  $l$  es la cantidad de líneas por surco,  $r$  es la distancia entre líneas y la constante  $k$  equivale a

$$k = \frac{10000 \cdot l}{r}. \tag{4.10}$$

Al realizar el cálculo y guardar los parámetros ingresados, la aplicación ya cuenta con datos suficientes para realizar una verificación de la descarga de los dosificadores. Para esto

el usuario indica la cantidad de líneas que desea controlar y la distancia a lo largo de la cual realizará la recolección de semillas. Estas muestras recolectadas se deben pesar e ingresar los valores en una tabla que permite determinar la densidad de siembra promedio o densidad efectiva, la desviación respecto a lo calculado y el desvío individual de cada alimentador, alertando en caso de que la densidad de siembra de algún cuarto arroje un desvío superior al 10 %.

En caso de que el usuario esté trabajando con un tipo de siembra de monograno, donde el distanciamiento entre semillas es un parámetro crítico, la aplicación permite verificar la calidad de distribución de esta variable. En caso de haber realizado el control de siembra, se dispone de un valor real de la distancia entre semillas o distancia entre semillas efectiva. Si no se realizó este control, se puede, o bien importar el valor calculado en la sección de parámetros o sino ingresar un nuevo valor. Luego el usuario dispone de una serie de contadores que debe incrementar dependiendo de la separación entre semillas que arrojó el dosificador para finalmente obtener los índices de calidad de distribución. El valor de contador  $D$  indica la cantidad de semillas que se encuentran separadas a menos de la mitad de la distancia correcta. El contador  $A$  indica la cantidad de semillas que están distanciadas dentro del intervalo de  $\pm 50\%$  del valor de separación correcto. Con los pulsadores de los contadores  $F_s$ ,  $F_d$  y  $F_t$  se cuentan las semillas que caen en los rangos (150 % – 250 %), (250 % – 350 %) y 350 % en adelante respectivamente. Las fórmulas de los índices de calidad de distribución se listan a continuación:

$$\begin{aligned} indice_d &= \frac{100 \cdot D}{A + 2 \cdot F_s + 3 \cdot F_d + 4 \cdot F_t} \\ indice_a &= \frac{100 \cdot (-D + A + F_s + F_d + F_t)}{A + 2 \cdot F_s + 3 \cdot F_d + 4 \cdot F_t} \\ indice_f &= \frac{100 \cdot (F_s + F_d + F_t)}{A + 2 \cdot F_s + 3 \cdot F_d + 4 \cdot F_t} \\ n &= A + F_s + F_d + F_t \end{aligned} \tag{4.11}$$

El control de dosificación se puede realizar tanto para semillas como para fertilizantes de línea y lateral y otros insumos siguiendo el procedimiento explicado anteriormente.

#### 4.3.4. Calculador de insumos y generación de reportes

El cálculo de insumos consiste en determinar las cantidades de semilla, fertilizante y otros productos en base a la densidad de aplicación, área de siembra y presentación, en cuanto a formato de venta de dichos productos. En el caso de las semillas, dependiendo de sus características se comercializan a granel o en envases de cierto peso y también por número de semillas. Lo mismo ocurre con los fertilizantes y otros insumos.

La aplicación cuenta con una sección para realizar estos cálculos y se puede emplear tanto los valores calculados e ingresados durante el cálculo de parámetros o verificación de la máquina como también ingresando directamente las cantidades en dicha sección. Luego de completar los datos se presenta al usuario una lista de los productos que debe contar para poder realizar la labor.

Todos los cálculos se pueden ir anexando a un reporte acumulativo y finalizado el mismo,

se guarda en la memoria interna del dispositivo móvil pudiendo exportarse luego como documento PDF para ser compartido externamente por otros medios como correo electrónico, servicios de mensajería o redes sociales.

## 4.4. Consideraciones finales

Entre las principales ventajas que emergen del uso de estos desarrollos, vale mencionar que la estructura y flujo de información establecidos generan una protocolización de cada procedimiento ordenando, en cierta medida, los pasos a seguir por el usuario. Además, la capacidad de almacenar y compartir la información garantiza la fiabilidad de los datos registrados y permite la interacción entre los diferentes actores involucrados en el proceso, es decir, profesionales, productores, operarios, entre otros.

Por otro lado, la informatización de los cálculos libera a los operarios de la responsabilidad de realizarlos en forma manual, reduciendo sensiblemente la posibilidad de errores. Además, la concentración de todas las utilidades en el dispositivo móvil reduce la cantidad de elementos necesarios para realizar los diferentes procedimientos.

Las aplicaciones, denominadas Criollo y Campero, tienen una gran difusión en el medio, dado que se han registrado más de 10.000 descargas en total y de acuerdo a la valoración de los usuarios, cuentan con puntuaciones superiores al 90 % en la tienda de aplicaciones Google Play.

El código fuente de la app Criollo permitió la derivación de otras dos aplicaciones especializadas, por un lado, en pulverización para especies arbóreas mediante atomizadores y por el otro, en fumigación con mochila. De igual modo, se planea extender una versión similar a Campero para los cálculos relacionados con la aplicación de fertilizantes.

Los desarrollos presentados en este capítulo consisten en software de aplicación que son ejecutados por el sistema operativo, en este caso, de teléfonos móviles. En el capítulo que sigue a continuación, se vuelve a descender un nivel de abstracción para tratar cuestiones referidas a los sistemas operativos, particularmente aquellos que son de tiempo real duros, a diferencia de los empleados en móviles, que se denominan blandos.



# Capítulo 5

## Sistemas de tiempo real

### 5.1. Motivación

En el contexto de los sistemas computacionales, las tareas de tiempo real son aquellas que deben ejecutarse antes de cierto vencimiento para producir resultados correctos [40]. Dependiendo del tipo de sistema y la naturaleza de las tareas a ejecutar, la planificación de las mismas puede ser *online* u *offline*. En el primer caso, es el sistema operativo el encargado de decidir qué tarea debe ejecutarse en cada instante de tiempo dependiendo de la regla de planificación implementada y para esto se cuenta con algoritmos de planificación dinámicos o estáticos. En el segundo caso, el diseñador calcula el esquema de planificación, también llamado tabla de tareas, en una instancia de diseño y el sistema una vez iniciado se encarga de ejecutar estas tareas en el orden estipulado. En este capítulo se evalúan metaheurísticas para determinar un esquema de planificación que logre un balance óptimo entre el factor de utilización del sistema y el ahorro de energía.

El análisis planteado para arquitecturas multicore se puede extender también para resolver algunas cuestiones referidas al problema de la transmisión de mensajes por diferentes canales de frecuencia y multiplexado en tiempo, tal como se detalla en el capítulo 6. Cuando los requerimientos asociados a las tareas tienen restricciones temporales es preciso definir políticas de planificación que resulten adecuadas para garantizar el correcto funcionamiento. Los resultados obtenidos en este capítulo son por lo tanto relevantes para todas las etapas que conlleva la adquisición, transmisión y procesamiento de la información.

Se presenta un método basado en metaheurísticas para resolver el problema de planificación de tareas de tiempo real con parte opcional y obligatoria en procesadores multinúcleo homogéneos con consideraciones de energía, un problema que es de tipo NP-difícil. Se introduce también el modelo de una función objetivo para maximizar el rendimiento del sistema considerando, por un lado la ejecución de tareas durante tiempo adicional y por el otro el beneficio de reducir la velocidad de los procesadores y el consumo de energía, teniendo en cuenta el costo por cambiar el contexto de ejecución de tareas mientras se cumplen las restricciones de tiempo real del sistema.

Las metaheurísticas utilizadas incluyen los Algoritmos Genéticos y la Optimización por Enjambre de Partículas. Los resultados se comparan contra métodos de Programación Lineal Entera, evaluando un conjunto de sistemas sintéticos generados a partir de las pautas

propuestas en trabajos anteriores. Finalmente se analizan los sistemas que contienen restricciones de precedencias y se proponen métodos para extender las metaheurísticas implementadas para resolver, a modo de ejemplo, distintos sistemas con estas características que fueron propuestos en la literatura.

## 5.2. Modelo del sistema

El tiempo se supone discreto, dividido en ranuras y se nota  $h = 1, 2, \dots, H$ . La duración de cada ranura es un parámetro de diseño que se determina en base a la naturaleza de las tareas y características del sistema [41]. Para sistemas embebidos como los propuestos en el capítulo 3, un valor típico para la duración de la ranura puede ser entre 10 y 100 ms. Todos los eventos del sistema se sincronizan con el comienzo de cada ranura de tiempo.

Luego se considera un conjunto de tareas apropiables, periódicas y en principio independientes, que se notan como  $\tau = 1, 2, \dots, N$ , siendo  $N$  el número total de tareas del sistema. Algunas de estas tareas implementan algoritmos *anytime*, es decir, luego de cierto tiempo de ejecución mínimo, pueden devolver resultados válidos aún si se interrumpen en algún momento previo a su finalización y la calidad del resultado mejora mientras más tiempo se ejecutan. Cada tarea entonces, está descripta por una tupla  $(m, o, P, D, r)$  donde  $m$  representa la cantidad de ranuras de tiempo que debe ocupar como mínimo obligatorio,  $o$  es una cantidad adicional de ranuras que puede ejecutar opcionalmente,  $P$  es el período, que indica el tiempo que transcurre entre cada instancia de ejecución de cada tarea y se supone que el vencimiento  $D$  es tal que  $P = D$ , es decir, las tareas tienen una ventana de tiempo de  $P$  ranuras para completar su ejecución. Para cada tarea existe una función  $f_o$  que computa la recompensa o el beneficio que se obtiene por la ejecución de cada ranura opcional. Por ejemplo, si una dada tarea  $\tau$  ejecuta  $x$  ranuras de tiempo dentro de una ventana de  $P$  ranuras, siendo  $x \geq m$ , entonces la recompensa será  $r = f_o(x - m)$ , ya que la cantidad de ranuras opcionales de  $\tau$  ejecutadas es  $x - m$ .

Cada tarea tiene un periodo de ejecución y el mínimo común múltiplo de los períodos de todas las tareas que componen el sistema, se denomina hiperperiodo y se nota  $H = mcm(P_1, P_2, \dots, P_N)$ . El hiperperiodo define una ventana de tiempo en la cual el sistema repite el orden de ejecución de las tareas, con lo cual, el esquema de planificación que debe definir el diseñador, contiene la asignación de tareas a ranuras para una tabla de  $H$  ranuras en total.

El modelo de sistema contempla arquitecturas con múltiples núcleos de procesamiento y que admiten distintos modos de potencia. El número de núcleos que tiene el procesador se nota  $M$ . Para reducir el consumo de energía, los núcleos de un procesador pueden reducir su frecuencia de operación independientemente [42]. Entonces, además de asignar tareas a las ranuras de tiempo de los distintos núcleos, es necesario indicar el modo de potencia al cual se ejecuta esa tarea y, eventualmente, la tarea consumirá más ranuras de tiempo para completarse. Por ejemplo, una tarea  $\tau$  que está definida tal que  $m_\tau = 3$ , requerirá 6 ranuras si el núcleo del procesador trabaja al 50 % de su frecuencia nominal y 12 si trabaja al 25 %. Debido a que reducir la frecuencia de operación de un núcleo de procesamiento tiene como consecuencia el aumento de asignación de ranuras, el ahorro de energía también produce una recompensa, al igual que la ejecución de partes opcionales, por lo tanto, la solución a calcular

debe tener un balance óptimo entre la ejecución de ranuras opcionales de las tareas que mayor función de recompensa poseen, a la vez que se ahorra energía reduciendo la frecuencia de operación de los núcleos. Se nota como  $\bar{F}$  al valor de la frecuencia promedio de ejecución de todo el sistema y se nota  $\bar{E}$  al valor opuesto, es decir, el ahorro de energía del plan de ejecución calculado, es decir  $\bar{E} = 1 - \bar{F}$ . El valor de  $\bar{F}$  se puede calcular como,

$$\bar{F} = \frac{\sum_{m=1}^M \sum_{h=1}^H f_{h_m}}{H \cdot M}. \quad (5.1)$$

Donde  $f_{h_m}$  indica el valor de frecuencia de operación del núcleo  $m$  relativo al valor nominal en la ranura  $h$ , por ejemplo puede tomar los valores 1, 0.5, 0.25, etc.

Como restricción se impone que las tareas no pueden iniciar una instancia en un núcleo del procesador y finalizar en otro núcleo distinto, ya que esto implicaría migrar datos entre las memorias caché de cada núcleo, lo cual representa un costo computacional apreciable. Por lo tanto, cada instancia de las tareas inician y terminan su ejecución en un mismo núcleo, aunque a lo largo del hiperperiodo, estas instancias pueden asignarse a distintos procesadores.

El factor de utilización del sistema es un valor que indica el nivel de ocupación de ranuras y puede expresarse en porcentaje o en escala 0 a  $M$ . Se nota  $UF$  y se calcula según,

$$UF = \sum_{\tau=1}^N \frac{m_{\tau}}{P_{\tau}}. \quad (5.2)$$

Debido a que la ejecución de tareas opcionales incrementa el factor de utilización total del sistema y en cada instancia se puede ejecutar distinto número de partes opcionales, se puede generalizar el valor del  $UF$  como,

$$UF_t = \sum_{\tau=1}^N \frac{k_{\tau}}{H}. \quad (5.3)$$

Donde  $k_{\tau}$  indica la cantidad de ranuras ocupadas por la tarea  $\tau$  en todo el hiperperiodo  $H$ . Si el valor de  $UF_t$  es tal que  $UF_t = M$ , significa que la ocupación de ranuras es del 100 %.

En la Figura 5.1 se muestra a modo de ejemplo, el esquema de planificación de un conjunto de 10 tareas que definen un hiperperiodo de 30 ranuras a ejecutarse en un sistema compuesto por núcleos homogéneos denominados  $\pi_1$  y  $\pi_2$ , y que poseen cuatro niveles de potencia cada uno. Puede observarse que el procesador  $\pi_1$  corre a frecuencia nominal todo el tiempo y el procesador  $\pi_2$  ejecuta las primeras dos ranuras del hiperperiodo al 50 % de frecuencia nominal y el resto lo hace al 25 %.

Para poder determinar una solución óptima, es necesario establecer un marco comparativo entre soluciones que permita evaluar cada una de manera de decidir cuál es la *mejor* solución. Para lograr esto se define una función objetivo que califica con un puntaje  $q$  cada posible solución del problema según la siguiente ecuación simplificada,

$$q = \alpha \cdot \sum_{\tau=1}^N r_{\tau} + \beta \cdot \bar{E}. \quad (5.4)$$

Donde  $\alpha$  y  $\beta$  son parámetros de ajuste que debe definir el diseñador en pos de priorizar soluciones que obtengan recompensa por la ejecución de partes opcionales de las tareas o bien que priorice el ahorro de energía total logrado respectivamente.

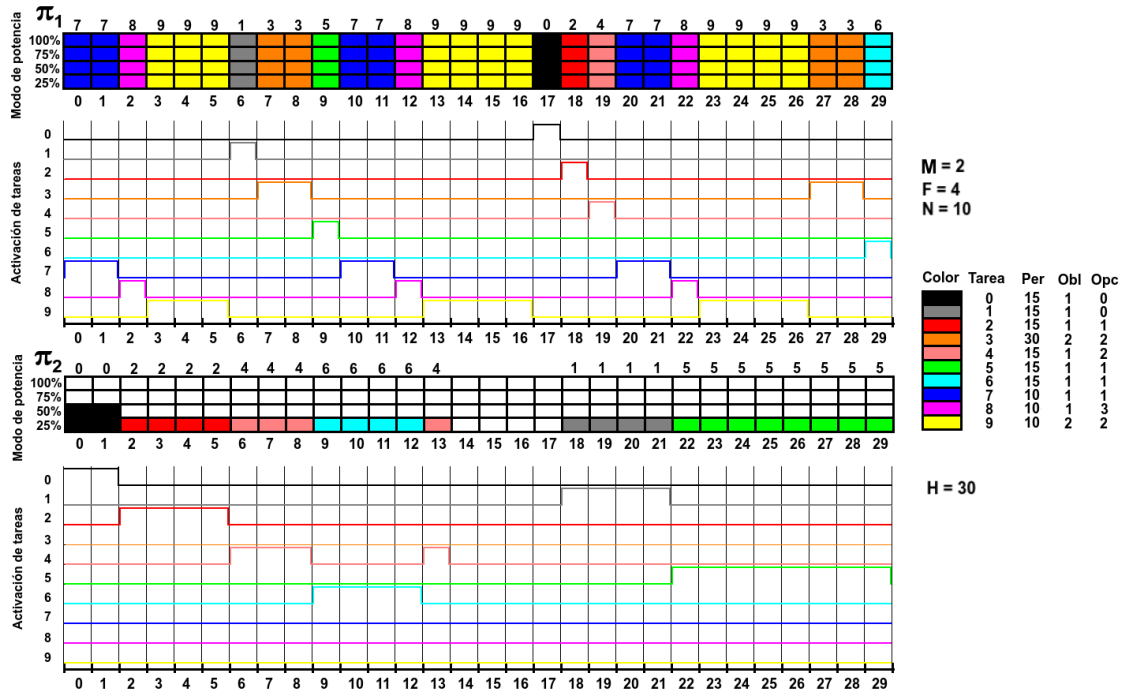


Figura 5.1: Esquema de planificación de 10 tareas en un sistema doble núcleo.

La principal condición que debe satisfacer el sistema es la factibilidad, que establece que para cada tarea  $\tau$  se ejecuten  $m_\tau$  ranuras dentro de todos los períodos  $P_\tau$ . Si se utiliza como variable de optimización al arreglo de asignación de tareas a ranuras (como en el ejemplo de la Figura 5.1), existe un gran número de soluciones posibles que no son factibles, por no cumplir la condición de tiempo real. Para reducir la complejidad del problema y buscar soluciones en un espacio de menores dimensiones, se emplea como variable de optimización, un arreglo en el que se indica, para cada instancia de cada tarea, la asignación del par procesador-frecuencia que le correspondería. Luego, mediante un algoritmo de planificación, se intenta distribuir las tareas en las ranuras empezando por las tareas de mayor factor de utilización. La Figura 5.2 ilustra mediante un diagrama de bloques, la separación entre los distintos algoritmos que forman parte de lo que sería el *pipeline* de optimización.

### 5.3. Eficiencia de Pareto

Tal como se explica en la sección anterior, el diseñador debe elegir valores para los parámetros  $\alpha$  y  $\beta$  para regular las características de la solución óptima que se necesite. Sin embargo, para estimar qué tan buena es una solución, independientemente de los parámetros de ajuste, es necesario establecer otro marco comparativo que trascienda la función objetivo. Para esto se define un espacio con las variables del problema que representan un beneficio, y que son el factor de utilización opcional y el ahorro de energía.

La Figura 5.3 muestra el espacio de soluciones donde se ubica en el eje horizontal, el factor de utilización y en la escala vertical, el ahorro de energía logrado. El valor del factor

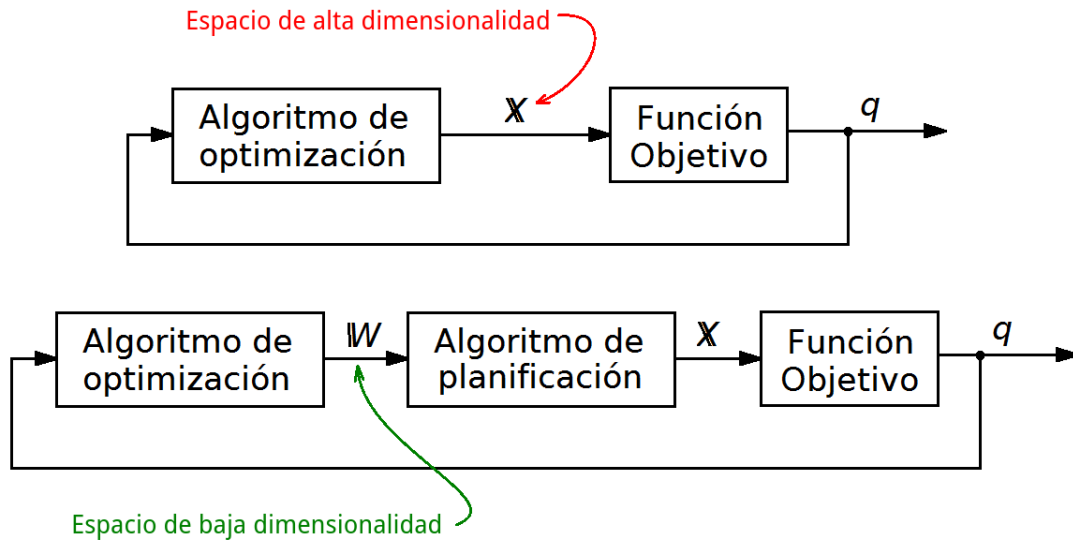


Figura 5.2: Diagramas de bloques del modelo de optimización con y sin algoritmo de planificación.

de utilización se divide entre el número de núcleos para obtener la misma escala independientemente del número de núcleos del sistema con el que se trabaja. Como se observa, existe un valor máximo de ahorro de energía posible que se logra a la mínima frecuencia de operación de los núcleos. La línea roja se denomina *Frontera de Pareto* del problema y las soluciones que se encuentren sobre esta línea son óptimos de Pareto, ya que no pueden mejorar en ningún aspecto sin desmejorar en otro. Por encima de este límite, es decir, hacia arriba o hacia la derecha, las soluciones no son factibles, porque no satisfacen la condición de vencimiento de partes obligatorias. Dado que para todos los sistemas es posible calcular la ecuación de la curva que define la Frontera de Pareto, es posible determinar la distancia de cualquier solución calculada a esta curva y por lo tanto estimar qué tan cerca se encuentra de un óptimo hipotético. Este esquema permite evaluar la solución de cualquier sistema, independientemente de sus parámetros como cantidad de tareas, cantidad de ranuras obligatorias y opcionales, períodos, recompensas, cantidad de núcleos y modos de potencia.

## 5.4. Métodos de optimización

En esta sección se describen tres técnicas de optimización aplicadas al problema y su desempeño.

### 5.4.1. Búsqueda aleatoria

El método de la búsqueda aleatoria (RS, por sus siglas en inglés) consiste en generar soluciones mediante alguna distribución aleatoria particular y recordar la mejor solución luego de muchas iteraciones. Si el espacio de búsqueda es discreto y muy reducido, entonces es probable dar con una solución óptima sin recurrir a un método complejo.

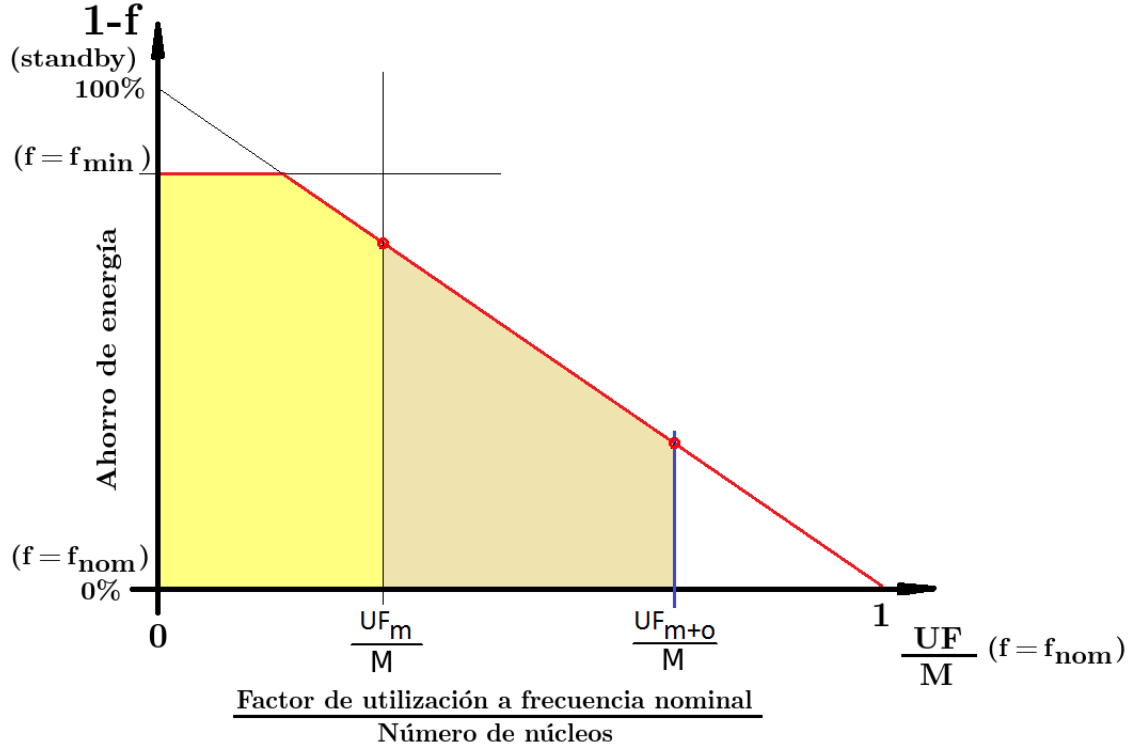


Figura 5.3: Frontera de Pareto del problema de optimización.

En el caso del problema tratado en este capítulo no es probable dar con soluciones aceptables mediante este método. Sin embargo, es útil para evaluar el comportamiento de los bloques de planificación y función objetivo, realizar una primer exploración del espacio de búsqueda y definir una escala de aceptabilidad para otros métodos de optimización heurísticos.

Este método también puede utilizarse para obtener una población de soluciones con ciertas características para inicializar otras técnicas, por ejemplo basadas en inteligencia de enjambre o que emplean una población de soluciones.

#### 5.4.2. Enjambre de partículas

El método de optimización por enjambre de partículas (PSO, por sus siglas en inglés) simula el comportamiento social de una población de individuos moviéndose por el espacio de búsqueda siguiendo un par de reglas muy simples, compuestas por las ecuaciones que se describen a continuación,

$$\vec{v}_t = \omega \vec{v}_{t-1} + \phi_g r_g (\vec{g} - \vec{x}_{t-1}), \quad (5.5)$$

$$\vec{x}_t = \vec{x}_{t-1} + \vec{v}_t, \quad (5.6)$$

donde  $x_t$  y  $v_t$  representan la posición y velocidad de cada partícula respectivamente.  $\omega$  y  $\phi_g$  son los parámetros de ajuste del método, que controlan el comportamiento del enjambre,  $r_g$

|             | RS    |              |        | PSO   |        |        | GA            |        |              |
|-------------|-------|--------------|--------|-------|--------|--------|---------------|--------|--------------|
|             | Q     | ET           | DP     | Q     | ET     | DP     | Q             | ET     | DP           |
| Primer set  | 87.96 | <b>7.3</b>   | 77.74  | 94.54 | 8.97   | 36.51  | <b>104.29</b> | 18.26  | <b>30.82</b> |
| Segundo set | 90.22 | <b>76.7</b>  | 139.79 | 90.78 | 106.91 | 91.83  | <b>100.11</b> | 135.51 | <b>35.92</b> |
| Tercer set  | 84.45 | <b>10.63</b> | 287.78 | 87.53 | 12.23  | 143.95 | <b>102.11</b> | 16.62  | <b>40.82</b> |
| Promedio    | 88.82 | <b>39.81</b> | 121.25 | 92.38 | 54.74  | 90.76  | <b>102.20</b> | 72.68  | <b>35.85</b> |

Cuadro 5.1: Resultados comparativos resaltando los mejores valores de cada métrica.  $Q$ =Calidad,  $ET$ =Tiempo de ejecución (seg.),  $DP$  = Distancia a la Frontera de Pareto.

es un número aleatorio uniformemente distribuido entre 0 y 1 y  $\vec{g}$  es la posición de la mejor solución hallada por todo el enjambre. Esta ecuación es una variante del método denominado PSO-VG, ya que sólo considera la inercia y la fuerza de atracción hacia la mejor posición descubierta por toda la población de partículas, a diferencia de otros enfoques en los que se incorporan otras componentes que guían a las partículas en otras direcciones [43]. En la Sección A.1 del Apéndice A se detalla el pseudocódigo utilizado por este algoritmo.

### 5.4.3. Algoritmos genéticos

Los algoritmos genéticos (GA, por sus siglas en inglés) logran el mecanismo de optimización numérica imitando el proceso de selección natural de las especies. Al igual que PSO, cuenta con una población de individuos a los que se los denomina cromosomas, que a la vez contienen una serie de genes que codifican la información de la variable de optimización. Mediante un algoritmo iterativo, los cromosomas se seleccionan para reproducirse si, en el contexto de la optimización, la calidad de los mismos es superior a la media, caso contrario se extinguen. La reproducción o entrecruzamiento de cromosomas es una combinación de dos soluciones donde se combinan los atributos de cada una en distintas proporciones para lograr una generación de descendientes que competirán durante el proceso de selección de la siguiente iteración. También se incluyen componentes aleatorias al método mediante un proceso denominado mutación, en el cual se alteran los valores de algunos genes de cada cromosoma, lo cual evita el estancamiento y acelera el descubrimiento de nuevas y mejores soluciones. En la Sección A.2 del Apéndice A se detalla el pseudocódigo de cada una de las instancias del algoritmo y el proceso iterativo con el que se calculan las generaciones.

## 5.5. Evaluaciones experimentales

Para una evaluación completa de los métodos presentados se tomaron tres conjuntos de problemas con descripción de distintos sistemas que representan diferentes situaciones en aplicaciones de tiempo real. Para los primeros dos sets se generaron 100 instancias con distintos factores de utilización y combinación de partes obligatorias y opcionales [44, 45]. El tercer set de problemas se compone de 11 tareas donde se varía el factor de utilización opcional del 0 al 80 % respecto del obligatorio [46]. Se considera para todos los casos, una arquitectura de doble núcleo con cuatro modos de potencia. El propósito de la evaluación es determinar qué tan rápidas y eficientes son las alternativas al ser comparadas contra métodos

como el de programación lineal entera (ILP, por sus siglas en inglés) propuesto en [41].

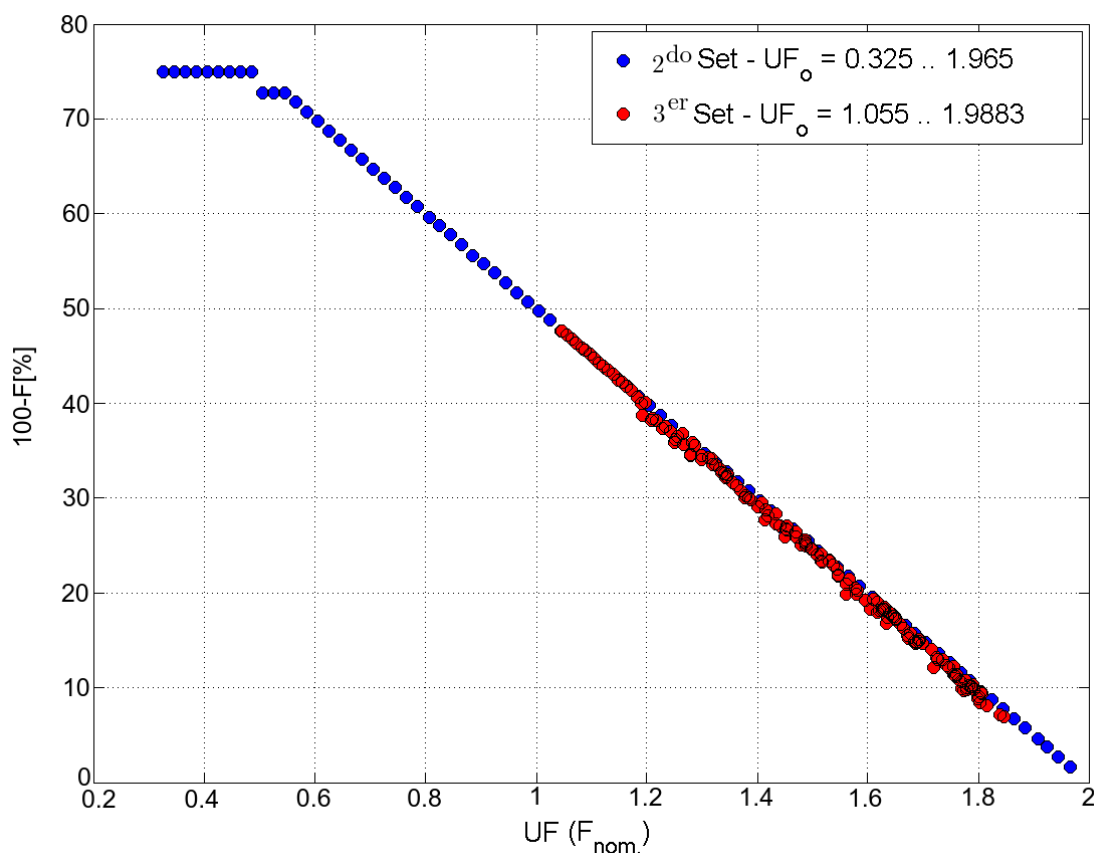


Figura 5.4: Resultados hallados en el espacio de soluciones simplificado.

El número de iteraciones realizado con cada método fue seleccionado tal que se mantenga la cantidad de evaluaciones de la función objetivo, constante para los tres algoritmos.

A partir de los resultados que se listan en la Tabla 5.1, se concluye que las mejores soluciones, en términos de distancia a la Frontera de Pareto (DP) y calidad (Q), calculado mediante la ecuación 5.4 se obtienen con el método GA. El método más veloz para completar el proceso iterativo es, naturalmente, la búsqueda aleatoria, pero se tiene un costo considerable si se tiene en cuenta la calidad de las soluciones.

En la Figura 5.4 se grafica el par  $UF, \bar{E}$  de cada solución. Como puede observarse, la gran mayoría de los resultados se ubican muy próximos a una línea recta que claramente puede identificarse como la Frontera de Pareto. Cambiando los valores de los parámetros de ajuste  $\alpha$  y  $\beta$  se puede lograr que una solución se mueva a lo largo de esta curva. Esto ocurre porque en el caso particular de los problemas utilizados, se tienen hiperperíodos extensos lo cual brinda cierta flexibilidad para distribuir las tareas y el problema se comporta como si fuera continuo. En el caso de un sistema con un esquema de planificación de pocas ranuras, se puede apreciar más las características discretas del modelo y también es más difícil alcanzar la Frontera de Pareto.

La Figura 5.5 muestra, para un problema particular, la distancia calculada de la solución a la Frontera de Pareto a medida que se varían los parámetros de ajuste. Como puede



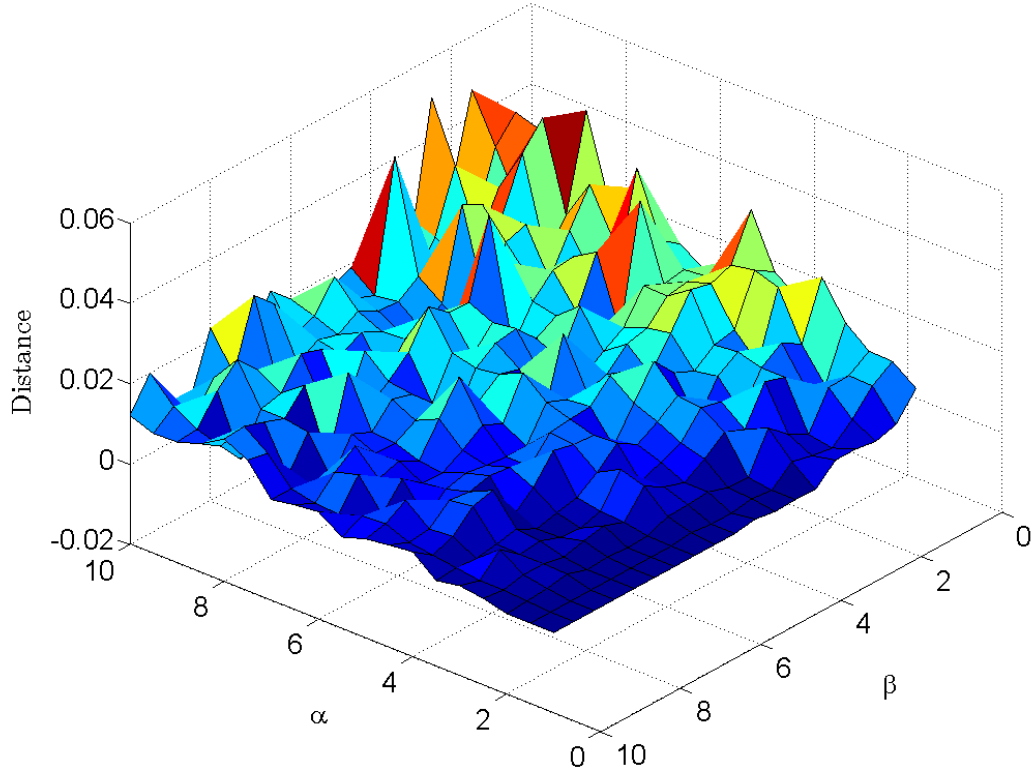


Figura 5.5: Distancia a la Frontera de Pareto al variar parámetros de ajuste.

observarse, valores altos para  $\alpha$  y bajos para  $\beta$  incrementan las distancias a la Frontera de Pareto. Esto se debe a que los resultados que se obtienen con  $\alpha \gg \beta$  buscan maximizar la recompensa obtenida por la ejecución de ranuras opcionales descuidando algunos huecos que no pueden ser ocupados por otras tareas y tampoco reduciendo la frecuencia de operación, lo que significa que la solución óptima calculada, se aleja de un óptimo ideal hipotético.

## 5.6. Consideraciones finales

En general, los procesos agropecuarios que implican el crecimiento y desarrollo de especies vegetales y animales son procesos lentos, con lo cual no es común la aplicación de conceptos de tiempo real a propósitos de este tipo, ya que apuntan a sistemas dinámicos en los que el tiempo es crítico. Sin embargo, la planificación y gestión de recursos es una problemática que está presente en todos los sectores productivos y las técnicas de resolución de problemas de optimización presentadas en este capítulo pueden extenderse perfectamente para resolver dichas cuestiones.

En este capítulo se presentó la aplicación de metaheurísticas bioinspiradas para calcular planes *offline* de ejecución de tareas para sistemas embebidos multinúcleo con restricciones temporales y de energía. Se aplicaron conceptos de Eficiencia de Pareto para establecer un

marco de evaluación general de las soluciones calculadas y para la evaluación de los métodos se emplearon distintos problemas extraídos de la bibliografía de sistemas de tiempo real.

El modelado del problema e implementación de soluciones computacionales ad-hoc representa una ventaja frente a técnicas basadas en ILP tanto desde el punto de vista del tiempo de cómputo como de la calidad de las soluciones obtenidas. Cuando se emplean métodos o librerías estándar, el modelo matemático del problema se aleja de la realidad en cierta medida y la implementación de los resultados obtenidos no es trivial o simplemente no funciona en la práctica. Por ello el desarrollo de modelos matemáticos que se ajusten a la problemática, permite simular la situación miles o millones de veces para lograr mejoras considerables al diseño.

En los capítulos siguientes se aplican procedimientos similares de modelado y simulación para resolver problemáticas referidas al diseño óptimo de redes de comunicación.

# Parte III

## Transmisión de datos

# Capítulo 6

## Redes de comunicación de alta latencia

### 6.1. Motivación

Las redes de comunicación en las cuales la ocupación del canal es sustancialmente menor a la demora de transmisión de datos extremo a extremo constituyen una clase particular de problemas en la que la transmisión y la recepción de los mensajes se realizan en instantes separados y sin que el transmisor pueda determinar, por la interferencia del canal de comunicación, la corrupción eventual del mensaje. Esta situación se plantea en diversos tipos de redes de comunicación como las Low Power Wide Area Networks (LPWN, por sus siglas en inglés), las redes acústicas subacuáticas y las transmisiones espaciales. Se diferencian de las redes inalámbricas basadas en protocolos como IEEE 802.11 e IEEE 802.15 en las cuales el sensado del canal de comunicación permite determinar más allá de los problemas de estación oculta y expuesta la probabilidad de éxito de la transmisión.

Las redes de sensores submarinos representan un importante campo de investigación debido a la gran diversidad de aplicaciones que es posible lograr con esta tecnología, como por ejemplo sistemas de alerta de tsunamis o derrames de petróleo, monitoreo de navíos o de ecosistemas submarinos. La mayoría de estos sistemas de monitoreo y alertas requiere de comunicación de tiempo real en redes de área amplia que tienen baja densidad de nodos. El medio de comunicación involucrado en estas redes es severo e impone serias restricciones al proceso de comunicación. En este escenario, la transmisión de información en tiempo real se logra principalmente usando señales acústicas, ya que los nodos de la red no se encuentran físicamente próximos. Las características y los requerimientos del entorno y el proceso de comunicación representan grandes desafíos para los diseñadores, tanto de los protocolos de comunicación como de los sistemas de monitoreo y alertas. La falta de modelos para representar estas redes es el principal obstáculo para la proliferación de los sistemas submarinos ubicuos.

En este capítulo se presenta un modelo de comunicación de tiempo real para redes de sensores acústicos subacuáticos (UW-ASN, por sus siglas en inglés) que son diseñados para cubrir grandes áreas con baja densidad de nodos usando un modelo de comunicación *any-to-any*. Este modelo es analítico, considera dos enfoques de solución para planificar los mensajes

de tiempo real y provee un análisis de las cotas de tiempo real para el desempeño de la red. Utilizando este modelo, los diseñadores de protocolos y sistemas submarinos ubicuos, pueden prototipar y evaluar rápidamente sus soluciones de manera evolutiva, con el fin de determinar la mejor solución al problema que se está resolviendo. La viabilidad de la propuesta se ilustra con un caso de estudio que muestra el rendimiento de las UW-ASN bajo distintas condiciones iniciales. Este es el primer modelo analítico para representar un sistema de comunicación en este tipo de red y por lo tanto, abre una ventana al desarrollo de sistemas submarinos ubicuos para distintos casos de aplicación.

El modelo de comunicación de tiempo real contempla mensajes periódicos que deben viajar desde transmisor a receptor dentro de una ventana de tiempo acotada. En este sentido, el desafío consiste en determinar una trama periódica dividida en ranuras de tiempo en la que los nodos que componen la red deben transmitir y recibir los mensajes evitando las interferencias y colisiones que se producen dada la naturaleza del medio. Cuanto menor es la cantidad de ranuras de la trama, el periodo de transmisión se reduce y por lo tanto, el rendimiento de la red es mayor.

## 6.2. Modelo del sistema

La propagación del sonido bajo el agua ha sido estudiada en la literatura y las leyes físicas que intervienen son bien conocidas, sin embargo, el establecimiento de un enlace de comunicación entre dos nodos sigue siendo relativamente difícil. En este trabajo se presentan los modelos propuestos para representar la propagación de mensajes, la sincronización, la transmisión de mensajes de tiempo real y el ruteo de los mismos por la red.

Las señales acústicas submarinas se propagan por un canal físico con límites en la superficie y en el fondo marino sobre las cuales la señal rebota generando réplicas del mensaje que llegan al destino en distintos instantes de tiempo, debido a que recorren mayores distancias que el frente de onda que viaja en línea recta. Este efecto se conoce como *multipath* o propagación multicamino y también se produce en señales de tipo electromagnéticas afectando a los sistemas de radiocomunicación. Aunque la potencia de las réplicas es menor que la del mensaje original, la colisión de mensajes en la posición en que se encuentra un nodo resulta en un tipo de interferencia destructiva lo cual impide al receptor decodificar el mensaje. La Figura 6.1 ilustra la situación.

El tiempo se considera una variable discreta y, en un esquema similar al modelo empleado en la sección 5.2 del Capítulo 5, puede representarse como una secuencia de ranuras numeradas  $t \in 1, 2, \dots, L$ , donde  $L$  es la duración de la trama de comunicación que es una ventana de tiempo en la que los eventos de comunicación del sistema se repiten constantemente. La duración de la ranura depende de varios factores como la duración de la transmisión y recepción de mensajes y los intervalos que se requieren para generarlos o procesarlos. Se supone que en una ranura de tiempo, un nodo puede, o bien transmitir un mensaje o recibirlo.

La topología de la red puede modelarse mediante un grafo dirigido  $G = (V, E)$  donde  $V$  es el conjunto de nodos de la red y  $E$  el conjunto de enlaces, que incluye el efecto de propagación multicamino. El peso de cada enlace del grafo modela el retardo de transmisión medido en ranuras de tiempo.

Las colisiones de mensajes son importantes sólo si se producen sobre uno de los nodos

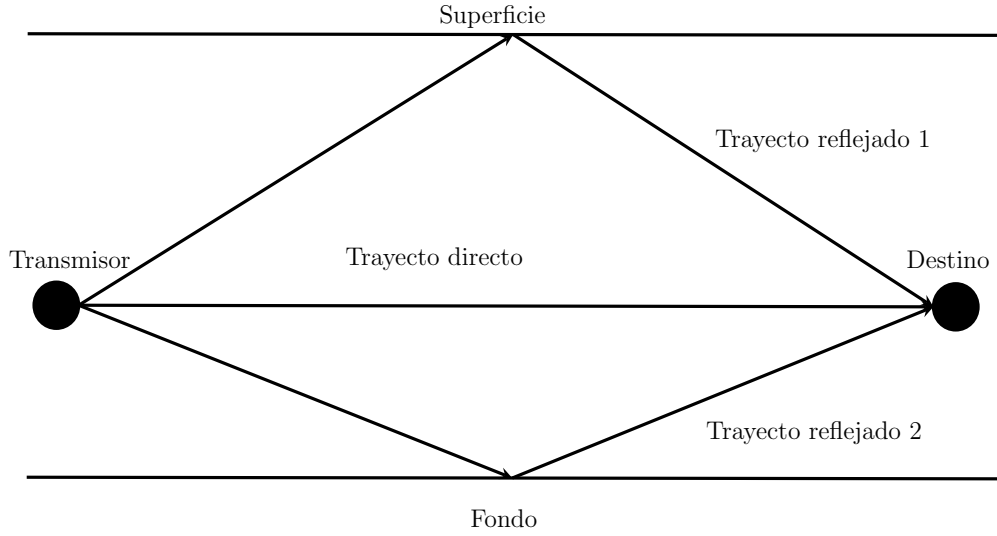


Figura 6.1: Ejemplo de propagación multicamino.

y como se enuncia en [47], existen cuatro situaciones diferentes que deben considerarse. El primer caso ocurre cuando dos o más mensajes llegan en simultáneo a un nodo y se denomina colisión Rx-Rx. El segundo caso se da ante la imposibilidad de un nodo de transmitir dos o más mensajes en simultáneo, llamado Tx-Tx. La tercera situación ocurre cuando un nodo intenta transmitir al mismo tiempo que se recibe un mensaje y este es el caso Tx-Rx. Finalmente, el cuarto caso se denomina Interferencia-Rx y, a diferencia del tipo Rx-Rx, se da cuando un nodo recibe un mensaje pero es interferido por otro mensaje cuyo destinatario no es el nodo donde se produce la colisión.

Debido a la propagación multicamino, las colisiones entre mensajes pueden producirse por cualquiera de las réplicas que viajan por el medio y por lo tanto, debe ser tenido en cuenta para la asignación de ranuras de la trama de comunicación. Cada vez que un nodo transmite un mensaje, todos los enlaces que vinculan dos de cualesquiera de los nodos del grafo  $G$  deben ser considerados.

### 6.3. Planificación y ruteo de mensajes

Se supone un conjunto de mensajes de tiempo real, es decir, que debe llegar a su destino antes de un vencimiento. Los mensajes se notan  $m_{ij}$  donde los índices  $i$  y  $j$  indican los nodos transmisor y receptor respectivamente. Cada mensaje tiene un periodo asociado  $P_{ij}$  y un vencimiento  $D_{ij}$ , ambos expresados en número de ranuras.

Cada nodo puede transmitir sólo en su ranura asignada dentro de la trama. Si hay más de un mensaje en espera para ser enviados, entonces se debe esperar el transcurso de varias tramas para ser transmitidos. En el caso de que ningún nodo tenga mensajes en espera, a un mensaje dado le tomará tantas ranuras llegar a su destino como la longitud de la trama multiplicada por la cantidad de saltos entre nodos. Esta es una condición necesaria del sistema para ser factible o planificable. Si se toma por ejemplo una red con la topología que se muestra en la Figura 6.2, donde por simplicidad no se contempla la propagación

multicamino y los enlaces tienen el mismo peso en ambas direcciones, la trama más compacta que se puede obtener es la que se presenta en la Tabla 6.1 y que se obtiene evaluando todas las combinaciones posibles. Como puede observarse, tiene una longitud de seis ranuras, con lo cual, para un mensaje  $m_{ae}$ , es necesario que transcurra al menos tres tramas o 18 ranuras en total hasta ser recibido por el nodo  $e$ .

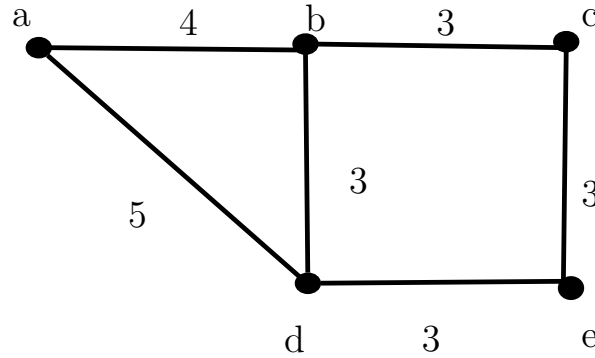


Figura 6.2: Una red de 5 nodos.

| Nodo | Ranuras de tiempo |         |         |         |         |         |
|------|-------------------|---------|---------|---------|---------|---------|
|      | 1                 | 2       | 3       | 4       | 5       | 6       |
| $a$  | $Tx(a)$           |         |         |         | $Rx(b)$ | $Rx(d)$ |
| $b$  | $Tx(b)$           |         |         | $Rx(d)$ | $Rx(a)$ | $Rx(c)$ |
| $c$  |                   |         | $Tx(c)$ | $Rx(b)$ | $Rx(e)$ |         |
| $d$  | $Tx(d)$           |         |         | $Rx(b)$ | $Rx(e)$ | $Rx(a)$ |
| $e$  |                   | $Tx(e)$ |         | $Rx(d)$ |         | $Rx(c)$ |

Cuadro 6.1: Asignación de nodos a ranuras de tiempo dentro de la trama de comunicación.

Las transmisiones subacuáticas son vulnerables a sufrir perturbaciones de distinto tipo y si se emplea una única ruta de menor distancia, como en el modelo de menor camino primero (SPF, por sus siglas en inglés), se incrementa el riesgo de perder un mensaje y tener que retransmitirlo. El modelo propuesto aprovecha el efecto de multicamino para incrementar la robustez de la decodificación de mensajes, ya que permite que si el mensaje original se pierde, éste puede ser recuperado a partir de la recepción de una réplica. Se incorpora también, un ruteo oportunista basado en un modelo epidémico.

En el ejemplo presentado en la Figura 6.2, si un nodo  $a$  debe enviar un mensaje a otro nodo  $e$ , mediante el modelo SPF, se selecciona la ruta  $a-d-e$ , que tiene un retardo de transmisión de 18 ranuras. Sin embargo, si el enlace  $a-d$  se rompe, el mensaje se pierde definitivamente. Por ello, al emplear un esquema de propagación epidémica, el mensaje puede llegar a destino recorriendo una ruta diferente. Este mecanismo requiere de mayor consumo de energía, pero el beneficio consiste en el incremento de la robustez del sistema de comunicación.

En la Figura 6.3 se muestra una red más compleja conformada por siete nodos, donde se incorporan además, enlaces duplicados para modelar el efecto de multicaminos. En este

segundo caso, la trama de comunicación óptima tiene 11 ranuras de duración y el orden de transmisión de cada nodo se ilustra en la tabla 6.2.

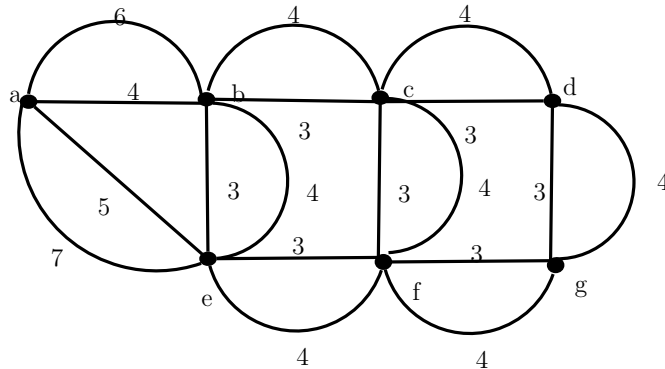


Figura 6.3: Red con siete nodos y consideración de multicamino para validar el modelo propuesto.

| Nodo     | Ranuras de tiempo |   |              |              |              |              |              |              |              |              |              |
|----------|-------------------|---|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
|          | 1                 | 2 | 3            | 4            | 5            | 6            | 7            | 8            | 9            | 10           | 11           |
| <i>a</i> |                   |   |              | <i>Tx(a)</i> | <i>Rx(b)</i> | <i>Rx(e)</i> | <i>Rx(b)</i> | <i>Rx(e)</i> |              |              |              |
| <i>b</i> | <i>Tx(b)</i>      |   |              | <i>Rx(e)</i> | <i>Rx(e)</i> | <i>Rx(c)</i> | <i>Rx(c)</i> | <i>Rx(a)</i> |              | <i>Rx(a)</i> |              |
| <i>c</i> |                   |   | <i>Tx(c)</i> | <i>Rx(b)</i> | <i>Rx(b)</i> | <i>Rx(f)</i> | <i>Rx(f)</i> |              |              |              |              |
| <i>d</i> |                   |   |              |              | <i>Tx(d)</i> | <i>Rx(c)</i> | <i>Rx(c)</i> | <i>Rx(g)</i> | <i>Rx(g)</i> |              |              |
| <i>e</i> | <i>Tx(e)</i>      |   |              | <i>Rx(b)</i> | <i>Rx(b)</i> | <i>Rx(f)</i> | <i>Rx(f)</i> |              | <i>Rx(a)</i> |              | <i>Rx(a)</i> |
| <i>f</i> |                   |   | <i>Tx(f)</i> | <i>Rx(e)</i> | <i>Rx(e)</i> | <i>Rx(c)</i> | <i>Rx(c)</i> | <i>Rx(g)</i> | <i>Rx(g)</i> |              |              |
| <i>g</i> |                   |   |              |              | <i>Tx(g)</i> | <i>Rx(f)</i> | <i>Rx(f)</i> | <i>Rx(d)</i> | <i>Rx(d)</i> |              |              |

Cuadro 6.2: Asignación de ranuras para el caso de la red de siete nodos.

## 6.4. Cálculo de la trama óptima

El cálculo de la trama óptima puede realizarse con técnicas de optimización lineal entera, ya que consiste en un problema combinatorio. Sin embargo, el costo computacional es muy elevado y el planteo de las condiciones del modelo es muy complejo. Se presenta como alternativa un algoritmo evolutivo que si bien no asegura hallar el óptimo absoluto, las comparativas contra el método ILP para una serie de problemas generados, arroja los mismos resultados en mucho menor tiempo de ejecución, aún siendo implementado con lenguajes interpretados.

Debido a que en la trama de comunicación debe transmitir un único nodo por vez, entonces la variable que define la solución es el orden en el que las ranuras de transmisión de cada nodo son asignadas. Por ejemplo, si el sistema tiene sólo tres nodos, entonces hay  $3! = 6$  posibles formas de ordenar la transmisión de cada uno y por lo tanto seis tramas



de comunicación posibles. Si la duración de la ranura es muy pequeña comparada con los retardos de transmisión, al método ILP le lleva mucho tiempo hallar la trama óptima dado que trabaja en un espacio de tantas dimensiones como ranuras de la trama y por lo tanto el costo computacional es muy elevado.

Partiendo de esta hipótesis se plantea la siguiente conjetura,

**Conjetura 1** *La trama de menor longitud puede hallarse mediante el Algoritmo 1, dado como entrada un arreglo de nodos ordenados adecuadamente.*

```

foreach  $n \in \text{Nodos}[N]$  do
     $h := 0$  ; // Número de ranura.
    while  $h$  ocupada do
         $h := h + 1$ 
    end
    asignar  $n$  a la ranura  $h$ 
    ocupar ranuras con recepciones de la transmisión
end

```

**Algoritmo 1:** Algoritmo para obtener la asignación de nodos a la trama.

El método para calcular la trama de menor longitud se reduce entonces a evaluar todas las posibles formas de ordenar la lista de nodos de la red. En los casos en los que la cantidad de nodos sea muy grande, entonces el tiempo que se demora en evaluar todas las combinaciones puede ser muy elevado y por lo tanto se requiere un método de búsqueda más eficiente.

Se plantea la búsqueda de la solución mediante Algoritmos Genéticos (GA), donde se codifica el ordenamiento del arreglo de nodos como cromosomas. En el método GA, la operación de entrecruzamiento tradicional puede dar como resultado arreglos con elementos repetidos, cosa que no puede ocurrir dado que la variable de optimización es un arreglo de elementos únicos. Para evitar esto se emplea el método de cruce para representación de orden (PMX, por sus siglas en inglés), que se presenta en [48]. Este método consiste en emplear un algoritmo reparador que elimina los elementos repetidos de la variable o cromosoma resultante del entrecruzamiento. De la misma manera, en la etapa de mutación, dentro del proceso iterativo del método GA, en lugar de alterar un gen, simplemente se permutan dos o más elementos dentro del arreglo del cromosoma.

El esquema general de optimización aplicado a este problema es muy similar al presentado en la Sección 5.2 del Capítulo 5, particularmente el descrito en la Figura 5.2, donde el bloque del algoritmo de planificación, en este caso contendría el Algoritmo 1 y la función objetivo simplemente retorna la longitud de la trama, ya que es el valor indicador de la calidad de la solución.

Como se explica en la Sección 6.2, la longitud de la trama define el periodo de comunicación de todo el sistema. Las últimas ranuras de cualquier trama calculada, están ocupadas por instantes de recepción en los nodos, sin embargo, es muy probable, que en algunos casos sea posible adelantar la transmisión en algunos nodos sin que se produzcan colisiones, es decir, es posible reducir el periodo de transmisión de cada nodo mediante el solapamiento de dos tramas consecutivas. En este punto se introduce la definición de *periodo efectivo*,

que representa el mínimo periodo de transmisión que se puede lograr para todos los nodos mediante el solapamiento de tramas consecutivas.

El solapamiento de las tramas no siempre es posible porque depende de cómo son asignadas las ranuras en el inicio y fin de la trama. En algunos casos puede darse que tramas más largas que la óptima, tienen un menor periodo efectivo si se admite el solapamiento. Para lograr esto, simplemente se modifica la función objetivo para que retorne el periodo efectivo calculando la cantidad de ranuras que es posible solapar entre tramas sin que haya colisiones de mensajes. De esta manera, el algoritmo de optimización busca soluciones en este nuevo contexto que logran redes con mayor *throughput*.

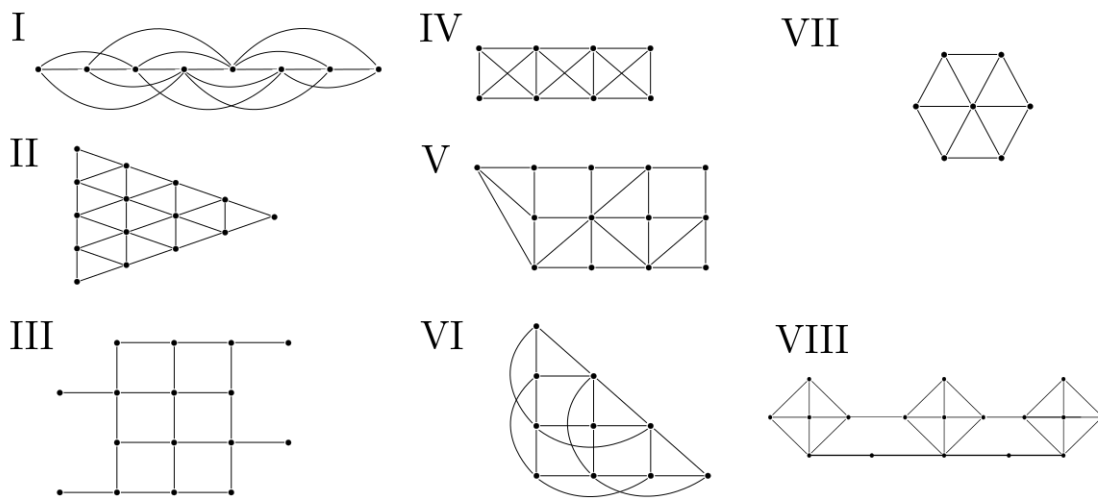


Figura 6.4: Distintas topologías evaluadas en las simulaciones.

Se generaron ocho situaciones de redes con distintas topologías con el fin de evaluar el desempeño de los algoritmos propuestos. La disposición de nodos y enlaces de cada red se muestran en la Figura 6.4. La Tabla 6.3 muestra la comparación de los resultados obtenidos mediante el método ILP, GA y GA con solapamiento de tramas, donde  $L$  indica la duración de la trama calculada y  $CT$  es el tiempo de ejecución o tiempo de cómputo, expresado en segundos. Como puede observarse, la eficiencia del método GA es mayor que la de ILP, dado que los resultados hallados, para los casos propuestos, son idénticos y el tiempo de ejecución es mucho menor.

## 6.5. Validación

Para evaluar el rendimiento del modelo propuesto en condiciones de tiempo real, se implementaron una serie de simulaciones mediante la librería SimPy [49], que consiste en un framework para simulaciones de eventos discretos orientado a procesos. Con esta herramienta se simulaban los enlaces acústicos entre los nodos de una red como la descrita en la Figura 6.3. Esta red tiene siete nodos y 18 enlaces incluyendo los caminos de reflexión. El valor de la trama mínima calculada es de 11 ranuras y la lista de mensajes se agrupa en el conjunto que se presenta en la ecuación 6.1.

| Instancia | ILP |          | GA |      | GA (Solap.) |      |
|-----------|-----|----------|----|------|-------------|------|
|           | L   | CT       | L  | CT   | L           | CT   |
| I         | 19  | 0.4      | 19 | 0.43 | 14          | 0.63 |
| II        | –   | $\infty$ | 12 | 0.93 | 10          | 2.90 |
| III       | 9   | 3.1      | 9  | 1.06 | 9           | 1.40 |
| IV        | 11  | 0.1      | 11 | 0.4  | 8           | 0.50 |
| V         | 13  | 38.9     | 13 | 0.43 | 10          | 1.00 |
| VI        | 14  | 2        | 14 | 0.58 | 10          | 0.70 |
| VII       | 9   | 12       | 9  | 2.47 | 9           | 1.08 |
| VIII      | 12  | 5        | 12 | 0.39 | 9           | 0.51 |

Cuadro 6.3: Análisis comparativo de métodos de optimización.

$$Z = \{m_{ag}(100, 300), m_{bd}(80, 200), m_{ce}(80, 200), m_{db}(80, 220), m_{ec}(80, 200), m_{ga}(100, 300)\} \quad (6.1)$$

Se evaluaron los siguientes parámetros:

1. Retardo de extremo a extremo.
2. Relación de recepción de paquetes, es decir, la cantidad de paquetes recibidos comparado con el número de enviados.
3. Rendimiento de la red a nivel de aplicación.
4. Número de mensajes en espera para ser retransmitidos.

Se simuló el sistema durante 30.000 ranuras, lo que equivale a la duración de unas 300 tramas. Por simplicidad, se muestran en la Figura 6.5 los retardos de las primeras 2.500 ranuras ya que ilustra los resultados más relevantes. El comportamiento de la simulación es coherente con lo esperado. Se puede observar por ejemplo que el retardo promedio para un mensaje  $m_{ga}$  es de 28 ranuras, mientras que para un mensaje  $m_{bd}$  es de 18 ranuras. La amplitud de oscilación de los retardos es de 11 ranuras, que coincide con la duración de la trama.

Para analizar el impacto de la duración de la trama de comunicación en los retardos de transmisión del sistema, se ejecutó una simulación empleando tramas de 11 a 14 ranuras de duración. La Figura 6.6 muestra la función de distribución acumulada empírica de los retardos de extremo a extremo en los distintos casos. Como se observa, la duración de la trama afecta considerablemente en los retardos de la red, principalmente en los casos donde la trama es de 13 y 14 ranuras.

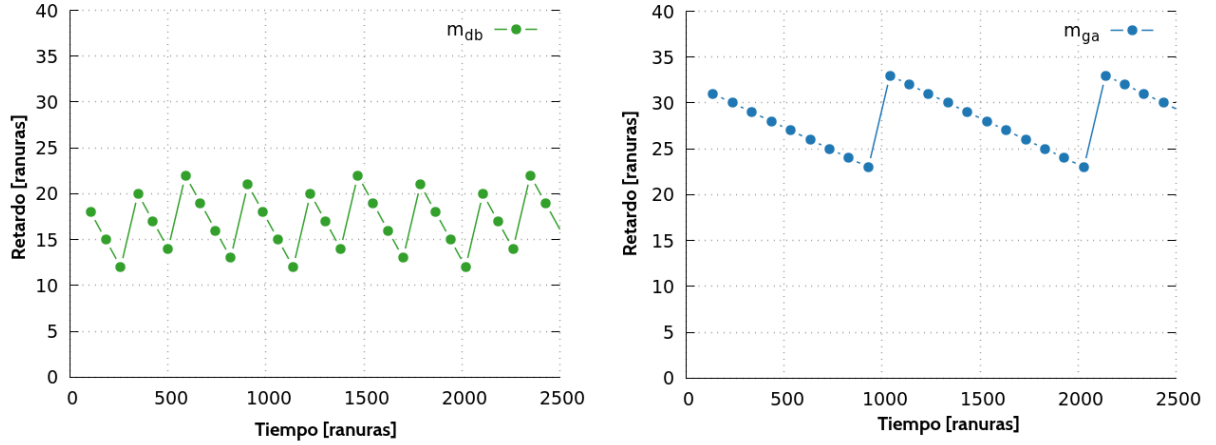


Figura 6.5: Retardos de extremo a extremo. Izquierda: nodo  $d$  a nodo  $b$ . Derecha: nodo  $g$  a nodo  $a$ .

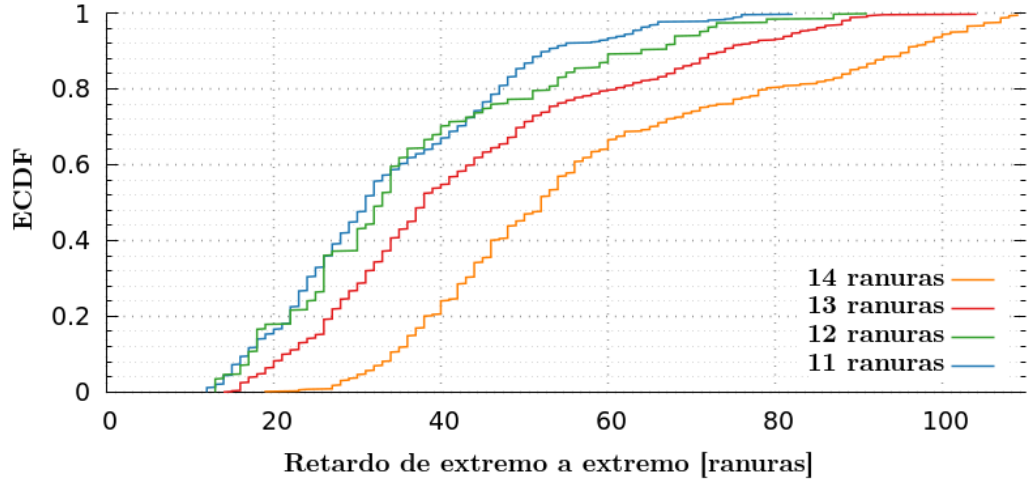


Figura 6.6: Retardos de extremo a extremo empleando tramas de distinta duración.

## 6.6. Consideraciones finales

El uso de redes submarinas acústicas se encuentran aún en desarrollo y lejos de lograrse un consenso en aspectos básicos como frecuencia de portadora o técnicas de modulación. Una de las limitaciones del desarrollo de este tipo de redes ha sido la falta de modelos para representar el rendimiento de la red en distintos escenarios de aplicación, particularmente en aquellos que requieren condiciones de tiempo real.

Los modelos teóricos desarrollados en este capítulo permiten analizar el comportamiento de una red de tiempo real en condiciones de comunicación muy restrictivas. El modelo fue validado mediante un simulador de eventos discretos que permitió acotar los retardos de propagación de mensajes en función de distintas situaciones hipotéticas de operación que pueden presentarse.

Continuando el tema de investigación que inicia en el capítulo anterior, se extendieron los

métodos de optimización para el caso de asignación de nodos de una red a ranuras dentro de una trama de comunicación TDMA. Esto demuestra la flexibilidad de los métodos heurísticos para resolver problemas de distinta naturaleza y con diferentes aplicaciones, obteniéndose en todos los casos, soluciones comparables frente a métodos determinísticos como ILP.

En el capítulo siguiente se aplican una vez más las heurísticas basadas en GA y PSO, también para resolver problemáticas relacionadas con las redes de comunicación y la gestión de recursos, pero en este caso, aplicadas a la distribución espacial de los nodos.

## Capítulo 7

# Redes de comunicación con topología configurable

El despliegue de redes de comunicación alternativas basadas en nodos móviles es un aspecto muy importante para lograr la comunicación en casos en los que no es posible contar con infraestructura establecida. En los últimos años, las redes aéreas ad-hoc (FANETs, por sus siglas en inglés) están cobrando relevancia en el ámbito de la producción agrícola, dada la incorporación de drones o UAVs para distintos propósitos. El costo de esta tecnología es cada vez menor y sus prestaciones permiten todo tipo de tareas. Entre las aplicaciones de las FANETs en actividades agrícolas se puede mencionar la exploración de cultivos, el monitoreo de hacienda e infraestructura, la aplicación de productos químicos, el relevamiento y mapeo de cultivos, el trazado y monitoreo de geocercas, el análisis de seguros agrícolas y la planificación y gestión de cultivos [50].

En general, las compañías de telecomunicaciones brindan servicios de conectividad en donde es posible tener clientes y sea posible amortizar la infraestructura. El problema de cobertura es por lo tanto uno tradicional en teoría de optimización y que resulta necesario volver a analizar y desarrollar en el marco de las nuevas tecnologías. En este capítulo se analizan algoritmos de aproximación para brindar cobertura mediante la ubicación de nodos móviles. Los resultados fueron validados en procesos de emergencia, donde la incorporación de esta tecnología resulta en un importante beneficio para las actividades de rescate. Sin embargo los resultados obtenidos son aplicables a la distribución de nodos en espacios abiertos para la disposición de redes de sensores aplicadas a la agricultura de precisión.

Mientras que han ocurrido importantes avances en las tecnologías de comunicación, particularmente de las inalámbricas, la provisión de soporte de comunicación para las actividades de socorro durante eventos de desastre natural sigue siendo un tema abierto. La literatura sobre investigación de casos de desastre enuncia una serie de restricciones que se deben cumplir para conducir actividades de rescate en zonas urbanas, dadas las limitaciones de las redes de telefonía y sistemas de radio para proveer comunicación digital en el espacio de trabajo de equipos de auxilio. Durante operaciones de búsqueda y rescate, los requisitos de comunicación son más estrictos, ya que los equipos de rescate deben depender de un sistema de comunicación confiable y de tiempo real para realizar sus actividades y coordinar esfuerzos entre varios equipos.

Situaciones de emergencia no sólo se producen en zonas urbanas. Por mencionar un

ejemplo de escenario similar que se ha dado en sectores rurales del SO bonaerense, se puede considerar el caso de los incendios en zonas de bosque como los de la región del sistema serrano de Ventania, que en muchos casos termina afectando una amplia zona agroprodutiva e incluso ha sido necesario alertar o evacuar algunas viviendas comprometidas por el avance del fuego. En este tipo de escenario de emergencia, disponer de un soporte de comunicación de tiempo real aporta un gran beneficio [51].

En este trabajo se sugiere que las redes de despliegue aéreo ad-hoc pueden proveer soporte de comunicación durante tales escenarios de desastre natural o producto de un descuido humano. La propuesta involucra el uso de las denominadas “unidades testigo” implementadas con drones que actúan como *gateways* para los equipos de rescate trabajando en distintas ubicaciones dentro del área afectada. La propuesta es denominada red de comunicación de tiempo real de despliegue aéreo y se presenta en este capítulo su factibilidad para proveer comunicación de tiempo real, por un lado, mediante el análisis de planificabilidad de entrega de mensajes, y por el otro, mediante simulaciones del sistema de soporte de comunicación en un entorno físico inspirado por incidentes reales. Los resultados obtenidos son muy positivos y consistentes, por lo tanto, esta propuesta representa también un paso adelante hacia la solución de este problema abierto.

## 7.1. Modelo del sistema

La red de comunicación está compuesta por dispositivos retransmisores capaces de brindar conexión desde un punto de acceso (AP, por sus siglas en inglés) hasta otra zona alejada. Estos dispositivos están montados en UAVs con autonomía suficiente para desplazarse dentro de un área denominada zona de trabajo y configurar la red automáticamente.

El principal inconveniente que se presenta al momento de realizar el despliegue de la red, es la irregularidad del terreno, debido a que los nodos deben estar ubicados en zonas suficientemente altas para tener la mayor cobertura y de este modo minimizar la cantidad de unidades requeridas.

Se considera un conjunto de  $N_f$  nodos con posiciones fijas dentro de la zona de trabajo. Estos nodos representan las posiciones de los actores que requieren de la infraestructura de comunicación. Por otro lado, se cuenta con un conjunto de  $N_m$  nodos móviles que modelan los UAVs repetidores de señal.

Formalmente, se define al arreglo  $X$  que contiene las coordenadas de cada uno de los  $N_f + N_m$  nodos que componen la red. Sea  $Z$  una estructura que modela el mapa de elevación de la zona de trabajo. Debido a que  $Z$  es un arreglo discreto, se puede ampliar su resolución empleando interpolación bilineal para determinar la altura del terreno en cualquier punto. Se supone que cada nodo tiene un alcance de comunicación definido por  $P_i$ , con  $i \in (1, \dots, N_f + N_m)$  que modela el rango máximo de recepción de señal inalámbrica. Se asume que si un par de nodos  $i$  y  $j$  se encuentran separados a mayor distancia que  $\min(P_i, P_j)$ , entonces no pueden comunicarse y requieren de una o más unidades repetidoras de señal. Además, los nodos  $i$  y  $j$  deben encontrarse en línea de vista (LOS, por sus siglas en inglés), no interceptado por el terreno. Entonces dada la distribución espacial de nodos en  $X$ , para determinar la topología de la red, se emplea la siguiente condición,

$$(\text{dist}(X_i, X_j) < \min(P_i, P_j)) \wedge \text{LOS}(X_i, X_j, Z), \quad (7.1)$$

para saber si puede haber un enlace entre los dos nodos  $i$  y  $j$ , donde la función  $\text{dist}(v, w)$  retorna la distancia Euclídea entre las posiciones  $v$  y  $w$  y la función  $\text{LOS}(v, w, z)$  determina si existe línea de visión entre las coordenadas  $v$  y  $w$  dado el mapa de elevación  $z$ . Si dos nodos de la red cumplen la condición 7.1, entonces se crea un enlace nuevo. Cada enlace entre dos nodos se enumera con  $\text{path}_i$  con  $i \in 1, \dots, N_p$  donde  $N_p$  es el número total de enlaces.

En este punto se define la siguiente función objetivo a minimizar:

$$f(X) = \alpha \cdot \sum_{i=1}^{N_p} \text{longitud}(\text{path}_i) + \beta \cdot N_m, \quad (7.2)$$

donde  $\alpha$  y  $\beta$  permiten regular la importancia de lograr una red con mayor velocidad reduciendo la longitud de los enlaces o una con menor número de nodos. No necesariamente una condición limita a la otra. La Figura 7.1 ilustra dos topologías distintas para conectar los puntos fijos **S1**, **S2** y **S3** con **IC** empleando nodos repetidores móviles cuando se varían los parámetros de ajuste  $\alpha$  y  $\beta$ .

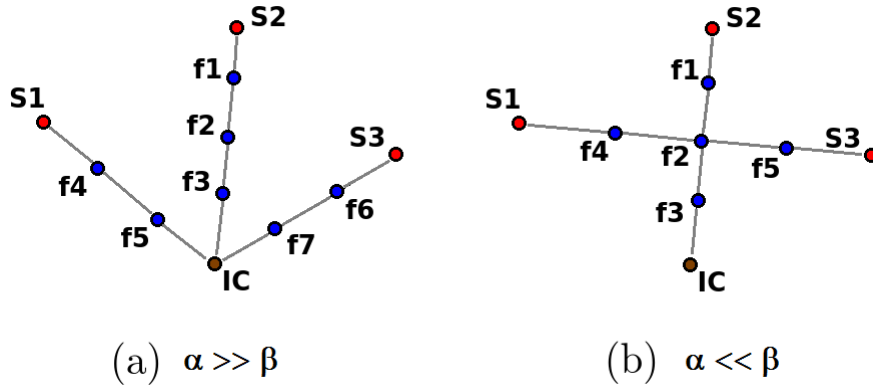


Figura 7.1: Dos topologías de red con (a) más nodos para mayor rendimiento y (b) menos nodos pero menor rendimiento.

### 7.1.1. Topología adaptativa basada en ahorro de energía

El ahorro de energía es también una cuestión clave, ya que determina el tiempo efectivo durante el cual se puede contar con la red de comunicación por tiempo ininterrumpido.

Por este motivo se propone un modelo de red dinámica en la cual la topología, una vez desplegada y funcionando, se adapta mediante pequeñas variaciones en función de reducir la longitud de algunos enlaces de mayor carga con el fin de disminuir la distancia y por lo tanto el consumo de energía, reduciendo la potencia de la señal. Este es un enfoque válido si se trata de una red de muchos nodos en los que existe un desbalance asimétrico en la distribución de la carga de la red. El modelo propuesto contempla las ecuaciones 7.4 y 7.3.



$$\vec{v}_i = b \cdot \vec{v}_i - \sum_{j \in \text{links}} k_j (\vec{x}_i - \vec{x}_j), \quad (7.3)$$

$$\vec{x}_i = \vec{x}_i + \vec{v}_i, \quad (7.4)$$

donde  $\vec{x}_i$  es la posición del nodo  $i$ -ésimo de la red y  $\vec{v}_i$  modela la fuerza atractiva que actúa sobre cada nodo y apunta hacia los nodos vecinos. El valor de las constantes  $k_j$  debe ser calculada de acuerdo a la tasa de transmisión de datos por el enlace  $j$ -ésimo. El parámetro  $b$  permite ajustar el tiempo de establecimiento con el fin de estabilizar las posiciones de los nodos lo más rápido posible.

La Figura 7.2 ilustra una situación en la que luego del despliegue inicial de la red, una alta carga en los enlaces  $d1$  y  $d2$  hace que los nodos móviles  $f1$  y  $f2$  se desplacen para reducir las distancias  $d1$  y  $d2$  de manera que se pueda mantener la transmisión de datos empleando menor potencia.

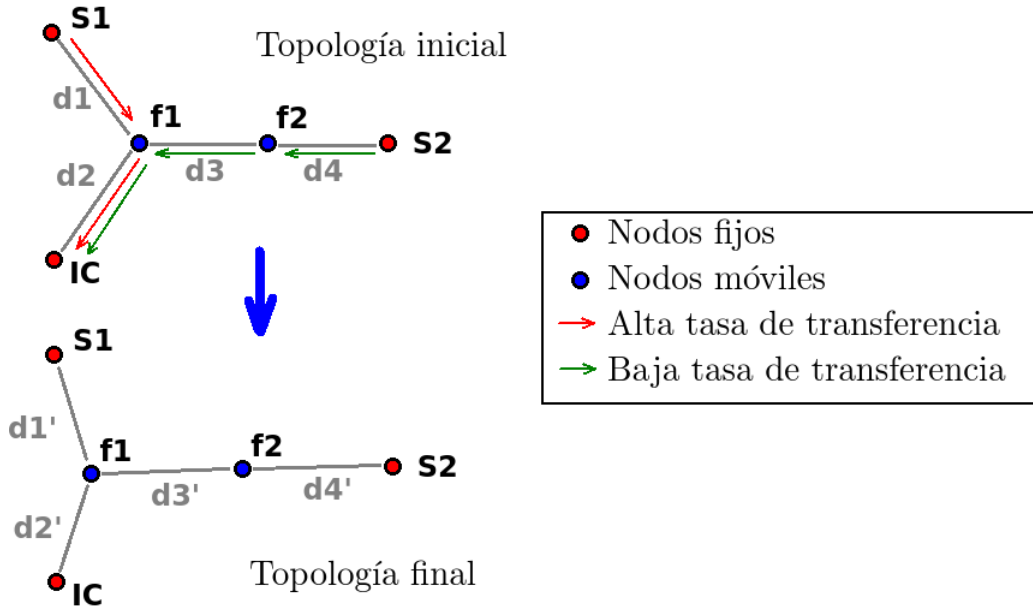


Figura 7.2: Dinámica del modelo de topología adaptativa.

## 7.2. Evaluación de rendimiento

Con el fin de evaluar el rendimiento del algoritmo propuesto para el despliegue de la red, se supone un terreno cuyo mapa de elevación se ilustra en la Figura 7.3. La topología de este terreno corresponde a una zona en las cercanías de la ciudad de Río de Janeiro, en Brasil, dado que es una región poblada que suele verse afectada por aludes, como ocurrió en 2011 [52]. Como parte inicial del procedimiento de respuesta, es necesario desplegar una red que permita a los equipos de búsqueda y rescate (SAR, por sus siglas en inglés), comunicarse con el comando de incidentes (IC, por sus siglas en inglés).

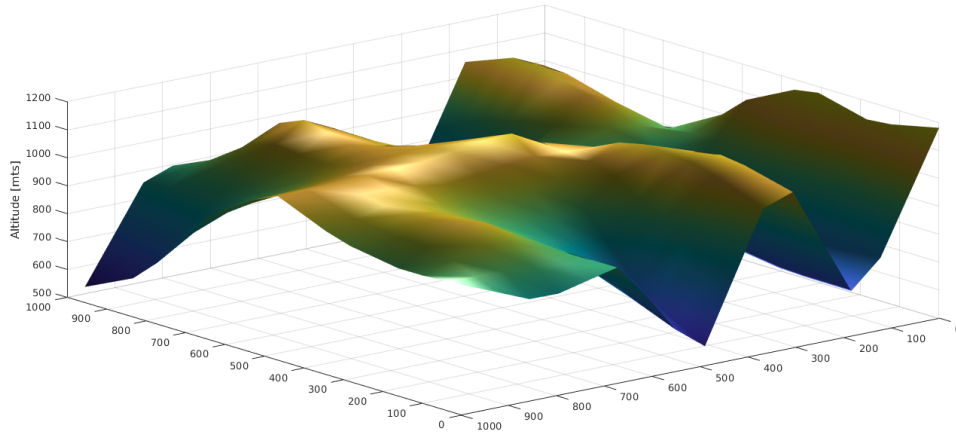


Figura 7.3: Mapa de elevación del terreno considerado para el ejemplo.

Inicialmente se definen las posiciones de tres grupos SAR y la posición del IC en ubicaciones fijas. Se supone que el radio de comunicación de cada nodo tiene un alcance de aproximadamente el 20 % del área total de despliegue, lo cual significa alrededor de unos 200 metros. Los parámetros de ajuste fueron definidos con los valores  $\alpha = \beta = 1$  y se aplicaron los métodos GA y PSO. Estos métodos requieren de la inicialización de una población inicial a partir de la cual cada algoritmo evoluciona mejorando la solución constantemente.

A partir de esta información, se ejecuta la simulación para calcular el mínimo número de nodos requeridos y sus posiciones. Estos resultados se ilustran en la Figura 7.4. Para el método GA, el tiempo de ejecución fue limitado a 3000 segundos durante el cual se iteró durante 173 generaciones y logrando una solución de nueve nodos móviles. Para el caso de PSO, en condiciones similares se logró una solución óptima de 11 nodos.

En las dos soluciones se observa que los despliegues calculados tienen la particularidad de rodear las áreas con pendientes, ya que en esas regiones, se debe emplear una mayor cantidad de nodos porque se dificulta mantener la línea de visión a lo largo de mayores distancias.

### 7.3. Consideraciones finales

En este capítulo se propuso un modelo de despliegue de una red inalámbrica compuesta por nodos aerotransportados, donde se analizan los aspectos referidos a la minimización del número de nodos de la red y al cálculo de una topología que presente un balance óptimo entre rendimiento y consumo de energía.

En la bibliografía existen numerosos enfoques para resolver el problema de optimizar el despliegue de una red de comunicación sobre un terreno con irregularidades que obstaculizan la línea de visión entre antenas. Sin embargo, en pocos casos se toma en cuenta la influencia de la topología en el rendimiento de la red y el consumo de energía de los nodos.

El modelo del problema que se presenta también puede extenderse para resolver diversas problemáticas vinculadas al trazado de un enlace para brindar servicios de comunicación en zona rural, lo cual es, hoy en día, un tema relevante dado el constante crecimiento de la



# Parte IV

## Procesamiento de datos

# Capítulo 8

## Análisis predictivo

### 8.1. Motivación

El agua y su dinámica en el suelo resulta de suma importancia ya que es la variable de mayor influencia en el desarrollo de las especies vegetales y otros organismos que contribuyen a su crecimiento, como así también afecta directamente en el rendimiento en cultivos de producción. Por ello, la determinación de la humedad natural del suelo es primordial para resolver problemas vinculados a las necesidades de agua de riego, consideraciones ambientales y determinación de los excedentes hídricos [53].

El cultivo de plantas bajo techo se realiza principalmente con la intención de crear las condiciones óptimas para el desarrollo de la especie y aumentar la productividad de la superficie de explotación. Para mantener la humedad del suelo o el sustrato sobre el cual se desarrollan las plantas se puede recurrir a sistemas automáticos que en general permiten configurar el caudal y momento de activación, aunque algunos sistemas más sofisticados emplean sensores de humedad del suelo y se activan sólo cuando es necesario [54].

Existen diversos métodos para la medición en vivo de la humedad del suelo y todos requieren de instrumental de mayor o menor costo según las condiciones y características del suelo, precisión y exactitud del sensor, vida útil, etc. [55].

Una de las técnicas más sencillas consiste en medir la conductividad eléctrica del suelo, es decir, la corriente que circula entre dos terminales metálicos enterrados que están a diferente potencial eléctrico. Este método es simple desde el punto de vista de la implementación, no requiere de instrumental costoso pero tiene la desventaja de ser impreciso y de tener poca vida útil, ya que el proceso de electrólisis al que se someten las sondas de medición produce desgaste de las mismas. Por este motivo resulta de importancia reducir lo más posible la frecuencia de medición de la humedad.

Si se conoce cómo afecta cada una de las variables meteorológicas a la variación de la humedad del suelo, sería posible predecir la evolución de esta variable durante períodos de tiempo suficientes de manera de ampliar el periodo de muestreo y prolongar la vida útil de las sondas de medición de conductividad.

En este capítulo se detalla un simple análisis realizado a una serie de datos recolectada durante un periodo de unos 25 días para obtener un modelo predictor. Se aplican conceptos de aprendizaje de máquina realizando el ajuste de un modelo lineal mediante optimización.

## 8.2. Objetivos

El objetivo general es calcular la influencia de las variables atmosféricas sobre la variación del porcentaje de humedad de suelo, con el fin de determinar el volumen de riego necesario para satisfacer los requerimientos hídricos de una planta y reduciendo lo más posible la frecuencia de medición directa de la humedad. Esto último, con el fin de reducir el desgaste de las sondas eléctricas. Se seguirá el procedimiento descrito a continuación:

- Realizar un seguimiento constante y automático de las variables atmosféricas más influyentes en la dinámica del agua dentro del sistema durante un periodo suficiente que garantice la obtención de un modelo de precisión aceptable.
- Obtener un modelo de la dinámica del agua del sistema aplicando regresión sobre los datos obtenidos.
- Validar el modelo calculado comparando los resultados de las simulaciones con datos reales y que no hayan sido incluidos en la regresión.

## 8.3. Materiales y Métodos

Para el desarrollo se utiliza como caso de estudio, una planta de interior y, aunque la especie a utilizar no resulta de interés productivo, los métodos empleados pueden extenderse a otras variedades o sistemas de producción como invernaderos, sistemas hidropónicos, entre otros. Si bien no será posible aplicar el modelo calculado en este trabajo a otros casos, las técnicas de adquisición y análisis de los datos y los métodos de modelado pueden replicarse a distintas situaciones que requieran conocer la dinámica del agua en un sistema suelo-planta-atmósfera.

### 8.3.1. Esquema de medición

Se dispone de dos recipientes idénticos con aproximadamente 1 Kg. de suelo cada uno. Uno de ellos tiene plantado un Potus Lemon (*Scindapsus neon*) con varios meses de desarrollo. Para medir humedad del suelo de cada maceta se emplean dos sensores compuestos por amplificadores YL-38 y sondas resistivas YL-69 que se conectan a un microcontrolador ATmega328 montado en una plaqueta de desarrollo Arduino Nano. Además, se cuenta con un sensor de humedad y temperatura ambiente DHT-22 y un sensor de temperatura y presión atmosférica BMP180. Para registrar las mediciones se utiliza una computadora personal tipo netbook programada para activarse cada media hora, solicitar el muestreo de datos al microcontrolador, guardar los mismos en un archivo de texto, sincronizar este archivo con un servicio de almacenamiento en la nube y volver a suspenderse hasta la siguiente muestra. Los riegos se realizaron mediante dosis de 30 c.c. de agua por vez teniendo el cuidado de que no haya escurrimiento a través de los orificios inferiores de las macetas, es decir, manteniendo el valor de humedad por debajo de la capacidad de campo del suelo y por encima del punto de marchitez permanente.



Figura 8.1: Arreglo del instrumental de adquisición de datos utilizado. (a) sensores de humedad de suelo, (b) sensores de humedad, temperatura y presión atmosférica, (c) microcontrolador, (d) computadora personal.

### 8.3.2. Análisis de los datos

Se registraron datos a razón de tres muestras, una cada cinco segundos y seguido de un período de 30 minutos, adquiriendo un total de 1200 muestras entre los días 3 y 29 de noviembre de 2017.

Cada muestra del *dataset* se compone de una tupla que contiene fecha y hora de medición y los valores de temperaturas, humedad ambiente, presión atmosférica y humedades de los suelos en ambas macetas. En primer lugar se promedian las muestras tomadas de a tres que corresponden al intervalo de 10 segundos que dura la toma tres datos seguidos. Los valores de temperatura que son obtenidos con distintos sensores también se promedian entre sí.

La variable de mayor interés del sistema es la razón de cambio de la humedad del suelo. Debido a que los valores medidos de esta variable tienen mucha variación entre muestras, se aplica un filtrado por método de Savitzky-Golay [56] para suavizar la curva de datos. Se emplea un filtro de orden 5 y una ventana de regresión de 13 muestras.

A continuación se calcula la derivada numérica mediante diferencias hacia adelante de los valores de humedad de suelo correspondiente al recipiente que no contiene la planta. El gráfico de la Fig. 8.3 muestra la relación entre la razón de cambio de la humedad de suelo y las demás variables del sistema.

Puede observarse que la mayor influencia sobre la tasa de cambio de la humedad del suelo corresponde al valor instantáneo de esta variable. Matemáticamente, esta relación puede expresarse como

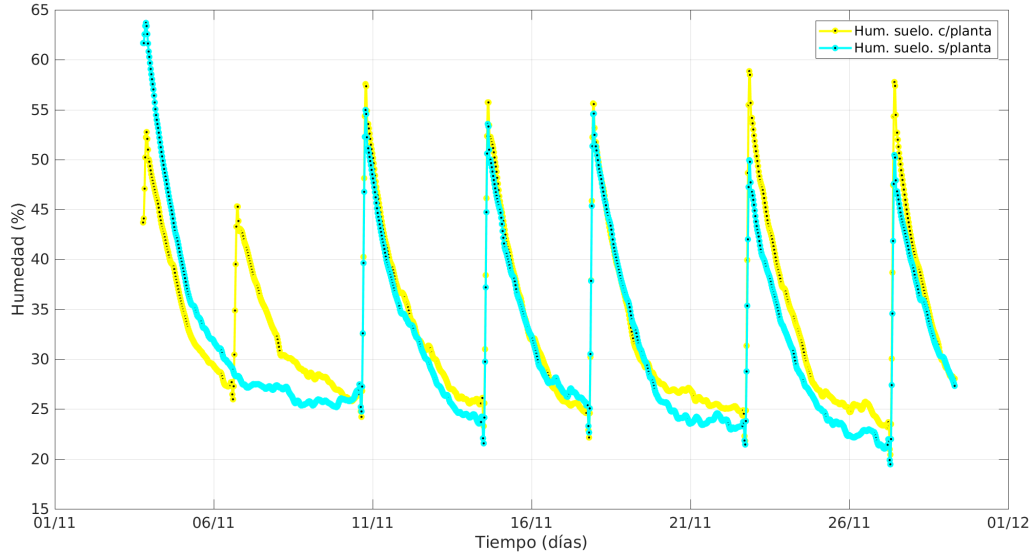


Figura 8.2: Gráfico de la humedad de los suelos medidas durante el periodo de muestreo.

$$\frac{dH_s}{dt} = w_1 + w_2 \cdot H_s, \quad (8.1)$$

donde  $H_s$  es el valor de la humedad del suelo y  $w_1$  y  $w_2$  son constantes a determinar. El resultado de esta ecuación diferencial es una relación exponencial de  $H_s$  con el tiempo, lo cual es coherente con lo que se observa en la Fig. 8.2.

### 8.3.3. Modelo del sistema

Para empezar se puede suponer que las demás variables influyen de manera lineal e independiente sobre la variación de  $H_s$ , entonces calculando una regresión lineal sobre los datos de la Fig. 8.3 puede estimarse un modelo del sistema según,

$$\frac{dH_s}{dt} = (w_{1a} + w_2 \cdot T_a) + (w_{1b} + w_3 \cdot P_a) + (w_{1c} + w_4 \cdot H_a) + (w_{1d} + w_5 \cdot H_s), \quad (8.2)$$

donde  $T_a$ ,  $H_a$  y  $P_a$  es la temperatura, humedad y presión atmosférica del ambiente y  $w_i$  son valores constantes reales. Agrupando los términos independientes se obtiene

$$\frac{dH_s}{dt} = w_1 + w_2 \cdot T_a + w_3 \cdot P_a + w_4 \cdot H_a + w_5 \cdot H_s, \quad (8.3)$$

donde

$$w_1 = w_{1a} + w_{1b} + w_{1c} + w_{1d}. \quad (8.4)$$

Este modelo se asume válido para la evapotranspiración de agua del suelo con o sin planta pero los parámetros son distintos en cada caso. Luego, obteniendo la diferencia entre ambas ecuaciones puede extraerse la contribución de la planta y su influencia en el flujo de agua dentro del sistema.



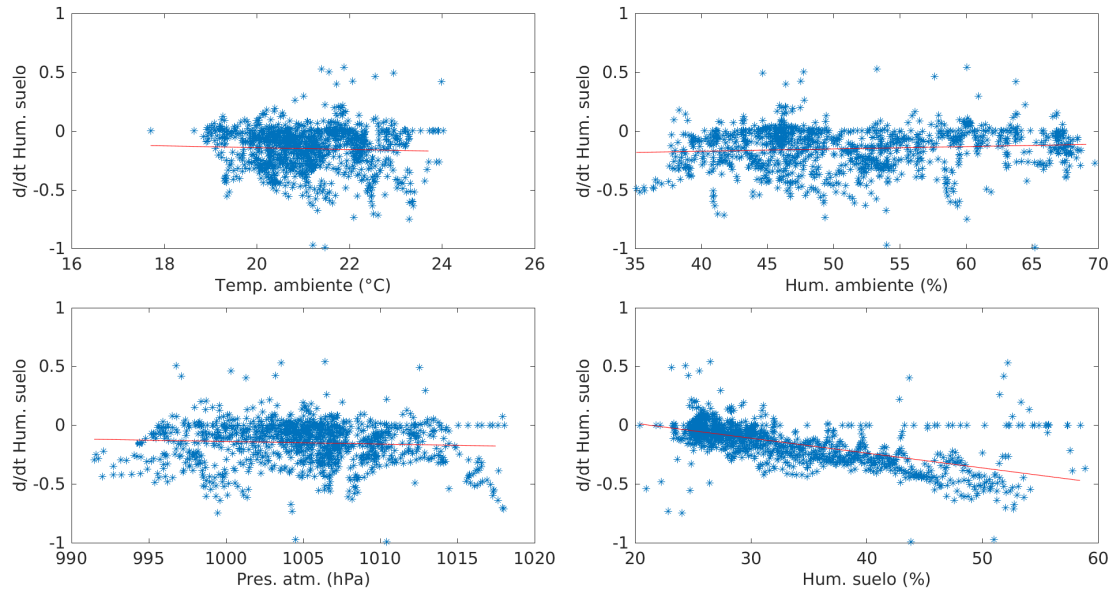


Figura 8.3: Relación entre la derivada de  $H_s$  (humedad del suelo) y las demás variables del sistema.

### 8.3.4. Simulación

Aplicando regresión lineal sobre los datos de la Fig. 8.3 y simulando el sistema mediante integración numérica de la ecuación 8.3 se obtiene una primera predicción del sistema. En las discontinuidades debidas a los instantes de riego se realiza una actualización directa de los valores de humedad de suelo, de manera que el sistema calcula la evolución de esta variable sólo durante el proceso de evapotranspiración del agua. Sin embargo, este modelo diverge en gran magnitud respecto de los valores medidos, con lo cual queda descartado.

Para remediar esta situación sin hacer cambios a la ecuación 8.3, se debe buscar calcular los parámetros  $w_1$  a  $w_5$  que minimicen el error de la comparación de los resultados simulados vs los valores medidos.

Para esto se construye una función objetivo que integra numéricamente la ecuación 8.3 mediante el método de Euler [57] y calcula el error cuadrático medio de la diferencia de los valores calculados y medidos. El esquema utilizado puede visualizarse a partir del diagrama de bloques de la Fig. 8.4.

La función objetivo es minimizada mediante el método de Nelder-Mead [58] provisto por la función `fminsearch` de Octave. La comparación entre valores simulados y medidos arroja el resultado que se muestra en la Fig. 8.5.

En este caso la simulación se comporta de manera muy cercana al sistema real, con lo cual puede adoptarse este modelo como una aproximación aceptable. La resolución de la ecuación diferencial 8.3, que por comodidad se repite más abajo, da como resultado una expresión muy compleja para analizar en dominio temporal, como muestra la ecuación 8.5.

$$\frac{dH_s}{dt} = w_1 + w_2 \cdot T_a + w_3 \cdot P_a + w_4 \cdot H_a + w_5 \cdot H_s,$$

$$\Rightarrow H_s(t) = w_1 \cdot \frac{e^{w_5 \cdot t} - 1}{w_5} + w_2 \cdot \int_0^t e^{w_5(t-u)} T_a(u) du + \dots$$

$$\dots + w_3 \cdot \int_0^t e^{w_5(t-u)} P_a(u) du + w_4 \cdot \int_0^t e^{w_5(t-u)} H_a(u) du, \quad (8.5)$$

donde  $t$  se expresa en horas a partir del instante en que se efectúa el riego.

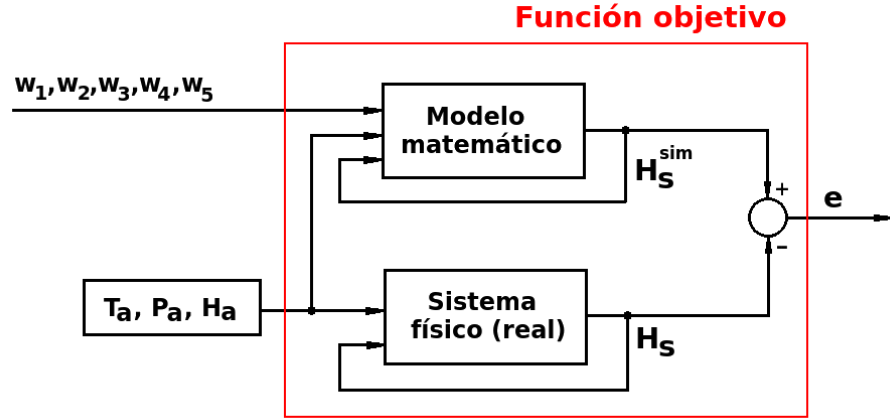


Figura 8.4: Modelo de la función objetivo utilizada.

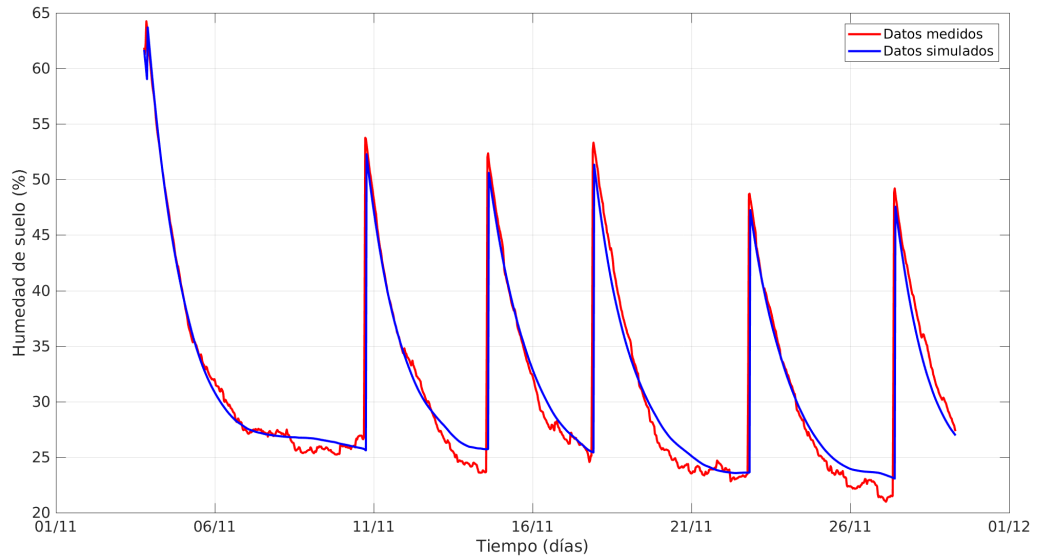


Figura 8.5: Comparativa de la simulación del modelo calculado mediante optimización vs. los datos medidos.

### 8.3.5. Análisis de los resultados

Se realizó el ajuste del modelo empleando distinto porcentaje de la totalidad de los datos, siempre comparando los resultados de la simulación con los valores reales durante todo el tiempo que duró la medición. En general se obtienen buenos resultados si el porcentaje de datos de optimización es superior al 75 %.

Los parámetros  $w_1$  a  $w_5$ , para el caso del suelo sin planta, que mejor aproximan el sistema real, utilizando un 80 % de los datos medidos, son:

$$w_1 = 1,4706, w_2 = -0,0017, w_3 = -0,0012, w_4 = 0,0030, w_5 = -0,0179. \quad (8.6)$$

En cambio, para el modelo que contiene la planta, los parámetros son:

$$w_1 = 2,3117, w_2 = -0,0089, w_3 = -0,0018, w_4 = 0,0026, w_5 = -0,0195. \quad (8.7)$$

Si se asume que ambos recipientes con tierra que simulan el suelo, tanto el que contiene vegetación como el que no, se comportan de manera idéntica, entonces es posible hallar el modelo de la influencia de la planta calculando la diferencia entre ambas ecuaciones, donde se obtiene un modelo con los siguientes parámetros:

$$w_1 = 0,8410, w_2 = -0,0072, w_3 = -0,0006, w_4 = -0,0004, w_5 = -0,0016. \quad (8.8)$$

Esto permite diferenciar el flujo de agua que se evapora o transpira directamente al ambiente y el flujo de agua que es absorbido por la planta. La Fig. 8.6 ilustra el modelo Vensim en el que es posible simular el sistema y evaluar cómo es cada flujo de agua en distintas situaciones.

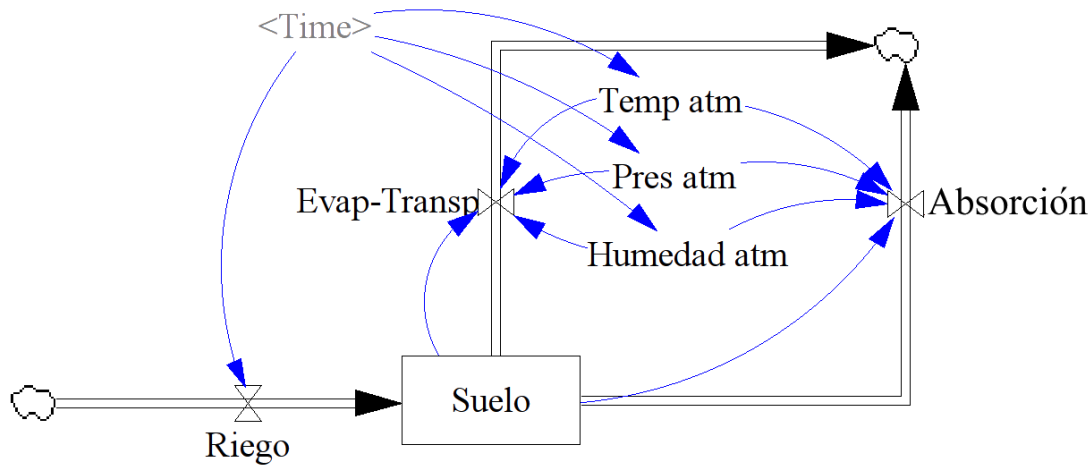


Figura 8.6: Modelo Vensim del sistema.

En este modelo pueden diferenciarse tres flujos que representan el movimiento del agua dentro del sistema y una variable de nivel que caracteriza el porcentaje de humedad del suelo. Se tiene una entrada de agua por riego que se especifica mediante una función dependiente del tiempo y suele ser una función impulsiva, es decir, que adopta valores altos durante

periodos de simulación cortos. El flujo “Evap-Transp” representa la humedad que se evapora o transpira desde o hacia el ambiente y depende tanto de la humedad del suelo como de las condiciones ambientales. De la misma manera, el flujo de agua “Absorción” representa el agua que es absorbida por la planta. Los flujos “Evap-Transp” y “Absorción” se modelan mediante la ecuación 8.3 y los parámetros 8.6 y 8.8 respectivamente.

## 8.4. Consideraciones finales

Mediante ajuste por regresión lineal de los datos medidos no fue posible calcular un modelo predictor que se comporte de manera aceptable, pero ajustando los parámetros del modelo comparando la respuesta temporal del mismo con los datos reales, se obtiene una mejora considerable. El método de modelado aplicado es un caso particular de aprendizaje de máquina pero en el que se empleó un esquema más simple y con menor cantidad de parámetros.

La principal observación es el buen comportamiento del modelo dada la simplicidad del mismo, ya que empleando sólo términos lineales es posible simular el proceso de secado del volumen de suelo contenido en los recipientes durante varios días.

Debido al poco tiempo de medición es posible que el modelo calculado no se comporte adecuadamente en casos de temperaturas o valores de humedad que estén por fuera del rango de los datos medidos. Sin embargo, esto es posible solucionarlo incorporando más mediciones y durante plazos más extensos de modo de contar con muchos períodos de riego y evaporación y dejando de lado una porción de los datos para realizar la validación del modelo.

Como trabajo futuro resta calcular el incremento de humedad del suelo en función del volumen de agua de riego, con lo cual sería posible simular periodos más prolongados y en caso de necesitar mayor precisión es posible incluir términos de mayor orden.

El modelo no contempla el desarrollo y crecimiento de la planta, es decir, se supone que los requerimientos hídricos de la misma se mantienen acotados y dependen únicamente de las condiciones ambientales. Sin embargo, al igual que cualquier modelo de aprendizaje de máquina, es posible estimar la evolución de cualquier sistema si se incorpora mayor número de datos y se emplean modelos con más grados de libertad.

El principal desafío en análisis predictivo es lograr un modelo numérico que aproxime tan bien el comportamiento del sistema real tal que sea posible predecir el estado de este sistema en condiciones que aún no han ocurrido y durante un lapso de tiempo extenso. En el capítulo siguiente se presenta un nuevo análisis de datos, pero en este caso, la intención no es predecir el comportamiento de un sistema, sino describirlo cuantitativamente para explicar cuestiones más profundas que escapan a las posibilidades de realizar un análisis numérico.

# Capítulo 9

## Análisis descriptivo

### 9.1. Motivación

La erosión hídrica y eólica del suelo sobre el que se realizan actividades de producción agrícola implican la pérdida de nutrientes y microorganismos importantes. La presencia de rastrojos o cultivos de cobertura reduce en gran medida la erosión eólica al frenar el viento sobre la superficie, reduce la erosión hídrica al impedir el escurrimiento de agua, evita la pérdida de humedad por evaporación al estar el suelo protegido de la radiación solar, y también se evita fenómenos como el “planchado” o “sellado” del suelo que impide la emergencia de los cultivos sembrados luego de lluvias intensas. La cobertura vegetal debe contar con ciertas características para optimizar la calidad del suelo y entre éstas se puede mencionar su altura y densidad volumétrica.

El estudio del suelo como sustrato sobre el cual se desarrollan los cultivos y pasturas resulta de interés en todos los ámbitos de la producción agrícola.

La agricultura de conservación (AC), que es una herramienta de gestión integrada para afrontar los retos del futuro, incluye labranza conservacionista, rotaciones de cultivos, manejo de residuos y cultivos de cobertura [59]. La conservación del suelo es de suma importancia para lograr una producción sustentable, por ello se considera que estas técnicas conducen a mitigar los efectos erosivos del viento y la lluvia, reducir la emisión de gases con efecto invernadero, mejorar la calidad del suelo, aumentar el almacenamiento de agua del suelo y la productividad de los cultivos [60].

La labranza conservacionista consiste en evitar que haya lo que se conoce como suelo desnudo, manteniendo siempre la presencia de cobertura vegetal sobre la superficie, por ello la retención de residuos es clave en los sistemas de AC.

El nivel crítico de cobertura debe ser establecido para cada sitio específico, según el tipo de suelo y condiciones del ambiente [61]. Según la FAO, para ser considerada AC la cobertura mínima del suelo debe alcanzar el 30 % de la superficie, límite por debajo del cual se hace más importante el riesgo de procesos erosivos. Más de la mitad de lotes bajo siembra directa (SD) evaluados en la región subhúmeda-semiárida del sudoeste bonaerense presentaron una cobertura del suelo menor a la indicada [61]. En el semiárido seco del partido de Villarino, solo la labranza cero (L0) supera ese límite, con labranza vertical apenas se lo alcanza, mientras que la labranza convencional se encuentra muy por debajo del umbral [62].



Figura 9.1: Pruebas a campo en Estación Experimental INTA. Escáner (izquierda) y preparación para la medición (derecha).

En este capítulo se detallan los resultados obtenidos a partir del análisis de un conjunto de datos recolectados a campo empleando como instrumento principal de medición, el escáner de cobertura cuyo desarrollo se detalló en la sección 3.3 del capítulo 3. El estudio se llevó a cabo a lo largo de dos períodos de siembra de cereales de invierno cultivados en las cercanías de la Estación Experimental INTA de Hilario Ascasubi.

El objetivo del trabajo es brindar apoyo, mediante el instrumento desarrollado, al análisis de la influencia que tiene la disposición vertical y estructura de los rastrojos (remanente luego de la cosecha) de cereales de invierno como trigo, avena y centeno, en las propiedades del suelo, la erosión del mismo y la velocidad de descomposición de la materia orgánica como parte de la preparación del suelo para la siguiente campaña de siembra.

## 9.2. Adquisición de los datos

Se realizaron múltiples escaneos para la obtención de imágenes en parcelas con rastrojo de trigo candeal (*triticum durum*) a los que se le aplicaron tres tratamientos distintos, denominados vertical (V), horizontal (H) y pastoreo (P). El tratamiento *V* consiste en preservar el rastrojo en estado natural desde finalizada la cosecha sin realizar labores posteriores. El tratamiento *H* se logra aplastando el rastrojo contra la superficie del suelo mediante un rodillo aplanador de tiro. Finalmente, el tratamiento *P* consiste en simular el estado del rastrojo luego de ser pastoreado por ganado, resultando en una altura de cobertura inferior al del tratamiento *V*, pero superior al *H*, ya que la parte de la materia vegetal que permanece erguida, no se descompone rápidamente.

La superficie total del lote de estudio es de unas 3 Ha y fue dividido en varios bloques dentro de los cuales a la vez están aplicados los tres tratamientos. Se realizaron tres repeticiones en cada bloque de los tres tratamientos obteniéndose un total de 27 muestras en cada ensayo realizado en dos oportunidades, los días 21 de febrero de 2019 y 26 de mayo de 2020. La misma porción de cobertura vegetal escaneada, fue recolectada a mano para obtener el peso seco de la materia vegetal y se separó en dos partes, por un lado la materia vertical, que consiste en la parte de rastrojo que permanecía erguida y por otro la materia vegetal

horizontal, que es la que se encuentra en los primeros 5 cm desde la superficie del suelo. Se realizaron también mediciones de temperatura del suelo en tres niveles de profundidad distintos, tanto sobre la línea de siembra como entre líneas. Además, con un anemómetro de mano se registró la velocidad del viento al nivel del suelo, a 30 cm y a 200 cm de la superficie del suelo en los distintos tratamientos durante varias semanas consecutivas, de manera de contar con velocidades mínima, máxima y promedio, así como la comparación de las registradas por la estación meteorológica más cercana.

Se configuró el escaner para obtener modelos con una resolución de 1800 muestras o píxeles por medición, demorando entre 10 y 15 minutos cada una. Los datos obtenidos con el escaner se registraron en archivos de texto plano al igual que los valores de peso seco, que se obtuvieron los días posteriores a los ensayos.

### 9.3. Características de los tratamientos

Los tratamientos analizados en este trabajo presentan distintas propiedades y simulan distintas condiciones de conservación de la cobertura vegetal. Como parte del análisis de estas características se muestran los valores de temperatura y velocidad del viento medidos en parcelas con distintos tratamientos. En la primera fila de la Figura 9.2 se ilustran los modelos generados computacionalmente a partir de los datos obtenidos con el escáner de cobertura.

Como puede observarse, el tratamiento *V* presenta una envoltura de mayor volumen donde se diferencian las hileras de siembra. El tratamiento *P* se caracteriza por tener un menor nivel de altura y densidad de cobertura por encima de los 5 cm del suelo y el tratamiento *H* posee el menor volumen y peso de materia vegetal de los tres tratamientos. Esto es debido a que el contacto de los restos del cultivo con el suelo aceleran el proceso de descomposición.

La arquitectura o disposición del rastrojo tiene efectos significativos sobre la dinámica de agua del suelo luego de una precipitación. La disposición horizontal de los rastrojos permite reducir o retardar la pérdida de humedad por evaporación durante los 13 días posteriores al evento [63].

La presencia de rastrojo presenta un marcado efecto sobre la temperatura de suelo y el  $\Delta T^\circ$ , que es mayor al originado por los cambios en la disposición del mismo. El tratamiento *H* reduce la temperatura de suelo y amortigua el efecto de las altas temperaturas ambientales. El rastrojo en posición vertical reduce la velocidad de la capa de viento próxima a la superficie de suelo.

Los resultados de los análisis presentados en [63] demuestran que se modifica el microclima de la interfase suelo-atmósfera mediante la gestión del rastrojo. Resulta crucial identificar, jerarquizar y categorizar los factores como temperatura y viento en función del peso relativo que tengan en cada sitio específico. En zonas semiáridas, con alta incidencia del viento y la temperatura sobre el balance de agua en el suelo, podría resultar conveniente plantear estrategias de manejo de los rastrojos, en términos de cantidad y arquitectura combinada (horizontal y vertical), para maximizar la eficiencia de uso del agua y la conservación del recurso suelo.

La Figura 9.3 presenta los valores de velocidad del viento y temperatura del suelo obtenidos en distintas oportunidades en las parcelas donde se realizaron los ensayos de medición



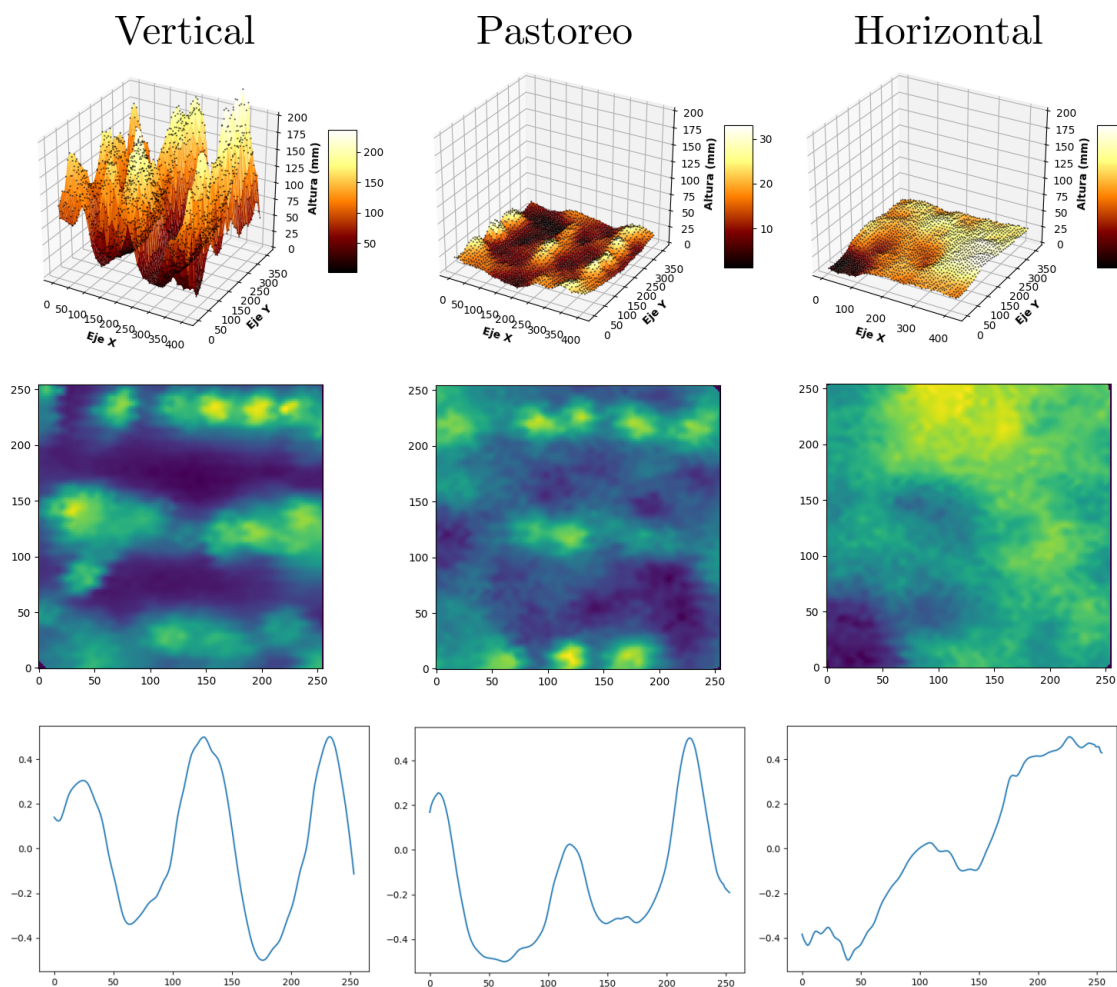


Figura 9.2: Imágenes de los tres tratamientos. De arriba hacia abajo, modelo 3D, imagen equivalente y sección transversal normalizada.

de cobertura.

La temperatura del suelo influye en la germinación de las semillas de los cultivos y el desarrollo de los microorganismos. En los primeros 10 cm de profundidad, la temperatura varía a lo largo del día por las condiciones climáticas, la incidencia del sol y también por la actividad de los microorganismos en la descomposición de la materia orgánica. Durante los meses de invierno existe un gradiente térmico positivo en función de la profundidad, es decir que la temperatura aumenta a mayor profundidad y ocurre lo contrario en los meses más calurosos.

Como puede observarse en el gráfico comparativo, el tratamiento *P* presenta mayor temperatura en la superficie del suelo que los otros dos. Los tres valores tienden a igualarse a mayor profundidad, dado que el contacto con la atmósfera es menor. El tratamiento *V* se iguala rápidamente al tratamiento *P* en los primeros centímetros de profundidad y el tratamiento *H* siempre presenta menores valores de temperatura. Una posible explicación para esto es que los tratamientos *P* y *V* dejan pasar los rayos solares en mayor medida de manera



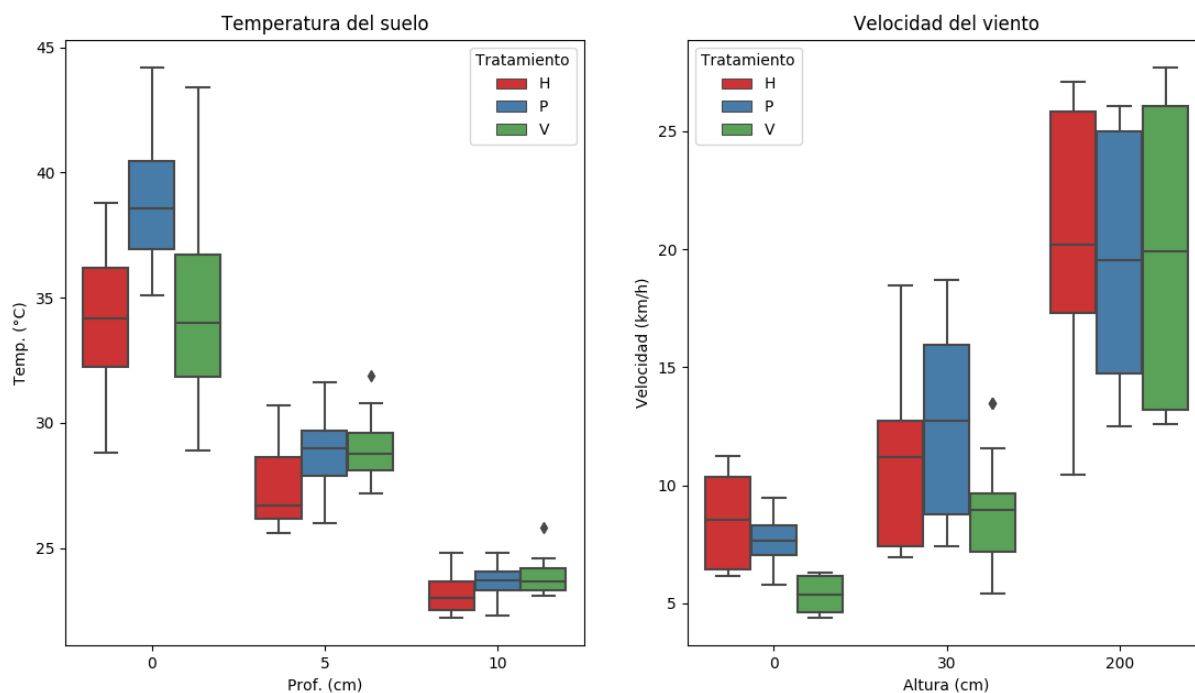


Figura 9.3: Temperatura del suelo a distintas profundidades (der.) y velocidad del viento a distintas alturas sobre el nivel del suelo (izq.) para cada tratamiento.

que la incidencia sobre el suelo provoca un aumento de temperatura.

Respecto a la intensidad del viento medida hasta una altura de 30 cm del suelo, puede observarse la diferencia que existe entre los tratamientos  $H$  y  $V$ , en donde se evidencia que la presencia de cobertura vegetal erguida reduce la velocidad del viento hasta en un 50 %. Esto significa que probablemente la erosión eólica también se ve disminuida en gran medida si se mantiene una buena altura de cobertura vegetal.

## 9.4. Estimación de materia seca

La Figura 9.4 muestra los valores de peso en Kg/ha de materia vegetal seca vertical recolectada y valores de volumen bajo la envoltura del modelo escaneado en litros. Como puede observarse a partir de esta gráfica, en las muestras de tratamiento  $V$  se correlaciona linealmente el volumen con la materia seca sin embargo esto no ocurre con los tratamientos  $H$  o  $P$ . Otra observación que permite visualizar esta figura es que la variable materia seca vertical por si sola, permite clasificar los tres tratamientos, es decir, los tratamientos se diferencian entre sí, principalmente por el valor de materia seca vertical.

Como se mencionó en la sección anterior, la importancia del peso de la materia seca vertical, radica en que influye en la velocidad del viento sobre la superficie del suelo y por lo tanto en la erosión y pérdida de materia orgánica. Por este motivo, contar con un mecanismo rápido para estimar este valor, agiliza la estimación de estas variables y además permite cuantificar estas medidas o establecer índices y marcos de comparación entre lotes.

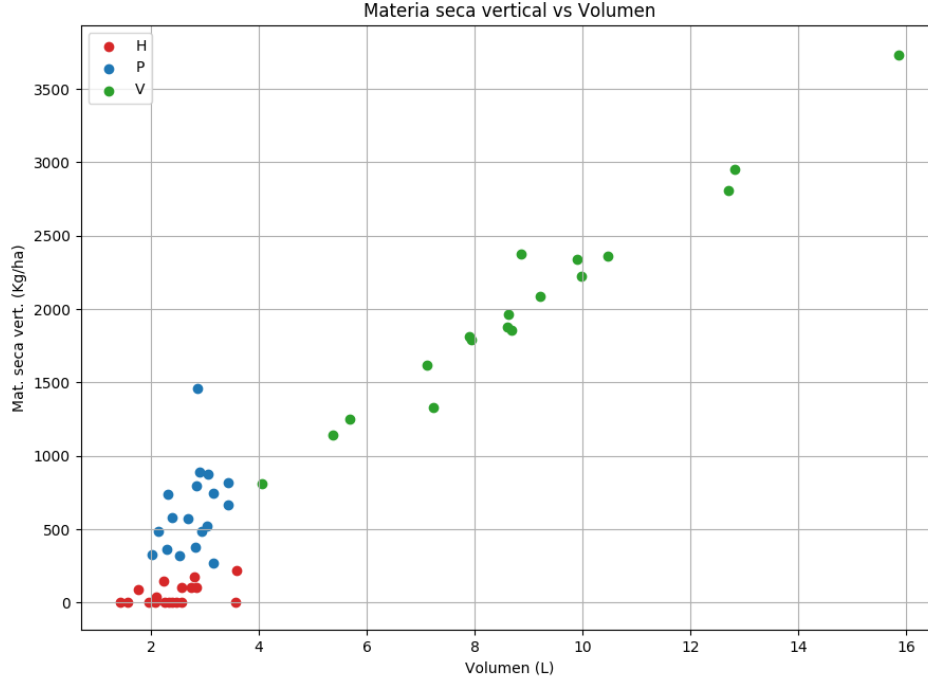


Figura 9.4: Gráfico de volumen bajo la envoltente vs peso de materia seca vertical.

Aplicando regresión lineal a los valores de volumen y materia seca medidos en los tratamientos  $V$ , se obtiene la siguiente expresión que relaciona estas variables:

$$M = 243,9 \cdot Vol - 163,8. \quad (9.1)$$

Con un coeficiente de determinación  $R^2 = 0,97$ , la ecuación 9.1 permite estimar el valor de materia seca vegetal vertical  $M$  en Kg/ha a partir del volumen  $Vol$  expresado en litros bajo la envoltente del modelo escaneado. Esto es válido para la materia vegetal recolectada a partir de los 5cm del nivel del suelo.

En el caso de los tratamientos  $H$  y  $P$ , no es posible estimar el peso, ya que como se explica en la sección 3.3 el escaner no arroja una medida de altura absoluta de la cobertura sobre el nivel del suelo, sino que es relativa al punto capturado más distante del sensor, que a veces puede presentar una medida inferior, sobre todo en los casos donde la densidad de cobertura no permite que el sensor capture la distancia al suelo.

## 9.5. Clasificación automática de tratamientos

Tal como se detalla en la sección anterior, el peso de la materia seca vertical es una variable que permite diferenciar con muy buena precisión los distintos tratamientos. Sin embargo, obtener este indicador no es inmediato sino que requiere un análisis posterior en laboratorio. El objetivo es poder clasificar los tratamientos únicamente a partir de las mediciones realizadas con el escaner de cobertura.

La intención es identificar cuáles características presentan los datos que se obtienen con el

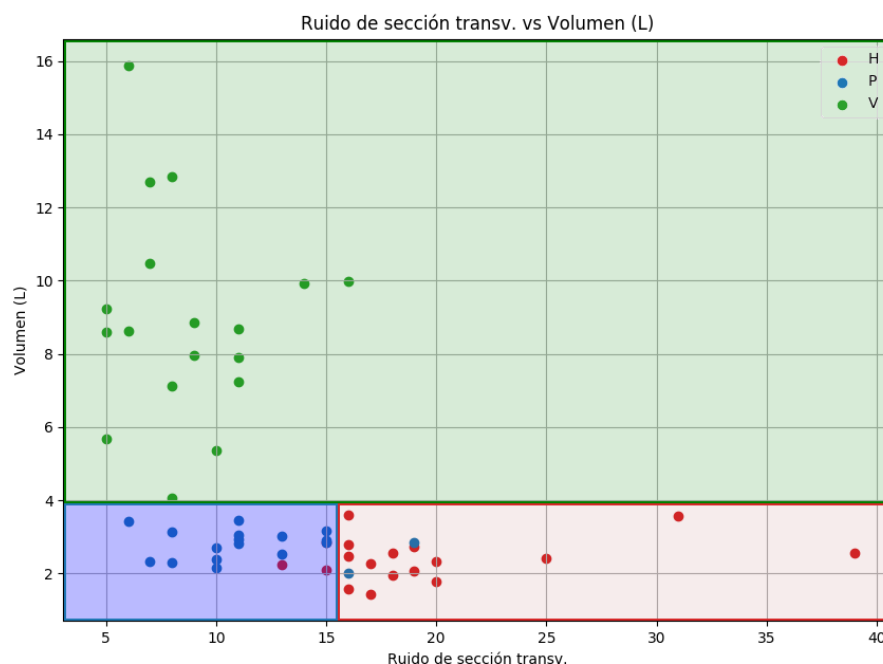


Figura 9.5: Clusterización de los modelos en el espacio volumen vs índice de ruido.

escaner de los distintos tratamientos de manera que sea posible luego estimar las propiedades de cualquier cobertura vegetal a partir del escaneo mediante el mismo instrumento.

En la Figura 9.4 se puede observar que el volumen del modelo escaneado permite diferenciar el tratamiento  $V$  respecto de los tratamientos  $H$  y  $P$ . Según los datos registrados, el límite que diferencia estos grupos se puede establecer en 4 L. De esta manera resta hallar un indicador que diferencie los tratamientos  $P$  y  $H$ .

El procedimiento para hallar un esquema de clasificación de estos modelos consistió en realizar un análisis discriminante lineal (LDA, por sus siglas en inglés) de todo el conjunto a medida que se fueron agregando más variables e indicadores. La técnica LDA es un método de aprendizaje de máquina supervisado que permite reducir las dimensiones del espacio de variables hasta lograr una combinación lineal de las mismas que permita maximizar las diferencias entre cada clase.

Estos tratamientos no pudieron ser clasificados mediante medidas estadísticas de centralización ni de dispersión como altura máxima, promedio, mediana ni desvío estándar de los datos que arroja el escáner. Otros indicadores más complejos basados en entropía de imágenes o transformada rápida de Fourier tampoco permiten diferenciar los modelos. A simple vista no resulta trivial establecer las características que diferencian las imágenes de los tratamientos, sin embargo, en la mayoría de los modelos de tratamiento  $P$  es posible distinguir las hileras de siembra.

Debido a que todas las mediciones se realizaron orientando el escáner de la misma manera que la dirección de siembra, se puede obtener una curva para visualizar la sección transversal del modelo. El inconveniente es que en algunas mediciones se alcanzó a capturar dos hileras y en otras tres. En la tercera fila de la Figura 9.5 se muestra la curva normalizada de cada modelo. Una particularidad que presentan estas gráficas de los tratamientos  $H$  son

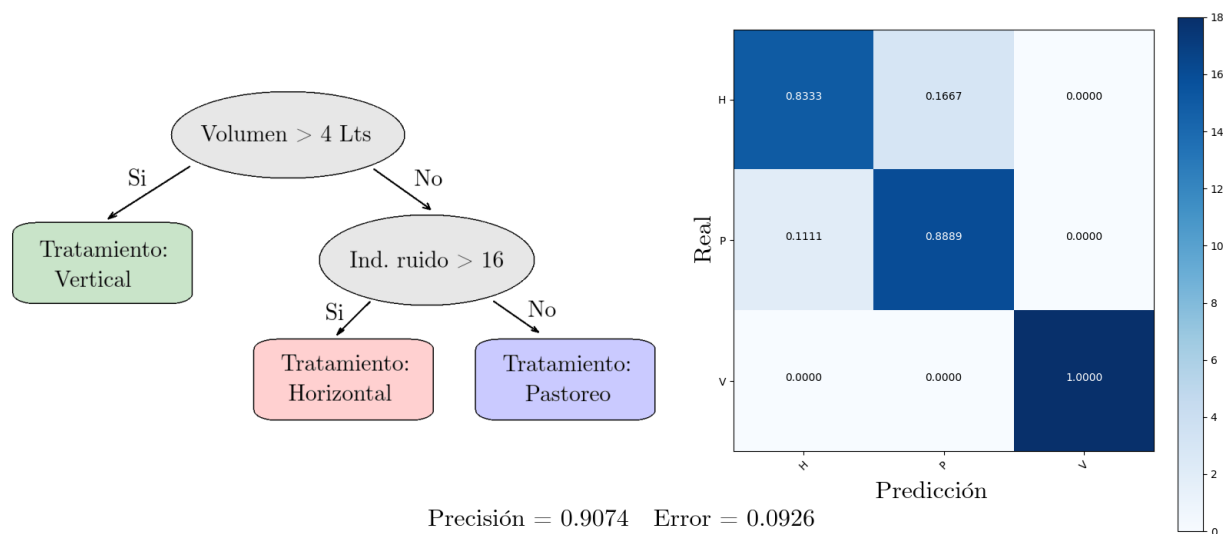


Figura 9.6: Algoritmo de clasificación de tratamientos en base a variables del escáner y matriz de confusión resultante.

oscilaciones de baja amplitud alrededor de la envolvente principal que se pueden extraer realizando el cálculo de la diferencia de la señal original con una versión suavizada mediante un filtro de media móvil o filtro promediador. El número de cruces por cero de esta resultante se correlaciona en gran medida con el tipo de tratamiento. A este índice se lo denominó “nivel de ruido” o “índice de ruido” de la sección transversal y en la Figura 9.5 se puede observar que permite diferenciar la mayoría de los casos.

Se estableció como límite para diferenciar los tratamientos  $H$  y  $P$  un valor de 16 cruces por cero de la señal de ruido. De esta manera se obtiene el árbol de la Figura 9.6 como algoritmo de clasificación para los tres tratamientos. La matriz de confusión de este algoritmo se muestra en la misma figura y como puede observarse, arroja más de un 90 % de efectividad empleando la totalidad del conjunto de datos. Al no tratarse de un esquema de aprendizaje de máquina y dado que no se cuenta con un número grande de observaciones, no se aplicaron estrategias de validación, pero dado a que las reglas de clasificación son muy generales y el árbol de decisión planteado no es profundo, el nivel de efectividad no se debe a un *overfitting*.

## 9.6. Consideraciones finales

En este capítulo se analizaron los datos de escaneo de cobertura vegetal, particularmente aplicado al caso del análisis de rastrojos de trigo. Debido a que las características morfológicas de los rastrojos de otros cultivos de invierno como avena, cebada o centeno son muy similares, los métodos y resultados presentados pueden extenderse a estos casos.

Se presentó una breve descripción de las características de los tres tratamientos a partir de una serie de datos medidos a campo, que son temperatura del suelo y velocidad del viento.

Una de las aplicaciones del instrumento es la capacidad de estimar el peso de la cobertura vegetal que se encuentra en disposición vertical, por encima de los 5cm del suelo. La fórmula de ajuste depende del tipo de cultivo y esto simplemente demuestra que para el caso de los

rastreros de invierno, el volumen de la envoltura escaneada es proporcional a la masa seca de la materia vegetal vertical.

El algoritmo de clasificación automático emplea sólo dos parámetros o índices, el volumen del modelo y el nivel o frecuencia de ruido. Estos valores se pueden ajustar automáticamente en función de la cantidad de datos para mejorar el grado de precisión, pero al tratarse de pocos grados de libertad, no es un esfuerzo que valga la pena. Este método tampoco requiere de un costo computacional muy alto y se puede implementar perfectamente tanto en las aplicaciones móviles que acompañan el dispositivo de adquisición como en el mismo firmware del microcontrolador. El cálculo del volumen del modelo escaneado consiste en multiplicar el área de escaneo por la altura promedio de los datos, por lo que también puede ser un valor que se calcula en forma continua y en tiempo real, a medida que se avanza realizando un escaneo a lo largo de una superficie de dimensiones indefinidas. Lo mismo ocurre con el cálculo del índice de ruido. Por este motivo, en este aspecto, el instrumento puede ser convertido en un clasificador de cobertura vegetal que arroje resultados en forma inmediata, donde la confiabilidad de la medición dependerá del área analizada.

En este capítulo concluye la presentación de los resultados referidos a la etapa del procesamiento de la información. Hasta aquí se han tratado problemáticas puntuales de los sistemas computacionales que, como se ha mencionado anteriormente, detallan aspectos de la adquisición, transmisión y procesamiento de la información. En el capítulo siguiente inicia la parte final de esta tesis donde se integran todos temas que han sido tratados hasta ahora bajo el contexto de los sistemas colaborativos.

# Parte V

## Sistemas colaborativos

# Capítulo 10

## Soporte LPWAN en entornos colaborativos de tiempo real

### 10.1. Motivación

Los primeros sistemas colaborativos estaban orientados al desarrollo de software, la redacción de informes técnicos, las presentaciones grupales. En forma, por lo general asincrónica, los distintos participantes trabajaban sobre un elemento común. Con la generalización del acceso a Internet, en especial con la llegada de las conexiones inalámbricas, y el crecimiento de los dispositivos inteligentes, el ámbito de aplicación de los sistemas colaborativos creció también de forma exponencial y hoy en día se encuentran en aplicaciones de turismo, trueque, culturales, salud, etc. En todos los casos, las personas representadas por sus dispositivos se comprometen a participar compartiendo información e incorporando conocimiento sobre el objeto del sistema. Para esto es preciso brindar garantías en relación a la privacidad y de seguridad de las mismas.

En este capítulo se presenta un modelo de interacción basado en Internet de las Cosas (IoT) construido sobre una LPWAN. El análisis se concentra sobre los nodos distribuidos que deben generar la información que maneja el sistema. En esta situación se deben desarrollar mecanismos que permitan definir con distinto tipo de parámetros a considerar cuáles son las variables que corresponde muestrear, la periodicidad y la distancia entre ellas para tener una muestra adecuada del estado del sistema. Cuando el enfoque está dado por ejemplo en el seguimiento de ganado en lotes o el monitoreo de infraestructura ganadera, la distribución de los nodos debe ser adecuada para capturar la información relevante, como ser el comportamiento atípico de un animal, daños o mal funcionamiento en el sistema de aguadas, entre otras. El sistema funciona distribuido a lo largo de una gran superficie de terreno que no cuenta con cobertura de comunicación ni provisión de energía, con lo cual debe ser robusto y contar con una autonomía duradera en las peores condiciones.

Esta infraestructura basada en IoT fue validada en una situación similar en cuanto a las condiciones de funcionamiento y este es el caso de un sistema de soporte para ciudadanos en situación de auto-evacuación en áreas urbanas, quienes podrían haber sido afectados por eventos catastróficos naturales o aquellos que pueden ser consecuencia de un descuido humano. En una situación catastrófica de estas características, no se garantiza el funcio-

namiento de las redes de comunicación ni de energía eléctrica. Sin embargo, las personas cuentan con dispositivos móviles con gran cantidad de procesamiento y ancho de banda de comunicación de corto alcance como el WiFi o Bluetooth. En este capítulo se detalla el elemento clave que permite la comunicación entre todas las personas en una situación con estas características.

Este sistema IoT es la base de un sistema basado en inteligencia de ambiente (AmI, por sus siglas en inglés) que se emplea como vínculo entre las personas y el centro de operaciones de emergencia que gestiona el procedimiento de evacuación de dichas personas. Particularmente, esta infraestructura guía a los ciudadanos hacia refugios y lugares seguros por medio de las denominadas unidades de asistencia (WU, por sus siglas en inglés) que se emplazan a lo largo de la ruta de evacuación y mediante aplicaciones móviles que actúan como accesos a las WU por parte de los usuarios. Cada WU actúa como *Access Point* o *Hotspot* y a la vez como servidor de una aplicación web que permite a los usuarios acceder a una interface de usuario muy sencilla.

En este trabajo se introdujeron diversos componentes, entre ellos una arquitectura de sistema que involucra los principales aspectos de diseño de la solución que la vuelve reutilizable por otros investigadores en distintos contextos, una implementación del sistema propuesto basado en dispositivos Raspberry Pi 3 con conectividad LoRa; una aplicación PWA por medio de la cual los usuarios pueden intercambiar información con el sistema a través de los *Access points*; un algoritmo simplificado de despliegue para la cobertura del área de interés con WUs y una evaluación experimental con simulaciones y mediciones reales. Vale la pena mencionar que el despliegue de este tipo de infraestructura basada en tecnología IoT tiene múltiples aplicaciones de inteligencia de ambiente, convirtiéndola en los pilares de soluciones para ambientes inteligentes.

## 10.2. Modelo del sistema

Para modelar las interacciones dentro del sistema basado en AmI, es necesario definir los actores y la forma en la que se relacionan. Estos actores pueden estar clasificados de distintas maneras. Dado su rol en el manejo de la información, se pueden diferenciar entre productores, consumidores o productores/consumidores, según su movilidad, pueden ser fijos o móviles y según su jerarquía dentro del sistema, se los puede clasificar como dispositivo final, concentradores o raíz.

En este sistema existen cuatro actores, ordenados de mayor a menor jerarquía: Centro de operaciones (E), Refugios (S), Unidades de asistencia (W) y Móviles (M). La Figura 10.1 presenta esquemáticamente los componentes del sistema y sus interacciones. Dentro de este sistema, los dispositivos móviles son los únicos actores que no tienen una posición fija. Todos los elementos pueden producir información y consumirla, ya sea obteniéndola a partir de las personas que interactúan con ellos o mediante el procesamiento de cierta información para generar nuevos resultados.





usuario da la vuelta y se dirige en sentido opuesto dando a entender que el camino está bloqueado. En el segundo caso, el usuario puede ingresar el obstáculo mediante la aplicación móvil.

4. El evacuado está constantemente informando sobre su posición a lo largo de la ruta y una vez que llega al refugio, con lo cual el sistema es capaz de evitar que las rutas y los refugios se colapsen. Esta es otra interacción implícita, ya que el usuario no debe hacer nada para informar su ubicación.

## 10.3. Implementación

Se implementó una prueba de concepto en el Laboratorio de Sistemas Digitales de la Universidad Nacional del Sur. Por este motivo se utilizaron varios componentes COTS como bloques constitutivos. Básicamente el sistema requiere de la implementación de cuatro tipos de nodos: la aplicación móvil, las unidades de asistencia, nodo de refugios y el servidor del centro de operaciones. Se describen a continuación los detalles y características de cada componente.

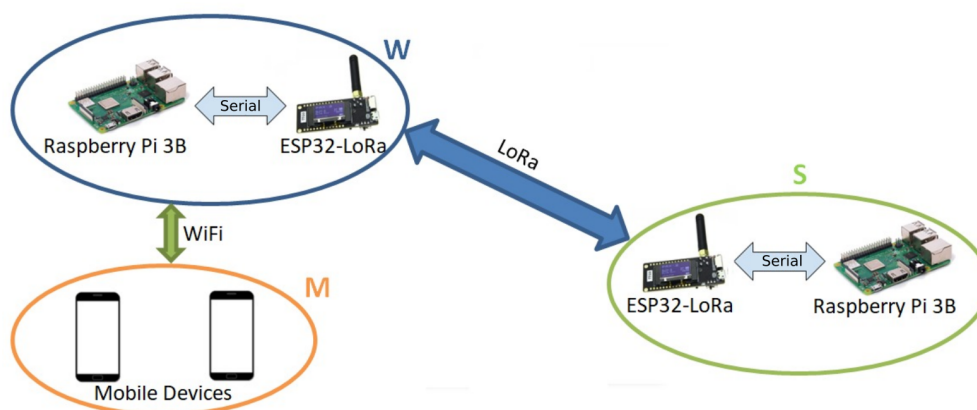


Figura 10.2: Diagrama de componentes del sistema

### 10.3.1. Unidades de asistencia

Los nodos W son los elementos principales del sistema y proveen el acceso a los usuarios a las rutas de evacuación. Se pueden entender como simples puntos de acceso (AP, por sus siglas en inglés) que a la vez son nodos finales o *endpoints* de la red LPWAN. Estos nodos están encargados de recibir reporte de nuevos obstáculos y calcular nuevas rutas de escape hacia el refugio más cercano. La cantidad de dispositivos móviles que se conectan a cada W es un indicador útil en el caso del sistema de evacuación porque indica la cantidad de personas que se están desplazando por las distintas rutas. Estos nodos actúan como *relays* o *brokers* entre el sistema y el usuario final y cuenta con capacidad de procesamiento autónoma para implementar ciertos algoritmos necesarios.

Se dispone de dos interfaces de comunicación. Una de ellas es un enlace LoRa que se emplea para el intercambio de información con el nodo S. La otra consiste en una antena WiFi que provee un punto de acceso o *hotspot* a los nodos M.

Estos dispositivos fueron implementados utilizando la placa de desarrollo Raspberry Pi 3B. La interface LoRa se implementó mediante la placa de desarrollo TTGO LoRa32 SX1276. La interacción entre estos dos elementos se realiza por medio de una interface serie.

La computadora Raspberry Pi es configurada como punto de acceso creando una red WiFi de acceso libre bajo un SSID que permita a los usuarios identificarla rápidamente, en caso de que no sea la única red presente. El acceso a la red se configura con un portal cautivo que inmediatamente abre un sitio web en el móvil del usuario. Convenientemente, este sitio web se trata de una aplicación con todas las instrucciones para el usuario y provee un enlace de instalación de una versión PWA que puede operar en el dispositivo móvil de manera *offline*, como se detalla en la siguiente sección.

La comunicación con el dispositivo móvil del usuario se realiza por medio de *web sockets*. Tanto en el nodo W como M se almacena una base de datos basada en ficheros con formato JSON. El nodo W posee un algoritmo de sincronización de estas bases de datos que determina cuál información es más reciente o relevante y actualiza esta información generando una única base de datos que es devuelta al dispositivo M que maneja el usuario. Si la base de datos del nodo W es actualizada con esta operación, los cambios son puestos en cola para ser despachados al nodo S correspondiente.

Mediante otro algoritmo de sincronización similar, se actualizan las bases de datos entre nodos W, ya que el mismo procedimiento es llevado a cabo en varios puntos simultáneamente y constantemente se va actualizando la información que maneja todo el sistema.

### 10.3.2. Dispositivos móviles

Los teléfonos inteligentes o tablets y cualquier otro dispositivo móvil con interface WiFi y soporte para la ejecución de aplicaciones basadas en webviews, puede actuar como nodos M. El dispositivo no necesariamente requiere de una aplicación específica preinstalada, ya que en el contexto de un sistema de evacuación en caso de emergencias, un evento catastrófico puede ocurrir en cualquier momento y es muy difícil lograr que un porcentaje alto de la población mantenga algún tipo de preparación previa en este sentido. La interface con la cual el usuario del móvil puede interactuar con el sistema es lo que realmente representa el componente central de los nodos M y se descargará al momento de ser necesaria. Si ya se cuenta con una aplicación instalada en el móvil, entonces el proceso de conexión y sincronización se agiliza considerablemente, sin embargo esto no siempre será posible y este factor es contemplado por el diseño propuesto.

Como se mencionó anteriormente, la aplicación mediante la cual interactúan los usuarios con el sistema son aplicaciones web que bien pueden estar previamente instaladas en los dispositivos o ser accedidas por medio del *hotspot* que generan los nodos W. Esta aplicación contiene un mapa interactivo del lugar sobre el cual se dibuja el recorrido que el evacuado debe realizar para alcanzar la posición donde se encuentra la siguiente unidad de asistencia o W. A lo largo del trayecto, el usuario puede marcar calles obstruidas tildando sobre el mapa que le presenta la aplicación y esta información es sincronizada con el sistema al conectarse el móvil a la red provista por la unidad de asistencia. En este sentido el usuario actúa como

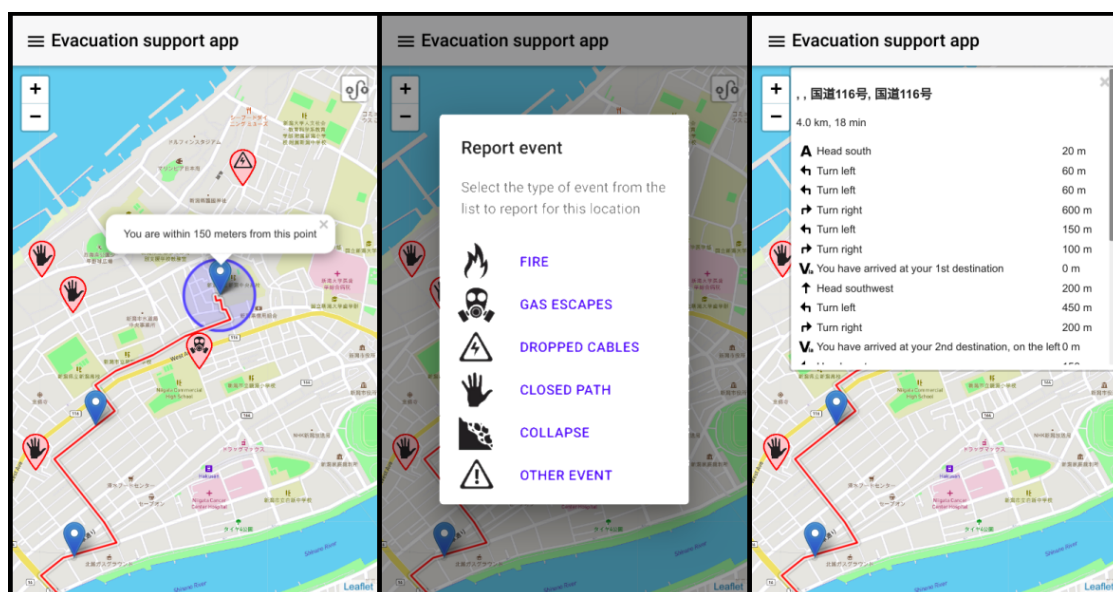


Figura 10.3: Capturas de pantalla de la aplicación móvil.

sensor humano [65].

En la Figura 10.3 se muestran tres capturas de pantalla de la aplicación web ejecutándose en un dispositivo móvil. Esta aplicación puede ser ejecutada incluso por el webview que permite mostrar un portal cautivo por defecto que viene preinstalado en cualquier sistema operativo, tanto de teléfonos inteligentes como computadoras personales, entre otros. La aplicación, desarrollada empleando puramente tecnologías web, posee una interface gráfica implementada con Framework7. El mapa interactivo es provisto por la librería Leaflet con distintos plugins que permiten el trazado de rutas sobre las calles.

### 10.3.3. Refugios

En el contexto del sistema colaborativo como soporte a la evacuación, se supone que las organizaciones gubernamentales ya cuentan con un protocolo organizado donde fueron definidos los emplazamientos de los refugios dependiendo de los riesgos existentes en el lugar. Los nodos S poseen un enlace satelital de conexión a la nube en caso de perder comunicación con el centro de operaciones y pueden operar con cierta independencia, sin embargo deben hacer un esfuerzo constante por restablecer la comunicación normal y evitar el uso del enlace satelital por el costo que ello implica en caso de tampoco disponer de suministro energético.

### 10.3.4. Centro de operaciones

El centro de operaciones contiene el servidor principal del sistema y actúa como maestro o nodo principal de la red. Dado que el sistema tiene capacidad para operar en forma distribuida, ciertas decisiones que deben ser tomadas por las autoridades a cargo se definen en el centro de operaciones y se inicia una propagación de estos comandos por la red del sistema para que surjan efecto.

## 10.4. Despliegue de la red

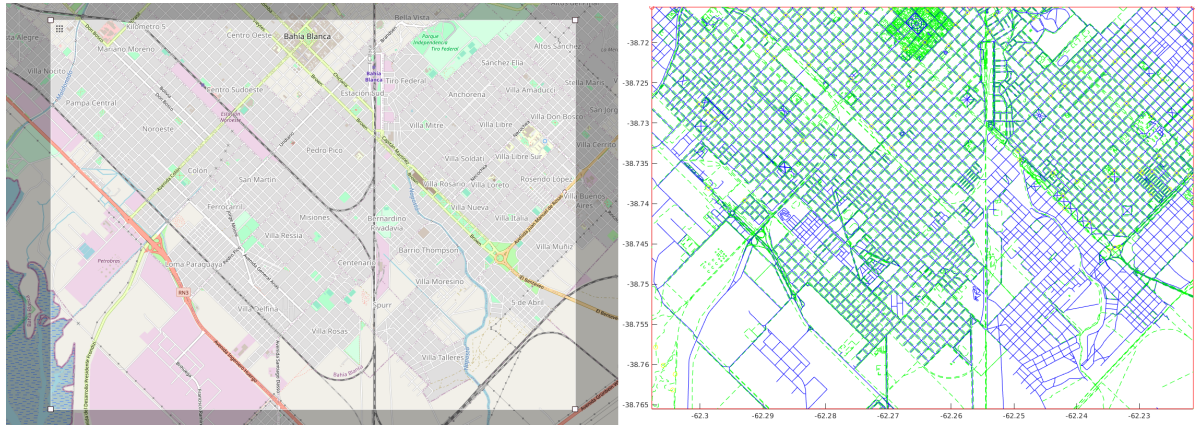


Figura 10.4: Región capturada del mapa de Bahía Blanca.

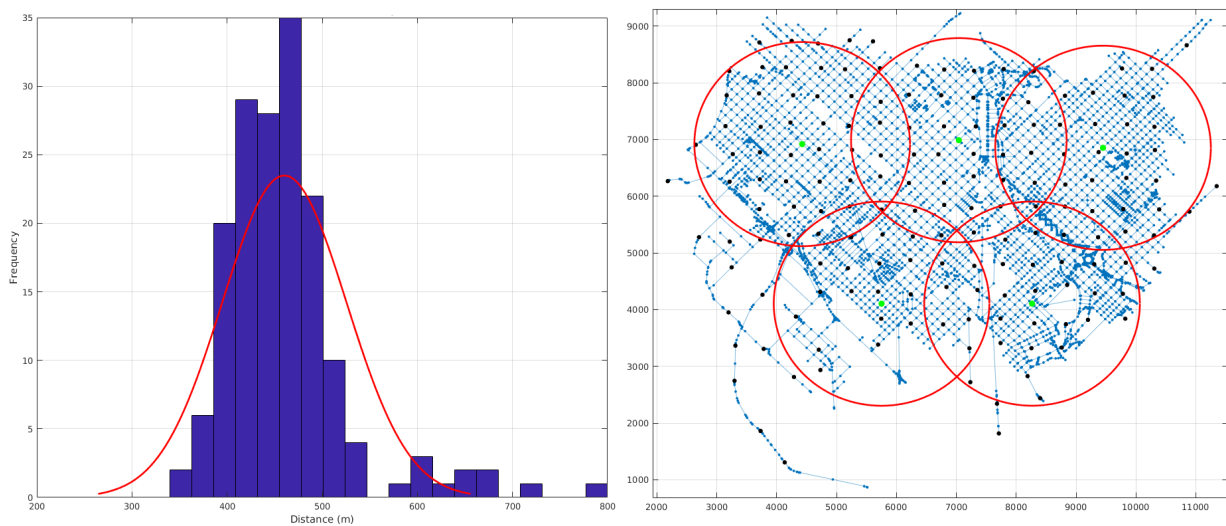


Figura 10.5: Distribución de distancia entre nodos vecinos (izq.). Mapa de nodos de la red sobre la ciudad de Bahía Blanca para  $D = 500m$  (der.).

La implementación del sistema depende de la cobertura del sistema AmI. En el contexto de un sistema de soporte a la evacuación, es necesario alcanzar la cobertura de la mayor área posible y con una densidad de nodos suficiente que permita a las personas ir de un punto al otro sin que les demande demasiado tiempo y corran el riesgo de perderse en el camino o desplazarse por lugares inseguros. Por ello, siguiendo los lineamientos presentados en el capítulo 7, se plantea un esquema de despliegue donde el parámetro limitante es la distancia  $D$  máxima que puede haber entre dos unidades de asistencia. El caso es similar al planteado anteriormente donde la limitación radicaba en la potencia de la señal de los nodos que tiene un alcance acotado. Otra diferencia, es que la distancia  $D$  no se calcula mediante la ecuación

de distancia euclídea sino que se refiere al tiempo que demanda desplazarse entre dos puntos y para esto se requiere como contexto el mapa de calles de la ciudad.

En un principio se planteó un modelo de optimización que permite calcular la distribución de nodos por toda una ciudad de manera que se verifique la única condición de que dos nodos adyacentes no se encuentren a una distancia de recorrido entre ellos superior a  $D$  metros. El inconveniente que presenta este método, es que el grafo de calles de una ciudad es un modelo muy complejo y requiere de un segmentado previo con el fin de indentificar los lugares donde es factible emplazar una unidad de asistencia de forma permanente.

Las Figuras 10.4 y 10.5 muestran, a modo de ejemplo, una distribución tentativa de nodos a lo largo de la ciudad de Bahía Blanca. Si bien esta ciudad no es propensa a sufrir fenómenos catastróficos como tsunamis o terremotos de gran intensidad, la zona urbana se encuentra próxima a un puerto y una zona industrial que ante incidentes pueden ocasionar una eventual evacuación de los habitantes en los barrios aledaños. La distribución de nodos para esta ciudad se obtiene superponiendo una grilla con distribución regular y ajustando las posiciones de los nodos para que coincidan con intersecciones de las calles del grafo. Como se puede observar en el gráfico de distribución de distancias de la Figura 10.5 izquierda, la gran mayoría de los nodos quedan distanciados a menos de  $D = 500\text{mts}$  entre sí.

## 10.5. Pruebas experimentales

Para realizar una validación de la capacidad de comunicación del sistema se implementó un enlace entre un nodo S y uno W como prueba de concepto. Se ubicó un nodo S emplazado en la terraza de un edificio a 30 m de altura mientras que se desplazó un nodo para simular una unidad de asistencia ubicada en distintos puntos de la ciudad. Los transceptores LoRa se configuraron con SF=12 y BW=125KHz. En la Tabla 10.1 se listan los parámetros medidos.

| Distancia [m] | RSSI [dBm] | SNR [dB] |
|---------------|------------|----------|
| 2000          | -113       | -12.95   |
| 1800          | -119       | -10.50   |
| 1300          | -116       | -14.80   |
| 1000          | -111       | -10.53   |
| 700           | -118       | -9.55    |
| 500           | -115       | -16.50   |

Cuadro 10.1: Parámetros medidos en distintas posiciones.

Para cada ubicación, se realizaron cinco experimentos para enviar 1, 5, 10, 15 y 20 eventos, presentando 11 mensajes en total. Las últimas peticiones del nodo S fueron respondidas con cuatro mensajes para enviar los 20 eventos. En todos los casos, la demora de transmisión fue de 33 segundos con un rendimiento efectivo de 26.9 bytes/seg. Las mediciones se realizaron en distintos puntos de la ciudad, combinando áreas residenciales con comerciales. Incluso ubicando la antena del nodo W en el techo de un auto en lugar de su posición ideal (entre 3 y 10 m de altura), el experimento muestra la factibilidad de la red propuesta y la baja tasa de error.

## 10.6. Consideraciones finales

En este capítulo se propuso un sistema AmI basado en componentes de IoT con soporte LPWAN. El sistema fue validado mediante una aplicación de soporte a tareas de evacuación en zonas urbanas que pueden verse afectadas por eventos catastróficos limitando la capacidad de comunicación de las personas, sin embargo los resultados pueden extenderse para implementar la misma arquitectura para otros propósitos.

El escenario hipotético que se plantea es el súbito colapso de los sistemas de comunicación y de provisión de energía eléctrica lo cual imposibilita a las personas que se encuentran en situación de emergencia, lograr una buena organización durante un eventual procedimiento de rescate. Excluyendo cualquier evento de desastre natural o artificial y considerando sólo el hecho de estar sin energía y sin comunicación, para el ciudadano esto puede significar un estado de emergencia, sin embargo, para el campesino es una situación más bien normal, ya que acostumbra a vivir en zonas con poco acceso a las redes de comunicación y al suministro ininterrumpido de energía. Por este motivo, un sistema AmI con soporte LPWAN permite implementar un esquema colaborativo con buen grado de robustez en una red conformada por actores con distintas funcionalidades en un espacio de zona rural. Ejemplos de esto pueden ser los sistemas de monitoreo y alertas de eventos meteorológicos, monitoreo de infraestructura para ganadería y vigilancia del estado general del ganado, entre muchos otros.

El modelo presentado cubre tanto las interacciones implícitas como explícitas entre el sistema y los usuarios finales. Se extendió el algoritmo de distribución de nodos presentado en capítulos anteriores para computar la topología de la red del sistema de manera de lograr una cobertura adecuada minimizando el número de unidades de asistencia. Finalmente se implementó una prueba de concepto mediante componentes COTS y se realizaron varias mediciones del rendimiento de los enlaces que permitieron verificar la factibilidad del sistema.

En el próximo capítulo se presenta el modelo de un sistema colaborativo que se aplica de forma directa a cuestiones relacionadas a la producción agrícola y donde se hace énfasis en la adquisición y el esquema de procesamiento de la información, más que en cuestiones de transmisión de los datos.

# Capítulo 11

## Sistema colaborativo para el mantenimiento de redes viales rurales

### 11.1. Motivación

La posibilidad de obtener datos del camino a partir de la información provista por el usuario es un tema en continuo desarrollo. En este campo hay numerosas aplicaciones como Waze que relevando la cantidad de suscriptos al sistema y su velocidad de desplazamiento computan el estado de congestión de las calles en una ciudad. La idea de que las personas por medio de sus dispositivos móviles se vuelven sensores se planteó en [66, 65]. La información obtenida de ese modo se denomina *crowdsensing* ya que son muchos los que proveen la información.

La red vial de Argentina, que se extiende a lo largo de más de 500.000 km de carreteras, se encuentra integrada por toda superficie terrestre, pública o privada, por donde circulan peatones y vehículos. Dependiendo del material de la superficie de los caminos, se distribuye aproximadamente según la siguiente proporción: [67].

- 39 % de rutas de asfalto.
- 33 % de caminos de tierra.
- 19 % de rutas de hormigón.
- 9 % de caminos de ripio.

La Clasificación Nacional de vías de circulación vehicular contempla tres tipos de red vial: primaria, secundaria y terciaria. La red vial primaria se refiere a las vías troncales, que conectan grandes centros urbanos, zonas productivas de relevancia, conexiones internacionales, puertos y aeropuertos. Esta red primaria, en el caso de Argentina, generalmente está pavimentada. Las rutas nacionales pertenecen a esta red primaria y parte de las rutas provinciales también. La red vial secundaria conecta centros urbanos y zonas productivas de menor tránsito que las primarias pero de relativa magnitud, en general son rutas provinciales y, de acuerdo a la provincia, están pavimentadas en mayor o menor grado. Una parte de las secundarias que no están pavimentadas pueden tener algún tipo de mejora. Por último,



la red terciaria, no cumple una función central de conectividad entre centros urbanos sino de conectividad más capilar con población rural, pequeños asentamientos humanos, escuelas rurales, puntos turísticos y establecimientos productivos, principalmente agropecuarios. Esta red terciaria también suele llamarse caminos vecinales, y en su mayor parte no están pavimentados ni mejorados. La jurisdicción de esta red varía entre provincias, en unas depende de Vialidad Provincial, en otras está delegado en Consorcios Camineros y en otras depende de los municipios [67].

La proporción de carreteras se distribuye según su jurisdicción, de la siguiente manera:

- 7 % pertenece a la red troncal primaria con jurisdicción nacional.
- 35 % es gestionada por la jurisdicción provincial.
- 58 % corresponde a la red terciaria administrada por municipios o comunas.

Las condiciones de la infraestructura de transporte y la accesibilidad constituyen elementos clave en la eficiencia de la mayoría de las organizaciones, dado que permiten vincular sectores que se encuentran geográficamente dispersos dentro del territorio y así concretar los movimientos. En este contexto, el mal estado de estas vías constituye un freno al desarrollo de la sociedad y frente a eso, se necesita una solución sostenible, con un correcto mantenimiento y desarrollo de obras básicas que garanticen su transitabilidad [68].

Las tareas de inspección y relevamiento del estado de los caminos implican recorridos regulares a lo largo de muchos kilómetros que significa la extensión de la red vial y que debe ser efectuado por parte de personal capacitado en la identificación de las distintas fallas que pueden presentarse en los caminos no pavimentados [69]. Dependiendo de las características del camino y las fallas que se presentan, se debe determinar el tipo de labor a realizar que también dependerá del presupuesto disponible [70].

En los caminos de bajo flujo vehicular y que, en general, no suelen estar pavimentados, las tareas de mantenimiento deben ser frecuentes ya que se ven afectados por eventos meteorológicos erosivos como la lluvia o el viento y la circulación de camiones o maquinaria pesada, que dañan rápidamente la regularidad de la vía dificultando la circulación de vehículos.

La contribución principal de este trabajo es la propuesta de un sistema colaborativo para la determinación del estado de los caminos, en especial los rurales que son de bajo tránsito y empleando como información de base, la medición y auto-reporte de las velocidades de circulación de los vehículos. El objetivo de este sistema es complementar el relevamiento del estado de los caminos que realizan las organizaciones encargadas del mantenimiento de la red vial y optimizar la gestión de recursos destinados a tal propósito. Si bien el sistema puede ser escalado a niveles provinciales y nacionales, tiene un objetivo más reducido, en el cual los caminos de una zona rural, que puede involucrar a varios municipios vecinos, son relevados tanto por los usuarios de los caminos como los encargados de realizar el mantenimiento. La propuesta incluye un modelo de determinación de mal estado junto a una validación cruzada en función del registro meteorológico de la zona.

## 11.2. Modelo del Sistema: Casos de usos

El objetivo es contar con información georeferenciada y actualizada del estado y calidad de los caminos rurales. Se parte de la hipótesis de que las variaciones en la velocidad de tránsito de los vehículos es indicador del estado de la vía, asumiéndose que cuanto menor es la velocidad respecto del promedio de cada vehículo a lo largo de cierto trayecto, peor es el estado del camino en ese trayecto. Dado que la velocidad de circulación puede depender de muchos factores entre los que se incluye el tipo de vehículo, obstáculos, visibilidad, hábito de manejo del conductor, etc., se normalizan los valores para establecer un marco comparativo general para todos los usuarios de la carretera.

A continuación se detallan los distintos aspectos del sistema, partiendo de un esquema general hasta llegar al detalle de cada actor, donde se sugieren las distintas tecnologías que permitirían la puesta en marcha del sistema colaborativo propuesto.

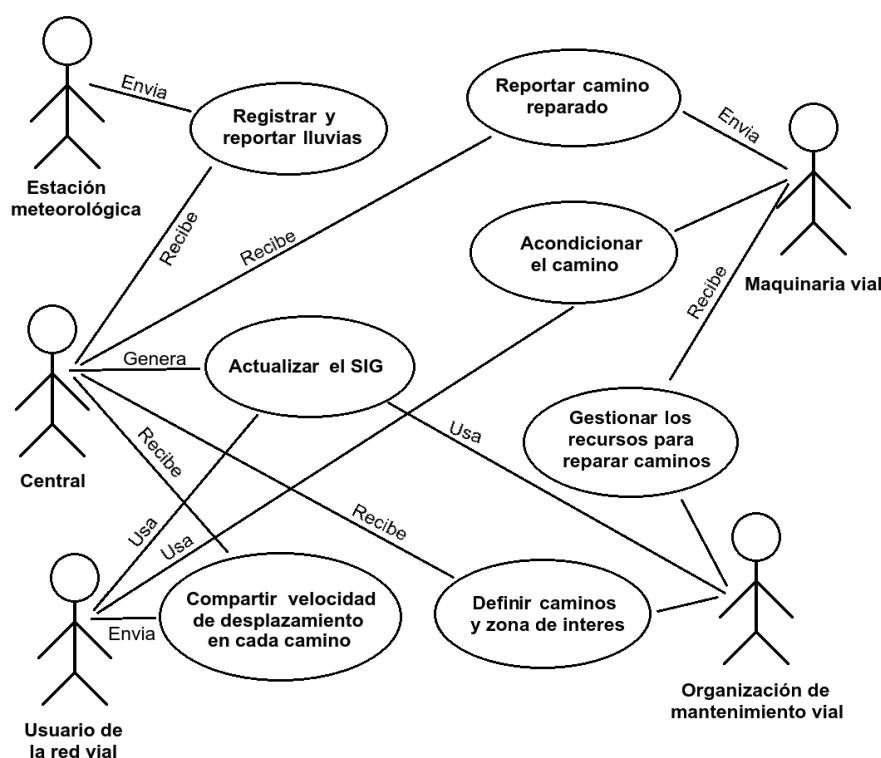


Figura 11.1: Diagrama de casos de uso.

### 11.2.1. Arquitectura

El sistema debe tener una arquitectura centralizada, ya que los principales actores, los dataloggers en vehículos particulares y maquinarias, en general, no operarán en forma simultánea y se encontrarán la mayor parte del tiempo fuera de línea. Es necesario que al menos uno de los actores o nodos de la red se encuentre todo el tiempo conectado y accesible

para evitar la pérdida de información. Además, para producir un resultado, el sistema requiere de procesar la mayor parte o toda la información que releva cada actor, lo cual obliga a que esta información se encuentre centralizada en un lugar.

La red que brinda conectividad a los actores no necesariamente debe estar presente todo el tiempo. Los dataloggers registran los datos a lo largo de todo el recorrido por un dado camino en forma offline y al hallar acceso a Internet, pueden sincronizar sus datos con el servidor de la central.

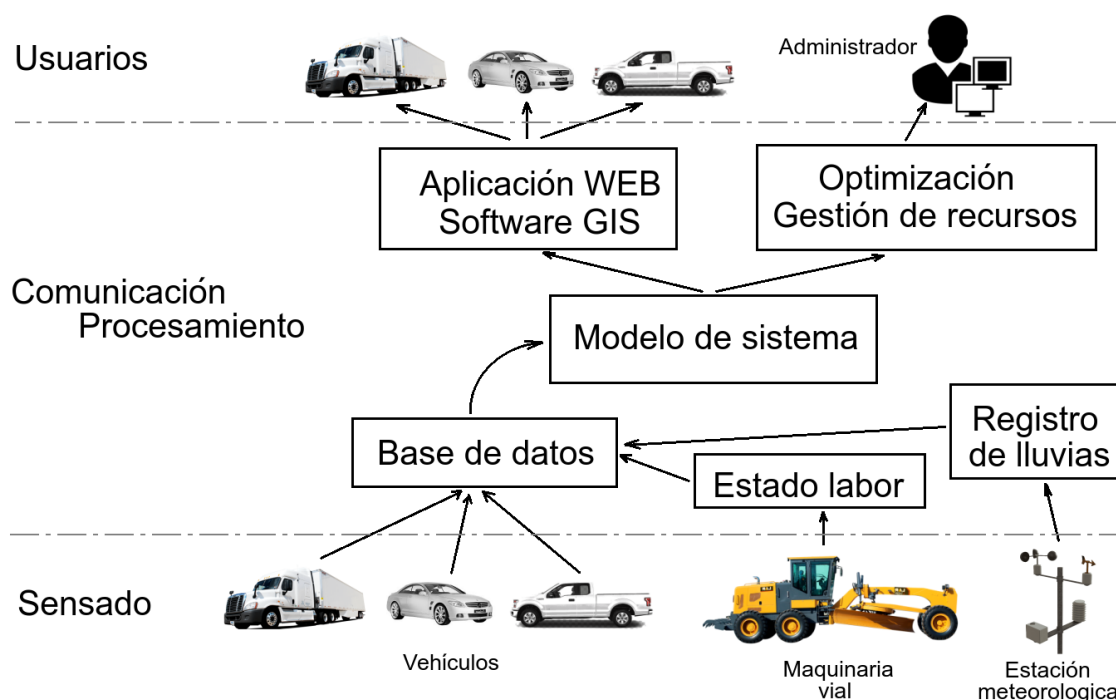


Figura 11.2: Esquema Internet de las Personas del Sistema Colaborativo propuesto

### 11.2.2. Actores

El sistema colaborativo cuenta con la participación de los siguientes actores cuyos casos de uso contemplan los siguientes roles:

**Usuario de red vial (datalogger):** reporta un conjunto de datos dispersos donde cada dato está compuesto por la tupla latitud, longitud y estampa de tiempo. La velocidad de desplazamiento puede o no estar presente. Puede ser un smartphone o un dispositivo ad-hoc.

**Maquinaria vial (datalogger):** reporta datos dispersos compuestos por la tupla latitud, longitud, identificador de maquinaria, y variables de estado de la máquina (combustible, presión de aceite, temperatura de motor, etc.). Este datalogger debe tener acceso a los sensores para monitorear el funcionamiento de la maquinaria con lo cual requiere de un dispositivo embebido ad-hoc.

**Estación meteorológica o servicio de meteorología:** registra y pone a disposición información actualizada sobre precipitaciones.

**Central de procesamiento:** analiza todos los datos y genera un modelo del estado de caminos (SIG).

**Organización de mantenimiento vial (trabajador):** recibe el reporte que genera el sistema para gestionar los recursos de mantenimiento de la red vial.

El sistema propuesto se puede enmarcar dentro de lo que se conoce en Internet de las Personas (IoP por su sigla en Inglés) [71] como una extensión de Internet de las Cosas. En el caso del sistema colaborativo, las personas por medio de sus equipos (dispositivos móviles) se transforman en sensores o productores de información que luego de ser procesada es presentada a los mismos usuarios como información ordenada que les sirve para tomar decisiones. En la Figura 11.2, se presenta el modelo de capas usualmente utilizado en IoP para representar el flujo de información. Como se puede apreciar, la capa de usuarios y sensado es prácticamente igual. La única diferencia entre ambas es que la estación meteorológica es un sensor tradicional cuyas mediciones sirven para predecir en forma probabilística el estado de los caminos cercanos. Con los datos obtenidos por los sensores e incorporados en una base de datos, el sistema procesa la información determinando el estado del camino. Esta información es puesta a disposición de los usuarios de la red vial que podrán optar por el camino en mejores condiciones por un lado y por el otro se aplica un algoritmo de optimización para poder generar las órdenes de trabajo para la maquinaria vial. Esta función final es tomada por el Administrador del sistema que ordenará la secuencia de las reparaciones considerando las ubicaciones de las máquinas y la prioridad del arreglo. El problema de optimización es similar al que se tiene con la planificación de tareas presentado en la Sección 5 en el sentido de que se trata de lograr una gestión óptima del tiempo dada una cierta limitación de recursos.

## 11.3. Composición del Sistema

La Figura 11.1 muestra el diagrama de casos de uso del sistema propuesto. Se puede apreciar el modo en que los diferentes actores interactúan entre sí y las funcionalidades que se deben considerar al momento de desarrollar el sistema. A continuación se presenta una descripción más detallada de cada componente del sistema con sus funcionalidades.

### 11.3.1. Dataloggers de vehículos

Los vehículos particulares que hacen uso regular de una dada red vial de interés en la propuesta de esta plataforma deben contar con dataloggers que registran los pares de coordenadas GPS con estampa de tiempo, que es la mínima información requerida para estimar la velocidad de desplazamiento georeferenciada a lo largo de los distintos caminos. La velocidad también puede ser calculada por el mismo datalogger e incluso aplicar correcciones utilizando acelerómetros o giróscopos, si es que estos sensores están disponibles. La combinación de datos de múltiples mediciones también permite mejorar la estimación de la velocidad del tránsito [72].

Con el objetivo de lograr un sistema que requiera de la menor infraestructura posible, se puede utilizar una aplicación para teléfonos móviles para convertirlos en dataloggers de este sistema. El consumo de energía de esta aplicación es considerable ya que se debe lograr

una alta tasa de actualización de la posición del vehículo, aún si estos datos no se reportan inmediatamente al servidor.

La alternativa al teléfono inteligente es diseñar dispositivos ad-hoc que cuenten con módulo GPS, módulo de conectividad GSM o WiFi para el acceso a internet y módulo de almacenamiento. Este dispositivo estaría permanentemente instalado en el vehículo y la ventaja es que se trata de un dispositivo específico que puede activarse o desactivarse automáticamente sin la intervención del usuario y evita que, tal como puede ocurrir con el caso del teléfono móvil, el usuario se olvide de activar la aplicación al realizar un recorrido por los caminos.

Si los dataloggers reportan datos dispersos, es necesario identificar y diferenciar los datos que corresponden a cada vehículo, por lo que es requisito contar con un identificador para cada datalogger y cuyo valor podría cambiar entre sesiones de registro de datos, de modo que no contenga ninguna referencia a la identidad del usuario. El hecho de que los usuarios del sistema sean anónimos, garantiza la privacidad de los mismos y evita la desconfianza que puede generar el brindar información personal sobre los desplazamientos de cada persona que requiere esta plataforma para funcionar.

Los datos registrados por los dataloggers no necesariamente serán reportados inmediatamente a la central, dado que no siempre habrá disponibilidad de acceso a la red pero deben tener una validez de tiempo limitado, dependiendo de las condiciones climáticas. Si se registran precipitaciones, el estado del camino puede cambiar rápidamente, con lo cual, dependiendo del milimetraje de lluvias, se acorta la validez de la información reportada por los vehículos particulares. De una u otra forma, será la central la que determina la validez de un informe.

Una alternativa al informe continuo de datos, es utilizar un medio de almacenamiento local de cada datalogger que permita grabar todo el recorrido realizado y que se envíe a la central un fichero con toda la información por medio de un protocolo apropiado, como ser FTP (por sus siglas en inglés, File Transfer Protocol). Esto evita la necesidad de contar con un sistema de identificación para cada tupla de datos y que la central deba luego agruparlos, pero requiere de un medio de almacenamiento de mayor capacidad en el datalogger para acumular varios registros.

Se supone que los usuarios son los responsables de la correcta utilización de los dataloggers con el fin de aportar información al sistema a cambio de una mejora del servicio de mantenimiento vial e información actualizada del estado de los caminos.

### **11.3.2. Estaciones meteorológicas**

Las precipitaciones afectan en mayor o menor medida al estado de los caminos no pavimentados dependiendo de muchos factores, entre ellos, la frecuencia de circulación y el tipo de vehículos, las propiedades del suelo, el nivel del terreno y la vegetación circundante [68].

En toda zona rural se cuenta con servicios de meteorología, por un lado, debido a que la mayoría de las actividades productivas que se llevan a cabo, se ven afectadas por el clima y contar con este servicio es fundamental para la toma de decisiones. Por otro lado, en las actividades que se realizan en zona rural como por ejemplo la cría de animales a galpón, minería, o el turismo, la producción también se ve afectada por el clima, pero en menor medida y las precipitaciones siguen siendo un factor muy importante ya que condiciona el acceso por medios terrestres a tales sectores.

Por este motivo se asume que siempre hay disponibilidad de estaciones meteorológicas que permitan, al sistema colaborativo propuesto, incluir las precipitaciones al modelo de datos sin que esto signifique un incremento de costo o esfuerzo computacional significativo. Si el servicio meteorológico de la zona no es suficientemente confiable, entonces se deben incluir pluviómetros que reporten datos georeferenciados a fines de que la información del SIG se mantenga actualizada.

### **11.3.3. Dataloggers de maquinaria vial**

Las maquinarias viales tales como motoniveladoras, topadoras, aplanadoras, retroescavadoras, etc., deben disponer de dataloggers que registren las zonas donde trabajan, para informar la labor realizada y opcionalmente parámetros de funcionamiento como combustible, vida útil del lubricante, aceite hidráulico, temperaturas y presiones de trabajo, etc., que permitan optimizar el mantenimiento de la mismas. Si bien estos dataloggers no aportan un gran volumen de información, la incorporación de este rol no implica un costo considerable a la infraestructura del sistema, con lo cual, es recomendable incluirla ya que contribuye al objetivo principal propuesto que es la optimización de la gestión de recursos para el mantenimiento vial.

Se asume que las porciones de camino sobre las que trabajó la maquinaria quedan en condiciones óptimas y serán procesadas por la central etiquetando dichas regiones como camino reparado. Naturalmente, a medida que transcurre el tiempo, el estado del camino se va degradando, por ello, la calificación se debe mantener hasta que circulen vehículos midiendo la condición real de manera que esto permite evaluar también el trabajo de reparación realizado.

### **11.3.4. Sistema de información geográfica**

Se debe contar con un modelo abstracto o SIG (por sus siglas, Sistema de Información Geográfica) para representar un mapa de los caminos y que contemple un esquema de clasificación para el estado de cada sección de los mismos, por ejemplo, empleando una capa temática que utilice un esquema de colores para dicha clasificación. Para garantizar cierta flexibilidad y permitir el acceso público a la información, el modelo debe ser compatible con entornos web como por ejemplo Google Maps ®, OpenStreetMap ® o Bing Maps ®, entre otros.

En el contexto de la red vial Argetina, un espacio digital apropiado para este propósito es el “IDE Transporte”, de la Secretaría de Planificación del Transporte, que según se detalla en su sitio oficial, “El objetivo es el Desarrollo de plataforma informática en red, que permita la publicación de Información Geográfica de Transporte conforme los estándares de interoperabilidad definidos por IDERA, Infraestructura de Datos Espaciales de la República Argentina” [73].

Un modelo dinámico de calles rurales de mayor complejidad podría contemplar en qué medida se ve afectada cada sección de los caminos ante las precipitaciones de manera que se pueda estimar, a partir del milimetraje precipitado, cuál será el estado de transitabilidad luego de la lluvia.

A medida que el sistema acumule datos, los parámetros de estimación pueden ajustarse en base a los nuevos resultados de manera de contar con un sistema de mejora continua de la predicción o aproximación. Los algoritmos específicos para la construcción de este modelo dependerán del software utilizado y deben ejecutarse periódicamente en el servidor que centraliza toda la información del sistema.

### 11.3.5. Modelo de datos y algoritmos

Si el modelo de datos se registra en una base de datos relacional, se requerirán al menos tres tablas. Una para la lista de caminos, otra para las velocidades promedio registradas por los usuarios y una tercera para los registros de precipitaciones.

Inicialmente, el administrador del sistema debe definir los caminos que forman parte de la zona de interés y especificar una serie de parámetros que permiten modelar el cálculo de la calidad de cada camino. Por ejemplo, si en la zona de interés existen caminos con suelos de distintas características, la afectación por parte del tránsito y las lluvias será diferente y la frecuencia de mantenimiento también.

El algoritmo 2 presenta el pseudo código que asigna un índice de calidad a cada camino, según las velocidades registradas y las precipitaciones. La propuesta es simple y puede ser mejorada sustancialmente por medio de técnicas de inteligencia artificial.

La variable  $Camino_i$  es una referencia a la mínima sección de calle que se debe evaluar y  $Caminos$  es el conjunto completo de caminos que forman parte de la zona de interés. Para cada camino, se busca en la base de datos todas las velocidades normalizadas recientes a las que se transitó por dicho lugar y se promedian los valores para obtener  $\bar{vel}$ . Seguido se debe buscar en la base de datos, el milimetraje de lluvia,  $mm_p$ , que se haya registrado en las cercanías de la posición del camino que se está evaluando. La afectación de las presipitaciones al estado del camino, es un parámetro que debe determinarse mediante un modelo adecuado. Finalmente, se calcula el índice de calidad del camino con una función paramétrica  $get\_q$  que toma en cuenta las variables obtenidas anteriormente y que se detalla en la Sección 11.3.6. Finalmente se actualiza el valor  $q$  en la base de datos.

```

for  $Camino_i$  in  $Caminos$  do
     $\bar{vel} \leftarrow DB.get\_avg\_vel(Camino_i)$    $mm_p \leftarrow DB.get\_mm(Camino_i)$ 
     $q \leftarrow get\_q(Camino_i, \bar{vel}, mm_p)$ ;
     $DB.set\_q(Camino_i, q)$ 
end

```

**Algoritmo 2:** Calificación estado del camino

El valor de  $q$  es un estimativo de qué tan buena o mala es la condición de ese camino para poder circular. A partir de este valor se puede establecer un esquema de escalas, “Bueno”, “Regular”, “Malo” por ejemplo, que simplifica interpretación para los usuarios.

La gestión de recursos contempla, al menos en parte, decidir en qué orden reparar los caminos. Para ello se debe tener en cuenta, por un lado, el grado de urgencia con el que hay que acondicionar una calle lo cual dependerá de la calificación reportada por el sistema y por el otro, el lugar en el que se encuentra la maquinaria actualmente, ya que el desplazamiento

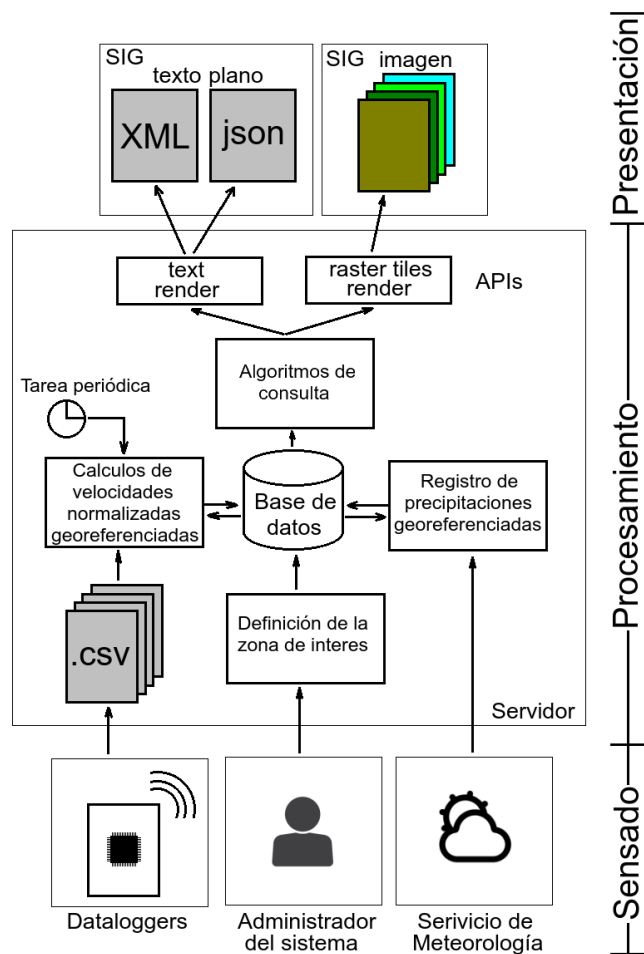


Figura 11.3: Arquitectura del sistema.

de la misma implica un costo considerable. La complejidad del problema dependerá del área total que abarca el sistema y la cantidad de recursos disponibles.

### 11.3.6. Central de procesamiento

La función de la central es procesar la información recolectada por todos los actores de manera que sea posible evaluar la calidad de porción de camino y presentar un informe actualizado regularmente. Consiste en un servidor que es el actor central del sistema. Los recursos computacionales de hardware y de software con los que contará este servidor dependerán de la escala de la red vial, que no solamente estará limitada a una única organización de mantenimiento vial, sino que puede abarcar un gran área de interés y proveer servicio a múltiples zonas.

En la Figura 11.3 se puede visualizar la arquitectura general del sistema donde se detalla la organización de los principales componentes de software mediante un esquema simplificado.

Con la información de posiciones y estampas de tiempo, la central deberá calcular la velocidad normalizada de desplazamiento de los vehículos, relacionar las velocidades con cada sección de los caminos y determinar la calidad de cada una calificando con una escala



numérica. Luego, se supone que los trabajadores viales asignan los recursos para reparar y acondicionar los caminos que se encuentren en peor estado de manera más eficiente, si es que cuentan con la información apropiada.

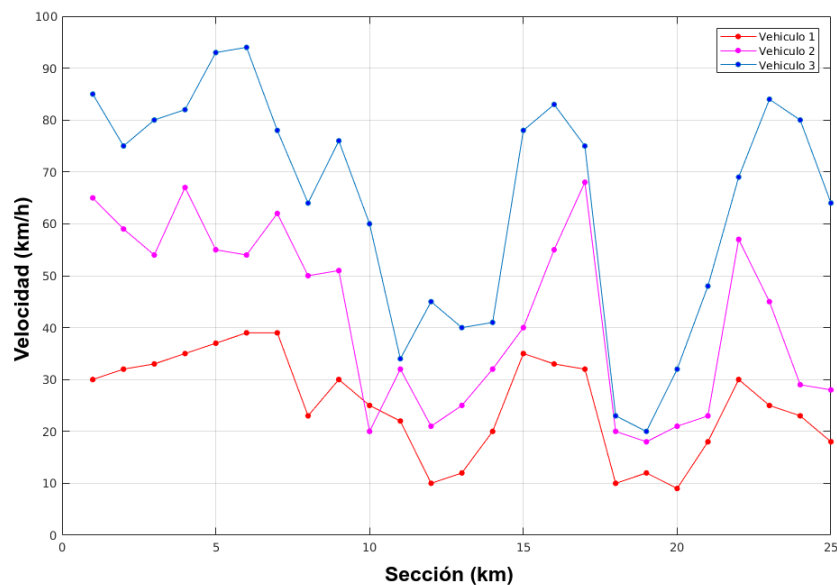


Figura 11.4: Gráfico de velocidades.

La principal carga de operaciones de escritura de datos se deberá al reporte de los dataloggers que transfieren registros de recorridos con posiciones, estampas de tiempo y velocidades. El servidor de la central debe procesar cada registro extrayendo la información útil, para lo cual utilizará un programa que se encargará de procesar los registros enviados por los dataloggers y subir los resultados a la base de datos. El procedimiento de segmentado de este programa consiste en:

- Determinar qué parte del recorrido reportado corresponde a caminos de interés. Debido a que los dataloggers pueden encenderse y medir en zonas que corresponden a propiedad privada por ejemplo.
- Calcular la velocidad de desplazamiento del vehículo en cada punto, si este dato no está presente en el registro.
- Normalizar los valores de velocidad dividiendo cada dato por la velocidad media de desplazamiento de todo el trayecto que corresponde a un camino de interés.
- Mapear el arreglo de velocidades normalizadas a cada sección del trayecto dentro del modelo de caminos.
- Convertir el resultado al formato adecuado y almacenar en la base de datos.

A modo de ejemplo, la Figura 11.4 muestra el gráfico de tres arreglos de datos de velocidades de tres vehículos diferentes, lento, intermedio y rápido, que reducen sus velocidades en

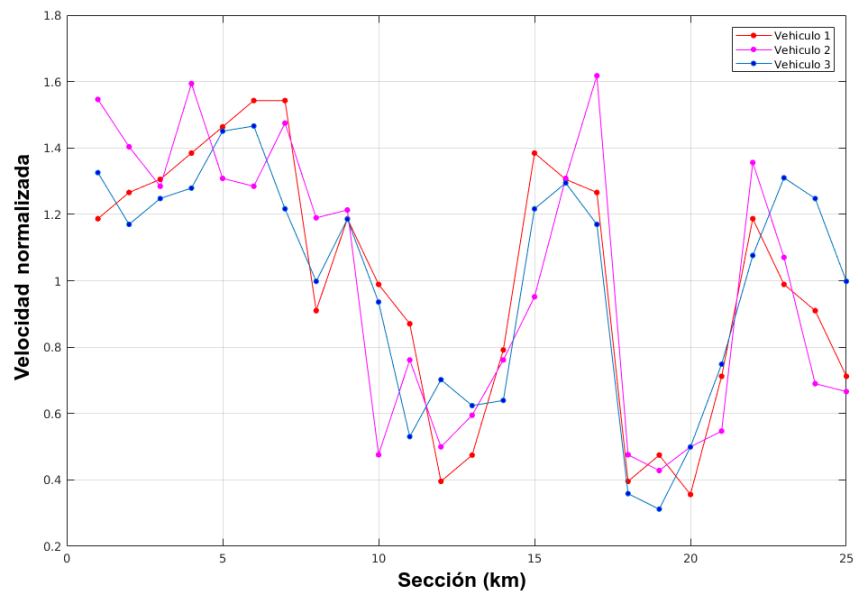


Figura 11.5: Gráfico de velocidades normalizadas.

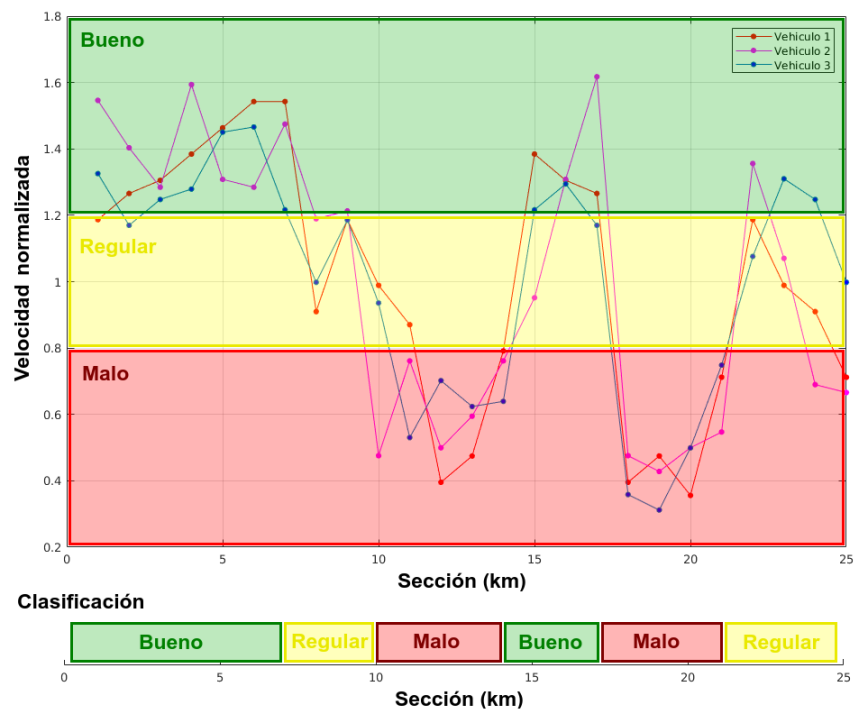


Figura 11.6: Gráfico de velocidades normalizadas y clasificación.

las mismas secciones de la trayectoria. Si se normalizan los valores dividiendo por el promedio de velocidad en cada caso, como muestra la Figura 11.5 se observa que las curvas quedan solapadas aunque las variaciones se definieron con valores aleatorios.

Si se utiliza un esquema de clasificación con escala discreta, se deben definir rangos de velocidades normalizadas para cada escala de clasificación del estado del camino, luego es posible etiquetar a cada dato, con lo cual, se puede determinar la calidad de cada sección del camino en base a la mayor cantidad de datos que contenga con cada etiqueta. Esto se ilustra en el gráfico de la Figura 11.6 donde se eligió un esquema de clasificación de tres escalas “Bueno”, “Regular” y “Malo”, donde la escala “Regular” contempla las velocidades normalizadas que se encuentran comprendidas en el rango 0,8 a 1,2.

De forma periódica, la central debe procesar el modelo de los caminos clasificando las zonas de rápida y lenta circulación y actualizar la capa correspondiente en el modelo del SIG.

Por otro lado, se debe contar con un servicio de suscripción a alertas por lluvias si la región que se ve afectada por precipitaciones contiene alguno de los caminos de interés. Para poder modelar la afectación de las lluvias a la calidad del camino, se debe contar, en principio, con datos de sucesivas mediciones, ya que esto permitiría correlacionar las velocidades de tránsito luego de las precipitaciones con el milimetraje de lluvia. De esta manera es posible contar con una estimación de cómo es el estado de cada sección del camino todo el tiempo, información que no solamente es útil para la gestión de recursos de mantenimiento vial, sino también que puede ser utilizada por todos los usuarios de la red vial que planean circular por la misma.

El costo del sistema incluye principalmente el mantenimiento de un servicio de cómputo y almacenamiento en la nube, el desarrollo y mantenimiento de una base de datos y algoritmos para la generación del SIG en tiempo real.

## 11.4. Discusión

La realización del sistema requiere de la participación ciudadana para que genere un beneficio. En los municipios o departamentos del interior, la actividad económica está vinculada a las tareas agropecuarias, mineras, turísticas y de otros tipos que se desarrollan en zona rural. Estas actividades involucran el transporte de materiales en vehículos de porte medio y grande por las dimensiones y el volumen de las cargas. Como ejemplos se puede mencionar el transporte y almacenamiento de granos, forrajes e insumos varios para éstas actividades, transporte de ganado vacuno, porcino, ovino, equino, etc. maquinaria de labor agrícola, minera u otras. Todos estos vehículos tienen al menos una última milla en caminos vecinales cuya calidad resulta muy importante para garantizar un tránsito eficiente.

Entre las aplicaciones más intensivas se puede mencionar a la producción láctea. Mientras la producción se centra en los tambos, se realiza transporte diario a las empresas de acopio o procesamiento y distribución. El movimiento de los transportes de carga que recojen diariamente la leche y la transportan a las empresas requiere de caminos transitables, a tal punto que el estado de las carreteras por las que se mueven las cargas puede influir en el costo de producción y valor del producto.

Estos caminos vecinales en ocasiones tienen jurisdicciones mixtas ya que atraviesan di-

versos municipios. El mantenimiento, entonces, resulta de interés tanto para los productores como para quienes negocian los productos y las autoridades intervinientes que deben garantizar la transitabilidad de las vías.

Los caminos además son la vía de comunicación para que las familias rurales, aquellas encargadas de cuidar y trabajar el campo (puesteros) puedan llegar a las ciudades por compras o atención médica como así también para acercar a los niños a las escuelas rurales.

Surge entonces que es posible instalar nodos colectores de la información proveniente de los dispositivos móviles que sensaron el estado del camino en las escuelas rurales, las estaciones de servicio más utilizadas por la población rural, posiblemente en las empresas productoras o acopiadoras de granos o estaciones de ferrocarril en donde se realiza el traspaso de cargas. De esta forma una estructura de actualización más dinámica con puntos de acceso al procesamiento más próximo es posible permitiendo a las autoridades realizar un orden adecuado para el establecimiento de las tareas de mantenimiento, permitiendo a la vez que los usuarios cuenten con una información más amplia que puede ser utilizada a la hora de decidir por qué caminos movilizarse.

## 11.5. Consideraciones finales

En este capítulo se detalló la infraestructura de una plataforma colaborativa que brinda información en tiempo real para complementar las tareas de relevamiento del estado de los caminos no pavimentados de la zona rural.

A diferencia de otras propuestas, este sistema no requiere de sensores específicos ni de instrumentos costosos y se puede lograr la puesta en marcha en plazos razonables. Al tratarse de un sistema basado en crowdsensing el costo de adquisición de la información es bajo pero se requiere que se garantice un buen servicio para mantener el incentivo a la participación de muchos usuarios. Cuanto mayor es el volumen de datos del que se disponga, más precisa y actualizada será la información que genere el sistema.

Como trabajo futuro se avanza con la realización de pruebas experimentales, validación de la hipótesis y propuesta a las autoridades viales la implementación del sistema en una región piloto para su evaluación.

Aquí finaliza el último de los capítulos en los que se presentan resultados dentro de este compendio que abarca cuestiones puntuales de los sistemas computacionales de soporte a la producción agrícola. En lo que resta de la tesis se exponen las conclusiones generales y futuros pasos.

# Parte VI

## Conclusión

## Capítulo 12

# Conclusiones y trabajo futuro

El sector agropecuario del SO bonaerense subsiste ante la competencia que genera el mercado de productos agrícolas con otras regiones del país que obtienen mayores volúmenes de producción por unidad de superficie gracias a la introducción de prácticas eficientes y sustentables, entre ellas, la de conservación del suelo. El productor agropecuario es consciente de la importancia de una buena toma de decisiones para lo cual requiere de una permanente capacitación y por supuesto, la incorporación de tecnología y es por ello que existe una demanda constante de soluciones tecnológicas en la región SO de la provincia de Buenos Aires.

Durante el desarrollo de esta tesis se analizaron algunos problemas puntuales referidos a los sistemas de adquisición, transmisión y procesamiento de datos. Se inició con el desarrollo de sistemas embebidos y aplicaciones móviles como sistemas computacionales para la adquisición de datos y sensado de variables. Se presentaron dos instrumentos basados en sistemas embebidos para brindar apoyo a las tareas de investigación agrícola y dos aplicaciones móviles para el soporte de cálculos de siembra y pulverización. Estas últimas aplicaciones apuntan a un grupo de usuario compuesto por productores, contratistas y profesionales del sector agropecuario.

Referido a la transmisión de los datos, se aplicaron técnicas de optimización numérica con el fin de mejorar algunos aspectos de diseño de las redes, como ser la distribución espacial y el rendimiento en tiempo de las mismas. Por último se aplicaron distintos métodos de procesamiento de la información para el modelado de sistemas complejos, como por ejemplo la dinámica del agua para predecir humedad del suelo en un sistema de riego automático y por otro lado el análisis de imágenes tridimensionales de la cobertura vegetal, con el fin de determinar sus propiedades e influencia de su estructura en la conservación del suelo durante el período de barbecho.

Los métodos de optimización numérica aplicados al diseño de sistemas, en general permiten explorar el espacio de soluciones posibles de una manera práctica y es una buena alternativa al cálculo determinístico del diseño óptimo, ya que éste último implica a veces analizar modelos muy complejos. Por este motivo, los métodos de optimización se aplican continuamente a los sistemas de inteligencia artificial y aprendizaje de máquina, donde se crea un entorno simulado en el cual se ajusta un modelo flexible en función de maximizar uno o más beneficios y/o minimizar uno o más costos. En algunos casos, no se busca un diseño óptimo dentro del universo de posibilidades, sino hallar un diseño al menos factible

o logable, para lo cual se plantean las reglas o requisitos que debe cumplir el sistema y mediante una función de proximidad a este objetivo, se realiza una búsqueda automática de dicha solución. Mediante este paradigma de desarrollo se logró resolver distintos problemas, desde la planificación de tareas de tiempo real en arquitecturas multicore, al cálculo de la trama de comunicación óptima en redes LPWN, hasta la distribución de nodos para brindar cobertura de red en espacios que no disponen de conectividad inalámbrica.

Finalmente, resta mencionar algunos de los aspectos explorados en el marco de los sistemas colaborativos. Estos sistemas requieren de una participación activa de varias personas que, mediante una infraestructura de cómputo y comunicación, colaboran en pos de un objetivo común logrando un mutuo beneficio mediante el intercambio de información. La primera propuesta, que involucra un sistema con soporte LPWAN, aborda las problemáticas relacionadas con el desafío que implica la coordinación entre actores compuestos por dispositivos de IoT para llevar a cabo una comunicación de tiempo real mediante un canal muy limitado, tanto en ancho de banda como en alcance de señal. En este trabajo se demostró, que más allá de las dificultades técnicas que significa la implementación de una red de tales características, existen cuestiones a resolver relacionadas con el comportamiento humano y factores que introducen el hecho de operar un sistema en condiciones tan críticas como los estados de emergencia producto de un desastre o catástrofe.

Por otro lado existen diversas aplicaciones de tipo colaborativas orientadas a ofrecer soluciones al sector agrícola y en este sentido se desarrollaron las bases para el diseño de un sistema que apunta a lograr una mejora en la eficiencia del mantenimiento de las vías de transporte en la zona rural, cuyo estado de transitabilidad es un importante factor que influye también en el precio de los productos agrícolas, por estar directamente vinculado a las cuestiones de logística. Se buscará validar la hipótesis sobre la que se planteó el desarrollo y también implementar una prueba piloto, debido a que la idea del proyecto es factible y logable en un plazo razonable.

En cualquier ámbito existe cierta brecha de conocimiento entre las distintas disciplinas y que puede verse acentuada por muchos aspectos que van desde las necesidades del sector productivo, el contexto socio-económico, la formación académica de los profesionales, la distribución geográfica de asentamientos, la organización de las instituciones educativas o de investigación, entre muchas otras razones. Por este motivo, articular actividades para el desarrollo tecnológico, como parte de los objetivos de esta tesis, fue un gran desafío para todos los investigadores que formaron parte de este trabajo.

Dentro de las problemáticas que fueron tratadas en el marco de esta tesis, se observa que aquellas que aportan un gran beneficio y que más interés despiertan en el sector agro-productivo, son las relacionadas con la extracción y procesamiento de la información. La tecnología con la que se cuenta hoy en día permite, en mayor o menor medida, adquirir y transmitir un gran volumen de datos sin demasiadas dificultades técnicas, pero es a través de estos datos que es posible extraer información de gran valor que puede ser aplicado en pos de lograr una mejor toma de decisiones.

En este sentido, a modo de trabajo futuro se buscará avanzar en la exploración de técnicas y métodos de procesamiento de la información intentando incorporar sistemas basados en inteligencia artificial que apunten a lograr una mejora en la toma de decisiones que realizan los productores, profesionales, asesores e investigadores, por mencionar algunos de los principales actores del ámbito agropecuario.

# Bibliografía

- [1] Max Roser. Employment in agriculture. *Our World in Data*, 2013. <https://ourworldindata.org/employment-in-agriculture>.
- [2] Hannah Ritchie and Max Roser. Land use. *Our World in Data*, 2013. <https://ourworldindata.org/land-use>.
- [3] Fernando López. *Dinámica de la cobertura de residuos en suelos bajo siembra directa: relación con la humedad del suelo y la transformación de las fracciones orgánicas lábiles*. PhD thesis, Universidad Nacional del Sur, 2017.
- [4] Sabine Grunwald. Fusion of soil and remote sensing data to model soil properties. *Advances in Agronomy*, 131:1–109, 01 2015.
- [5] K.H. Kim. Challenges and future directions of cyber-physical system software. In *Computer Software and Applications Conference (COMPSAC), 2010 IEEE 34th Annual*, pages 10 –13, july 2010.
- [6] Manfred Broy. Requirements engineering for embedded systems.
- [7] Edward A. Lee. Cyber physical systems: Design challenges. In *Proceedings of the 2008 11th IEEE Symposium on Object Oriented Real-Time Distributed Computing*, pages 363–369, Washington, DC, USA, 2008. IEEE Computer Society.
- [8] John A. Stankovic, Insup Lee, Aloysius Mok, and Raj Rajkumar. Opportunities and obligations for physical computing systems. *Computer*, 38:23–31, November 2005.
- [9] Unified modeling language. <http://www.uml.org>, Last visited 20th October 2014.
- [10] Daniel Riesco, Paola Martellotto, and German Montejano. Uml and the unified process. chapter Extension to UML Using Stereotypes, pages 273–293. IGI Global, Hershey, PA, USA, 2003.
- [11] Alistair Cockburn. Structuring use cases with goals. *Journal of Object-Oriented Programming*, Sep-Oct, Nov-Dec 1997.
- [12] Ivar Jacobson and Pan-Wei Ng. *Aspect-Oriented Software Development with Use Cases*. Addison-Wesley Professional, 2004.



- [13] Kexing Rui and Greg Butler. Refactoring use case models: The metamodel. In *Proceedings of the 26th Australasian Computer Science Conference - Volume 16*, ACSC '03, pages 301–308, Darlinghurst, Australia, Australia, 2003. Australian Computer Society, Inc.
- [14] J. Xu, W. Yu, K. Rui, and G. Butler. Use case refactoring: a tool and a case study. In *11th Asia-Pacific Software Engineering Conference*, pages 484–491, Nov 2004.
- [15] Claudia Marcos, Alejandro Rago, and Jorge Andres Diaz Pace. Improving use case specifications by means of refactoring. *IEEE Latin America Transactions*, 13(4):1135–1140, 2015.
- [16] Miguel A. Teruel, Elena Navarro, Víctor López-Jaquero, Francisco Montero, and Pascual Gonzalez. An empirical evaluation of requirement engineering techniques for collaborative systems. Technical Report DIAB-11-01-1, Universidad de Castilla - La Mancha, 2011.
- [17] Jesús M. Almendros-Jiménez and Luis Iribarne. Describing use-case relationships with sequence diagrams. *Comput. J.*, 50(1):116–128, January 2007.
- [18] David D. Zhang. Use case modeling for real-time application. In *WORDS '99: Proceedings of the Fourth International Workshop on Object-Oriented Real-Time Dependable Systems*, Washington, DC, USA, 1999. IEEE Computer Society.
- [19] Mohamed Kimour and Djamel Meslati. Deriving objects from use cases in real-time embedded systems. *Information and Software Technology*, 47(8):533 – 541, 2005.
- [20] Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino, and David Mellis. *Proyecto Arduino*, 2005 (accedido 17 de julio de 2020).
- [21] Eben Upton. *Raspberry*, 2006 (accedido 17 de julio de 2020).
- [22] Texas Instruments. *Beaglebone*, 2011 (accedido 17 de julio de 2020).
- [23] Ricardo E. Garbers and Yi Erh Chen. Costos operativos de maquinaria agrícola, síntesis básica para el cálculo.
- [24] Terry Kastens. Farm machinery operation cost calculations.
- [25] Leonardo Venturellia, Andrés Moltoni, Silvina Esparza Ignacio Huerga Gabriela Gior-dano, Pedro Valderrama, and Juan Regazzoni. Tracción de implementos en la agricultura familiar. desarrollo de instrumental para medir fuerza de tiro y potencia.
- [26] Amalia González Cadenas. Diseño y construcción de un medidor electrónico de consumo de combustible para tractores agrícolas.
- [27] *Chango, un microtractor diseñado para la agricultura familiar*, 2019. <https://intainforma.inta.gob.ar/chango-un-microtractor-disenado-para-la-agricultura-familiar/>.

- [28] G.B. Swella, P.R. Ward, K.H.M. Siddique, and K.C. Flower. Combinations of tall standing and horizontal residue affect soil water dynamics in rainfed conservation agriculture systems. *Soil and Tillage Research*, 147:30 – 38, 2015.
- [29] RepRap Project - Project Description. <http://reprap.org/>. Último acceso: 2018-11-15.
- [30] D. Villarroel, F. Scaramuzza, J. P. Vélez, M. Bianco Gaido, and L. Cuevas. Gestión remota de datos a partir de aplicaciones y plataformas en el nuevo contexto de la agricultura digital. *Equipo Técnico, Agricultura de Precisión EEA INTA Manfredi*.
- [31] J. Izquierdo and M. Rodríguez Fazzzone. Buenas prácticas agrícolas: En busca de la sostenibilidad, competitividad y seguridad alimentaria. 2006.
- [32] S. A. Suarez and A. M. Suarez. The impact of mobile phone apps in the agricultural production. *DAAAM International Scientific Book*, 35:629–636, 2013.
- [33] González Tena, P. A.; Medel, R. R., D. M. Sangerman-Jarquín, Cruz Castillo, and J. J. G.; Díaz. Extensionismo agrícola en el uso de tecnologías de la información y comunicación (tic) en chiapas y oaxaca. *Revista Mexicana de Ciencias Agrícolas*, 6(1):175–186, 2015.
- [34] International Organization for Standardization. 5 equipment for crop protection — sprayer nozzles — colour coding for identification. *ISO*, 1062, 2005.
- [35] Onorato Agustín and O. Tesouro Mario. Pulverizaciones agrícolas terrestres. 2006.
- [36] P. D. Leiva. Protocolo de calibración del equipo pulverizador terrestre. 2009.
- [37] Android Studio. <https://developer.android.com/studio>. Último acceso: 2020-09-10.
- [38] AngularJS. <https://angularjs.org/>. Último acceso: 2020-08-15.
- [39] Framework7. <https://framework7.io/>. Último acceso: 2020-08-15.
- [40] J. A. Stankovic. Misconceptions about real-time computing. *IEEE Computer*, 21(17):10–19, 1988.
- [41] I. Méndez-Díaz, J. Orozco, R. Santos, and P. Zabala. Energy-aware scheduling mandatory/optional tasks in multicore real-time systems. *International Transactions in Operational Research*, 24(1-2):173–198, 2017.
- [42] Johan Pouwelse, Koen Langendoen, and Henk Sips. Dynamic voltage scaling on a low-power microprocessor. In *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking*, MobiCom '01, pages 251–259, New York, NY, USA, 2001. ACM.
- [43] M. Pedersen. *Tuning & Simplifying Heuristical Optimization*. Phd. thesis, University of Southampton, 6 2010.

- [44] Enrico Bini and Giorgio C. Buttazzo. Measuring the performance of schedulability tests. *Real-Time Syst.*, 30(1-2):129–154, May 2005.
- [45] P. Gai, M. Di Natale, G. Lipari, A. Ferrari, C. Gabellini, and P. Marceca. A comparison of mpcp and msrp when sharing resources in the janus multiple-processor on a chip platform. In *Real-Time and Embedded Technology and Applications Symposium, 2003. Proceedings. The 9th IEEE*, pages 189–198, 2003.
- [46] Hakan Aydin, Rami Melhem, Daniel Mosseé, and Pedro Mejía-Alvarez. Optimal reward-based scheduling for periodic real-time tasks. *IEEE Trans. Comput.*, 50(2):111–130, February 2001.
- [47] Y. Guan, C. C. Shen, and J. Yackoski. Mac scheduling for high throughput underwater acoustic networks. *IEEE Wireless Communications and Networking Conference*, pages 197–202, 2011.
- [48] S.N. Sivanandam and S. Deepa. *Introduction to Genetic Algorithm*. Springer Science & Business Media, 2007.
- [49] SimPy-Development-Team. Simpy: discrete-event simulation in python. <https://simpy.readthedocs.io/>. Accessed: 2017-05-10.
- [50] Georgios A. Kakamoukas, Panagiotis G. Sarigiannidis, and Anastasios A. Economides. Fanets in agriculture - a routing protocol survey. *Internet of Things*, page 100183, 2020.
- [51] R. M. Santos, J. Orozco, S. F. Ochoa, R. Meseguer, and D. Mosse. Providing real-time message delivery on opportunistic networks. *IEEE Access*, 6:40696–40712, 2018.
- [52] Joaquin Toro, Fernanda Senra de Moura, Frederico Ferreira Pedroso, and Felissa Marques. Avaliacao de perdas e danos: Inundacoes e deslizamentos na regio serrana do rio de janeiro. Technical report, World Bank, 2012.
- [53] Fernando Muñoz Arboleda. Importancia del agua en la nutrición de los cultivos. *Carta Trimestral. Cenica na v.*, 31(3):16–18, 2009.
- [54] Marie France Leroux. *Design of an Automated Irrigation System*. McGill University, Dept. Of Bioresource Engineering. [online].
- [55] Edward C. Martin. *Métodos para medir la humedad del suelo para la programación del riego*. Extensión Cooperativa de La Universidad de Arizona . [online].
- [56] M. J. E. Golay Abraham Savitzky. Smoothing and differentiation of data by simplified least squares procedures. *Anal. Chem.*, 36(8):1627–1639, 1964.
- [57] Raymond P. Canale Steven C. Chapra. Método de integración de euler-cauchy. In *Métodos Numéricos para Ingeniería 5ta Ed*, page 720. ISBN-13:978-970-10-6114-5.
- [58] J. A. Nelder R. Mead. A simplex method for function minimization. *The Computer Journal*, 7(4):308–313, January 1965.

- [59] Lotfollah Abdollahi and Lars Juhl Munkholm. Tillage system and cover crop effects on soil quality: I. chemical, mechanical, and biological properties. *Soil Science Society of America Journal*, 2013.
- [60] Lilian Wanjiru Mbuthia, Veronica Acosta-Martínez, Jennifer DeBruyn, Sean Schaeffer, Donald Tyler, Evah Odoi, Molefi Mpheshea, Forbes Walker, and Neal Eash. Long term tillage, cover crop, and fertilization effects on microbial community structure, activity: Implications for soil quality. *Soil Biology and Biochemistry*, 89:24 – 34, 2015.
- [61] Fernando López, Matías Duval, Juan Manuel Martínez, and Juan Alberto Galantini. Cobertura en el sudoeste bonaerense en suelos bajo siembra directa. *Ciencia del Suelo*, 33:273–281, 2015.
- [62] R. Agamennoni, JI Vanzolini, O Reinoso, and A. Logiúdice. Labranzas en el sur de Buenos Aires: I. Efecto en la cobertura del suelo. *XIX Congreso Latinoamericano de la ciencia del suelo. XXIII Congreso Argentino de la ciencia del suelo*, 2012.
- [63] L. Zubiaga, J.I. Vanzolini, L.G. Dunel, R. Storniolo, D.A. Ombrosi, M.A. Cantamutto, and J.A. Galantini. Efecto de la arquitectura del rastrojo de cultivos de invierno bajo labranza cero. In Asociación Argentina de Ciencia del Suelo, editor, *XXVII Congreso Argentino de la Ciencia del Suelo*, 2020.
- [64] Todorka Glushkova and Stanimir Stoyanov. Ambient-oriented modeling of intelligent context-aware systems. *Bulgarian Comput. Sci. Commun*, 7(1):53–61, 2018.
- [65] Sergio F. Ochoa and Rodrigo Santos. Human-centric wireless sensor networks to improve information availability during urban search and rescue activities. *Information Fusion*, 22:71 – 84, 2015.
- [66] A. Monares, S.F. Ochoa, V. Herskovic, R. Santos, and J.A. Pino. Modeling interactions in human-centric wireless sensor networks. pages 661–666, 2014. cited By 8.
- [67] Ministerio de Transporte. Datos transporte argentina. <https://datos.transporte.gob.ar/>. en línea; accedido 10 de junio de 2020.
- [68] G. I. Lequizamón. *Logística y accesibilidad en caminos rurales análisis del anegamiento en el sudeste de la provincia de Buenos Aires*. Civil engineering thesis, Unidad de Enseñanza Universitaria de Quequén, UNICEN, Tandil, Argentina, 2019.
- [69] J. N. Cardenas Robles. *Estudio Comparativo de Metodologías de Relevamiento de Fallas en Caminos No Pavimentados*. Civil engineering thesis, Facultad de Ingeniería, Escuela Profesional de Ingeniería Civil, Universidad Ricardo Palma, Lima, Peru, 2012.
- [70] J. Castagnino, L. Castagnino, and C. Zanini. Guía de mantenimiento de caminos rurales, 2018.
- [71] Javier Miranda, Niko Mäkitalo, Jose Garcia-Alonso, Javier Berrocal, Tommi Mikkonen, Carlos Canal, and Juan M Murillo. From the internet of things to the internet of people. *IEEE Internet Computing*, 19(2):40–47, 2015.

- [72] C. Wang, Z. Xie, L. Shao, Z. Zhang, and M. Zhou. Estimating travel speed of a road section through sparse crowdsensing data. *IEEE Transactions on Intelligent Transportation Systems*, 20(9):3486–3495, 2019.
- [73] Ministerio de Interior. Resolución 792/14 del ministerio del interior y transporte. dirección de observatorio, estudios y sistemas, de la subsecretaria de coordinación y planificación del transporte dependiente de la secretaría de planificación del transporte.



# Apéndice A

## Algoritmos de optimización

### A.1. Optimización por enjambre de partículas

```
q ← -∞ // Óptimo global
forall partícula i-ésima do // Para cada partícula
    W1,τ,ρ ~ U(0, M) // Posición inicial aleatoria
    if U(0, 1) < Qr then // Con probabilidad Qr, hacer:
        | W2,τ,ρ ~ U(0, F) // Asignar modo de energía aleatorio
    else
        | W2,τ,ρ = 0 // Asignar máxima frecuencia (probabilidad 1 - Qr)
    end
    if Fobj(xi) > -∞ then // Si la partícula actual es solución factible
        | xi ← W // Agregar a la población inicial
        if Fobj(xi) > g then // Si la calidad supera al óptimo
            | g ← xi // Recordar mejor posición del enjambre
            | q ← Fobj(xi) // Recordar el valor óptimo del enjambre
        end
    end
end
end
while no se verifica el criterio de terminación do
    for partícula i-ésima do
        rg ~ U(0, 1) // Valor aleatorio entre 0 y 1
        vi,τ,ρ,1 ← ωcvi,τ,ρ,1 + φcrg(gi,τ,ρ,1 - xi,τ,ρ,1) // Actualizar velocidad
        vi,τ,ρ,2 ← ωfvi,τ,ρ,2 + φfrg(gi,τ,ρ,2 - xi,τ,ρ,2) // Actualizar velocidad
        xi ← xi + vi // Actualizar posición
        if Fobj(xi) > fg then // Actualizar óptimo
            | g ← xi
            | q ← Fobj(xi)
        end
    end
end
end
```

Algoritmo 3: Algoritmo PSO

## A.2. Algoritmos genéticos

**Function** *init*():

```

for  $i \leftarrow 1..N_{cr}$  do // Para cada cromosoma
     $V(\tau, \rho) \sim U(0, M)$  // Asignar genes inicialmente aleatorios
    if  $U(0, 1) < Q_r$  then // Con probabilidad  $Q_r$ , hacer:
         $W(\tau, \rho) \sim U(0, F)$  // Modo de potencia aleatorio
    else
         $W(\tau, \rho) = 0$  // Modo de potencia máximo (con probabilidad  $1 - Q_r$ )
    end
    if  $F_{obj}(x_i) > -\infty$  then // Si el cromosoma es factible,
         $x_i \leftarrow (V, W)$  // Agregar a la población inicial
    end
end

```

**Algoritmo 4:** Método GA: Inicialización.

**Function** *selection*():

```

for  $i \leftarrow 1..N_{cr}$  do
     $q_i \leftarrow F_{obj}(x_i)$  // Determinar la calidad de cada cromosoma
end
Sort( $x_i, f_{i_i}, i \leftarrow 1..N_{cr}$ ) // Ordenar cromosomas según calidad
for  $i \leftarrow 1..\frac{N_{cr}}{2}$  do // Obtener descendencia de la mejor mitad de la población
     $\bar{q}_i \leftarrow \sum_{i=0}^{\frac{N_{cr}}{2}-1} q_i$   $osg_i \leftarrow \|\frac{q_i}{\bar{q}_i}\|$ 
end
// Duplicar la lista de cromosomas de acuerdo al número de descendencia
de cada uno
 $c \leftarrow 0, idx \leftarrow 0$ 
for  $i \leftarrow \frac{N_{cr}}{2}..N_{cr}$  do
    if  $osg_{idx} > c$  then
         $x_i \leftarrow x_{idx}$   $c \leftarrow c + 1$ 
    end
     $c \leftarrow 0$   $idx \leftarrow idx + 1$ 
end
Blend( $x_i, i \leftarrow \frac{N_{cr}}{2}..N_{cr}$ ) // Desordenar la lista de cromosomas

```

**Algoritmo 5:** Método GA: Selección



**Function** *crossover()*:

```

    for  $i \leftarrow \frac{N_{cr}}{2} : 2 : N_{cr}$  do // Alternar genes de los cromosomas de acuerdo a la
        probabilidad de entrecruzamiento
            for  $\tau \leftarrow 0..N$  do
                if  $U(0,1) < Q_{core}$  then
                    | Swap( $x_{i,\tau,1}, x_{i+1,\tau,1}$ )
                end
                if  $U(0,1) < Q_{freq}$  then
                    | Swap( $x_{i,\tau,2}, x_{i+1,\tau,2}$ )
                end
            end
        end
    end
end

```

**Algoritmo 6:** Método GA: Entrecruzamiento

**Function** *mutation()*:

```

    for  $i \leftarrow \frac{N_{cr}}{2}..N_{cr}$  do
        for  $j \leftarrow 0..N$  do
            for  $k \leftarrow 0 : \frac{H}{P_j}$  do
                // Realizar alteraciones aleatorias con probabilidades  $Q_{core}$ 
                y  $Q_{freq}$ 
                if  $U(0,1) < Q_{core}$  then
                    |  $x_{i,j,k,1} \sim U(0, M)$ 
                end
                if  $U(0,1) < Q_{freq}$  then
                    |  $x_{i,j,k,2} \sim U(0, F)$ 
                end
            end
        end
    end
end

```

**Algoritmo 7:** Método GA: Mutación.

*init()*

**while** *No se verifica el criterio de terminación* **do**

```

    selection()
    crossover()
    mutation()

```

**end**

**Algoritmo 8:** Método GA: Iteración