



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

SISTEMA DE GENERACIÓN DE DOSEL URBANO PARA SIMULACIÓN
ATMOSFÉRICA WRF A PARTIR DE DATOS OBTENIDOS DE OPENSTREETMAP

TESIS PARA OPTAR AL GRADO DE
MAGÍSTER EN CIENCIAS, MENCIÓN COMPUTACIÓN

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL EN COMPUTACIÓN

MATÍAS ALEJANDRO BRAVO VALENZUELA

PROFESORES GUÍA:
NANCY HITSCHFELD KAHLER
LAURA GALLARDO KLENNER

MIEMBROS DE LA COMISIÓN:
AIDAN HOGAN
PATRICIO POBLETE OLIVARES
ANDREA RODRIGUEZ TASTETS

SANTIAGO DE CHILE
2016

RESUMEN DE LA TESIS PARA OPTAR
AL GRADO DE MAGÍSTER EN CIENCIAS, MENCIÓN COMPUTACIÓN
POR: MATÍAS ALEJANDRO BRAVO VALENZUELA
FECHA: 2016
PROF. GUÍA: NANCY HITSCHFELD KAHLER, LAURA GALLARDO KLENNER

SISTEMA DE GENERACIÓN DE DOSEL URBANO PARA SIMULACIÓN ATMOSFÉRICA WRF A PARTIR DE DATOS OBTENIDOS DE OPENSTREETMAP

La existencia de efectos generados por la interacción entre las zonas urbanas y la atmósfera, hace que la urbanización sea un tema de interés. Estos efectos provocan variados fenómenos, como aumento de temperatura y cambios de turbulencias dentro de la urbes, los cuales repercuten de manera tanto directa como indirectamente en la calidad de vida de las personas. Características como la geometría de la ciudad y el cambio en el uso de suelo que genera, son aspectos que provocan ciertamente cambios en los flujos de gases de la atmósfera. El uso de simuladores como el *Weather Research and Forecasting Model* (WRF), intentan reproducir por medio de sus modelos urbanos esta interacción, y de esta forma poder entender y predecir el comportamiento de la atmósfera frente a distintos factores urbanos. Para ello es imperativo definir de la mejor forma la zona urbana en la región, para lo cual se utiliza un dosel urbano que contiene toda la información relevante. Sin embargo, no todas las zonas urbanas poseen la cantidad suficiente de datos disponibles, siendo ésta la principal barrera para realizar estudios. El presente trabajo trata sobre la implementación de un sistema de generación de dosel urbano llamado **CityCanopy**, el cual busca generar de manera automática la información necesaria que modele la zona urbana de estudio, a partir de fuentes de mapas digitales de libre acceso. Para ello se propone un algoritmo capaz de extraer y generar información de ciudades a partir de *OpenStreetMap*. Se diseña e implementa un programa computacional que lleva a cabo la tarea extracción y generación del dosel urbano utilizando como núcleo el algoritmo propuesto. Luego, el programa se pone a prueba con la ciudad de Santiago de Chile y Buenos Aires, observando y verificando los resultados. Se utiliza el dosel generado de la ciudad de Santiago de Chile para incorporarlo en una simulación utilizando el simulador WRF. Los resultados que se obtienen permiten observar los efectos que provoca la incorporación de datos de ciudad. Finalmente se presentan las conclusiones y trabajos propuestos.

Dedicado a Daniela Lorca y Jose Ignacio Bravo.

Agradecimientos

Quiero agradecer primero a la gente mas cercana. A mi polola Daniela Lorca que me ha apoyado en todo mi paso por la Universidad, y a quién dedico todo este esfuerzo por nuestro futuro. A mi hermano José Ignacio Bravo, mi madre Patricia Valenzuela y mi padre Claudio Bravo que me dieron la posibilidad de estudiar, que confiaron en mi en todo momento y que siempre han estado para ayudarme y apoyarme en todo. A los padres de mi polola, Myriam Puebla y Mario Lorca, que en todo momento estuvieron ahí para entregarme su ayuda y apoyo. A mis familiares y amigos que me hacen mi vida mas alegre. A la gente del equipo de básquetbol y su entrenador Julio Figueroa, por darme la oportunidad de entrenar y pasarla bien en tan hermoso deporte.

Quiero también agradecer a mis profesoras guía Nancy Hitschfeld y Laura Gallardo, con quienes trabajé en conjunto para llevar a cabo este trabajo de título, las cuales se preocuparon que no me faltase nada en ningun momento, que me dieron la oportunidad de trabajar y viajar tanto al extranjero como dentro del país, y que me tuvieron paciencia en el transcurso de estos años. Agradecer a la gente del Centro del Clima y la Resiliencia, en especial a Andrea Orfanoz, por enseñarme y ayudarme en todo respecto al área del clima, además de facilitarme su trabajo de tesis para que yo pudiese desarrollar el mío. También a César Ordoñez por ayudarme y facilitarme su trabajo para poder graficar los resultados.

Finalmente agradecer a todos los que no he mencionado y que estuvieron ahí para darme sus energías, muchas gracias.

Agradecimientos especiales

- Center for Climate and Resilience Research (FONDAP15110009)
- Powered@NLHPC: Esta investigación fue parcialmente apoyada por la infraestructura de supercómputo del NLHPC (ECM-02)
- Powered@NLHPC: This research was partially supported by the supercomputing infrastructure of the NLHPC (ECM-02)

Tabla de Contenido

1. Introducción	1
1.1. Conceptos básicos	5
1.2. Objetivo general	6
1.3. Objetivos específicos	6
1.4. Metodología	7
1.5. Contenido del trabajo	8
2. Recursos computacionales	9
2.1. Simulador WRF	9
2.1.1. Implementación computacional	9
2.1.2. Modelos urbanos	11
2.2. Mapas digitales urbanos	13
2.2.1. Google Maps	13
2.2.2. OpenStreetMap	14
2.2.3. CGIAR-CSI	15
2.3. Visualización con OpenGL	17
3. Diseño e implementación	21
3.1. Planificación de requerimientos	21
3.1.1. Requerimientos del Usuario	21
3.1.2. Requerimientos del Sistema	22
3.1.3. Casos de uso	22
3.1.4. Diagrama general	23
3.2. Análisis y selección de recursos disponibles	23
3.2.1. Elección de datos disponibles para generación del dosel	24
3.2.2. Elección del nivel de modelo urbano a utilizar y los datos a generar	25
3.2.3. Elección de lenguaje de programación	26
3.3. Flujo del Sistema	26
3.4. Diseño de Clases	27
3.4.1. Modelo	28
3.4.2. Vista	28
3.4.3. Controlador	29
3.5. Procesos relevantes	29
3.5.1. Procesamiento de datos OSM/XML	29
3.5.2. Dibujo de figuras	34
3.5.3. Acoplamiento de datos a WRF	37

3.6. Implementación final	39
4. Aplicación Sistema Dosel Urbano	45
4.1. Definición	45
4.2. Resultados	46
4.3. Verificación dosel urbano Santiago	53
5. Impacto sobre flujo atmosférico	57
5.1. Contexto	57
5.2. Configuración	58
5.3. Resultados	62
5.3.1. Temperatura	63
5.3.2. Magnitud de Viento	75
5.4. Análisis de resultados	87
6. Conclusión	90
Bibliografía	91
A. Archivos de configuración de WRF	96
A.1. Archivo namelist.wps	96
A.2. Archivo namelist.input	97
B. Series de tiempo	100
B.1. Temperatura	100
B.2. Magnitud de viento	120

Índice de Ilustraciones

1.1. Figura obtenida desde [Hidalgo et al., 2008] para esquematizar algunos efectos que provocan las urbes sobre la atmósfera.	2
2.1. Diagrama de flujo de datos en WRF [wrf, 2010a].	10
2.2. Diagrama WPS [wrf, 2010c].	11
2.3. Diferencias entre distintos niveles [Chen et al., 2011]	12
2.4. Automóvil con cámara incorporada para capturar imágenes posteriormente utilizadas en Street View [goo, 2016b].	13
2.5. Vista ejemplo Street View [goo, 2016a].	14
2.6. Interfaz para extraer datos según coordenadas del área deseada.	14
2.7. Portal CGIAR-CSI para datos 90m resolución [cgi, 2016b].	16
2.8. Etapas en el flujo de procesos en OpenGL.	17
2.9. Esquema de visualización en pantalla del espacio virtual.	18
2.10. Pequeña triangulación de un conjunto de puntos.	18
2.11. Número de triángulos dependiendo del nivel de detalle [Luebke, 2003].	19
2.12. Rasterización de un triángulo [tri,].	19
2.13. Diagrama general del modelo de iluminación de Phong [pho,].	20
3.1. Proceso de desarrollo iterativo del trabajo	21
3.2. Diagrama general de caso de uso.	24
3.3. Diagrama general de flujo de datos.	27
3.4. Diagrama de clases.	27
3.5. Dibujo ejemplo de representación de matriz de NxM. Línea negra representa a la matriz y sus puntos. Línea amarilla discontinua representa el subárea que abarca cada punto.	31
3.6. Dibujo ejemplo donde se muestra el proceso de búsqueda del centroide por elemento (2) y su ubicación espacial dentro de la grilla (3).	32
3.7. Ejemplo de obtención de normales. Flechas en negro representan el sentido horario de los vértices de los triángulos. En rojo las normales de cada triángulo.	35
3.8. Ejemplo de obtención de normales para una cara del paralelepípedo. En negro se representan las aristas que definen al paralelepípedo. De izquierda a derecha en la segunda figura, se puede observar en rojo la orientación de la triangulación por cara. En la tercera figura, se puede observar en rojo las normales de cada triángulo.	36
3.9. Escritura de <i>write_geogrid.c</i> [wps, 2010].	37
3.10. División de matriz [wps, 2010].	38

3.11. Adición de líneas en archivo <i>GEOGRID.tbl</i>	38
3.12. Toma de imágenes de la topografía del terreno obtenido desde dos perspectivas distintas.	40
3.13. Primera figura toma de imagen con un acercamiento a al topografía. La segunda figura muestra el dosel urbano generado sin topografía.	41
3.14. Toma de imágenes del dosel urbano generado sin topografía desde dos perspectivas distintas.	42
3.15. Toma de imágenes de la topografía y el dosel urbano generado desde dos perspectivas distintas.	43
3.16. Toma de imágenes de la topografía y el dosel urbano generado desde dos perspectivas distintas.	44
4.1. OpenStreetMap: Ciudad de Santiago, Chile.	47
4.2. Aplicación del primero (izquierda) y segundo (derecha) criterio sobre Santiago.	47
4.3. Variación de tamaño de grilla en factor $1/2^n$ en Santiago.	48
4.4. Mediciones de tiempo de ejecución del algoritmo sobre Santiago con respecto al ancho de grilla, con eje X en escala logarítmica.	49
4.5. Mediciones de tiempo de ejecución del algoritmo sobre Santiago con respecto al número de puntos generados, con eje X en escala logarítmica.	49
4.6. OpenStreetMap: Ciudad de Buenos Aires, Argentina.	50
4.7. Aplicación del primero (izquierda) y segundo (derecha) criterio sobre Buenos Aires.	50
4.8. Variación de tamaño de grilla en factor $1/2^n$ en Buenos Aires.	51
4.9. Mediciones de tiempo de ejecución del algoritmo sobre Buenos Aires con respecto al ancho de grilla, con eje X en escala logarítmica.	51
4.10. Mediciones de tiempo de ejecución del algoritmo sobre Buenos Aires con respecto al número de puntos generados, con eje X en escala logarítmica.	52
4.11. Dosel urbano generado manualmente.	54
4.12. Dosel urbano generado por CityCanopy utilizando el primer criterio.	55
5.1. Dominios de la zona de estudio.	58
5.2. Acercamiento al tercer dominio de la zona de estudio.	59
5.3. Niveles verticales de la configuración. La primera Figura muestra los 30 niveles verticales, mientras que la segunda realiza un acercamiento de los primeros 10 niveles.	60
5.4. Diferencia entre valores promedio de temperatura de UCM y SLAB, a las 03:00 y 15:00 hrs, a 2m de elevación.	63
5.5. Diferencia entre valores promedio de temperatura de OSMUCM y SLAB, a las 03:00 y 15:00 hrs, a 2m de elevación.	64
5.6. Diferencia entre valores promedio de temperatura de BEP y SLAB, a las 03:00 y 15:00 hrs, a 2m de elevación.	64
5.7. Diferencia entre valores promedio de temperatura de OSMBEP y SLAB, a las 03:00 y 15:00 hrs, a 2m de elevación.	65
5.8. Corte vertical de la diferencia entre valores promedio de temperatura de UCM y SLAB, a las 03:00 y 15:00 hrs.	66
5.9. Corte vertical de la diferencia entre valores promedio de temperatura de OSMUCM y SLAB, a las 03:00 y 15:00 hrs.	67

5.10. Corte vertical de la diferencia entre valores promedio de temperatura de BEP y SLAB, a las 03:00 y 15:00 hrs.	68
5.11. Corte vertical de la diferencia entre valores promedio de temperatura de OSM-BEP y SLAB, a las 03:00 y 15:00 hrs.	69
5.12. Serie de tiempo de variable Temperatura para estación Puente Alto en el periodo entre 23-08-2013 y 29-08-2013.	70
5.13. Serie de tiempo de variable Temperatura para estación Quilicura en el periodo entre 23-08-2013 y 29-08-2013.	71
5.14. Serie de tiempo de variable Temperatura para estación Santiago en el periodo entre 23-08-2013 y 29-08-2013.	72
5.15. Serie de tiempo de variable Temperatura para estación Talagante en el periodo entre 23-08-2013 y 29-08-2013.	73
5.16. Diferencia entre valores promedio de magnitud de viento de UCM y SLAB, a las 03:00 y 15:00 hrs, a 10m de elevación.	75
5.17. Diferencia entre valores promedio de magnitud de viento de OSMUCM y SLAB, a las 03:00 y 15:00 hrs, a 10m de elevación.	76
5.18. Diferencia entre valores promedio de magnitud de viento de BEP y SLAB, a las 03:00 y 15:00 hrs, a 10m de elevación.	76
5.19. Diferencia entre valores promedio de magnitud de viento de OSMBEP y SLAB, a las 03:00 y 15:00 hrs, a 10m de elevación.	77
5.20. Corte vertical de la diferencia entre valores promedio de magnitud de viento de UCM y SLAB, a las 03:00 y 15:00 hrs.	78
5.21. Corte vertical de la diferencia entre valores promedio de magnitud de viento de OSMUCM y SLAB, a las 03:00 y 15:00 hrs.	79
5.22. Corte vertical de la diferencia entre valores promedio de magnitud de viento de BEP y SLAB, a las 03:00 y 15:00 hrs.	80
5.23. Corte vertical de la diferencia entre valores promedio de magnitud de viento de OSMBEP y SLAB, a las 03:00 y 15:00 hrs.	81
5.24. Serie de tiempo de variable Magnitud de Viento para estación Puente Alto en el periodo entre 23-08-2013 y 29-08-2013.	82
5.25. Serie de tiempo de variable Magnitud de Viento para estación Quilicura en el periodo entre 23-08-2013 y 29-08-2013.	83
5.26. Serie de tiempo de variable Magnitud de Viento para estación Santiago en el periodo entre 23-08-2013 y 29-08-2013.	84
5.27. Serie de tiempo de variable Magnitud de Viento para estación Talagante en el periodo entre 23-08-2013 y 29-08-2013.	85
B.1. Serie de tiempo para SLAB de variable Temperatura para estación Puente Alto en el periodo entre 23-08-2013 y 29-08-2013.	100
B.2. Serie de tiempo para SLAB de variable Temperatura para estación Quilicura en el periodo entre 23-08-2013 y 29-08-2013.	101
B.3. Serie de tiempo para SLAB de variable Temperatura para estación Santiago en el periodo entre 23-08-2013 y 29-08-2013.	102
B.4. Serie de tiempo para SLAB de variable Temperatura para estación Talagante en el periodo entre 23-08-2013 y 29-08-2013.	103
B.5. Serie de tiempo para UCM de variable Temperatura para estación Puente Alto en el periodo entre 23-08-2013 y 29-08-2013.	104

B.6. Serie de tiempo para UCM de variable Temperatura para estación Quilicura en el periodo entre 23-08-2013 y 29-08-2013.	105
B.7. Serie de tiempo para UCM de variable Temperatura para estación Santiago en el periodo entre 23-08-2013 y 29-08-2013.	106
B.8. Serie de tiempo para UCM de variable Temperatura para estación Talagante en el periodo entre 23-08-2013 y 29-08-2013.	107
B.9. Serie de tiempo para OSMUCM de variable Temperatura para estación Puente Alto en el periodo entre 23-08-2013 y 29-08-2013.	108
B.10. Serie de tiempo para OSMUCM de variable Temperatura para estación Quilicura en el periodo entre 23-08-2013 y 29-08-2013.	109
B.11. Serie de tiempo para OSMUCM de variable Temperatura para estación Santiago en el periodo entre 23-08-2013 y 29-08-2013.	110
B.12. Serie de tiempo para OSMUCM de variable Temperatura para estación Talagante en el periodo entre 23-08-2013 y 29-08-2013.	111
B.13. Serie de tiempo para BEP de variable Temperatura para estación Puente Alto en el periodo entre 23-08-2013 y 29-08-2013.	112
B.14. Serie de tiempo para BEP de variable Temperatura para estación Quilicura en el periodo entre 23-08-2013 y 29-08-2013.	113
B.15. Serie de tiempo para BEP de variable Temperatura para estación Santiago en el periodo entre 23-08-2013 y 29-08-2013.	114
B.16. Serie de tiempo para BEP de variable Temperatura para estación Talagante en el periodo entre 23-08-2013 y 29-08-2013.	115
B.17. Serie de tiempo para OSMBEP de variable Temperatura para estación Puente Alto en el periodo entre 23-08-2013 y 29-08-2013.	116
B.18. Serie de tiempo para OSMBEP de variable Temperatura para estación Quilicura en el periodo entre 23-08-2013 y 29-08-2013.	117
B.19. Serie de tiempo para OSMBEP de variable Temperatura para estación Santiago en el periodo entre 23-08-2013 y 29-08-2013.	118
B.20. Serie de tiempo para OSMBEP de variable Temperatura para estación Talagante en el periodo entre 23-08-2013 y 29-08-2013.	119
B.21. Serie de tiempo para SLAB de variable Magnitud de Viento para estación Puente Alto en el periodo entre 23-08-2013 y 29-08-2013.	120
B.22. Serie de tiempo para SLAB de variable Magnitud de Viento para estación Quilicura en el periodo entre 23-08-2013 y 29-08-2013.	121
B.23. Serie de tiempo para SLAB de variable Magnitud de Viento para estación Santiago en el periodo entre 23-08-2013 y 29-08-2013.	122
B.24. Serie de tiempo para SLAB de variable Magnitud de Viento para estación Talagante en el periodo entre 23-08-2013 y 29-08-2013.	123
B.25. Serie de tiempo para UCM de variable Magnitud de Viento para estación Puente Alto en el periodo entre 23-08-2013 y 29-08-2013.	124
B.26. Serie de tiempo para UCM de variable Magnitud de Viento para estación Quilicura en el periodo entre 23-08-2013 y 29-08-2013.	125
B.27. Serie de tiempo para UCM de variable Magnitud de Viento para estación Santiago en el periodo entre 23-08-2013 y 29-08-2013.	126
B.28. Serie de tiempo para UCM de variable Magnitud de Viento para estación Talagante en el periodo entre 23-08-2013 y 29-08-2013.	127

B.29.Serie de tiempo para OSMUCM de variable Magnitud de Viento para estación Puente Alto en el periodo entre 23-08-2013 y 29-08-2013.	128
B.30.Serie de tiempo para OSMUCM de variable Magnitud de Viento para estación Quilicura en el periodo entre 23-08-2013 y 29-08-2013.	129
B.31.Serie de tiempo para OSMUCM de variable Magnitud de Viento para estación Santiago en el periodo entre 23-08-2013 y 29-08-2013.	130
B.32.Serie de tiempo para OSMUCM de variable Magnitud de Viento para estación Talagante en el periodo entre 23-08-2013 y 29-08-2013.	131
B.33.Serie de tiempo para BEP de variable Magnitud de Viento para estación Puen- te Alto en el periodo entre 23-08-2013 y 29-08-2013.	132
B.34.Serie de tiempo para BEP de variable Magnitud de Viento para estación Qui- licura en el periodo entre 23-08-2013 y 29-08-2013.	133
B.35.Serie de tiempo para BEP de variable Magnitud de Viento para estación San- tiago en el periodo entre 23-08-2013 y 29-08-2013.	134
B.36.Serie de tiempo para BEP de variable Magnitud de Viento para estación Ta- lagante en el periodo entre 23-08-2013 y 29-08-2013.	135
B.37.Serie de tiempo para OSMBEP de variable Magnitud de Viento para estación Puente Alto en el periodo entre 23-08-2013 y 29-08-2013.	136
B.38.Serie de tiempo para OSMBEP de variable Magnitud de Viento para estación Quilicura en el periodo entre 23-08-2013 y 29-08-2013.	137
B.39.Serie de tiempo para OSMBEP de variable Magnitud de Viento para estación Santiago en el periodo entre 23-08-2013 y 29-08-2013.	138
B.40.Serie de tiempo para OSMBEP de variable Magnitud de Viento para estación Talagante en el periodo entre 23-08-2013 y 29-08-2013.	139

Capítulo 1

Introducción

De acuerdo a [Nations, 2014], globalmente existe una mayor cantidad de personas que reside en zonas urbanas con respecto a zonas rurales, superando un 54 % de la población total hasta el año 2014. Dado los datos que se tienen desde el año 1950, en donde 30 % del mundo se asentaba en zonas urbanas, se proyecta que para el año 2050 la población mundial que vive en zonas urbanas alcance hasta el 66 %. Para Chile la situación se acentúa más, logrando un porcentaje de población urbana de hasta un 87 % para el año 2012 [de Estadísticas INE, 2012]. Este crecimiento conlleva un aumento de la demanda de necesidades dentro de estas zonas, tales como vivienda, transporte y energía, entre otras cosas. Una serie de fenómenos que afectan tanto directa como indirectamente a la atmósfera y el clima local aparecen debido a este crecimiento de las urbes, y de los cuales no se tuvo consideración en el pasado. En [Hidalgo et al., 2008] se realizan la exposición de distintas maneras en la cuales las ciudades influyen en la termodinámica atmosférica y el transporte de contaminantes en el aire. Se muestra que la turbulencia y el intercambio energético de los gases dentro de la atmósfera se ven alterados con respecto a zonas rurales. Aquí se hace una clara mención al fenómeno de calor urbano interno (UHI), recurrente en muchas partes del mundo, en donde se observa un aumento de temperatura de hasta $10^{\circ}C$, bajo ciertas condiciones, al interior de las zonas urbanas con respecto a sus alrededores. Trabajos como [Kalnay and Cai, 2003] miden y observan diversos factores influyentes en el UHI, como cambios en el uso de suelo y densidad demográfica, en donde se relaciona la urbanización y su desarrollo a través del tiempo con los cambios atmosféricos que se presentan. Otras observaciones realizadas en una ciudad de Italia en [Morabito et al., 2016], relacionan los efectos del cambio de uso de suelo, específicamente la aparición de construcciones y viviendas, con el aumento de temperatura dentro de la ciudad. El trabajo permite entender la influencia que tiene este factor en los cambios y efectos que se generan en la atmósfera que rodea a la ciudad. Por otro lado, no sólo la atmósfera y el clima local se ven afectados por la urbanización. En [von Glasow et al., 2013] se exponen algunos de los efectos que provocan megaciudades costeras por sobre el clima y el ecosistema que las rodean. Los contaminantes y su influencia química sobre el agua, el UHI y la variación de la temperatura, y el cambio en la turbulencia de los vientos, afectan en diversas formas el entorno y el ecosistema. Ciertamente, todos estos fenómenos repercuten en variados campos tales como la economía, por ejemplo mayor gasto energético, y la salud y calidad de vida de las personas. Es por esto que toda esta situación debe ser tomada en

consideración y estudiada de cierta forma que se puedan tomar medidas para poder prevenir estos sucesos. Es aquí donde el rol de estudios científicos llegan a ser una herramienta clave para la toma de decisiones.

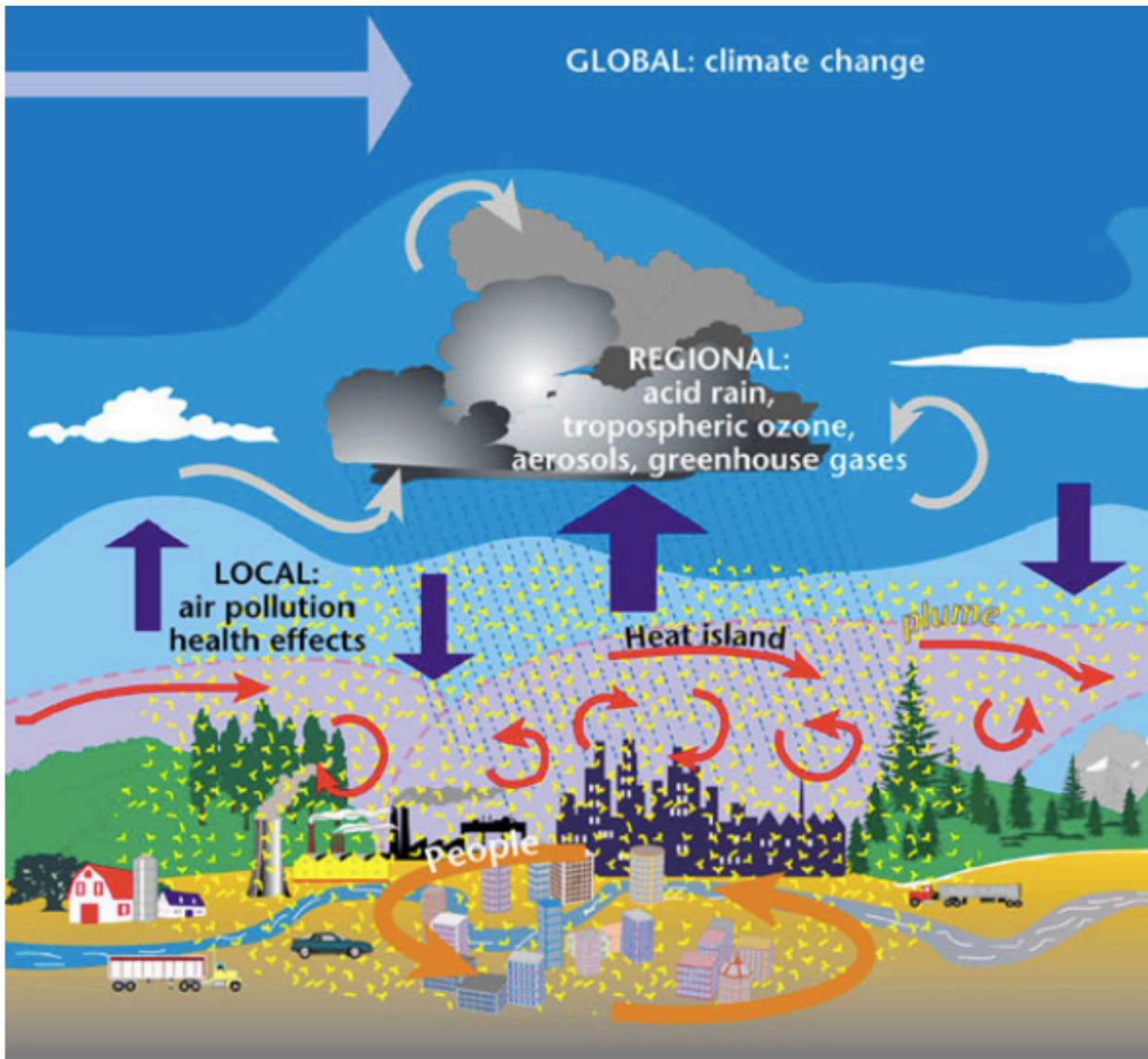


Figura 1.1: Figura obtenida desde [Hidalgo et al., 2008] para esquematizar algunos efectos que provocan las urbes sobre la atmósfera.

Es común que en estudios referentes a la atmósfera se haga uso de simuladores atmosféricos para observar y predecir su comportamiento. Algunos de estos en la actualidad ya cuentan con modelos urbanos acoplados que permiten realizar predicciones del comportamiento de la atmósfera tomando en consideración las zonas urbanas. Muchos de estos simuladores intentan reflejar los efectos del UHI y cambios en la turbulencia, entre otros fenómenos, para poder ajustar los resultados de la simulación. En [Takahashi et al., 2011] se expone un simulador llamado MSSG, acoplado a otro ya existente llamado EARTH SIMULATOR, capaz de tener en consideración modelos de ciudades complejas a muy alta resolución. En [Baklanov et al., 2008], se realiza un acoplamiento urbano al simulador ENVIRO-HIRLAM,

en donde se demuestran los cambios y mejoras que se provocan al adicionar estos fenómenos en el modelo a resolver. Otro acoplamiento urbano se ve en [Chen et al., 2011], donde se muestran los cambios y pruebas realizadas al simulador WRF (*Weather Research and Forecasting Model*) con el fin de verificar su funcionamiento y sus mejoras en las estimaciones.

Si bien se mencionaron distintos modelos, todos requieren ciertos parámetros básicos para poder representar las zonas urbanas. Entre estos parámetros se destaca en primera instancia el uso de suelo urbano. Esta información permite a los distintos modelos urbanos predecir el comportamiento de la atmósfera circundante en las distintas sub zonas, por medio de la clasificación de lugares como zonas residenciales o zonas comerciales, los cuales presentan en la realidad distinto comportamiento como emisión de gases y acumulación de energía calórica. En segunda instancia destaca también la geometría de la ciudad, la cual toma en cuenta valores como alturas de edificios, anchos de calles, disposición de techos, etc. Todo esto genera lo que se denomina "rugosidad", lo cual produce una resistencia a los distintos vientos de la región, provocando efectos de relentización en la magnitud del viento y turbulencia. En [Tursilowati et al., 2012] se muestra la influencia del uso de suelo frente al fenómeno de UHI en Jakarta, Indonesia, utilizando distintos tipos de uso de suelo urbano dentro de simulaciones realizadas con WRF. Aquí, se desprende que la variación entre porcentajes de fracciones de suelo utilizados con construcciones y vegetación, son importantes en la toma de decisiones sobre la planificación futura de la ciudad. En [Chen et al., 2004] se realiza un estudio sobre la zona urbana de Houston Texas, EE.UU., utilizando una de las primeras implementaciones del modelo urbano dentro de WRF. Algunas de las conclusiones relevantes son la importancia en el uso de suelo que se utiliza y la representación geométrica de la ciudad que permite definir los flujos de gases, en donde la rugosidad es un factor bastante influyente en el momento de ver los efectos que provoca la inclusión de una ciudad en la región.

La necesidad de tener datos urbanos de entrada lo más actualizados y robustos posible permite ajustar las simulaciones y de esta forma mejorar los resultados. Trabajos como [Lee et al., 2011], realizan comparaciones entre los resultados obtenidos con anterioridad y nuevas estimaciones con datos generados desde distintas fuentes que antes no se tenían. La adición y utilización de esto permitió ajustar las estimaciones con respecto a las observaciones reales, teniendo por ejemplo mejoras de hasta $1-2^{\circ}C$. Sin embargo, no toda la información necesaria está disponible para todas las zonas urbanas del mundo, y tampoco todas las simulaciones necesitan la misma cantidad información. Por ejemplo, en [Tewari et al., 2006] se realiza una revisión general de la implementación del modelo urbano en WRF, exponiendo algunos parámetros que se utilizan para realizar los cálculos. Datos aparentemente simples como la altura de construcciones, y que son importantes para simular los efectos de rugosidad y resistencia al viento de la ciudad, no siempre se encuentran disponibles para todas las zonas urbanas, y por ende puede ser una situación contraproducente al momento de realizar estudios. Variados trabajos buscan dar solución a esta problemática de disponibilidad de los datos, creando nuevos sistemas y generando diversa información de entrada para los simuladores. En [Kim et al., 2008] se estudia el impacto de utilizar datos satelitales para mejorar la estimación del uso de suelo dentro del simulador WRF, específicamente para utilizarlo con modelos urbanos acoplados. Algo similar se propone en [Burian et al., 2004] con el uso de sistemas aéreos de foto- radares y sistemas láser para generar datos de mayor resolución, como la geometría de la ciudad, e introducirlos en el modelo urbano del simulador WRF. Una propuesta un poco más robusta es [Glotsfelty et al., 2013], basada en la creación de una base de

datos llamada NUDAPT, la cual trata sobre modelos de ciudades mantenidos/actualizados en forma comunitaria creados específicamente para el simulador WRF, construida con información proveniente de agencias y centros de investigación dedicadas a distintas áreas de la ciencia y las matemáticas. Esta base de datos es capaz de reproducir la morfología necesaria de variadas ciudades del mundo para el modelo urbano, es decir, mantiene datos estadísticos que ayudan a crear un modelo probabilístico de la geometría de las ciudades, sin necesidad de mantener una representación exacta. Otro caso parecido es la base de datos expuesta en [González-Aparicio et al., 2010] y utilizada en [Sievinen et al., 2010], donde se intenta reproducir de igual forma la morfología de las ciudades, pero difieren de algunos parámetros necesarios requeridos en otros simuladores como el ENVIRO-HIRLAM. La búsqueda de nuevos métodos y maneras de generar datos siempre es de mucha utilidad, tomando en cuenta que se complementan para generar información más aproximadas de las zonas urbanas.

Como bien se expuso, algunos trabajos proponen el uso de imágenes satelitales y aéreas, provenientes del procesamiento de datos obtenidos de la manipulación de instrumentos más avanzados de detección. Una forma de procesar estos datos se ve en [Gao et al., 2012], en donde a partir de una nube de puntos generada por la utilización sistemas de láser sobre zonas urbanas, se pueden detectar distintos tipos de alturas de murallas, construcciones y hasta árboles. También en [Yu et al., 2011] se puede ver la utilización de la nube de puntos, en donde además de detectar objetos se logran separar los puntos que corresponden a un elemento del otro. Otro tipo de procesamiento se ve en [Ortner et al., 2007] haciendo uso de mapas DEM (digital elevation model). Aquí se reconstruye las ciudades a partir de imágenes con datos de elevación, haciendo una separación de objetos mediante algoritmos probabilísticos y luego creando lo que se llama *building footprints* o contornos de las construcciones. Un trabajo posterior utilizando esto es [Lafarge et al., 2010], el cual a partir de los contornos generados y algoritmos probabilísticos, levanta modelos en 3-dimensiones que representan las casas y construcciones de la ciudad. Por otro lado, el uso de base de datos con información relevante a ciudades es una propuesta distinta en donde se utilizan datos ya generados, los cuales se corrigen y se entregan a disposición del público. Utilizando esta directriz, existen tipos de base de datos de mapas digitales enfocadas a entregar información de zonas urbanas en el mundo, pero no con un enfoque específicamente para simuladores. Ejemplo de esto son OpenStreetMap y Google Maps, grandes aplicaciones computacionales dispuestas para el público general que adicionan y actualizan día a día su información de mapas urbanos. El potencial de estas herramientas proviene principalmente de su velocidad de crecimiento en cuanto a tamaño y cantidad de información. Además OpenStreetMap tiene la ventaja de ser de libre acceso a los datos dada la naturaleza colaborativa del proyecto. En [Haklay and Weber, 2008] se hace la presentación de OpenStreetMap, sus visiones del proyecto, sus metas y los avances que ha tenido hasta la fecha a través de los años. Sin duda, el futuro de los datos va a provenir de estas plataformas colaborativas, y se deben tener en consideración para los distintos proyectos a realizar. En [Latif et al., 2011] se ven distintas situaciones de emergencia y cómo se puede utilizar OpenStreetMap para proyectar soluciones en ciudades de países como Bangladesh. La poca cantidad de datos existentes en estas regiones del mundo, hace que estos mapas digitales sean de gran utilidad, esperando que su crecimiento pueda aumentar significativamente la cantidad disponible de este tipo de información. En comparación con la toma de datos de instrumentos, en donde se lleva a cabo proyectos muchas veces costosos para el uso específico de alguna ciudad, estos datos colaborativos son una alternativa que no tiene un costo monetario para el usuario y que, además, presentan un acceso rápido

a ellos. Calles, casas, construcciones, estadios, universidades y edificios públicos, entre otros, se pueden utilizar para generar la información deseada, según sea su disponibilidad.

El presente trabajo busca realizar un aporte en el estudio del impacto de zonas urbanas sobre la atmósfera y el clima, mediante la creación de **un sistema computacional de uso gratuito encargado de la obtención y generación automática de datos urbanos para simuladores atmosféricos**, con el propósito de complementar la información ya existente. Se hará uso de los datos provenientes de mapas digitales como OpenStreetMap y Google Maps para la generación de los datos de ciudad a utilizar en el simulador. La razón de esto es la libertad del uso de los datos, la constante actualización que mantienen y la posibilidad de adquirir datos de cualquier zona urbana en el mundo. La evaluación de los resultados se realizará aplicando el sistema sobre dos ciudades Sudamericanas importantes por su tamaño (Santiago de Chile y Buenos Aires). Finalmente se realizará una simulación atmosférica en un periodo determinado utilizando WRF sobre Santiago de Chile y utilizando los datos generados de esta ciudad, para de esta forma observar los efectos que tiene el acoplamiento de la información a los ya existentes. Se espera que el sistema sea influyente sobre los resultados, viendo cómo se ajustan de mejor forma al momento de compararlos con observaciones obtenidas de distintas estaciones meteorológicas para ese periodo determinado.

1.1. Conceptos básicos

Se presentan a continuación algunos conceptos básicos a ser utilizados a menudo en el presente trabajo.

- **Modelo atmosférico:** Un modelo atmosférico es un conjunto de ecuaciones matemáticas provenientes de distintas áreas de la ciencia, tales como la física y la química, que buscan principalmente representar y predecir el comportamiento de la atmósfera a través de la interacción de sus distintos componentes.
- **Simulación atmosférica:** Una simulación atmosférica es un procedimiento mediante el cual se resuelve el sistema de ecuaciones representados en un modelo atmosférico en particular utilizando parámetros iniciales definidos, obteniendo como resultado una aproximación al comportamiento de la atmósfera en un periodo de tiempo determinado. Un simulador atmosférico es una implementación computacional que realiza la tarea de cálculo y resolución de la simulación atmosférica.
- **Dosel urbano:** Se entiende por dosel urbano como el conjunto de datos que describen una zona urbana dentro de un modelo específico. Es común que la información del dosel urbano se presenta a través de una estructura de datos dispuesta en forma de grillas, en donde cada una de estas contiene en forma simultánea información referente a una variable. Por ejemplo, una zona urbana delimitada puede ser representada por una grilla para que contenga los datos de altura de terreno por cada punto, otra grilla para los datos de altura de edificios, otra para los tipos de árboles por punto, otra grilla para los datos de población por cada punto, etc. Hay que tener en cuenta que los datos para definir el dosel van a depender ciertamente de cómo se esté definiendo la zona

urbana, dado que muchas implementaciones requieren distintas variables. Un ejemplo de esto puede ser el uso de alturas, mapas de calor de personas, energía por zona de la ciudad, etc.

- **Mapas digitales:** Son un conjunto de datos que representan información espacial y atributos, almacenados en el ordenador. Una definición más exacta es ser el almacenamiento de información espacial como dibujos electrónicos hechos a base de elementos gráficos sencillos (líneas, puntos, círculos, etc.) organizados en capas, con el objetivo de una salida impresa o por pantalla.

1.2. Objetivo general

El objetivo general del presente trabajo es el desarrollo e implementación de un sistema computacional llamado **CityCanopy**, el cual se utilizaría para la generación, extracción de parámetros y visualización de un dosel urbano o grilla de datos de cualquier ciudad, que permita ser utilizada como información de entrada para simulaciones de modelos atmosféricos. En particular, se desea generar datos de la ciudad de Santiago de Chile para luego ser utilizados en simulaciones utilizando el modelo *Weather Research and Forecasting Model (WRF)* y ver los efectos que provoca la zona urbana en su entorno.

1.3. Objetivos específicos

A continuación se presenta los objetivos específicos que se esperan al término de la realización del presente trabajo.

1. Desarrollar un algoritmo para generación de dosel urbano a partir de mapas digitales.
2. Diseñar un modelo de datos que sea fácil de actualizar si a futuro se dispone de mayor cantidad de datos reales sobre ciudades.
3. Diseñar e implementar **CityCanopy**, un sistema computacional de generación y visualización de dosel urbano, en el que sea fácil agregar nuevas funcionalidades de extracción de parámetros, visualización de figuras, adquisición de datos externos y salida de datos a simuladores.
4. Realizar una simulación utilizando el modelo WRF, configurado en un periodo de tiempo determinado y haciendo uso de los datos del dosel urbano generado de la ciudad de Santiago.

1.4. Metodología

La metodología se divide en distintas fases que permiten conseguir los objetivos expuestos en la sección 1.3. Estas se presentan enumeradas a continuación:

1. Investigación y búsqueda de temas relacionado con el trabajo. Para ello se toman en cuenta distintos tópicos:
 - Conceptos relacionados al tema de simulaciones atmosféricas, que permitan entender el funcionamiento y objetivos de éstas.
 - Trabajos anteriores existentes, que se relacionen directamente con la generación de datos de doseles urbanos.
 - Recursos disponibles en donde se puedan obtener datos que puedan ser de utilidad para la implementación.
2. Diseño e implementación de **CityCanopy**. Se utiliza una metodología de desarrollo iterativo para el caso, en donde se reiteran todos los pasos hasta conseguir el producto final deseado. Una iteración completa de esta fase conlleva lo siguiente:
 - Proceso de planificación de requerimientos.
 - Diseño de clases para llevar a cabo los requerimientos de esta iteración.
 - Proceso de programación e implementación de clases definidas.
 - Generación y análisis de resultados obtenidos. Para validar los resultados referentes a la generación del dosel, estos se contrastan con un dosel urbano de referencia obtenido de forma manual.
3. Simulación y observación de resultados utilizando los datos generados de dosel urbano como entrada para el modelo atmosférico. En esta fase se planifica lo siguiente:
 - Generación de dosel urbano de la ciudad de Santiago de Chile utilizando **CityCanopy**.
 - Establecimiento de parámetros y variables de una simulación base o corrida control en un período determinado de donde se tengan datos observacionales.
 - Simulación de corrida control, con y sin acoplamiento de los datos del dosel urbano generado.
 - Análisis de los datos obtenidos contrastando las simulaciones realizadas, con y sin dosel urbano, y datos atmosféricos reales.
4. Análisis y conclusiones del trabajo realizado.

1.5. Contenido del trabajo

El presente trabajo se divide en 6 diferentes capítulos para llevar a cabo los objetivos propuestos:

- Capítulo 1: se hace una introducción al contenido, realizando una revisión de la bibliografía disponible y exponiendo los objetivos y metodología para el desarrollo del trabajo.
- Capítulo 2: se hace una revisión de los recursos disponibles relacionados con la implementación, para que el usuario pueda entender mejor las herramientas que se utilizan.
- Capítulo 3: se expone el desarrollo del diseño y la implementación del sistema computacional final.
- Capítulo 4: se realiza la aplicación del sistema sobre un set de datos, con el fin de observar los resultados generados.
- Capítulo 5: se hace uso de la información generada, acoplando los datos generados con los existentes y simulando en un caso de estudio en una zona específica, con el objetivo de observar los efectos generados por el dosel urbano en la simulación.
- Capítulo 6: se presentan las conclusiones del trabajo realizado, con respecto a los resultados obtenidos.

Adicionalmente se cuenta con una sección de bibliografía y apéndice para dar soporte al entendimiento de materias no expuestas y literatura consultada.

Capítulo 2

Recursos computacionales

Para llevar a cabo la implementación de **CityCanopy**, se deben tener en cuenta los recursos y datos disponibles a la fecha. El presente capítulo consta de la exposición de estos recursos computacionales que fueron considerados para el trabajo, realizando una presentación de sus características para un entendimiento general de estos.

2.1. Simulador WRF

El Weather Research and Forecasting Model (WRF) [Skamarock et al., 2008], tal como lo indica su nombre, es un simulador de predicción atmosférica a nivel de meso escala de uso gratuito y código abierto desarrollado para el estudio de la atmósfera. La generación de resultados se basa principalmente en la resolución de variadas ecuaciones en derivadas parciales (EDPs) sobre espacios grillados extensos, que dada su complejidad para ser resueltas por la cantidad de variables existentes, se calculan de manera computacional utilizando métodos numéricos. A modo de ejemplo, algunas de estas ecuaciones corresponden a las que se utilizan en áreas de la termodinámica y fluido-dinámica, e intentan modelar el comportamiento de los distintas variables de la atmósfera tales como temperatura, presión, turbulencia, velocidad de viento, etc. La constitución modular del simulador permite que los usuarios puedan agregar nuevas funcionalidades y modificar los distintos procedimientos del modelo, además de poder variar entre los existentes. Es decir, la implementación entrega la libertad de elegir, modificar y/o acoplar distintas EDPs según se estime conveniente para el caso de estudio.

2.1.1. Implementación computacional

La Figura 2.1 muestra a grandes rasgos cómo fluyen los datos dentro de la simulación módulo a módulo para la resolución de problemas dependiendo del uso que se quiere dar al modelo. Se puede observar que existen cuatro divisiones de módulos importantes. La primera, *External Data Source*, trata sobre los datos de entrada a la simulación. Estos deben ser obtenidos por el usuario y procesados de tal forma que cumplan las condiciones necesarias

para ser comprendidas dentro del programa. Hay que tener en cuenta que existe una gama limitada de categorías a utilizar para los cálculos, por lo que el usuario debe restringirse a éstas solamente. Las casillas *Alternative Obs Data* y *Conventional Obs Data* corresponden a datos observados, los cuales corresponden a mediciones indirectas o directas respectivamente de las distintas variables. Luego *WRF Terrestrial Data* corresponde a los datos estáticos de la simulación, los cuales no tienen una variación significativa en el tiempo (ej: topografía). Por último *Gridded data* corresponde a los valores de las distintas variables como temperatura, magnitud de viento, humedad, etc., los cuales sirven como condiciones iniciales para el cálculo de las simulaciones.

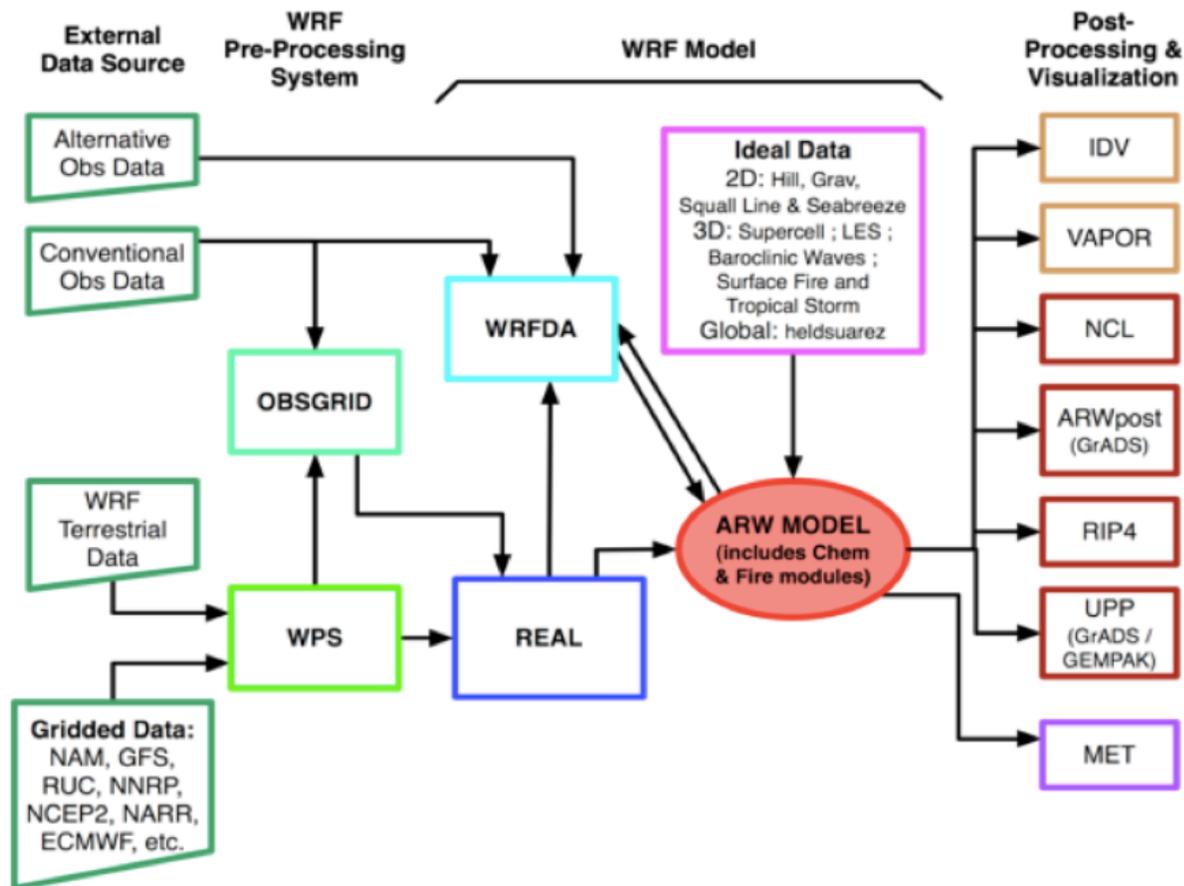


Figura 2.1: Diagrama de flujo de datos en WRF [wrf, 2010a].

La segunda división, *WRF Pre-Processing System*, es el pre-procesamiento de archivos llevado a cabo por el sistema llamado WPS (WRF Preprocessing System), compuesto de tres componentes esenciales: geogrid, ungrib y metgrid. La Figura 2.2 muestra como los archivos de entrada son encapsulados e integrados dentro de ficheros más complejos que representan las distintas capas de datos en forma de grillas, con el fin de ser procesados por la simulación. Los información de entrada proviene de dos tipos de fuente. La primera son datos geográficos (*static geographical data*) que hace referencia a datos como topografía de suelo, uso de suelo, etc. La segunda son datos en forma de grilla (*gridded data*) con las distintas condiciones iniciales de la simulación, como temperatura, velocidad de viento, concentraciones de fórmulas como dióxido de carbono y nitrógeno en la atmósfera, etc. Con grillas se hace referencia a

la división espacial que se realiza sobre la zona de estudio dependiendo de los parámetros entregados por el usuario. Luego cada una de las fuente es procesada por *geogrid* y *ungrib* respectivamente, para luego ser acopladas en un sólo archivo generado por *metgrid*. Por otro lado existe *OBSGRID*, módulo en el cual se logra ensamblar los datos generados por *WPS* con los datos observacionales entregados como *Alternative Obs Data* y *Conventional Obs Data*, con el fin de mejorar los resultados de la simulación.

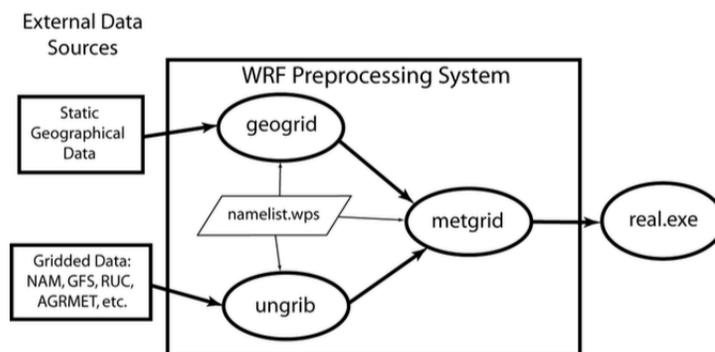


Figura 2.2: Diagrama WPS [wrf, 2010c].

La tercera división, *WRF Model*, es el ensamble y simulación del modelo. Esta sección conlleva la preparación de las condiciones iniciales y de borde del problema, para luego aplicar la resolución de ecuaciones complejas mediante métodos numéricos. Se pueden realizar 2 tipos de simulaciones dentro de *WRF*: ideales y reales. Las primeras (ideales) utilizan datos y condiciones iniciales entregadas por el usuario para casos en donde se tengan ciertas consideraciones especiales. Aquí se puede ver que el módulo *Ideal Data* presenta distintos tipos de simulaciones que se pueden realizar con estos parámetros. Las segundas (reales) utilizan datos y condiciones reales las cuales se procesan en el módulo *REAL* para crear los archivos de condiciones iniciales y de borde de las simulaciones. Luego estos archivos generados, ideales o reales, son utilizados por el módulo *ARW*, el cual realiza todo el proceso de cálculo de la simulación. Como alternativas se pueden utilizar el módulo *WRFDA* para poder manipular y consultar los datos *Alternative Obs Data* y *Conventional Obs Data* con el fin de mejorar las predicciones. Finalmente se obtienen los resultados finales de la simulación en archivos de formato abierto, los cuales pasan a una cuarta división opcional, *Post-Processing & Visualization*, en donde se pueden observar distintos programas computacionales como *IDV*, *VAPOR*, *NCL*, *GRaDS*, *MET*, los cuales facilitan al usuario las tareas de visualización y análisis de resultados obtenidos.

2.1.2. Modelos urbanos

Existen en la actualidad procedimientos definidos para el acoplamiento de datos urbanos al modelo, los que deben ser activados con la opción *sf_urban_physics* dentro del archivo *namelist.input*. Con acoplamiento se refiere a la adición de la información a los datos de entrada de *WRF* y su utilización efectiva dentro de la simulación. Se pueden diferenciar en tres tipos de niveles, donde cada uno de estos agrega efectos adicionales al problema:

1. Urban Canopy Model (UCM): 3 categorías distintas para uso de suelo (bajo residencial, alto residencial y comercial), en donde se agrega el efecto de superficie de techos, murallas, calles y un parámetro opcional de techos verdes. Esta aplicación del modelo se encuentra en [Chen et al., 2004], [Lee et al., 2011], [Chen et al., 2011], [Tewari et al., 2005], [Tewari et al., 2006], [Kim et al., 2008].
2. Building Environment Parametrization (BEP): dosel urbano de multicapas adicional a UCM, que contempla tanto la capa inferior como la superior de la atmósfera. Se tiene también un manejo diferente de los efectos de temperatura a nivel de suelo con respecto al modelo UCM. Esta aplicación del modelo se encuentra en [Liao et al., 2014], [Gutiérrez et al., 2013], [Chen et al., 2011].
3. Building Energy Model (BEM): agrega a BEP el ciclo de energía de sistemas de calefacción y enfriamiento de las construcciones y viviendas. Esta aplicación del modelo se encuentra en [Gutiérrez et al., 2013], [Chen et al., 2011].

La Figura 2.3 muestra en forma resumida los distintos efectos que modela cada nivel urbano anteriormente expuesto. Se puede observar a la izquierda el primer nivel UCM y a la derecha el nivel BEP+BEM.

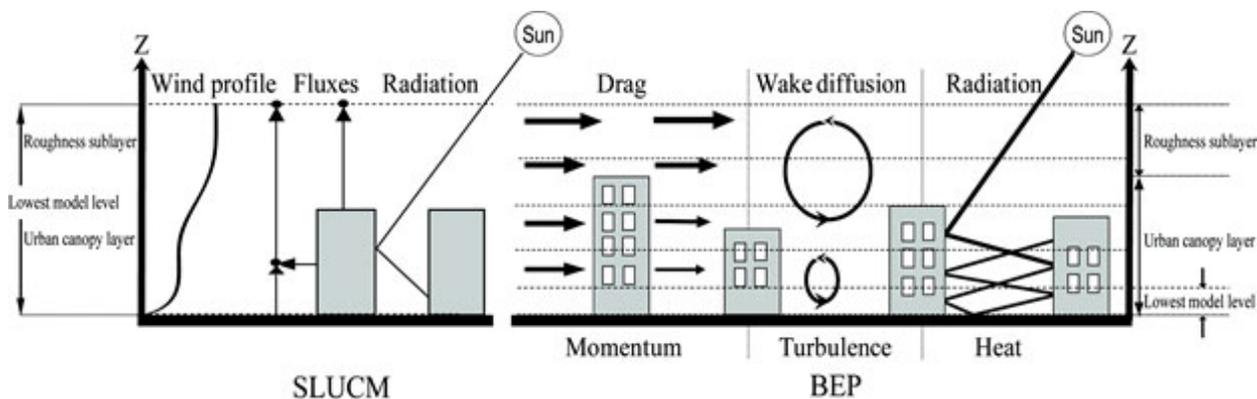


Figura 2.3: Diferencias entre distintos niveles [Chen et al., 2011]

Como la simulación utiliza un modelo de capas y grillas para la resolución de ecuaciones, no es posible definir la geometría de cada construcción como una malla de polígonos, por lo que el modelo urbano se basa principalmente en las categorías de uso de suelo de la zona ¹. La tabla de categorías urbanas es una modificación a la tabla estándar, en donde se adicionan tres nuevas: 31. *Low Intensity Residential*, 32. *High Intensity Residential* y 33. *Industrial or Commercial*. Luego para cada categoría se define un modelo estadístico de parámetros tales como altura promedio de techos, desviación estándar de alturas de techo, ancho promedio de techos, desviación de ancho promedio, etc. Estos vienen con valores predeterminados entregados por el autor, pero el usuario puede (y es altamente recomendable) modificarlos en el archivo *URBPARM.TBL* dependiendo de la zona de estudio.

¹El uso de suelo se utiliza para determinar que categoría pertenece cada punto de la grilla que representa un espacio determinado. Las categorías pueden variar dependiendo del modelo, en donde WRF puede utilizar tanto USGS con 24 categorías y/o MODIS con 20.

2.2. Mapas digitales urbanos

A continuación se hace una presentación de los mapas digitales con datos urbanos y topográficos que se consideraron convenientes para el actual trabajo, y que por esta razón es una cantidad pequeña de la gran gama existente.

2.2.1. Google Maps

La empresa multinacional Google, fundadora del sitio de búsqueda del mismo nombre, entrega a los usuarios mapas digitales por medios de su aplicación Google Maps. Aquí el usuario puede hacer uso gratuito de su interfaz visual y de esta forma poder ver distintas ciudades de todas partes del mundo, teniendo la posibilidad tanto de observar mapas satelitales como mapas representativos de las zonas urbanas. El propietario de la aplicación es el encargado de mantener los datos, es decir, constantemente realiza cambios y modificaciones al mapa al momento de existir en la vida real, por lo que estos no pueden ser utilizados a libertad del usuario, sino que su uso es restringido según las políticas del sitio. Por otro lado, se ofrece una gran gama de herramientas útiles de tanto de forma gratuita como pagadas, de lo que se puede sacar bastante provecho.

Visualización en Google Street View

La herramienta Street View es una de las características que entrega la aplicación Google Maps. La idea es que el usuario pueda visitar cualquier parte del mundo a través de una base de datos de imágenes, obtenidas mediante un trabajo manual que incluye el recorrido físico de la zona y su posterior captura a través de una cámara especial con vista periférica (ejemplo del dispositivo en la Figura 2.4).



Figura 2.4: Automóvil con cámara incorporada para capturar imágenes posteriormente utilizadas en Street View [goo, 2016b].

Si bien puede tener un uso recreacional, se puede utilizar para otros fines como por ejemplo la corrección de modelos 3D de monumentos/edificaciones o el cálculo aproximado de altura de una construcción.



Figura 2.5: Vista ejemplo Street View [goo, 2016a].

2.2.2. OpenStreetMap

La fundación OpenStreetMapi [Haklay and Weber, 2008] es una organización internacional, de carácter comunitario, dedicada a la obtención y procesamiento de información de mapas, además de su visualizador que entregan en línea. La diferencia, con respecto a otros de su especie, es que todos los datos son de libre acceso, dado que los propios usuarios son los que de forma voluntaria agregan y rectifican los datos sobre construcciones y calles, entre otras cosas.

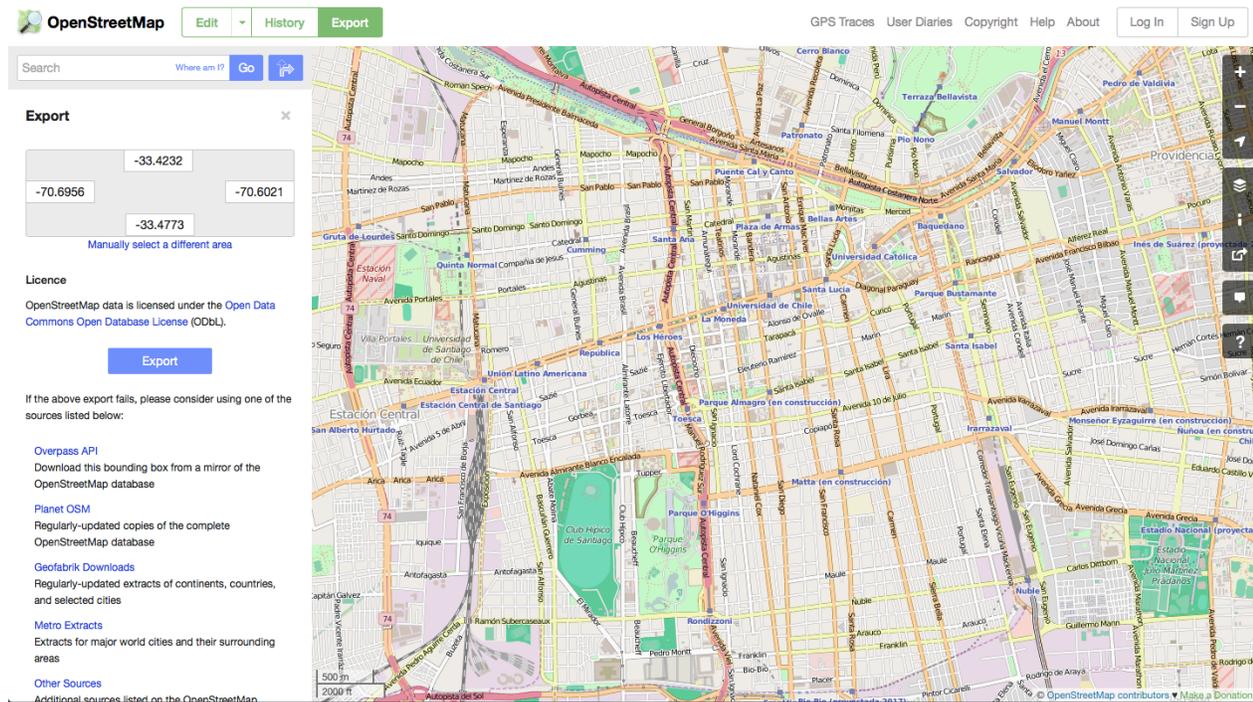


Figura 2.6: Interfaz para extraer datos según coordenadas del area deseada.

En la Figura 2.6 se puede ver como el sitio web [ope, 2016a] entrega una interfaz bastante intuitiva y amigable para que el usuario pueda extraer y descargar información de su base de datos. Como se explica en la documentación [ope, 2016b] la descarga se realiza a través de su sitio web, generando un archivo en formato estándar *XML* en el que se encuentra toda

la información respecto a la zona escogida. Dentro de este fichero se puede encontrar tres elementos fundamentales dentro de la estructura de datos: los nodos *<node>*, los caminos *<way>* y las relaciones *<relation>*. Los primeros representan un punto específico en el espacio, lo cual puede ser objetos como señalizaciones, semáforos, numeraciones de calles, etc. Por otro lado, los segundos son una lista ordenada de los primeros, con lo cual se pueden crear formas geométricas para representar elementos más complejos. Entre estos pueden ser simples líneas y curvas para las calles y caminos, y polígonos para las construcciones y parques. Los terceros son una agrupación de los segundos y que por ende permiten establecer objetos que tienen mayor cantidad de componentes. Como elemento adicional se encuentran las etiquetas *<tag>*, las cuales llevan la tarea de mantener información respecto al objeto real que se representa.

Lamentablemente el usuario al momento de ingresar algo nuevo, no está obligado a completar todos los datos relevantes, por lo que queda a criterio de él qué cosas va a definir y colocar. Esto perjudica al uso de la información dado que no se encuentra completa en su totalidad, pero por otro lado simplifica y ayuda bastante a mantener actualizado el sistema por parte de la comunidad.

2.2.3. CGIAR-CSI

La *CGIAR* es una organización dedicada a dar soporte a distintos tipos de centros de investigación con el fin de reducir la pobreza y el hambre, mejorar la calidad de vida humana y la nutrición, y ayudar a mejorar la resiliencia del ecosistema a través de investigación agrícola. Dentro de este contexto, la *Consortium for Spatial Information* o *CSI* es una comunidad dedicada a la ciencia espacial que facilita a *CGIAR* el uso de herramientas para el análisis espacial (GIS) y teledetección. El sitio web [cgi, 2016a] ofrece un portal para la obtención de datos de topografía a 90m de resolución, tal como se muestra en la Figura 2.7. De acá el usuario puede obtener en formato de archivo de texto *ASCII* una grilla que representa la elevación de terreno. Como la topografía no sufre grandes cambios en el tiempo a gran escala, estos mapas no son actualizados constantemente, pero por temas de visualización, y además al ser de libre acceso, son una herramientas útil para cualquier tipo de aplicación.

The CGIAR Consortium for Spatial Information (CGIAR-CSI)

Applying GeoSpatial Science for a Sustainable Future...

CSI HOME SRTM MAIN HELP

SRTM Data Selection Options Chinese users : [中国用户可通过中国科学院镜像站点下载](#)

1. Select Server: CGIAR-CSI (USA) HarvestChoice (USA) JRC (IT) King's College (UK) TeraScience (USA)

2. Data selection method: Multiple Selection Enable Mouse Drag Input Coordinates

Many tiles can be selected at random locations. These selected tiles are listed in the results page for download.

Decimal Degrees (ie 34.5, -100.5) Degrees: Minutes: Seconds (ie 34 30 00 N, 100 30 00 W)

Longitude - min: max: Longitude - min: Eas max: Eas

Latitude - min: max: Latitude - min: Nori max: Nori

Longitude: 46.31 Latitude: 46.26 Tile X: 46 Tile Y: 3

3. Select File Format: GeoTiff ArcInfo ASCII

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24

Figura 2.7: Portal CGIAR-CSI para datos 90m resolución [cgi, 2016b].

2.3. Visualización con OpenGL

Para la visualización computacional se utiliza *OpenGL*. Esta es una librería computacional gráfica, que se encarga de entregar al usuario todos los procedimientos y herramientas para la proyección y visualización de los elementos que se desean mostrar al espectador a través de la pantalla. La Figura 2.8 muestra la línea de pasos que sigue esta librería para llevar a cabo sus tareas. En cada paso se llevan a cabo distintos procedimientos o tareas, que reciben como entrada datos de la etapa anterior, los procesan y los entregan a la siguiente. Cada uno de estos puede ser modificado por el usuario, pero es común sólo manipular las etapas de *Vertex Shader* y *Pixel Shader*. Aún así, se van a explicar de forma general cada proceso de la línea de trabajo.

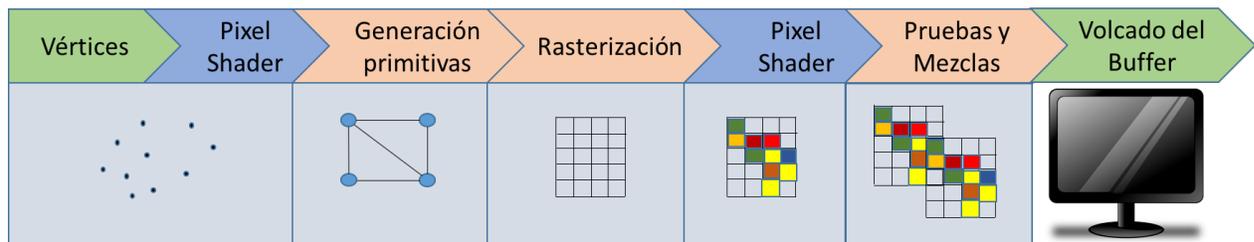


Figura 2.8: Etapas en el flujo de procesos en OpenGL.

Los vértices y sus parámetros adicionales son el conjunto de datos de entrada para el proceso completo. Como parámetros adicionales, se consideran vectores normales y coloración de cada punto definidos por el usuario entre otras cosas, que serán útiles para las distintas etapas. Estos datos de entrada los recibe en primera instancia el proceso de *Vertex Shader*, el cual para todos los vértices aplica algoritmos **definidos por el usuario** para posicionar en pantalla los distintos puntos que se encuentran en el espacio virtual. La Figura 2.9 muestra de forma general el esquema básico de visualización. Se puede ver que el *viewpoint* es la cámara con que mira el usuario la escena a dibujar, la que posee una posición en coordenadas espaciales (u, v, w) y una orientación dado por los vectores x , y y z . Con esto luego se definen los planos de visualización, en donde se definen los límites que logra ver la cámara. Aquí se encuentran dos tipos de límites: el *near clipping plane* que es el plano más cercano con su ancho y su largo, y el *far clipping plane* o plano más lejano también con su ancho y su largo. Con estos parámetros el proceso de *Vertex Shader* entrega la posición de los distintos vértices dentro del plano de visualización al siguiente paso.

El proceso de *Generación Primitivas*, tal como dice su nombre, genera las primitivas o polígonos que conforman una malla final a partir de los puntos recibidos. El algoritmo de triangulación es el más utilizado, el cual a partir de los puntos entregados genera una malla de triángulos que definen una representación. Un ejemplo sencillo de triangulación en 2 dimensiones se puede observar en la Figura 2.10. La idea de triangular es poder utilizar la malla generada para llevar a cabo distintos cálculos en etapas posteriores, los que permiten obtener mejoras significativas en la visualización final de las figuras.

El tamaño y la cantidad de triángulos a utilizar varía dependiendo de la cantidad de puntos que se entregan o desean tener. Es común que el aumento considerable de la cantidad de puntos y elementos dentro de la malla, se debe principalmente al aumento en el nivel de

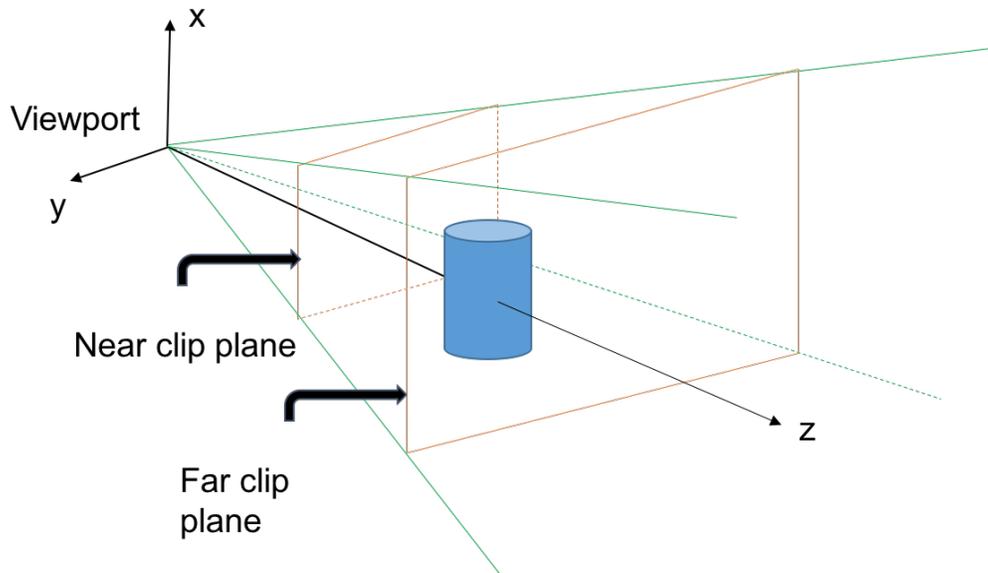


Figura 2.9: Esquema de visualización en pantalla del espacio virtual.

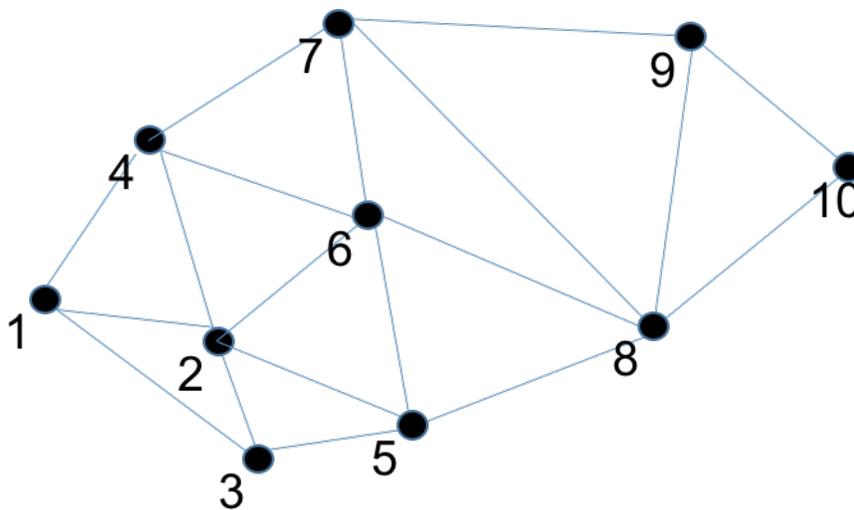


Figura 2.10: Pequeña triangulación de un conjunto de puntos.

detalle de la representación. La Figura 2.11 muestra un ejemplo de cómo varía el resultado de la triangulación de un mismo objeto a distintos niveles de detalle, en donde el aumento de esta característica conlleva la necesidad de elevar también el número de elementos de la malla.

En el proceso de *Rasterización*, se realiza el marcado de *pixeles*² en la ventana de visualización. Un ejemplo de este proceso se observa en la Figura 2.12, en donde a partir de una ventana que contiene un elemento triangular en 2 dimensiones, se aplica una rasterización,

²El píxel puede definirse como la unidad más pequeña de las unidades homogéneas en color que componen una imagen de tipo digital. La cantidad de pixeles que se disponen se encuentra limitado por la capacidad de *hardware* que se tenga.

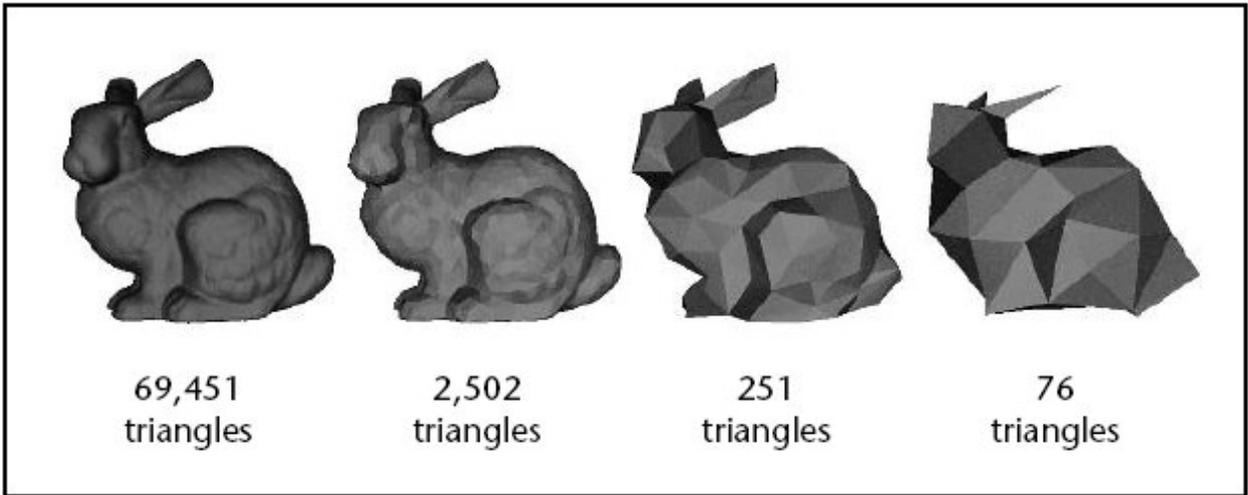


Figura 2.11: Número de triángulos dependiendo del nivel de detalle [Luebke, 2003].

y se obtiene un marcado de *pixels*. El conjunto de *pixels* marcados en este proceso van a pasar entonces al siguiente paso en conjunto con la malla de polígonos.

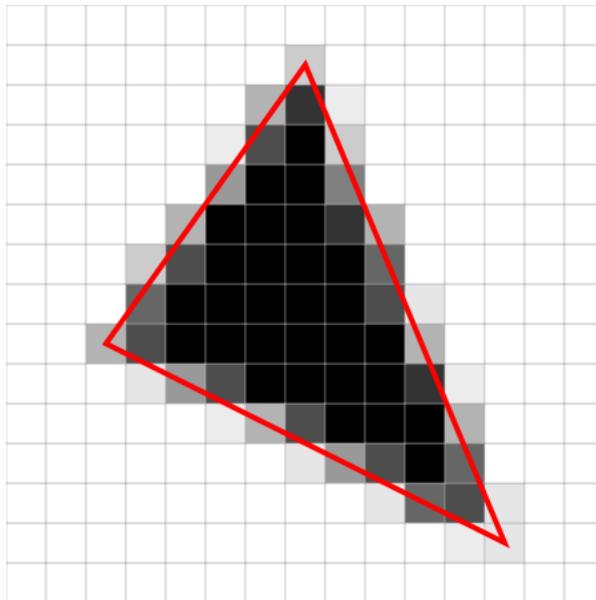


Figura 2.12: Rasterización de un triángulo [tri,].

En la etapa de *Pixel Shader* se realiza el proceso de iluminación, sombreado y coloración de la representación. Para ello, se aplican a todos los vértices de la malla algoritmos **definidos por el usuario**, que calculan las distintas componentes para obtener los colores finales de la figura. Luego, de forma interna, aplica los resultados al raster de puntos obtenidos en el proceso anterior de rasterización, utilizando procedimientos de interpolación para los puntos interiores de los polígonos. Uno de los modelos simples de sombreado e iluminación es el de *Phong*, el cual se puede explicar a través de sus componentes vectoriales de forma general expuestas en la Figura 2.13. Suponiendo que se tiene una fuente de luz y una representación geométrica. Se comienza proyectando el haz de luz inicial desde la fuente como el vector L que apunta hacia algún punto de la representación, el cual incide en un ángulo β respecto a la

normal N . Este se refleja en un ángulo β generando el vector R . A su vez, produce un vector V con un ángulo α con respecto a R . La cantidad de luz, color y sombra se genera a partir de este último vector V , y depende principalmente de componentes y parámetros del algoritmo según sea el material del objeto. Por ejemplo, objetos de materiales metálicos reflejan mayor cantidad de luz que objetos de materiales con superficies irregulares. Cabe destacar que este modelo es un modelo empírico, y que si bien puede simular el comportamiento de la luz con motivos de visualización, no tiene ciertamente una base científica.

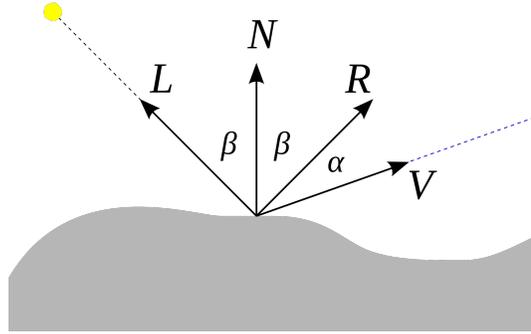


Figura 2.13: Diagrama general del modelo de iluminación de Phong [pho,].

El proceso final de *Pruebas y Mezclas* utiliza distintas opciones que permiten optimizar el proceso de dibujo, tales como pruebas de profundidad de las figuras y eliminación de píxeles sobrepuestos, clipping y eliminación de elementos que no se encuentran en pantalla, entre otras cosas. Luego, este proceso se encarga de traspasar los datos de los *píxeles* a la memoria de volcado gráfico o *buffer*, la cual finalmente se utiliza para dibujar y visualizar en pantalla.

Capítulo 3

Diseño e implementación

El presente capítulo trata sobre el diseño e implementación del sistema computacional **CityCanopy**, exponiendo los resultados finales luego de aplicar la metodología de la sección 1.4. Un diagrama general de los pasos se presenta en la Figura 3.1, en donde se comienza desde la planificación de requerimientos y se continúa hasta la generación y análisis de resultados, en donde dependiendo de los resultados se puede volver o no al primer paso.

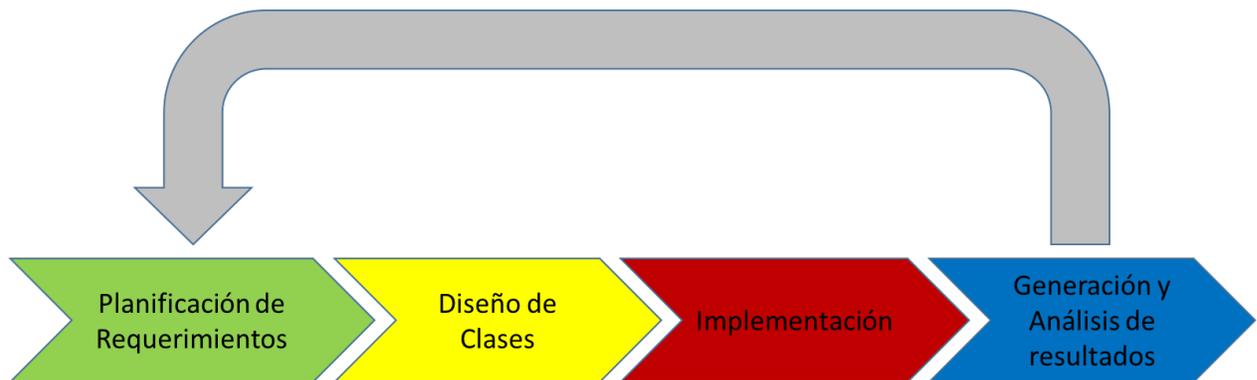


Figura 3.1: Proceso de desarrollo iterativo del trabajo

3.1. Planificación de requerimientos

Las siguientes secciones exponen de manera general los requerimientos y funcionalidades que debe realizar la implementación **CityCanopy**, las cuales permiten cumplir los objetivos expuesto en la sección 1.3.

3.1.1. Requerimientos del Usuario

El Usuario debe ser capaz de llevar a cabo las siguientes tareas:

1. Generar un dosel urbano a partir de los datos de entrada.
2. Visualizar la información generada del dosel urbano.
3. Visualizar la topografía donde se encuentra el dosel urbano.
4. Generar archivos para la simulación deseada a partir de la información obtenida.

3.1.2. Requerimientos del Sistema

El Sistema debe ser capaz de llevar a cabo los siguientes procesos funcionales que permiten dar soporte a las acciones del Usuario:

1. Desplegar una interfaz gráfica para el usuario.
2. Leer datos de entrada dependiendo del formato entregado.
3. Extraer y procesar datos del dosel urbano a partir de datos de entrada.
4. Generar la visualización de figuras del dosel urbano.
5. Generar la visualización de la topografía en donde se encuentra el dosel urbano.
6. Exportar los datos hacia archivos de salida para la simulación deseada.

Además, el Sistema debe tener ciertas características no funcionales deseables que permitan mejorar su extensibilidad y mantenibilidad:

1. Puede agregar más formatos de entrada a medida que aparezca nuevos datos.
2. Puede agregar nuevos métodos de visualización dependiendo del formato de entrada.
3. Puede generar más formatos de salida dependiendo de la simulación objetivo.

3.1.3. Casos de uso

A continuación se presentan los distintos casos de uso del sistema en una descripción general de alto nivel. Estas descripciones enumeradas son las acciones que se deben generar en un curso normal desde que el Usuario ingresa al Sistema. Hay que considerar que el único actor dentro de las distintas acciones es el Usuario, y es el que gatilla el inicio de cada una.

- **Caso de Uso (1): construir y visualizar dosel urbano**

1. El sistema despliega menú principal.
2. El usuario aprieta botón para agregar datos urbanos de entrada.

3. El sistema pide ruta de archivo de entrada.
 4. El usuario ingresa el archivo de entrada.
 5. El sistema procesa la información y genera dosel.
 6. El sistema despliega la visualización del dosel generado.
 7. El usuario visualiza el dosel.
- **Caso de Uso (2): visualizar topografía**
 1. El sistema despliega menú principal.
 2. El usuario aprieta botón para agregar datos de topografía.
 3. El sistema pide ruta de archivo de entrada.
 4. El usuario ingresa el archivo de entrada.
 5. El sistema procesa la información. Si existe visualización de dosel, el terreno se coloca por debajo. Si no, genera visualización del terreno sola.
 6. El usuario visualiza el terreno generado.
 - **Caso de Uso (3): exportar datos para simulación**
 1. Considerando que se realiza el caso de uso (1).
 2. Si el dosel generado es el deseado, el usuario aprieta el botón de exportar. Si no, sale del sistema.
 3. El sistema genera archivos de salida para simulador.
 4. El usuario agrega los datos al simulador.

3.1.4. Diagrama general

El diagrama general de caso de uso se presenta en la Figura 3.2, en donde se exponen los casos de usos descritos en la sección 3.1.3.

3.2. Análisis y selección de recursos disponibles

A continuación se hace un pequeño análisis sobre ciertos aspectos de los recursos disponibles expuestos en el capítulo 2, para de esta forma realizar una decisión de qué características serán utilizadas en la implementación de **CityCanopy**

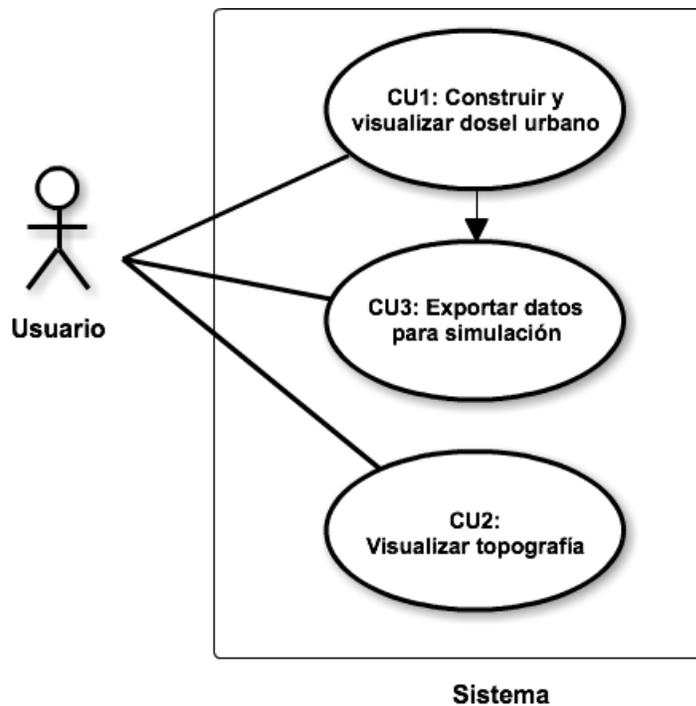


Figura 3.2: Diagrama general de caso de uso.

3.2.1. Elección de datos disponibles para generación del dosel

En la sección 2.2 se presentaron un grupo de distintas fuentes de datos (Google Maps, OpenStreetMap, CGIAR-CSI), desde donde se puede extraer información útil para el sistema. Sin embargo, cada una conlleva distintas restricciones, por lo que se debe hacer una comparación en base a ciertos aspectos que presentan cada una.

1. Tipo de datos: Google Maps y OpenStreetMap presentan mapas con información sobre las zonas urbanas, como la geometría de calles, viviendas, construcciones y parques. Además, en OpenStreetMap se le da un valor agregado a los elementos utilizando etiquetas para poder entregar información adicional como nombres, enumeraciones, tipo de vivienda, etc. Por otro lado, en CGIAR-CSI existen distintos tipos de mapas referentes a variadas mediciones, en donde se puede señalar la topografía de terrenos.
2. Disponibilidad de datos: OpenStreetMap y CGIAR-CSI presentan un uso totalmente libre de los datos, dado que las organizaciones encargadas intentan entregar esto para el uso de la ciencia y la comunidad. Por otro lado Google Maps restringe y/o limita el uso de sus datos, en donde se debe pagar una licencia adicional en ciertos casos para poder obtener cierta información.

Tomando en cuenta estos aspectos, se debe tener en consideración el uso de libre acceso que debe tener la implementación, dado que se busca eliminar de cierta forma las barreras monetarias de soluciones costosas para el estudio, pudiéndose utilizar sin pago de licencias y uso exclusivo. Los datos provenientes de OpenStreetMap son la mejor opción para el caso, en donde existe una mejor accesibilidad de los datos. Por otro lado, la herramienta Streetview

no se debe descartar del todo para trabajos futuros, dado que su utilidad para rectificar los datos de manera visual parece ser una muy buena opción para estudio de lugares remotos. Concerniente al tema topográfico de la visualización, la información desde CGIAR-CSI es la opción a elegir, de donde los datos se obtienen de manera libre y con muy buena resolución para todo tipo de aplicaciones.

3.2.2. Elección del nivel de modelo urbano a utilizar y los datos a generar

Dado que el *Weather Research and Forecasting Model* dispone de tres niveles de modelos urbanos distintos, expuestos en la sección 4, se debe hacer una elección de cuál es el modelo objetivo y que tipos de datos se van a generar. Realizando una mirada en la bibliografía, en [Liao et al., 2014] se muestra el uso del modelo urbano de WRF sobre ciudades en la zona del Yangtze River Delta, China. Se utilizan distintos niveles del modelo con el fin de realizar comparaciones a las distintas configuraciones existentes, y de este modo poder ajustar los resultados y determinar cuáles presentan mejoras a las estimaciones. En [Tewari et al., 2005] se lleva a cabo un estudio sobre la región de Salt Lake City, USA, haciendo uso del simulador WRF para observar la influencia del cambio de uso de suelo sobre los resultados correspondiente a temperatura y vientos de la zona. Las conclusiones indican que generando variaciones dentro de la ciudad se pueden obtener también pequeñas mejoras en los resultados estimados. En [Chen et al., 2004] también se observa como la influencia del uso de suelo y la representación geométrica de la ciudad producen efectos en la estimación de los datos.

Todos estos estudios utilizan de cierta forma parámetros en común para definir los doseles urbanos, pero hay que tener en cuenta ciertas restricciones para el presente trabajo:

1. Disponibilidad de datos: si bien los datos de OpenStreetMap proveen información de geometría y características propias de los elementos, la cantidad de datos que se pueden extraer es bastante limitada. Parámetros como materiales de construcciones, uso de energía, cobertura vegetal u otras característica, no siempre están disponibles y es escasa. Es posible que para esta situación se necesiten mapas digitales más especializados en el tema, y que en muchos casos no se encuentren disponibles para todo público.
2. Tipos de datos a utilizar: a medida que se aumenta de nivel en el modelo, se comienzan a necesitar distintos tipos de datos para mejorar los resultados. Un ejemplo de esto es el intercambio térmico a partir del uso del aire acondicionado, el cuál es un factor a considerar dentro del modelo BEP+BEM, pero que no se utiliza en UCM. Sin embargo, cada nivel utiliza el nivel inferior para poder realizar sus cálculos, por lo que la información utilizada en UCM servirá luego para BEP y para BEP+BEM.

Los distintos niveles pueden mejorar ciertamente la calidad de los resultados de las simulaciones, pero los datos actualmente disponibles son limitados a los existentes en OpenStreetMap. Si bien se espera poder agregar todos los parámetros posibles que definan a una zona urbana, el trabajo se va a concentrar en una implementación inicial que sólo considera la generación del **uso de suelo urbano** con los **3 tipos de uso de suelo básicos utilizados por los modelos urbanos del simulador**, los cuales son: (31) bajo residencial, (32) alto

residencial y (33) comercial. Esto permite hacer uso de la geometría plana y/o contorno de los elementos que conforman a la zona urbana, además de la información y descripción que los caracteriza dentro de los datos. La razón principal de esta decisión es que, dado que la implementación del sistema se encuentra en un desarrollo temprano, se desea comenzar con la generación de los parámetros básicos y común para todos los modelos, para luego ir agregando nuevos en implementaciones futuras. Según los trabajos expuestos en el capítulo 1, el uso de suelo es uno de los elementos principales para poder definir la zona urbana dentro del simulador, y por ende es la base para el dosel urbano completo. Dado lo anterior, no es necesario utilizar los niveles más complejos del modelo y bastará con hacer uso de los dos primeros nivel, **UCM** y **BEP**, para observar los efectos que se producen y poder determinar si existe o no un cambio significativo con el acoplamiento de los datos.

3.2.3. Elección de lenguaje de programación

Se utilizarán dos lenguajes de programación para el presente trabajo. La implementación será desarrollada en su totalidad utilizando el lenguaje C/C++ con el uso de las librerías de Qt/C++. Esto dado por dos razones: la comodidad que se tiene para la manipulación de ficheros XML con *QtXML* y la creación de ventanas de visualización gracias al uso de *QtCreator* con *QtOpenGL*, en conjunto con otras herramientas que se proveen; y la eficiencia en el uso de memoria y rapidez inherente que tiene el lenguaje para realizar los cálculos necesarios sobre matrices extensas. Para el caso, el uso típico de matrices extensas será de valores de punto flotante *float*, con dimensiones aproximadamente de 1.000.000 de puntos. En la parte de experimentación se va a utilizar en su totalidad el lenguaje Python. El objetivo específico es llevar a cabo la creación de los distintos gráficos que se necesiten para poder realizar las observaciones sobre los resultados, dado que contiene variadas librerías para la manipulación de los ficheros de salida de la simulación WRF, como por ejemplo *netCDF4* y *matplotlib*.

3.3. Flujo del Sistema

La Figura 3.3 representa el flujo de datos en donde el Sistema obtiene la información necesaria, la procesa y la utiliza para generar los archivos correspondientes.

La obtención se realiza a partir de dos recursos. El primero es un archivo *OSM/XML* generado a partir de la *API* de OpenStreetMap. El segundo es un archivo de texto *ASCII* proveniente de las base de datos de la *CGIAR-CSI* que representa la topografía. Finalmente los datos son procesados por el sistema y se generan los archivos binarios para el preprocesador *WPS*, para luego ser acoplados a los datos originales y posteriormente utilizados como entrada en *WRF*.

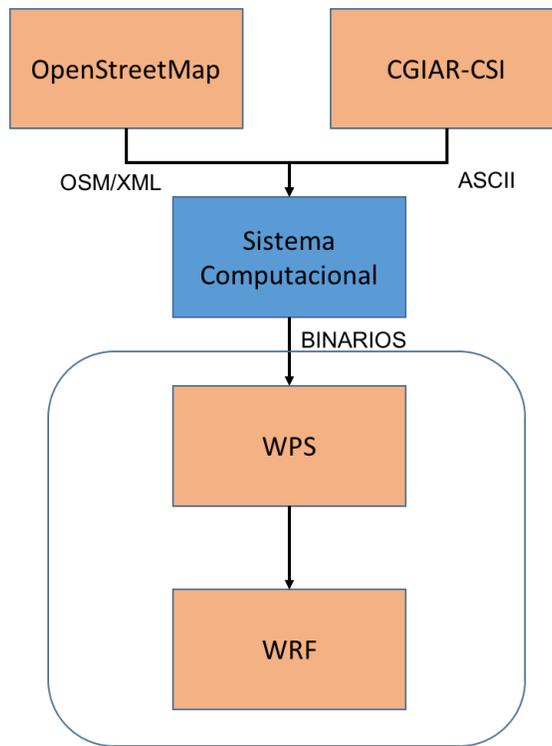


Figura 3.3: Diagrama general de flujo de datos.

3.4. Diseño de Clases

Las clases del sistema computacional siguen un patrón de arquitectura de software de tipo *Modelo Vista Controlador* como se presenta en la Figura 3.4. El motivo de esto es poder separar las distintas funcionalidades del programa y así facilitar el desarrollo, aumentar la escalabilidad y mejorar el mantenimiento de la implementación.

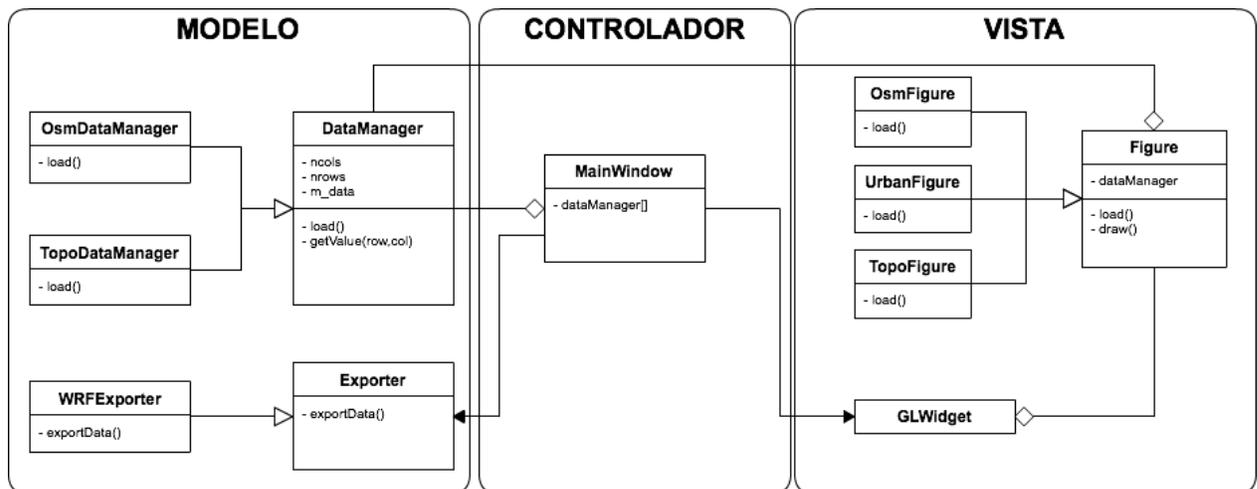


Figura 3.4: Diagrama de clases.

3.4.1. Modelo

Clase DataManager

Encargada de la obtención y mantención de los datos. Esta es una clase base que contiene todo lo necesario con respecto a los datos, tanto la matriz de información, como la cantidad de filas y columnas, el espaciado de grilla, los límites espaciales de los datos y el valor específico de dato "nulo". Las clases derivadas actualmente implementadas son las expuestas en la Figura 3.4, en donde los nombres de cada una hacen referencia al formato de entrada que procesan. Los métodos relevantes son:

1. **void load(string filename)**: Este método debe redefinirse en cada clase derivada dependiendo del formato de entrada que espera recibir. Su fin es leer los datos desde la ruta o el archivo *filename* y luego guardarlos junto con sus metadatos. La información se mantiene de forma matricial, por lo que se utiliza el vector *float* m_data* y los parámetros de número de columnas (ncols), número de filas (nrows), valor nulo (NODATA_value), espaciado de grilla (cellsize), valor de longitud mínima (xllcorner) y valor de latitud mínima (yllcorner).
2. **float getValue(int i , int j)**: Entrega el valor de la matriz en la coordenada (i,j) .

Clase Exporter

Encargada de exportar los datos generados a un archivo en formato específico para ser utilizado por el simulador. Las clases que derivan de ésta especifican en su nombre el simulador de salida al cuál se desean generar los archivos. Cabe destacar, que hasta el momento sólo se encuentra implementado *WRFExporter* (formato de salida exclusivo para el simulador WRF), pero se deja la posibilidad de agregar nuevos para casos futuros. Los métodos relevantes son:

1. **void exportData(DataManager* dataManager)**: Genera los ficheros en formato de salida específico para el simulador requerido a partir de los datos almacenados en *dataManager*.

3.4.2. Vista

Clase GLWidget

Encargada de generar la visualización de figuras en la *Vista*, la cual aprovecha el uso de recursos propios de la librería *Qt* en conjunto con *OpenGL*. Esta clase contiene métodos definidos para toda la inicialización del entorno, como la configuración del *viewport*, el *near plane*, el *far plane*, las dimensiones espaciales y los métodos de dibujo, entre otros.

Clase Figure

Encargada de la generación de malla de triángulos a partir de los datos encapsulados en *DataManager*, y de la que desprenden las clases derivadas que van a depender del tipo de figura que se desea generar. Los métodos relevantes de la clase *Figure* son:

1. **void load(DataManager* dataManager)**: Este método debe definirse en cada clase derivada dependiendo de los datos de entrada que se reciben para generar la figura. Su fin es leer los datos desde *dataManager* y transformarlos en una triangulación de puntos, para obtener los datos de normales y de esta forma seguir la línea de trabajo estándar definido por *OpenGL*. La figura se almacena en un arreglo para ser utilizado por los otros métodos.
2. **void draw()**: Realiza todos los procedimientos y llamadas a funciones para cargar o dibujar en pantalla del usuario la figura desde el arreglo almacenado por el método *load()* de la clase.

3.4.3. Controlador

Clase MainWindow

Encargada de la inicialización de las distintas clases que conforman la implementación y la interacción entre usuario y sistema. Es el puente principal entre las acciones que provee el usuario y las que genera el sistema a partir de estas.

3.5. Procesos relevantes

3.5.1. Procesamiento de datos OSM/XML

El objetivo de este procedimiento es construir una grilla regular (de igual espaciado entre puntos tanto a lo largo como a lo ancho) que define el dosel urbano a partir de los datos provenientes de OpenStreetMap. Para ello se procesan los archivos según la documentación de su sitio web ¹, y luego se generan los datos de uso de suelo utilizados en el simulador WRF. Los siguientes pasos explican de manera general como funciona el procedimiento:

1. **Inicialización de la grilla**: Se divide el área total en forma de una grilla regular. Para ello, se obtiene en primera instancia las coordenadas que definen el área rectangular que contienen los datos provenientes del archivo de entrada. Luego, una vez calculado el largo y el ancho del rectángulo, se procede a dividir estos por un segmento R que define la resolución de la grilla, obteniendo de esta forma M puntos a lo largo y N puntos a lo ancho. Finalmente se inicializa una matriz de $N \times M$ puntos que representa la nueva

¹<https://wiki.openstreetmap.org/>

grilla a poblar con datos. Cabe destacar, que cada punto de la matriz va a representar el centroide de cada subárea cuadrada que conforman el área total de cobertura como se muestra en la Figura 3.5.

- 2. Lectura y limpieza de datos:** Dado que los datos contienen información que no se utiliza para el caso, se realiza una limpieza inicial para extraer sólo los elementos de interés. La estructura del archivo de entrada se encuentra en un formato XML por lo que, utilizando una librería que se adapte a esta (librería QXml), se puede diferenciar entre los distintos objetos, más específicamente entre *node* y *way*. El primero contiene como atributos el *id* de identificación y las coordenadas *lat* (latitud) y *lon* (longitud). El segundo contiene un conjunto ordenado de *ids* que representan los nodos que conforman el camino o polígono, y un conjunto de *tags* que contienen descripciones sobre el objeto real al que representan. De esta manera se van guardando en memoria por un lado los *nodes*, en un diccionario ordenado de rápido acceso según su *id* con sus coordenadas respectivas, y por otro lado sólo los *ways* que definan un polígono y que contengan en su descripción la característica *building*, la cual hace referencia a una construcción (vivienda o edificio).
- 3. Poblar datos en la grilla:** Una vez limpios los datos, se procede a poblar la grilla con las categorías urbanas que se utilizan en el uso de suelo de WRF. Estas son por defecto: zona bajo residencial (31), zona alto residencial (32) y zona comercial (33). La idea de este paso, es hacer un recuento de cuántos elementos existen de cada tipo de categoría en cada punto de la grilla y luego, bajo cierto criterio, determinar cuál es el más influyente y representativo de la zona y así clasificarla como una sola categoría. Por lo tanto, por cada elemento *ways* se calcula el centroide del polígono que describe a partir de los *nodes* que lo conforman, y luego se le utiliza para posicionarlo espacialmente en la grilla. Un ejemplo de esto se puede observar en la Figura 3.6. Utilizando la descripción del *tag* se determina a qué tipo de categoría pertenece y posteriormente se adiciona al recuento local de objetos por categoría dentro de la zona. Cabe destacar que *OpenStreetMap* **no posee las 3 categorías o no las provee de forma directa** como tal. Aquí se debe hacer uso del atributo llamado *building*, el cual no siempre se encuentra presente en los elementos, y de donde se pueden ver 2 tipos posibles: *residential* y *commercial*. De esta forma, se toma en cuenta como bajo residencial (31) a todos los polígonos que no posean el atributo *building* designado, mientras que se tomará como alto residencial (32) a los con atributo *building* de tipo *residential*, y finalmente como comercial (33) a los con atributo *building* de tipo *commercial*. Por último, después de procesar todos los elementos, se recorre punto por punto de la grilla y se aplica un **criterio** determinado para establecer su categoría final. Este criterio será definido en la sección 4 en donde se busca aplicar distintos procedimientos para poder escoger el que presente mejores resultados.

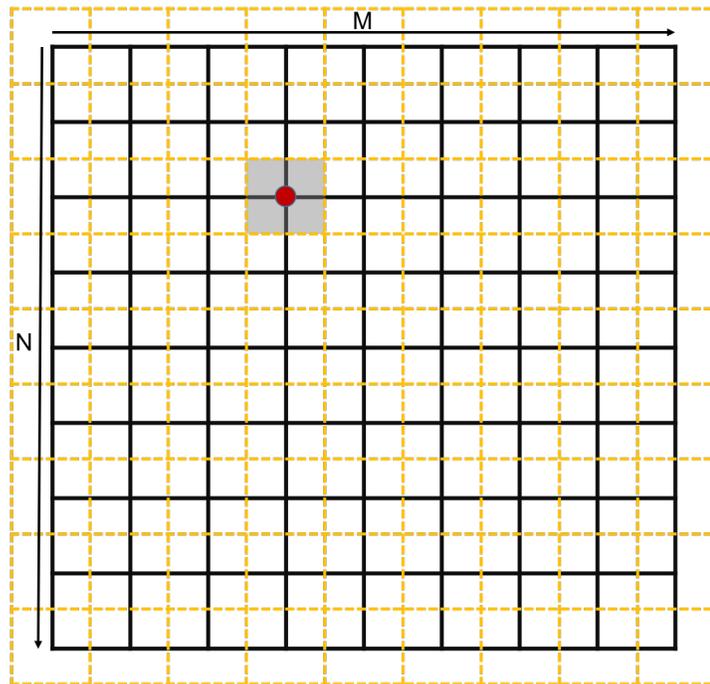


Figura 3.5: Dibujo ejemplo de representación de matriz de $N \times M$. Línea negra representa a la matriz y sus puntos. Línea amarilla discontinua representa el subárea que abarca cada punto.

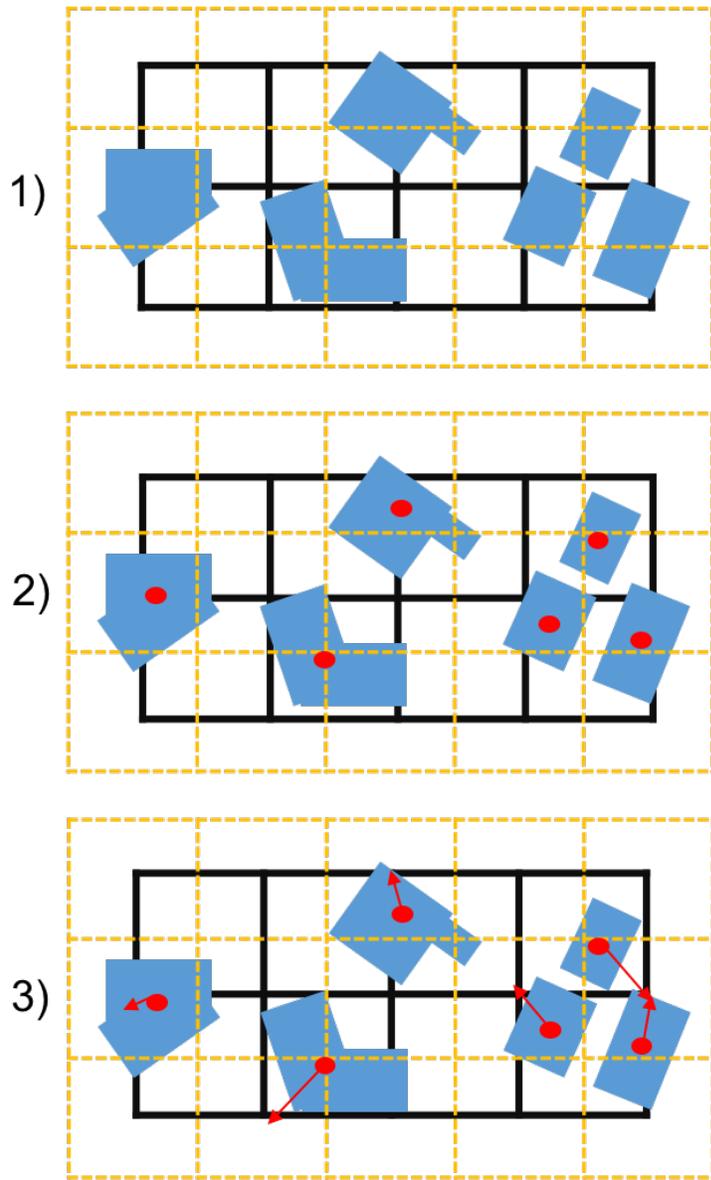


Figura 3.6: Dibujo ejemplo donde se muestra el proceso de búsqueda del centroide por elemento (2) y su ubicación espacial dentro de la grilla (3).

Traspasando estos pasos a un procedimiento, queda como el Algoritmo 1:

Data: Archivo OSM/XML F , Resolución grilla R , Criterio C .

Result: Grilla de $N \times M$ con datos de uso de suelo.

$F \leftarrow$ abrir F ;

$metadatos \leftarrow$ extraer metadatos de F ;

$N \leftarrow (metadatos.maxLatitud - metadatos.minLatitud) / R$;

$M \leftarrow (metadatos.maxLongitud - metadatos.minLongitud) / R$;

$G \leftarrow$ Iniciar grilla de $N \times M \times 4$;

$nodes \leftarrow []$;

$buildings \leftarrow []$;

foreach elemento e de F **do**

if e es un 'node' **then**

$nodes \leftarrow nodes + e$;

 continuar;

end

if e es un 'way' y contiene descripción 'building' **then**

$buildings \leftarrow buildings + e$;

 continuar;

end

end

foreach elemento e de $buildings$ **do**

$centroide \leftarrow$ calcular centroide de e ;

$categoria \leftarrow$ extraer categoría de e ;

$indice \leftarrow$ índice más cercano de G a $centroide$;

$G[indice][categoria] \leftarrow G[indice][categoria] + 1$;

end

foreach punto p de G **do**

$numComercial \leftarrow p['comercial']$;

$numAltoResidencial \leftarrow p['alto_residencial']$;

$numBajoResidencial \leftarrow p['bajo_residencial']$;

$categoriaFinal \leftarrow$ aplicar criterio;

$CRITERIO(numComercial, numAltoResidencial, numBajoResidencial)$;

$p['categoria_final'] \leftarrow categoriaFinal$;

end

Algorithm 1: Algoritmo para generación del dosel urbano.

El orden del algoritmo se calcula a continuación. El proceso inicial de asignación e inicialización es de $O(1)$, por tanto el orden final será entregado por los ciclos *foreach* que existen. Suponiendo E el número de elementos del archivo y R ancho de la grilla deseada. En el primer ciclo, el número de elementos extraídos del archivo depende del número de líneas que se lean del archivo, por tanto la obtención de cada elemento tiene un costo de lectura en memoria principal. La asignación e inserción de nuevos elementos a las listas es de $O(1)$, por tanto el primer ciclo queda como $O(E) * (O(1) + O(1)) = 2 * O(E) = O(E)$. En el segundo ciclo, las asignaciones, extracción de parámetros e inserción son $O(1)$. La consulta sobre los índices más cercanos, conociendo los límites de la grilla y la resolución de la grilla regular, se puede realizar en una sola consulta que toma $O(1)$, y por tanto el segundo ciclo, al ser dependiente del número de elementos queda $O(E)$. En el tercer ciclo, el criterio se considera como un procedimiento de $O(1)$, que calcula de manera directa a través de una ecuación el valor final proveniente de tres valores entregados. Por tanto, el orden es dependiente del número de puntos en la grilla, que suponiendo un área rectangular acotada A , entonces quedaría cómo $O(A/R^2)$.

El orden final del algoritmo estaría entregado por $O(1) + O(\text{ciclo}_1) + O(\text{ciclo}_2) + O(\text{ciclo}_3) = O(1) + O(E) + O(E) + O(A/R^2) = 2 * O(E) + O(A/R^2)$ lo que da como resultado final $O(E) + O(A/R^2)$. Si se realiza la conversión de $A/R^2 = N$, en donde N es el número de puntos de la grilla, quedaría $O(E) + O(N)$.

3.5.2. Dibujo de figuras

Las figuras utilizan como entrada una estructura de datos en forma matricial provenientes de la carga de datos. La matriz de entrada debe corresponder a la figura que se desea representar, dado que los valores en las distintas matrices no significan lo mismo en una y en otra. Es decir, mientras los datos de uso de suelo contienen números que representan categorías, los de topografía son números que representan alturas de terreno medidas en metros. Por lo tanto, se utilizan procedimientos distintos para la triangulación y obtención de normales. Una vez obtenidos estos datos, se traspasan al flujo de OpenGL descrito en la sección 2.3 para luego ser visualizados en pantalla.

Figura de topografía

Este procedimiento busca generar una representación del terreno a partir de los datos de altura provenientes de la matriz de entrada. Para ello, suponiendo que estos datos matriciales representan una grilla regular, hay que utilizar esto como una ventaja. Se puede obtener una triangulación sin la necesidad de generar un procesamiento de los datos adicional, dado que la regularidad entrega de forma indirecta una malla ya construida para utilizar. Por otro lado, el acceso rápido a los vecinos de cada punto permite simplificar el proceso de obtención de normales. Para ello, se construye una matriz auxiliar con las dimensiones de la matriz de topografía. Por cada punto de la grilla se construyen en sentido horario los dos triángulos que lo comparten como vértice, utilizando al primer vecino de la derecha, el primer vecino de abajo y el primer vecino diagonal derecha abajo. Notar que cada vértice va a tener las coordenadas

reales (x,y,z) , por lo que las dimensiones de los triángulos y sus posiciones espaciales varían, y por tanto, sus normales también. Luego, se calcula el vector normal de cada triángulo y se suman en la matriz auxiliar, en las posiciones de los puntos utilizados. La Figura 3.7 muestra un ejemplo de cómo se procede para un solo valor de la grilla y las normales resultantes.

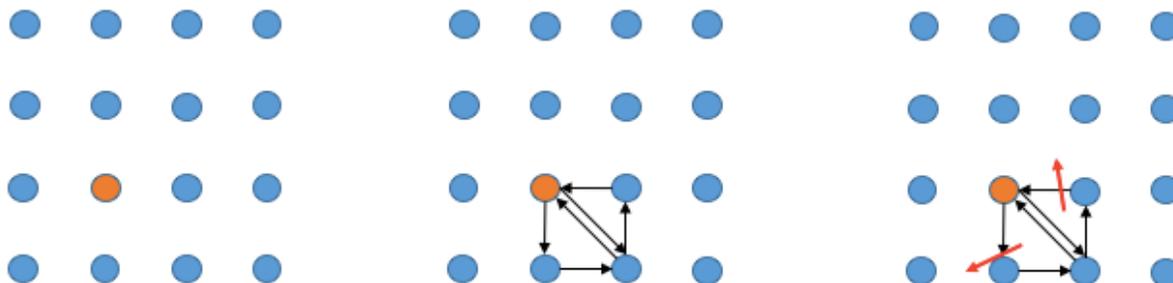


Figura 3.7: Ejemplo de obtención de normales. Flechas en negro representan el sentido horario de los vértices de los triángulos. En rojo las normales de cada triángulo.

Al final del proceso se obtienen en cada posición de la matriz auxiliar la suma de las normales de todos los triángulos que comparten aquel punto. Finalmente se aplica un proceso de normalización para obtener valores unitarios finales.

Figura de dosel urbano

Este procedimiento busca generar una representación del dosel a partir de los datos de categorías provenientes de la matriz de entrada. Hay que recordar que cada punto de la matriz representa en cierta forma el centro de un área cuadrada que la contiene, por lo que se busca una representación parecido a ello. Así, el objetivo es que por cada punto que contenga un dato, se crea un paralelepípedo que representa su categoría por medio de un color definido, en donde cada uno de estos paralelepípedos centra su cara inferior en el punto de la grilla, tal como se hace en los datos de entrada. De igual forma que el procedimiento anterior, se hace uso de la ventaja de la grilla regular. El procedimiento pasa por cada punto de la matriz de entrada, y construye un paralelepípedo regular o cubo, en donde el largo de la arista corresponde al ancho de resolución de la grilla. Utilizando nuevamente el sentido horario de los vértices, se calcula la normal de cada cara y se suma a la normal de cada uno de los vértices que las componen. A cada cara se le da un color específico que indica a cuál categoría pertenece. La Figura 3.8 muestra cómo se verían las normales de una sola cara del cubo y la asignación de la normal a sus vértices.

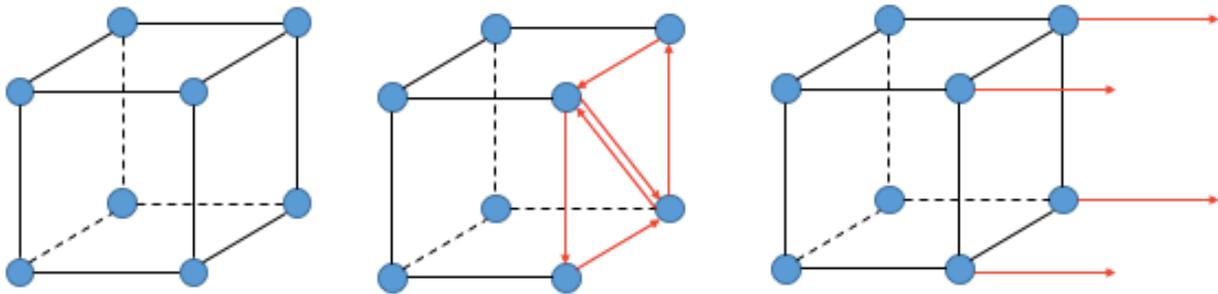


Figura 3.8: Ejemplo de obtención de normales para una cara del paralelepípedo. En negro se representan las aristas que definen al paralelepípedo. De izquierda a derecha en la segunda figura, se puede observar en rojo la orientación de la triangulación por cara. En la tercera figura, se puede observar en rojo las normales de cada triángulo.

3.5.3. Acoplamiento de datos a WRF

La clase *WRFFExporter* es la encargada de generar todo el contenido a exportar específicamente para *WRF*. Esto implica la transformación de datos a ficheros binarios para ser utilizados como entrada al flujo del simulador. Para el traspaso a binarios existe un procedimiento previamente definido por el simulador llamado *write_geogrid.c*, el cual recibe como entrada un arreglo de datos y los escribe al formato utilizado en el *WPS*.

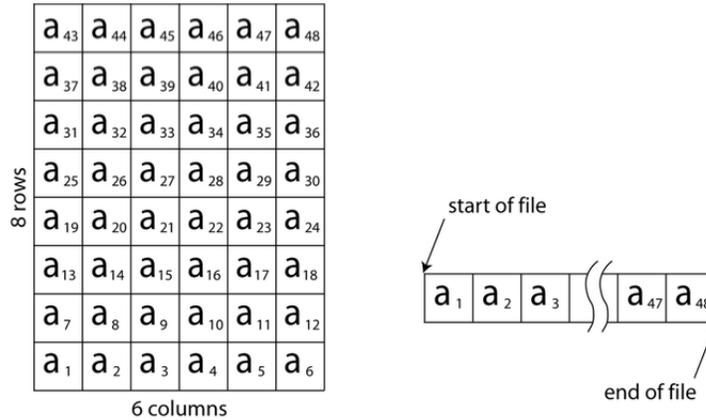


Figura 3.9: Escritura de *write_geogrid.c* [wps, 2010].

El ejemplo en la Figura 3.9 muestra cómo se escribe un arreglo de 48 elementos a un archivo binario. Internamente, la clase implementada, tiene especial cuidado con el orden de los datos que se entregan, dado que los primeros valores del arreglo corresponden a las últimas filas de la matriz. Si se entrega una matriz que excede los límites definidos por el proceso, entonces *write_geogrid.c* lo divide en archivos más pequeños y le entrega a cada uno un nombre con las coordenadas que contiene. La Figura 3.10 muestra cómo un arreglo de 3200 filas y 2960 columnas se divide en 4 archivos distintos (*Tile A*, *Tile B*, *Tile C*, *Tile D*), los cuales poseen un nombre en formato **columna_inicial-columna_final.fila_inicial-fila_final** que definen que cuadrante de la matriz contienen como información. La clase también se encarga de generar el archivo *indexi*, que hace referencia a los metadatos de los binarios. Éste define todo lo requerido por *WPS* para saber qué se está entregando y cómo procesarlo, como por ejemplo que tipo de datos contiene, la proyección de la tierra que se utiliza, el ancho de grilla tanto en el eje *x* como en el *y*, la cantidad total de columnas y filas, entre otros.

Una vez generados los archivos se procede a agregar estos al diccionario de datos de *WPS*. Se debe modificar el fichero *GEOGRID.tbl* dentro del directorio *geogrid*, agregando nuevas líneas de texto como las que muestra el ejemplo de la Figura 3.11. El campo *name* se refiere al nombre de la capa de datos al que se desean acoplar, en este caso *LANDUSEF* significa la capa de uso de suelo. El campo *priority* indica cuál es la prioridad por sobre las otras capas del mismo tipo de datos, y que en este caso toma el valor 2 para superponer los que se entregan por sobre los existentes. Hay que tener en cuenta que el valor 1 significa sobrescribir, lo que puede llevar a cometer errores en la simulación dado que algunos puntos de la grilla no necesariamente pueden tener datos. El campo *dest_type* trata sobre qué tipo de variable estadística son los datos incluidos en la grilla, que en este caso es un tipo de dato limitado

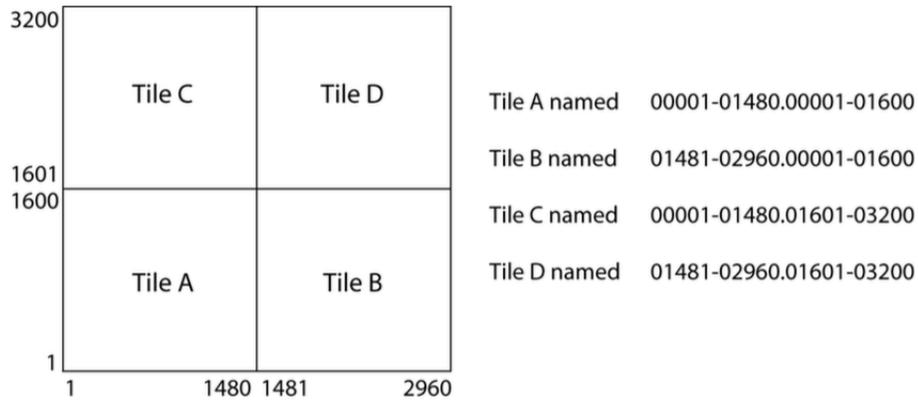


Figura 3.10: División de matriz [wps, 2010].

sólo a un valor dentro de un número finito de categorías por lo que se debe colocar *categorical*. El campo *z_dim_name* apunta a la variable de la simulación para la cual se están agregando los datos, que para el caso sería el parámetro *land_cat*. El campo *interp_option* es como se deben interpolar los datos en el caso de ajustar la grilla a una resolución distinta, por lo que basta mantener el valor de la grilla original *default:nearest_neighbour* que interpola simplemente al punto más cercano. Finalmente el campo *rel_path* es la ruta de directorios relativa a la raíz donde la implementación va a ir a buscar los archivos generados. Para el caso estos binarios generados con su *index* se encuentran dentro de la carpeta *landuse_urban* en el directorio explicitado en el *namelist.wps* del preprocesador.

```
name=LANDUSEF
  priority=2
  dest_type=categorical
  z_dim_name=land_cat
  interp_option=default:nearest_neighbor
  rel_path= default:landuse_urban/
```

Figura 3.11: Adición de líneas en archivo *GEOGRID.tbl*.

3.6. Implementación final

A continuación las Figuras 3.12, 3.13, 3.14, 3.15, 3.16 son algunas de las tomas de imágenes de la implementación final. Al centro se observa el espacio donde la aplicación realiza el dibujo de figuras. Aquí el usuario tiene la posibilidad de rotar las figuras y hacer zoom para poder observar mejor los datos. Al lado derecho es el menú de acciones disponibles para realizar, donde se cuenta con los siguiente botones:

- *Add Topo*: al apretarlo el sistema pide un archivo de topografía en el formato ASCII propuesto en CGIAR-CSI. Luego genera la visualización de estos datos como terreno.
- *Add OSM*: al apretarlo el sistema pide un archivo OSM/XML propuesto en OpenStreetMap. Luego procesa este fichero y genera el dosel urbano. Finalmente genera la visualización de estos datos utilizando 3 colores para representar la información de uso de suelo: rojo para categoría comercial (33), verde para categoría alto residencial (32) y azul para categoría bajo residencial (31).
- *Export*: una vez generado el dosel urbano, se puede realizar la exportación a través de este botón. El sistema pregunta donde se desean exportar los datos y luego genera los archivos binarios necesarios para WPS.
- *Clear*: al apretarlo limpia la pantalla de dibujos y elimina la memoria almacenada por el dosel para comenzar con información limpia.

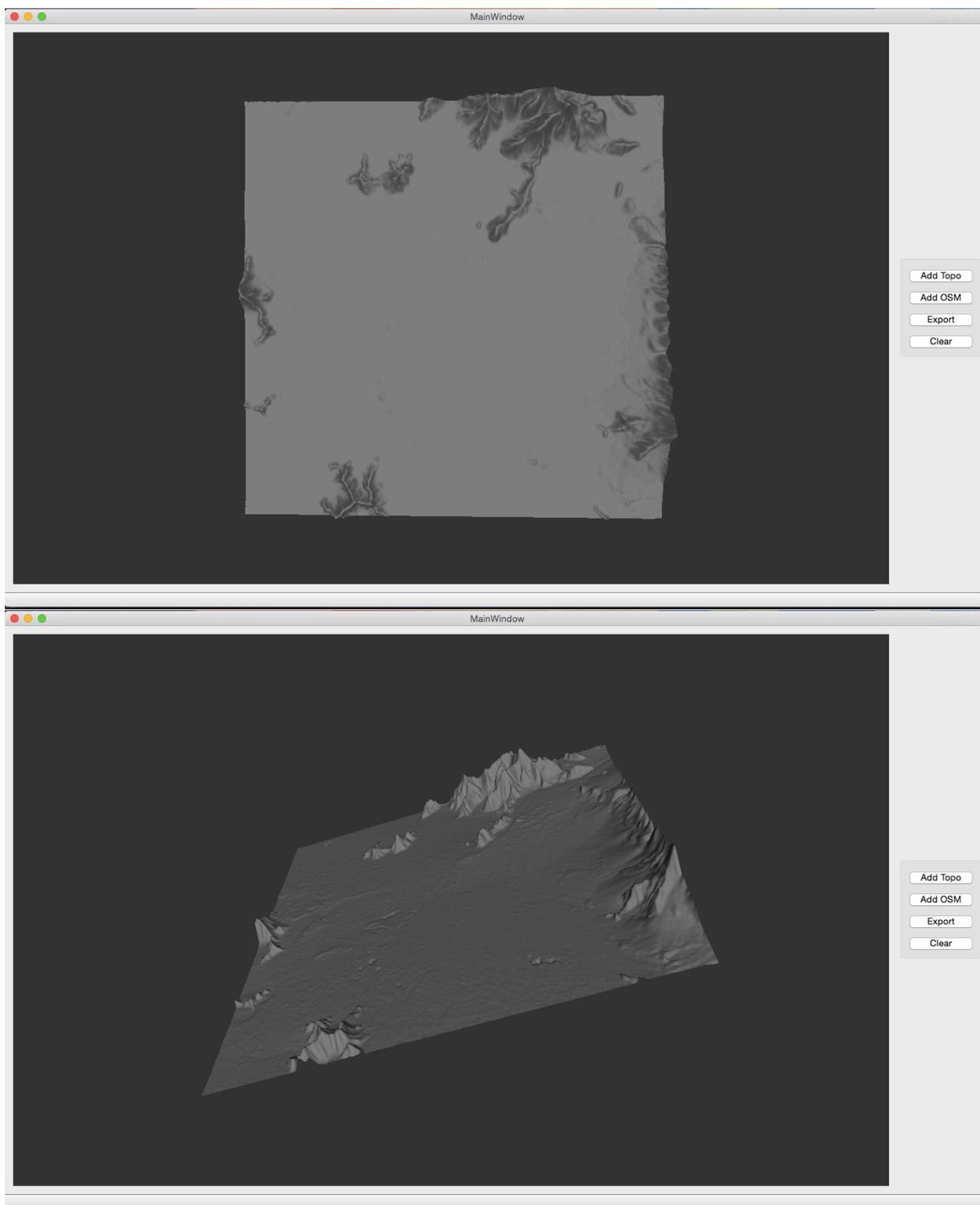


Figura 3.12: Toma de imágenes de la topografía del terreno obtenido desde dos perspectivas distintas.

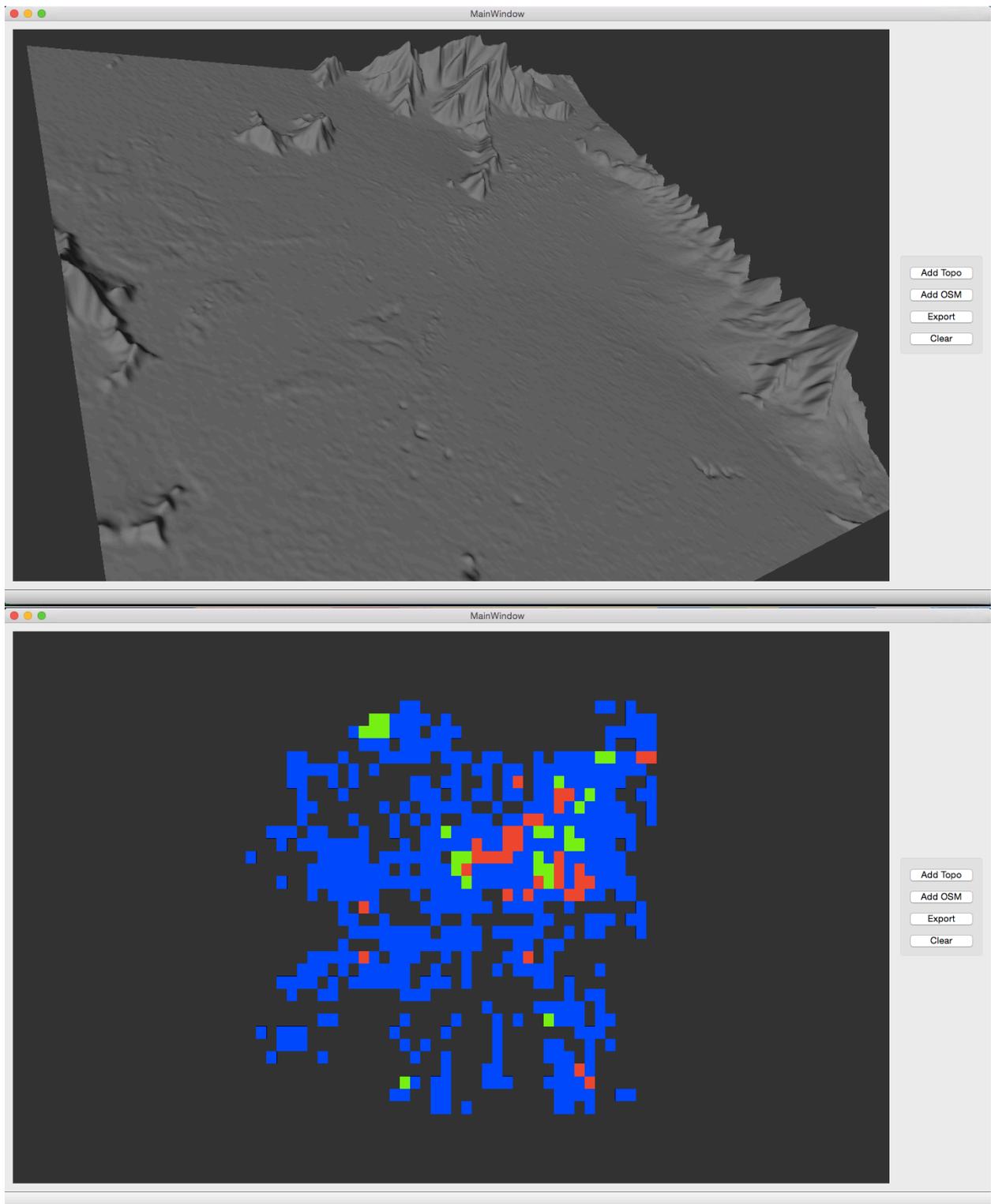


Figura 3.13: Primera figura toma de imagen con un acercamiento a al topografía. La segunda figura muestra el dosel urbano generado sin topografía.

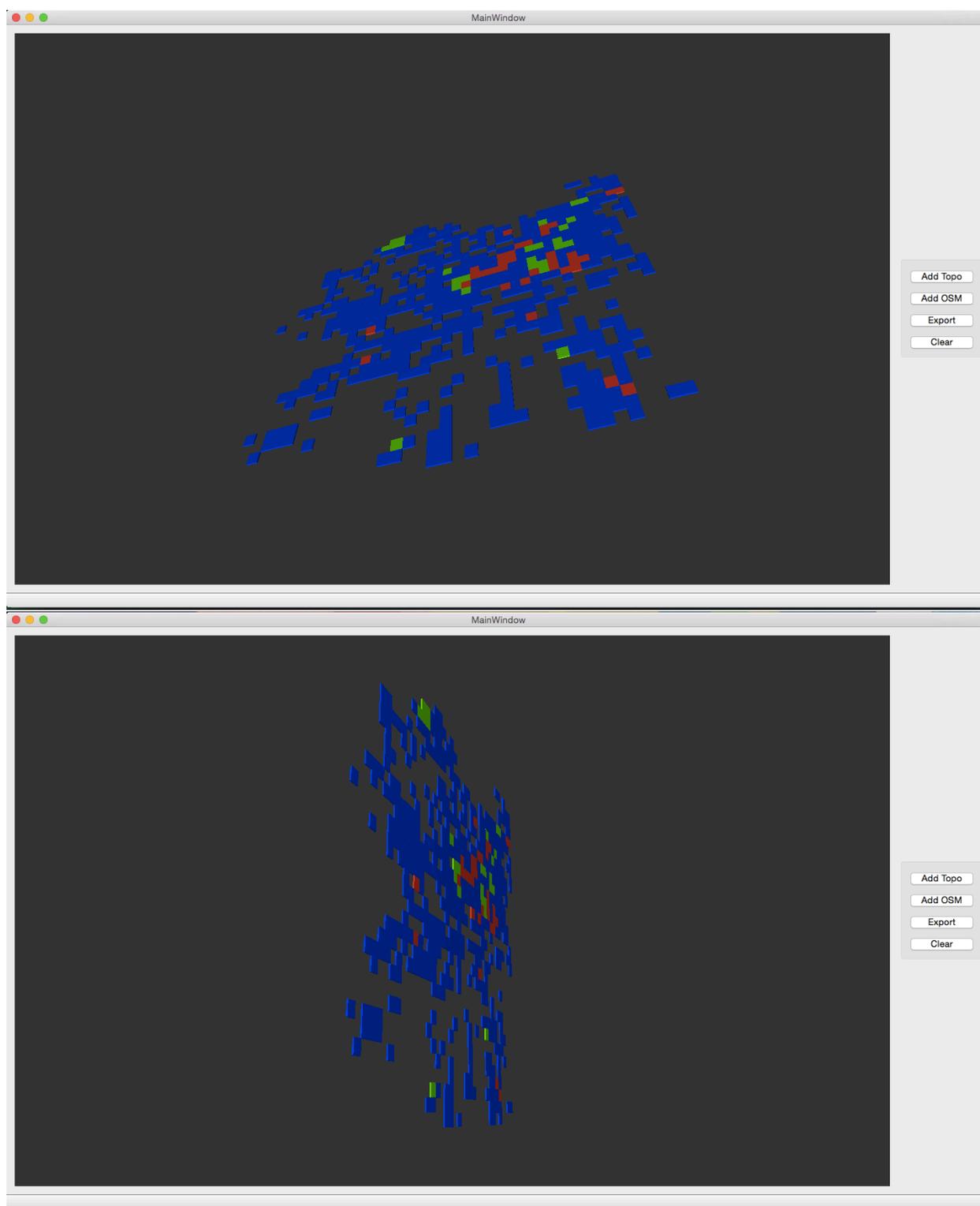


Figura 3.14: Toma de imágenes del dosel urbano generado sin topografía desde dos perspectivas distintas.

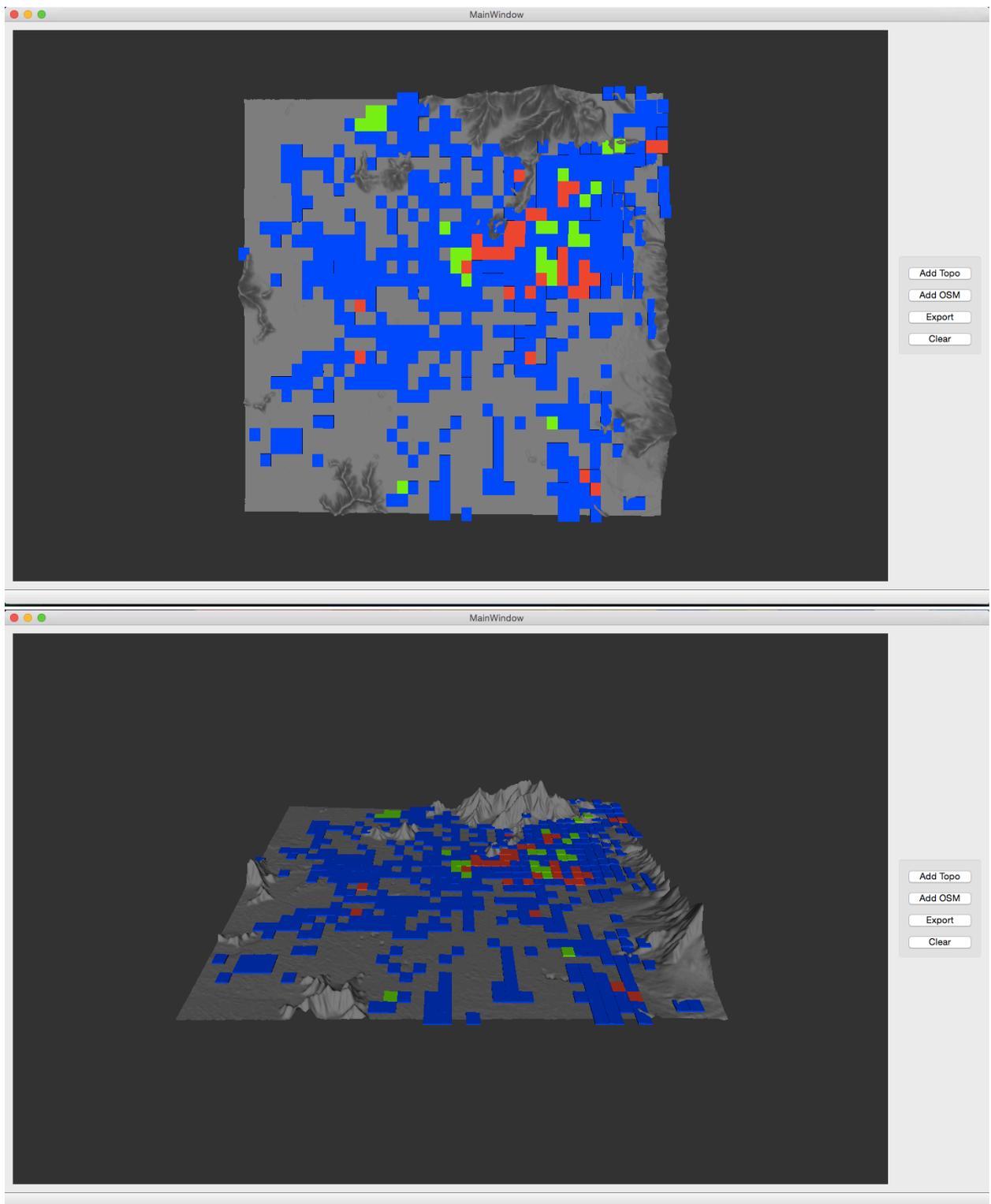


Figura 3.15: Toma de imágenes de la topografía y el dosel urbano generado desde dos perspectivas distintas.



Figura 3.16: Toma de imágenes de la topografía y el dosel urbano generado desde dos perspectivas distintas.

Capítulo 4

Aplicación Sistema Doseles Urbano

El presente capítulo trata sobre la aplicación de **CityCanopy** utilizando como entrada distintos mapas obtenidos desde OpenStreetMap. Se generan doseles a distintas resoluciones y criterios, para de este modo poder ver por un lado la información generada y, por otro, los alcances y la eficiencia que tiene la implementación actual a medida que varía la grilla del dosel urbano generado.

4.1. Definición

El objetivo de la aplicación de **CityCanopy** es poder observar cómo se comporta actualmente la implementación computacional utilizando dos conjuntos de datos provenientes de OpenStreetMap, los cuales representan a las ciudades de **Santiago, Chile**, y **Buenos Aires, Argentina**. Para ello se realizan variaciones en dos aspectos mencionados en la sección 3.5.1: el criterio de determinación de la grilla final y la variación del ancho de grilla.

Los dos criterios definidos para determinar la categoría final de cada punto de la grilla se exponen a continuación:

1. **Primer criterio:** suponiendo que se tienen n elementos de las tres categorías. Se asigna la categoría al punto según la aparición de al menos un elemento en orden de prioridad: COMERCIAL, ALTO RESIDENCIAL y RESIDENCIAL. Es decir, si existe al menos un elemento que sea COMERCIAL, entonces la categoría del punto es COMERCIAL. Si no, se sigue con la siguiente en la fila de prioridad. Este criterio busca poder dar énfasis a las zonas con pocos datos o elementos. No tiene ciertamente una base estadística, pero se espera que las categorías de los distintos barrios sean en efecto bastante parecidos a las categorías de los pocos elementos que aparezcan en él. Por otro lado, es usual ver para la categoría comercial, que los barrios comerciales no contienen una gran cantidad de elementos de estos por sobre otros de tipo residencial, pero sí son lugares bastantes transitados y utilizados para temas de comercio. Con estas suposiciones se espera que este criterio pueda entregar un resultado aceptable.

2. **Segundo criterio:** suponiendo que se tienen n elementos de las tres categorías. Si el 10 % del total de elementos corresponde a categoría COMERCIAL, entonces la categoría del punto es comercial. Si no, si el 20 % corresponde a ALTO RESIDENCIAL, entonces la categoría es ALTO RESIDENCIAL. Si ninguna de las otras condiciones se cumple, pero existen elementos en la zona, entonces la categoría es RESIDENCIAL. Este criterio busca determinar de una forma más proporcional la manera en que se eligen las distintas categorías en la zona urbana. No tiene ciertamente una base estadística, al igual que la primera categoría, pero se espera que maneje de una mejor forma los barrios en donde aparezcan un mayor número de elementos, dado el reconocimiento que se utiliza. Por último, los valores porcentuales son medidas experimentales iniciales entregadas por la implementación, pero deben ser ajustados según la zona urbana que se utilice como entrada dado que cada ciudad presenta diferentes comportamientos.

Las resoluciones de grilla utilizadas se calculan a partir de un ancho de grilla inicial de $1\text{km} = 0.008333\text{decdeg}^1$. Luego se va reduciendo el tamaño de manera polinomial utilizando el factor $1/2^n$, en donde n varía entre $(1, 2, 3\dots)$, hasta llegar al máximo n que puede soportar la aplicación. Se espera que con esto se logre aumentar el número de puntos de la grilla lo más rápido posible y poder observar el orden del algoritmo $O(E) + O(A/R^2)$ expuesto en la sección 3.5.1.

4.2. Resultados

Se presentan a continuación los resultados obtenidos mediante la aplicación de **CityCanopy** sobre las ciudades de **Santiago, Chile**, y **Buenos Aires, Argentina**. Los datos extraídos desde la plataforma de OpenStreetMap se pueden visualizar en las Figuras 4.1 y 4.6. La Tabla 4.1 muestra un resumen de las características de los distintos archivos que se utilizaron para generar el dosel urbano, tales como el tamaño del fichero *xml* descargado (en MB), el número de líneas que posee este fichero, el número de elementos que poseen (*node*, *way*, *relation*) y el área total que cubren tanto en medidas decdeg^2 como en km^2 . Las Figuras 4.2 y 4.7 muestran la aplicación de **CityCanopy** con los dos criterios existentes sobre los dos conjuntos de datos. Las Figuras 4.3 y 4.8 muestran los resultados al aplicar el primer criterio utilizando distintos anchos de grilla definidos en la sección 4.1, viéndose por orden de mayor a menor, de izquierda a derecha y luego de arriba para abajo. Las Figuras 4.4 y 4.9 muestran el tiempo de ejecución del algoritmo de generación del dosel urbano con respecto a la variación del ancho de grilla. Las Figuras 4.5 y 4.10 muestran el tiempo de ejecución del algoritmo de generación del dosel urbano con respecto a la cantidad de puntos de la grilla.

Ciudad	Tamaño fichero	Nro.Líneas	Nro.Elementos	Área decdeg^2	Área km^2
Santiago	172 MB	2847272	792205	0.0873	1081
Buenos Aires	245 MB	3919393	1176109	0.5206	6451

Tabla 4.1: Tabla características de archivos provenientes de OpenStreetMap

¹Los mapas digitales usualmente trabajan en valores en grados decimales (decimal degrees), en donde 111.32m son equivalentes a 0.001 grados. Si lo pasamos a 1km, esto corresponde aproximadamente a 0.008333decdeg.

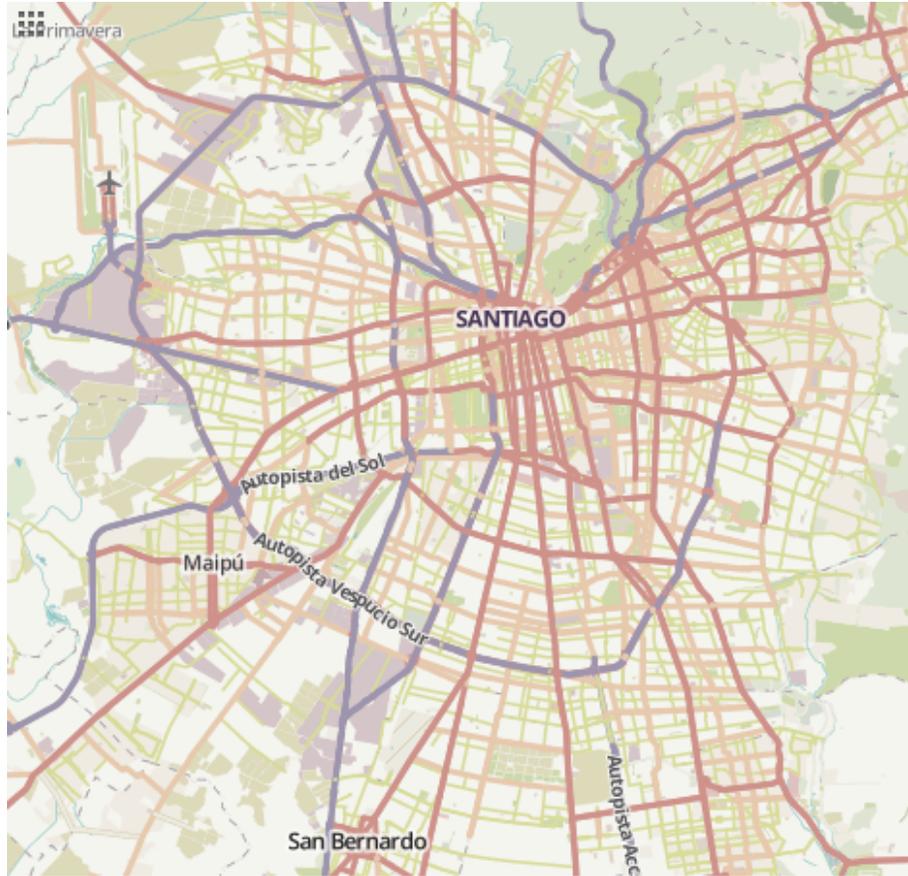


Figura 4.1: OpenStreetMap: Ciudad de Santiago, Chile.

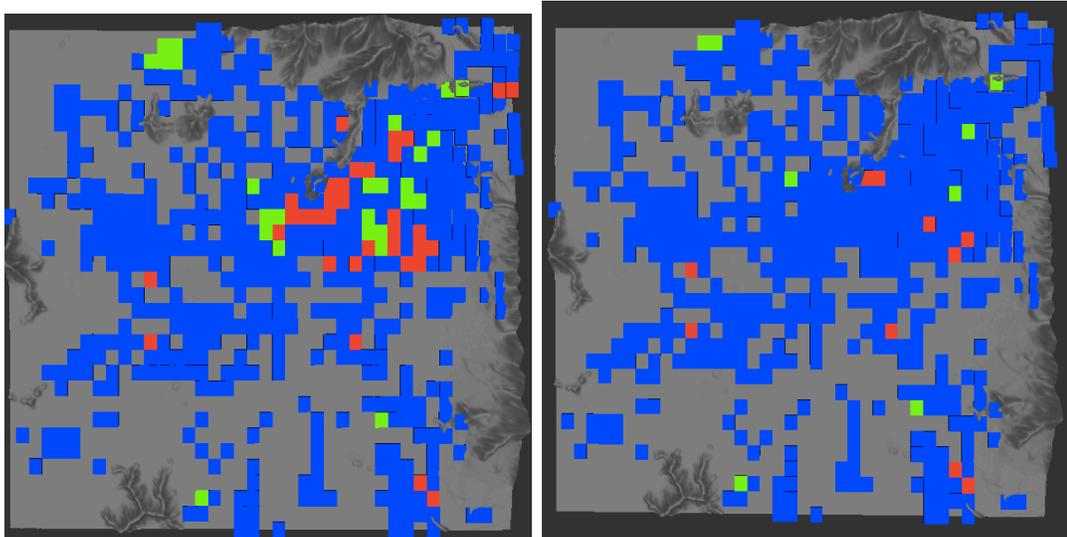


Figura 4.2: Aplicación del primero (izquierda) y segundo (derecha) criterio sobre Santiago.

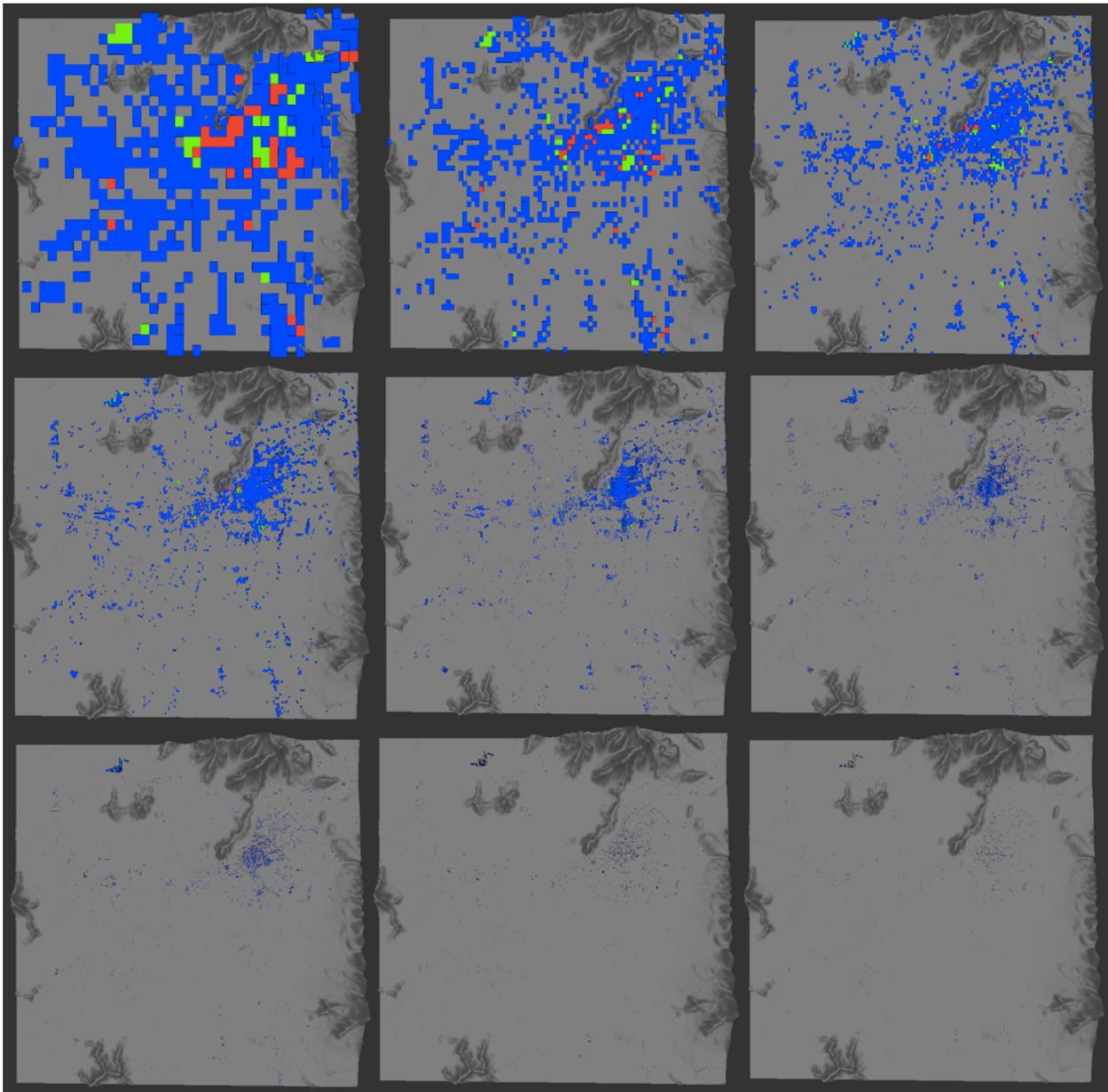


Figura 4.3: Variación de tamaño de grilla en factor $1/2^n$ en Santiago.

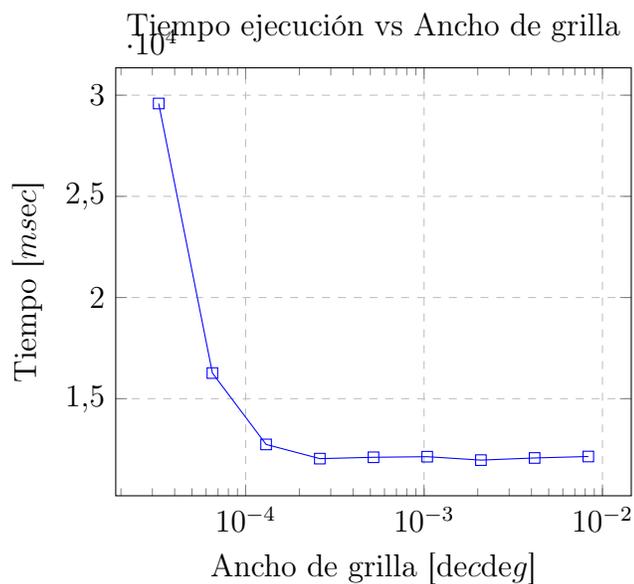


Figura 4.4: Mediciones de tiempo de ejecución del algoritmo sobre Santiago con respecto al ancho de grilla, con eje X en escala logarítmica.

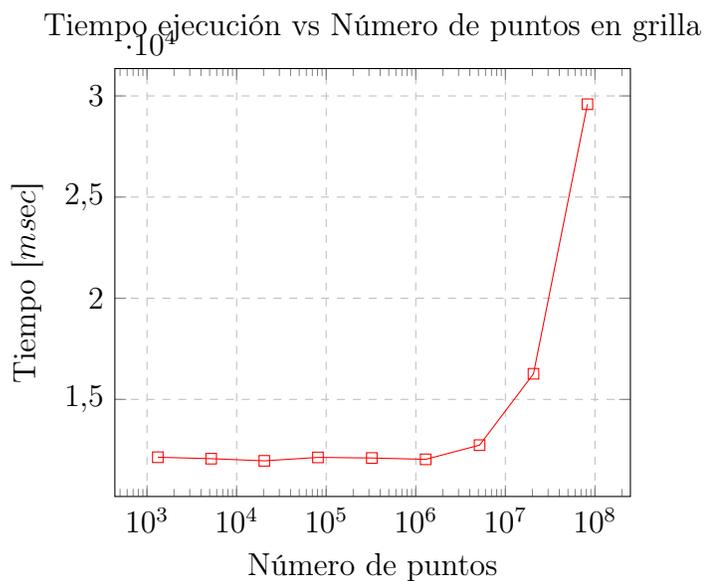


Figura 4.5: Mediciones de tiempo de ejecución del algoritmo sobre Santiago con respecto al número de puntos generados, con eje X en escala logarítmica.

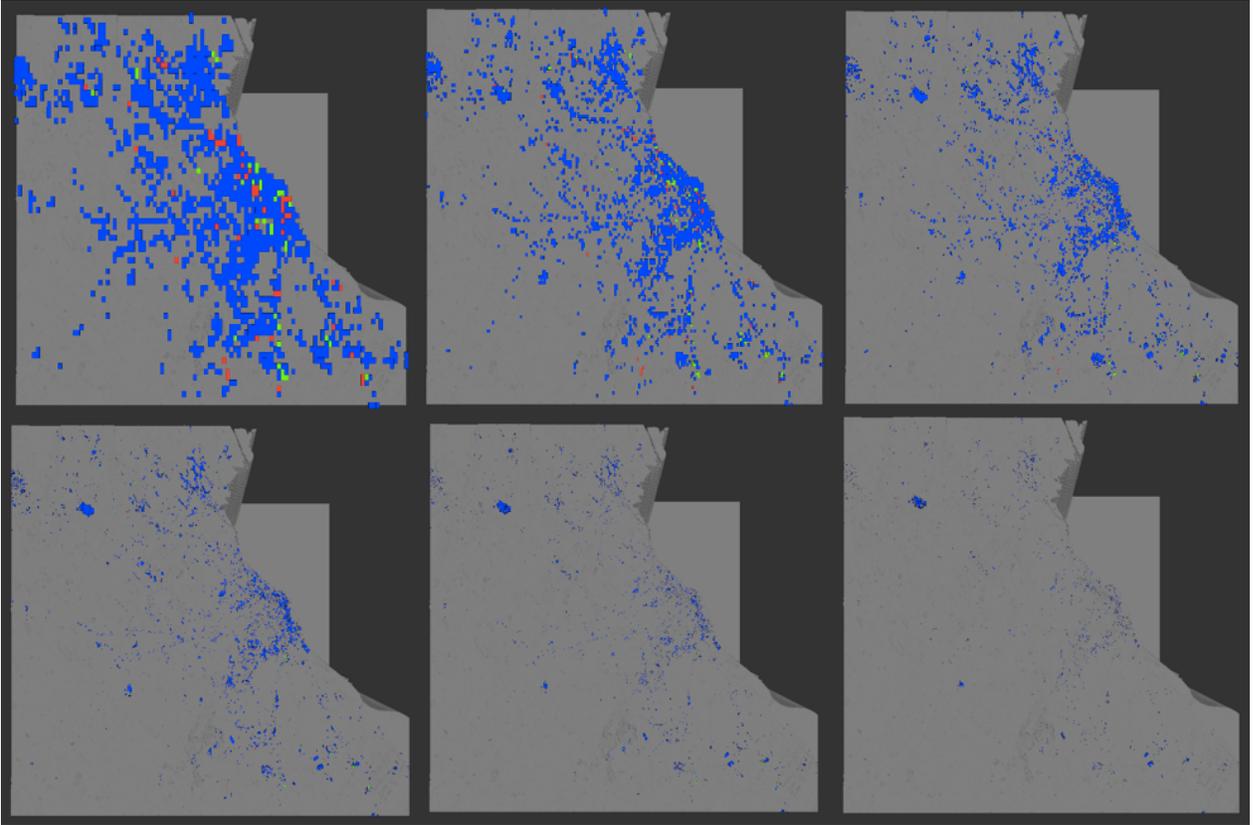


Figura 4.8: Variación de tamaño de grilla en factor $1/2^n$ en Buenos Aires.

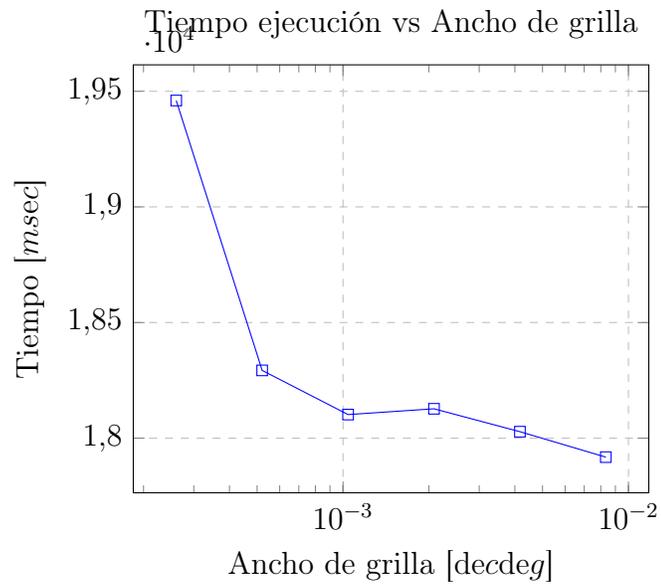


Figura 4.9: Mediciones de tiempo de ejecución del algoritmo sobre Buenos Aires con respecto al ancho de grilla, con eje X en escala logarítmica.

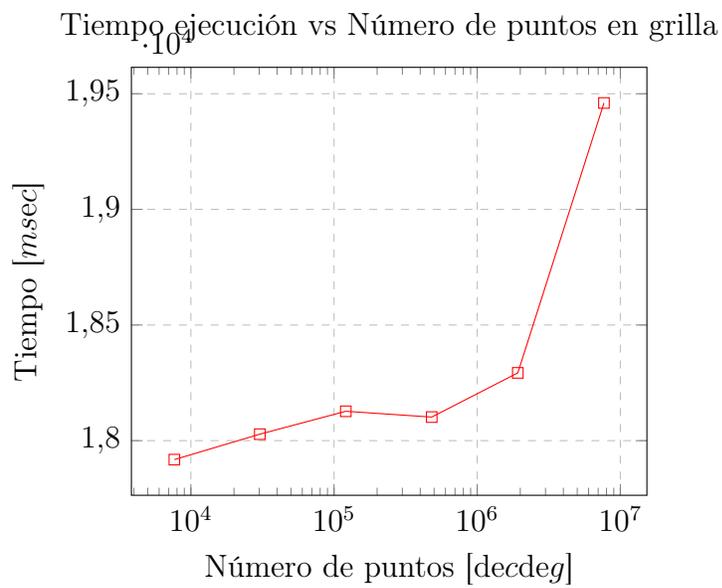


Figura 4.10: Mediciones de tiempo de ejecución del algoritmo sobre Buenos Aires con respecto al número de puntos generados, con eje X en escala logarítmica.

4.3. Verificación dosel urbano Santiago

Para poder verificar los datos del dosel urbano generado por **CityCanopy** para Santiago, se debe realizar una comparación con un set de datos que sean lo más aproximado posible con la realidad. Para ello, se generó de forma manual un archivo con los datos de uso de suelo de la ciudad utilizando la misma resolución de grilla (**1km aprox**) y las mismas coordenadas de zona observada. La forma en que se completó esta información, fué en primera instancia utilizar *GoogleMaps* y dividir la zona de estudio en la grilla propuesta utilizando las mismas coordenadas que el dosel urbano generado. Posteriormente se observó a través de *Google StreetView* qué tipo de categoría urbana correspondía cada punto de esta grilla (bajo residencial, alto residencial y comercial) para luego ir completando la información dentro del archivo generando una matriz lo más aproximada posible. La Figura 4.11 es la información final obtenida del procedimiento anteriormente explicado. Los distintos colores corresponden a las categorías urbanas, en donde azul es bajo residencial, verde representa alto residencial y rojo equivale a comercial.

Utilizando estos datos se comparan con los obtenidos por **CityCanopy** en la Figura 4.12. El objetivo principal de la comparación se basa en observar cuanta información de los datos generados manualmente fue detectada de igual forma por la implementación, o cuanto se asemeja la información generada por el sistema con respecto a nuestra mejor aproximación. Dado que los dos conjuntos de datos utilizan la misma cantidad de puntos y dimensiones de grilla, se puede realizar las distintas comparaciones utilizando la información de cada punto. Para un mejor entendimiento, se define como **conjunto-real** al dosel generado manualmente, y **conjunto-generado** al dosel generado por la implementación. Como definición se va a llamar cobertura *COB* de un conjunto, a la cantidad de puntos de un conjunto en donde se tenga alguna categoría urbana. Los siguientes aspectos se definen para poder ver la similitud:

1. **Porcentaje acierto de cobertura:**

$$\frac{COB_{conjunto-generado} \cap COB_{conjunto-real}}{COB_{conjunto-real}}$$

2. **Porcentaje acierto de categoría y cobertura coincidente:**

$$\frac{COB_{conjunto-generado} \cap_{\equiv categoria} COB_{conjunto-real}}{COB_{conjunto-generado} \cap COB_{conjunto-real}}$$

3. **Porcentaje de innovación:**

$$\frac{COB_{conjunto-generado} \cap (UNIVERSO - COB_{conjunto-real})}{COB_{conjunto-real}}$$

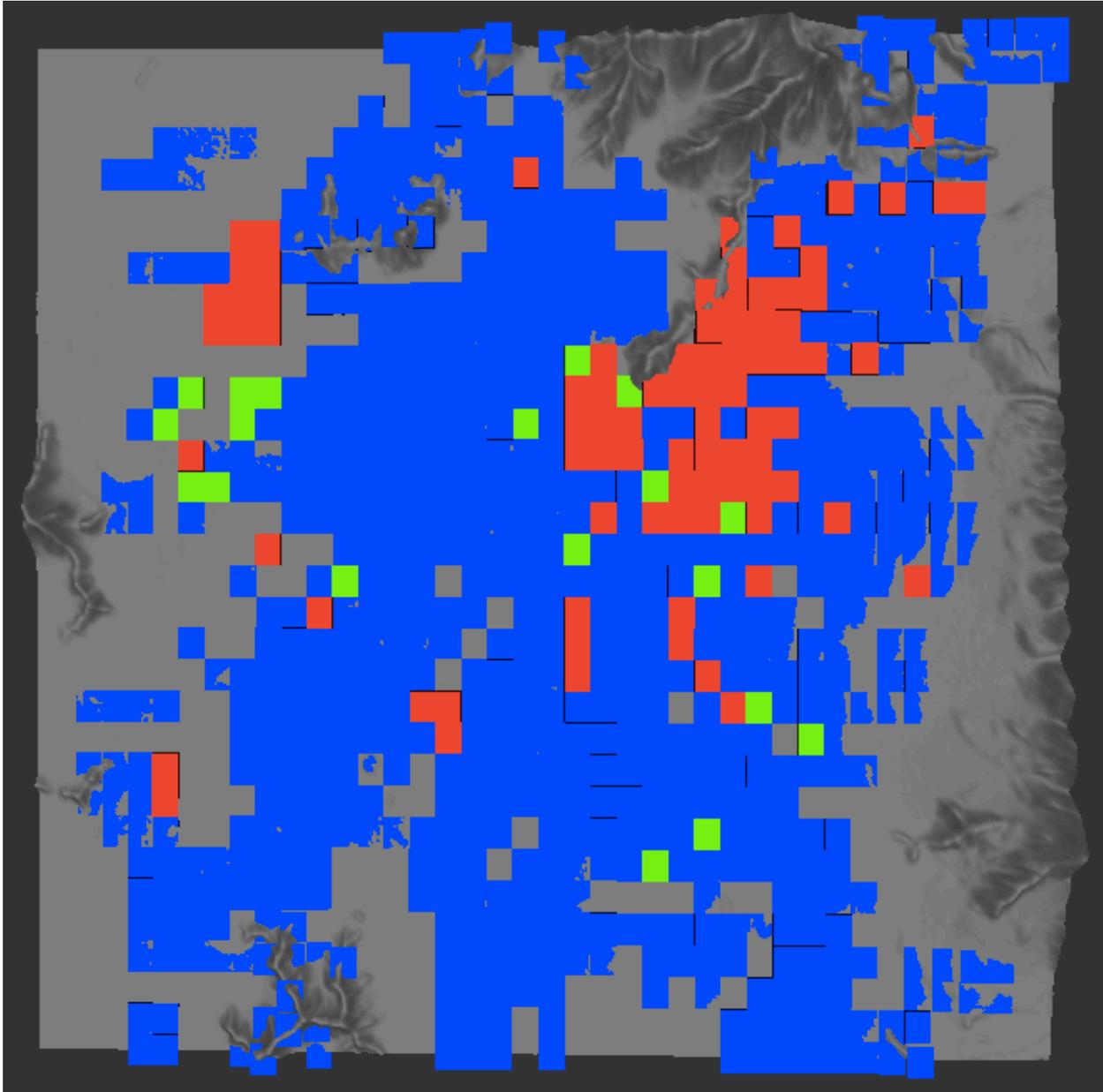


Figura 4.11: Dosel urbano generado manualmente.

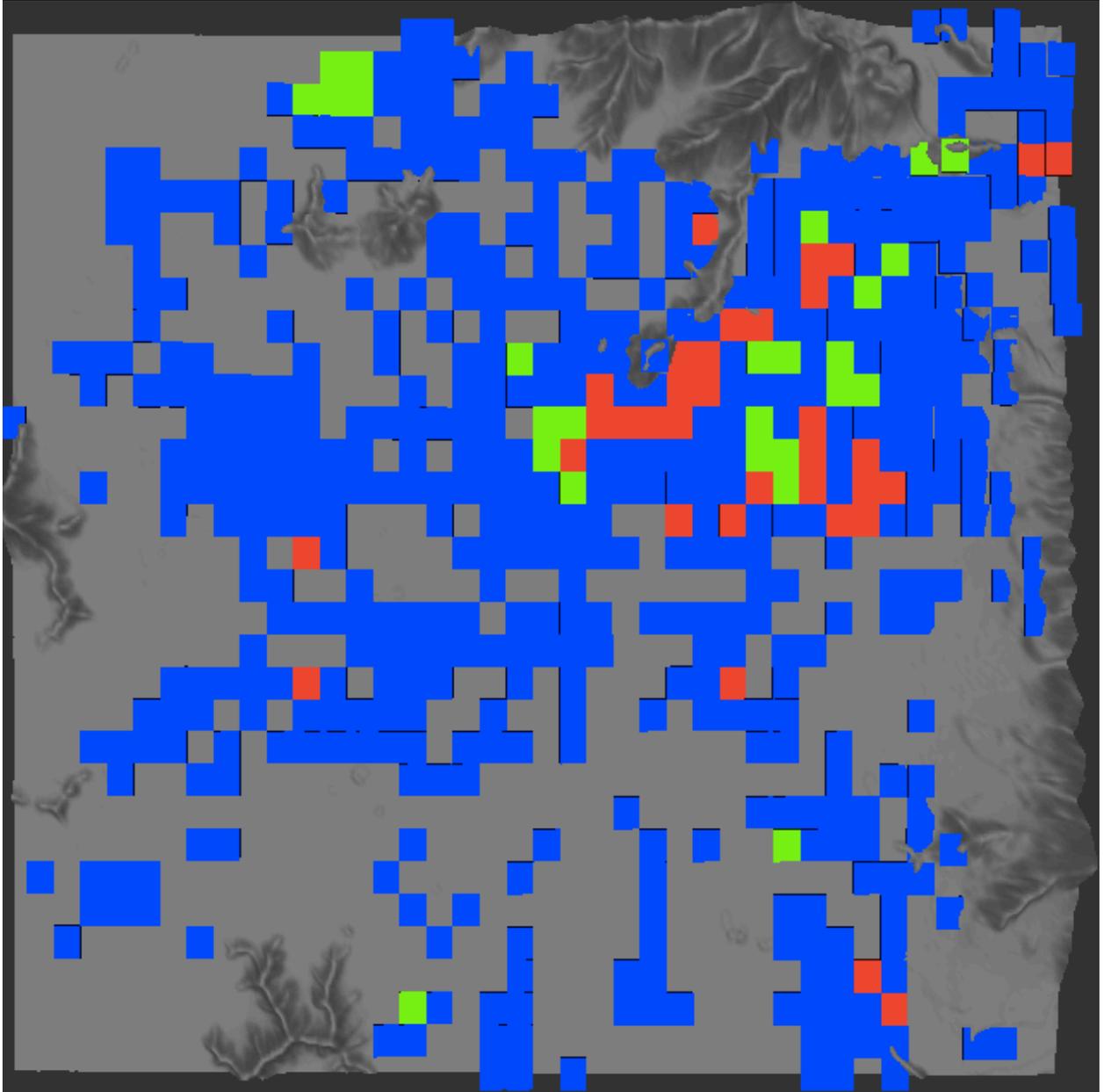


Figura 4.12: Doseil urbano generado por CityCanopy utilizando el primer criterio.

La Tabla 4.2 muestra los resultados obtenidos al comparar los doseles anteriormente mencionados. Se puede observar que según el porcentaje de acierto de cobertura los datos generados automáticamente por el sistema logran cubrir un valor cercano al 54 % de los datos esperados. Esta cifra muestra ser una falencia en cuanto a la completitud de los datos, dado que en gran parte esto puede ser provocado por la falta de información que provee la fuente de los datos. Si se observa la Figura 4.12 en comparación con la Figura 4.11 se pueden apreciar varios espacios sin completar, los cuales corresponden a zonas alejadas del centro de la ciudad, en donde posiblemente existan menos datos. Sin embargo, el porcentaje de acierto de categoría y cobertura coincidente indica que el 78 % de los datos cubiertos logró estimar de manera certera la categoría, considerando esto un aspecto positivo dado la automatización del proceso. Esto puede deberse en cierta forma a dos factores. Uno es la cantidad de zonas residenciales bajas que existen, dado que la ciudad posee un centro muy marcado para lo comercial y el resto se conforman por residencias, por lo que existe mayor probabilidad de encontrar zonas residenciales bajas que cualquier otra categoría. El otro es la cantidad de información que se encuentra en este centro comercial marcado, lo que permite diferenciarlo mejor de otras zonas, y por ende existe una probabilidad elevada de encontrar categorías comerciales al centro de la ciudad. Por último, el porcentaje de innovación del 17 % permite ver cuanta información nueva se generó a partir del sistema automático de detección.

Total de puntos grilla	1332
Total puntos con categoría conjunto-real	773
Total puntos con categoría conjunto-generado	551
Porcentaje de acierto de cobertura	54 %
Porcentaje de acierto de categoría y cobertura coincidente	78 %
Porcentaje de innovación	17 %

Tabla 4.2: Resultados de comparación entre el dosel generado manualmente y el dosel generado por el sistema.

Capítulo 5

Impacto sobre flujo atmosférico

El presente capítulo trata sobre la incorporación de los datos generados por **CityCanopy** para la ciudad de Santiago de Chile, dentro de distintas simulaciones atmosférica utilizando el simulador WRF. Se considera la utilización de distintos modelos urbanos para la obtención de los resultados, para luego compararlos y poder observar la influencia de la información generada dentro de los distintos modelos.

5.1. Contexto

La experimentación utiliza como base los archivos de configuración de un trabajo de tesis previo realizado por Andrea Orfanoz [Orfanoz, 2015], en donde se utilizó el simulador *WRF* para llevar a cabo corridas o simulaciones sobre un dominio que contiene a la Ciudad de Santiago de Chile.

El objetivo principal del presente experimento es poder observar los efectos causados por el acoplamiento del dosel urbano generado específicamente para la ciudad de Santiago de Chile. Para ello se realiza primero una **corrida control** con los efectos urbanos desactivados como base para las comparaciones. Luego se llevan a cabo cuatro corridas distintas con los efectos urbanos activados: dos utilizando el modelo **UCM** con y sin acoplamiento de los datos generados; y dos utilizando el modelo **BEP** con y sin acoplamiento de los datos generados. Por motivos de orden, las corridas se van a etiquetar como: **slab** (modelo sin efectos urbanos), **ucm**, **osm-ucm**, **bep** y **osm-bep**, en donde el prefijo **osm-** corresponde al acoplamiento de los datos generados de la ciudad de Santiago. Finalmente se observan las diferencias entre los valores obtenidos de la corrida control y cada una de las corridas llevadas a cabo.

5.2. Configuración

Para el caso del presente trabajo, se utiliza el simulador *WRF* en su versión **3.7.1**. Los archivos de configuración *namelist.wps* y *namelist.input* se encuentran en el Apéndice A los cuales presentan algunos cambios con respecto a los archivos del trabajo previo. El periodo de simulación consta de 10 días entre las fechas 20/08/2013 y 29/08/2013. La Figura 5.1 muestra los tres dominios sobre los cuales se trabaja, en donde el dominio más grande es el primero, el mediano el segundo y el pequeño el tercero. La Figura 5.2 realiza un acercamiento al tercer dominio, observando la inclusión de la ciudad de Santiago de Chile. Se van a considerar los resultados obtenidos del dominio más pequeño, para poder observar los cambios generados por la consideración de los modelos urbanos.

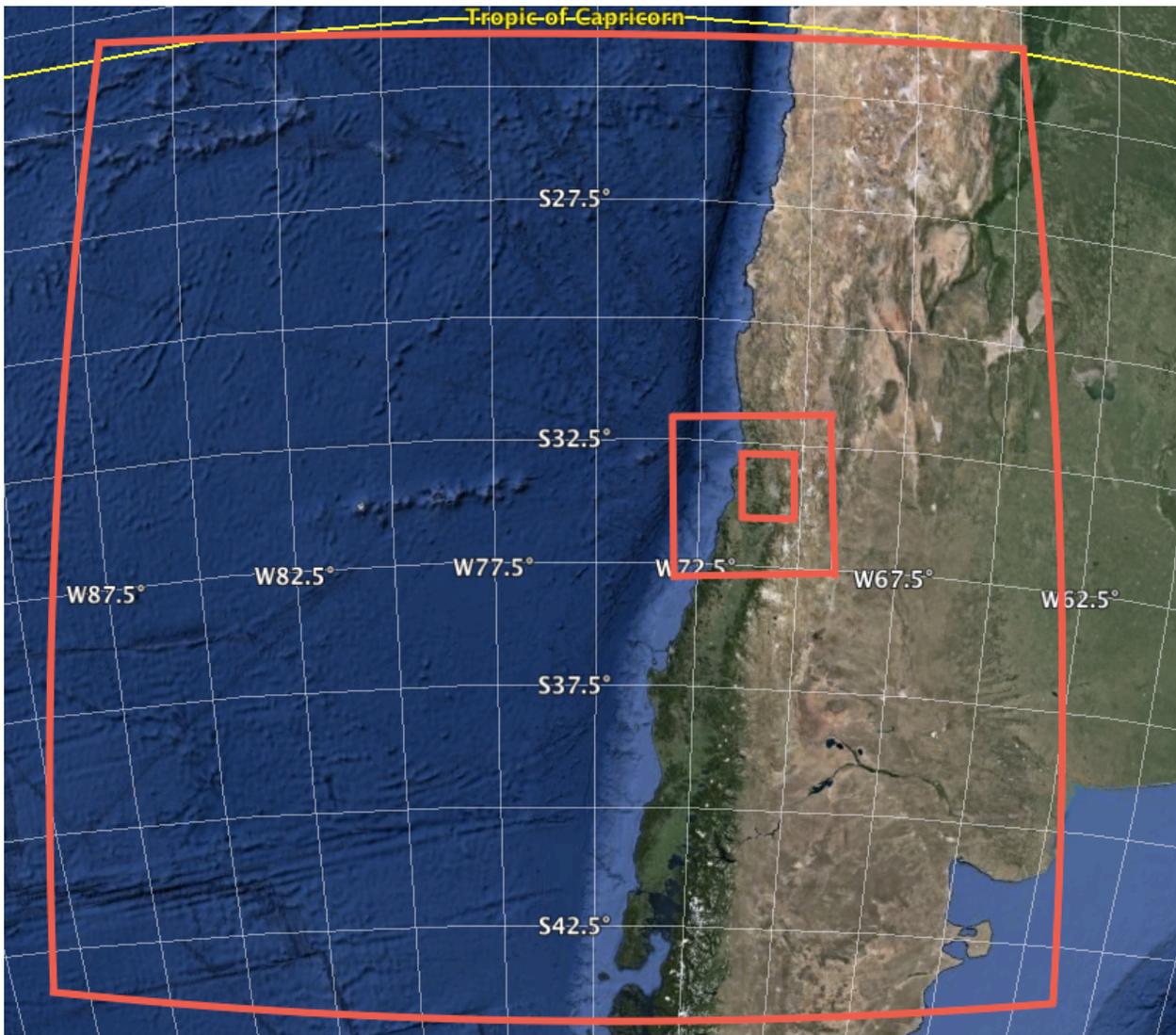


Figura 5.1: Dominios de la zona de estudio.



Figura 5.2: Acercamiento al tercer dominio de la zona de estudio.

La Figura 5.3 muestra los niveles verticales de la simulación, utilizando un corte orientación Oeste-Este que pasa por la mitad del tercer dominio. Se consideran para el caso 30 niveles verticales distintos, los cuales son generados automáticamente por el simulador.

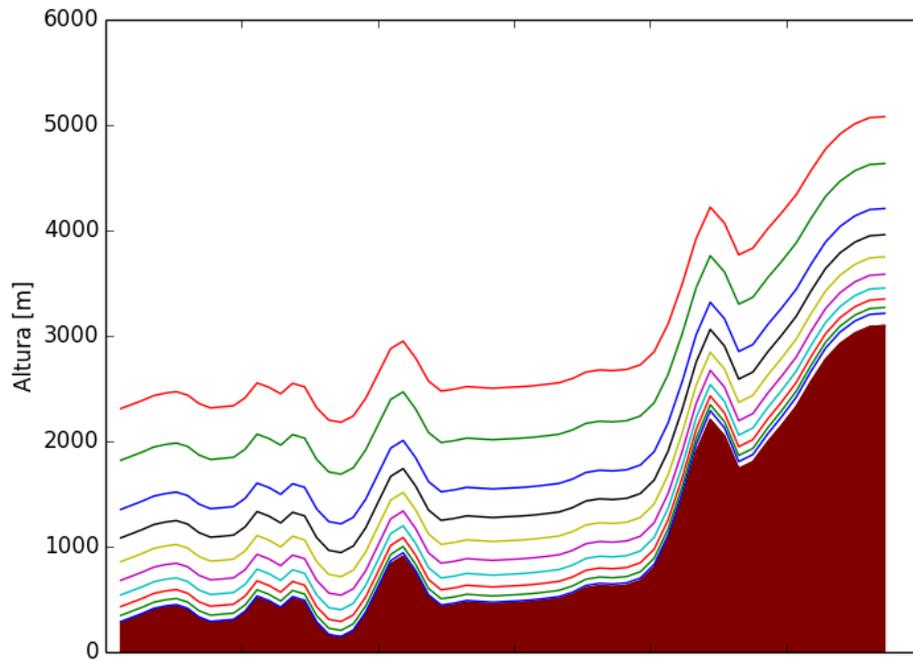
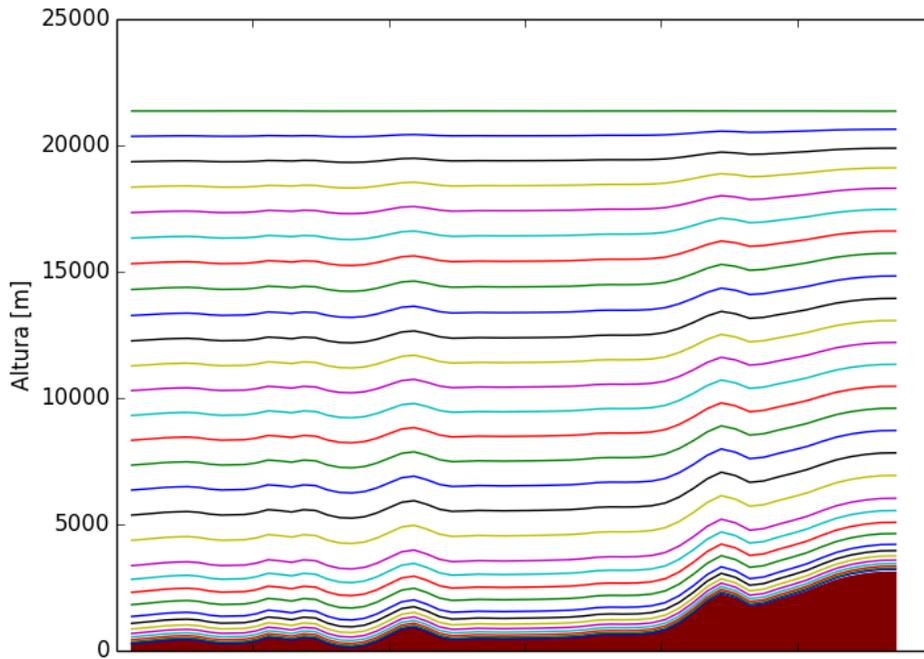


Figura 5.3: Niveles verticales de la configuración. La primera Figura muestra los 30 niveles verticales, mientras que la segunda realiza un acercamiento de los primeros 10 niveles.

Según sea el modelo urbano que se va a requerir, **UCM** o **BEP**, se deben realizar cambios en la configuración para poder activarlos dentro de la simulación. Estos cambios permiten al simulador utilizar distintos modelos físicos que se requieren para utilizar los modelos urbanos acoplados. Para una mayor información sobre los distintos parámetros se puede ver el manual de *WRF* [wrf, 2010b]. A continuación se presentan los parámetros necesarios para los experimentos que se realizan:

1. **sf_sfclay_physics**: determina el coeficiente de intercambio para predecir el comportamiento de cambio de energía y temperatura a 2 metros de la superficie, y de magnitud de viento a 10m de la superficie. Valores (2) para *Monin-Obukhov (Janjic Eta) scheme*, y (5) para *MYNN*.
2. **sf_surface_physics**: determina el modelo de capas de suelo a utilizar, en donde se predice el comportamiento térmico a partir del uso de suelo y las distintas capas. Valor (2) para *Unified Noah land-surface model*.
3. **bl_pbl_physics**: determina el comportamiento de mezcla y flujo de los gases dentro de la capa límite. Valores (5) para *MYNN 2.5 level TKE*, y (8) para *Bougeault and Lacarrere (BouLac) TKE*.
4. **sf_urban_physics**: determina el modelo urbano a utilizar vistos en la sección . Valores (1) para *UCM* y (2) para *BEP*.

La Tabla 5.1 muestra la configuración de las distintas corridas que se realizan.

Corrida	sf_sfclay_physics	sf_surface_physics	bl_pbl_physics	sf_urban_physics
slab	5	2	5	0
ucm	5	2	5	1
osm-ucm	5	2	5	1
bep	2	2	8	2
osm-bep	2	2	8	2

Tabla 5.1: Tabla de comparación entre distintos parámetros

5.3. Resultados

A continuación se presentan los resultados obtenidos de las distintas simulaciones corridas con los modelos expuestos. Se dividen los resultados en variables para poder observar de manera ordenada los efectos que se generan en los distintos modelos. En primera instancia se observan las diferencias de las variables a 2m de elevación, para el caso de temperatura, y a 10m de elevación, para el caso de magnitud de viento, por sobre la altura del terreno. Estas corresponden a las capas más bajas de la atmósfera observada, en donde se logran resaltar los efectos urbanos. Las diferencias entre los variables corresponden a la resta aritmética del promedio de los valores obtenidos en cada hora del día durante el período de simulación, entre las distintas corridas realizadas y la corrida control, es decir **a cada corrida realizada se le resta la corrida control**. Se eligen 2 momentos del día para poder observar el comportamiento de los resultados: uno que corresponde a las 03:00 horas para ver los efectos diurnos en las variables, y el otro, con 12 horas de diferencia, a las 15:00 horas para ver los efectos vespertinos. Luego, utilizando los mismo promedios y momentos del día, se observan las diferencias a través de un corte vertical de oeste a este que pasa por la zona media del dominio. Con esto se busca observar hasta que punto se puede o no ver el efecto que produce la inserción del modelo urbano dentro de la simulación.

Por último se exponen las series de tiempo en conjunto a datos observacionales de 4 estaciones meteorológicas distintas (Quilicura, Santiago, Puente Alto, Talagante) dispuestas en distintos puntos de la ciudad de Santiago. Esta información por estación fue obtenida desde la *Red de monitoreo de calidad del aire para la Región Metropolitana de Santiago* [red, 2015]. Se utilizan los valores de los resultados en cada punto geográfico en donde se tienen los datos reales, con el fin de observar que tan bien se ajustan los resultados de las simulaciones con respecto a la información real obtenida en el mismo período observado. Por temas de ajuste en los cálculos del simulador, se recomienda que los primeros 3 días de simulación no sean considerados, dado que los resultados en estos pueden contener variaciones y perturbaciones fuera de los esperados. Por esta razón, estos días no son considerados en los resultados que se exponen. Se adjuntan adicionalmente tablas con datos estadísticos por estación, en donde se puede observar la siguiente información:

1. **ObsMean**: valor promedio de la variable obtenido de las observaciones de la estación durante el periodo.
2. **ModelMean**: valor promedio de la variable obtenido de los resultados de la corrida en el periodo y punto geográfico de la estación.
3. **BIAS**: diferencia entre **ObsMean** y **ModelMean**. Representa cuanto se aleja la predicción de los datos observados.
4. **RMSE**: valor de la raíz cuadrada del error cuadrático medio de la variable.

En el Apéndice B se encuentran las series de tiempo separadas por corridas para una mejor visualización.

5.3.1. Temperatura

Las Figuras 5.4, 5.5, 5.6, 5.7 muestran la diferencia existente entre las distintas corridas y la corrida control, utilizando los promedios de temperatura a 2m de elevación dentro del periodo completo de simulación a las 3:00 horas y a las 15:00 horas.

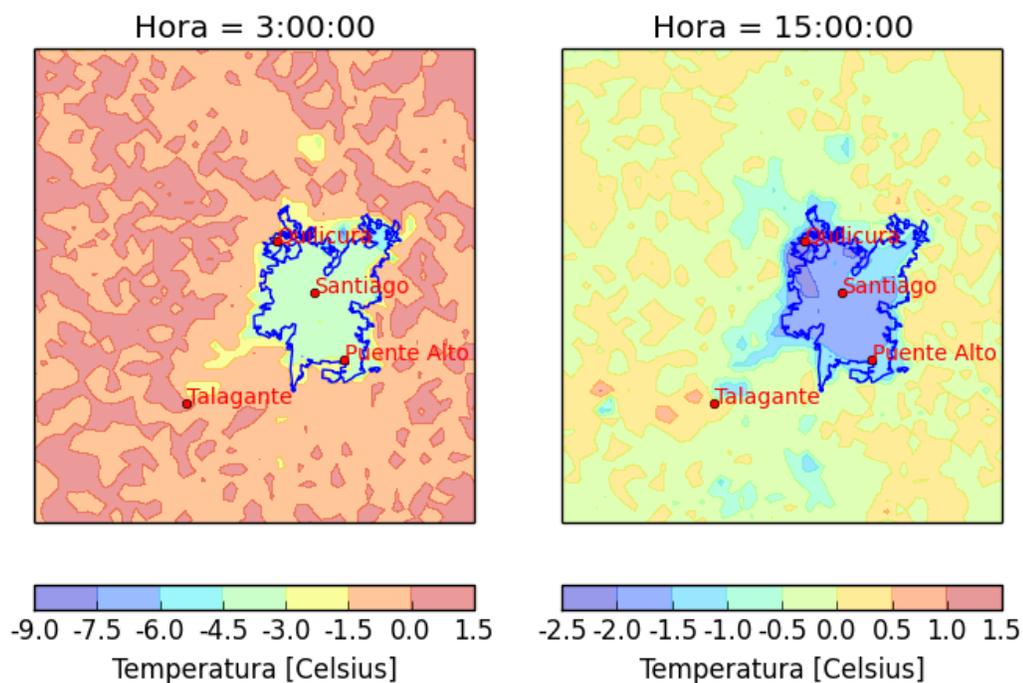


Figura 5.4: Diferencia entre valores promedio de temperatura de UCM y SLAB, a las 03:00 y 15:00 hrs, a 2m de elevación.

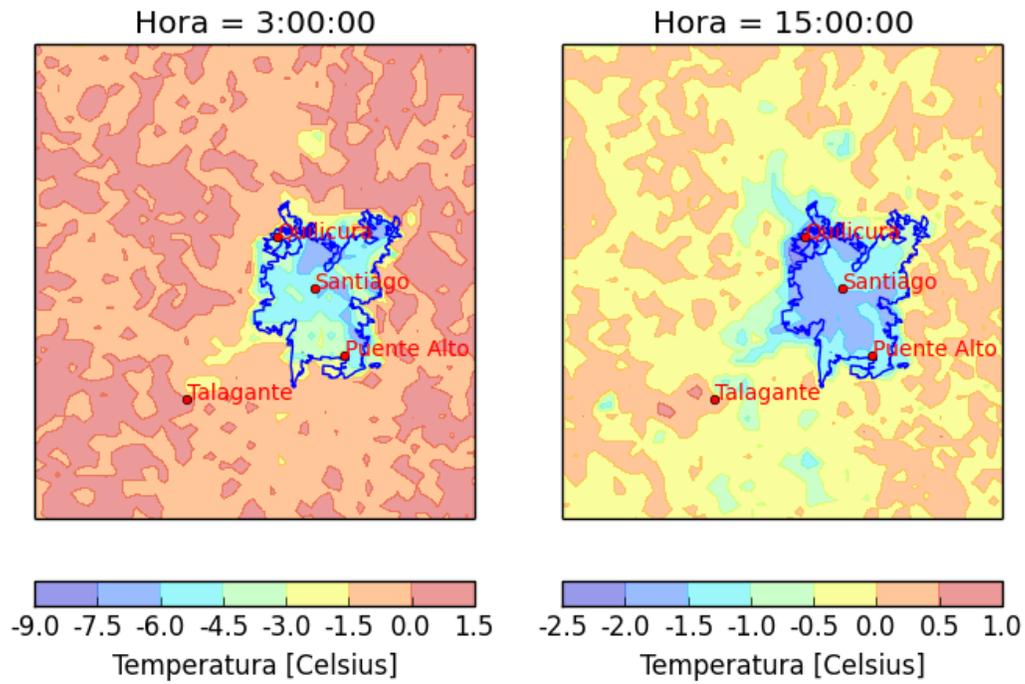


Figura 5.5: Diferencia entre valores promedio de temperatura de OSMUCM y SLAB, a las 03:00 y 15:00 hrs, a 2m de elevación.

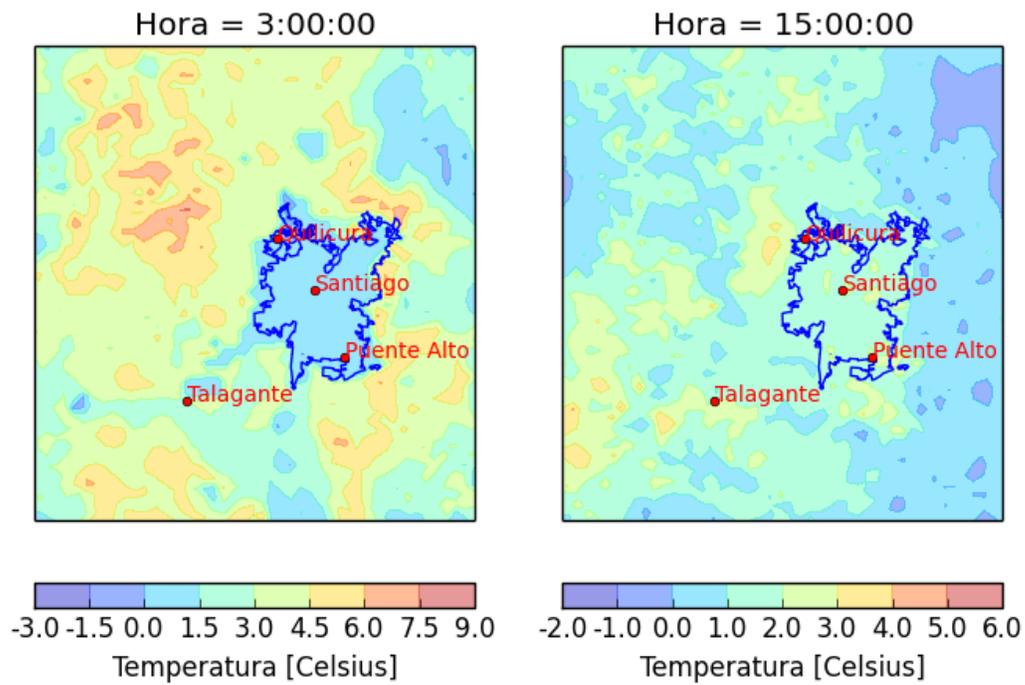


Figura 5.6: Diferencia entre valores promedio de temperatura de BEP y SLAB, a las 03:00 y 15:00 hrs, a 2m de elevación.

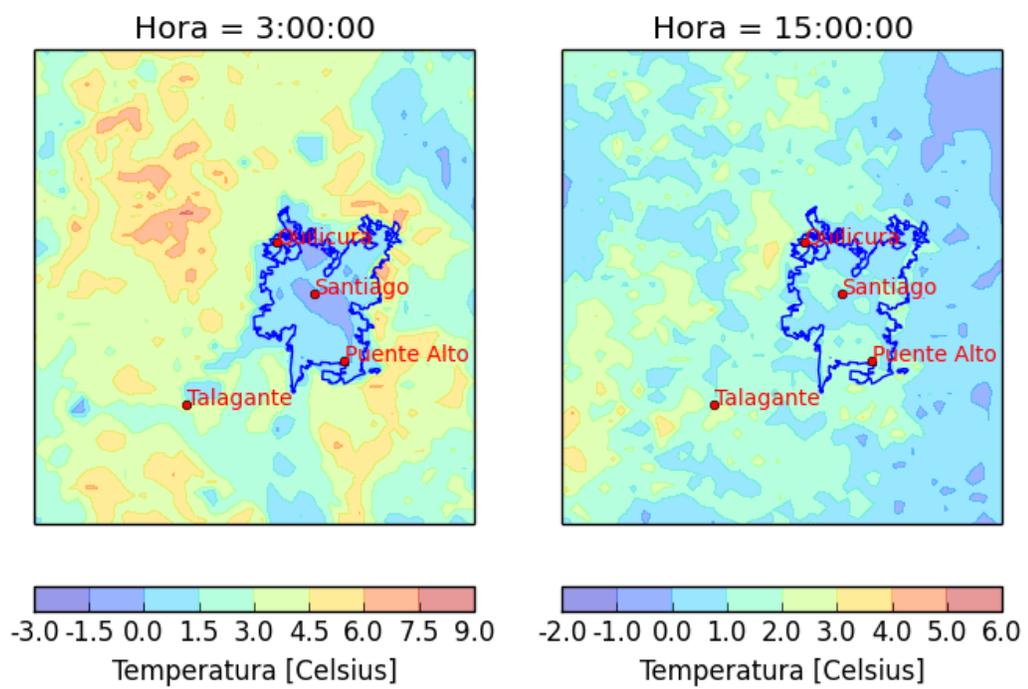


Figura 5.7: Diferencia entre valores promedio de temperatura de OSMBEP y SLAB, a las 03:00 y 15:00 hrs, a 2m de elevación.

Las Figuras 5.8, 5.9, 5.10, 5.11 muestran un corte vertical de oeste a este en la zona media del dominio, utilizando la diferencia de promedios de temperaturas a las 3:00 horas y a las 15:00 horas, entre las distintas corridas y la corrida control.

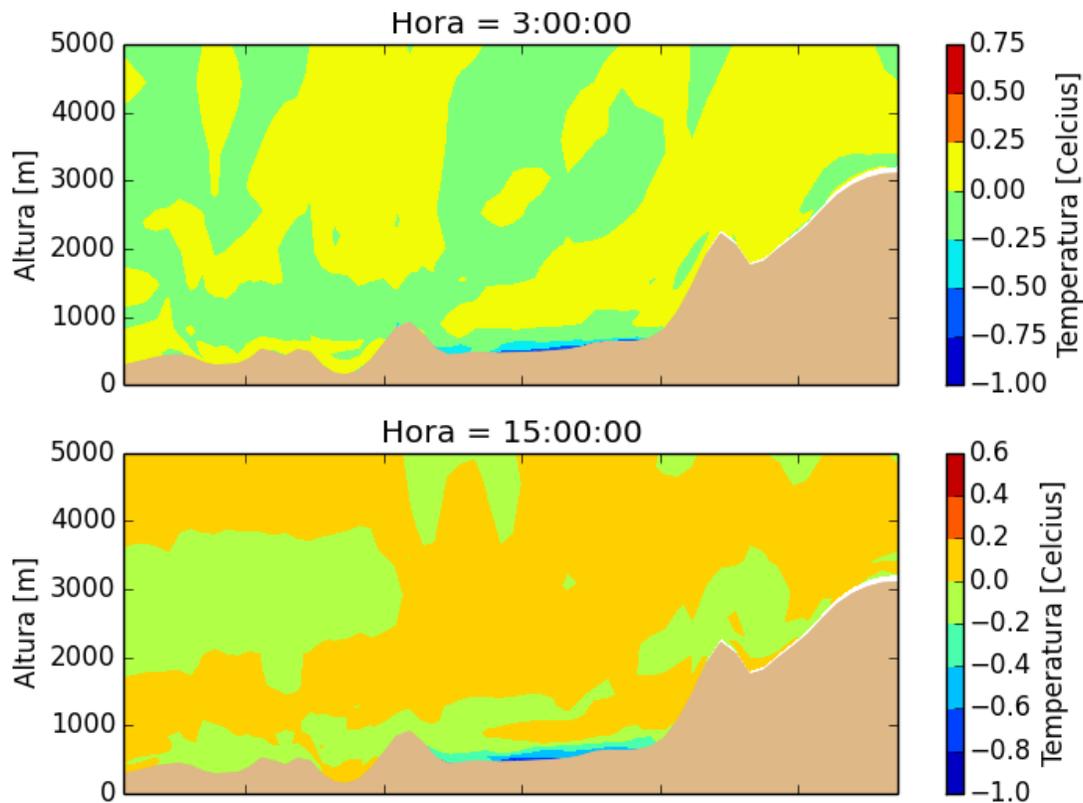


Figura 5.8: Corte vertical de la diferencia entre valores promedio de temperatura de UCM y SLAB, a las 03:00 y 15:00 hrs.

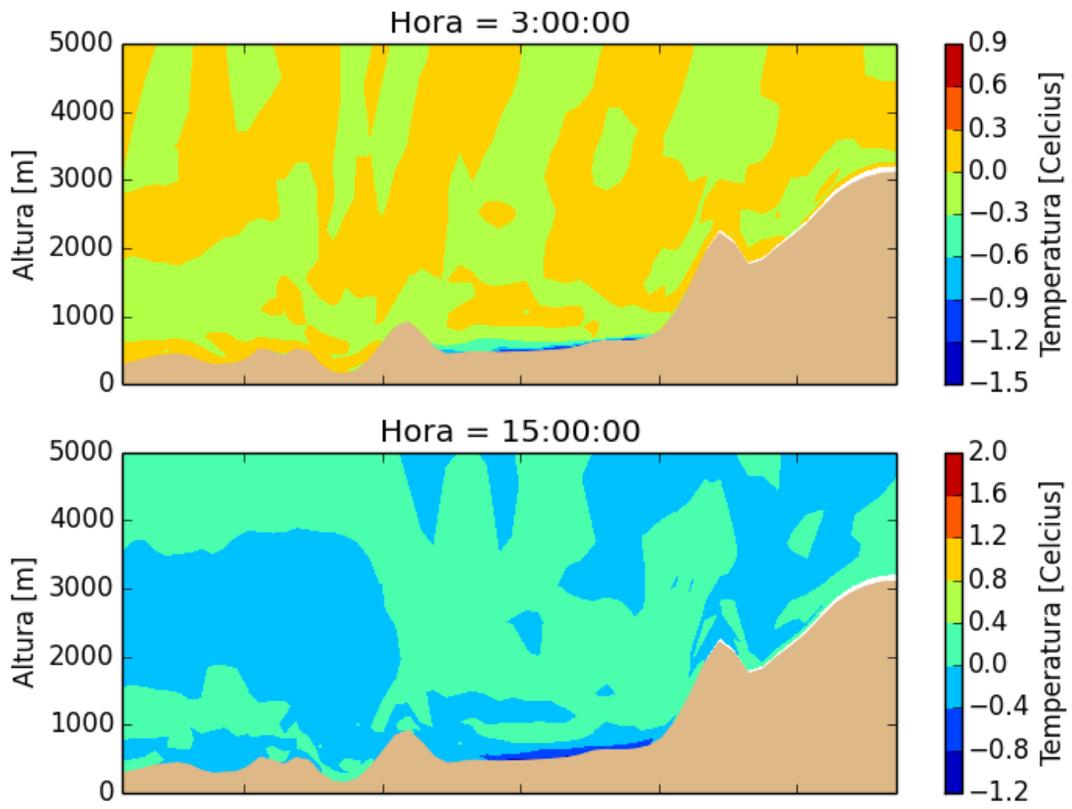


Figura 5.9: Corte vertical de la diferencia entre valores promedio de temperatura de OS-MUCM y SLAB, a las 03:00 y 15:00 hrs.

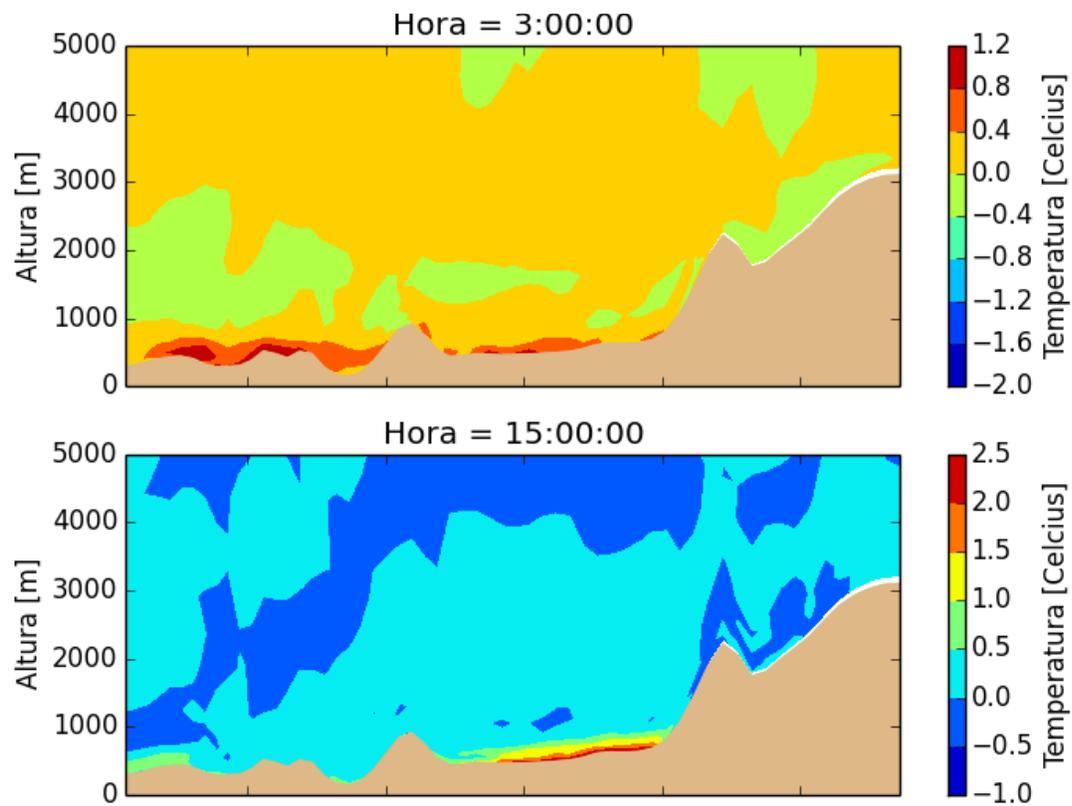


Figura 5.10: Corte vertical de la diferencia entre valores promedio de temperatura de BEP y SLAB, a las 03:00 y 15:00 hrs.

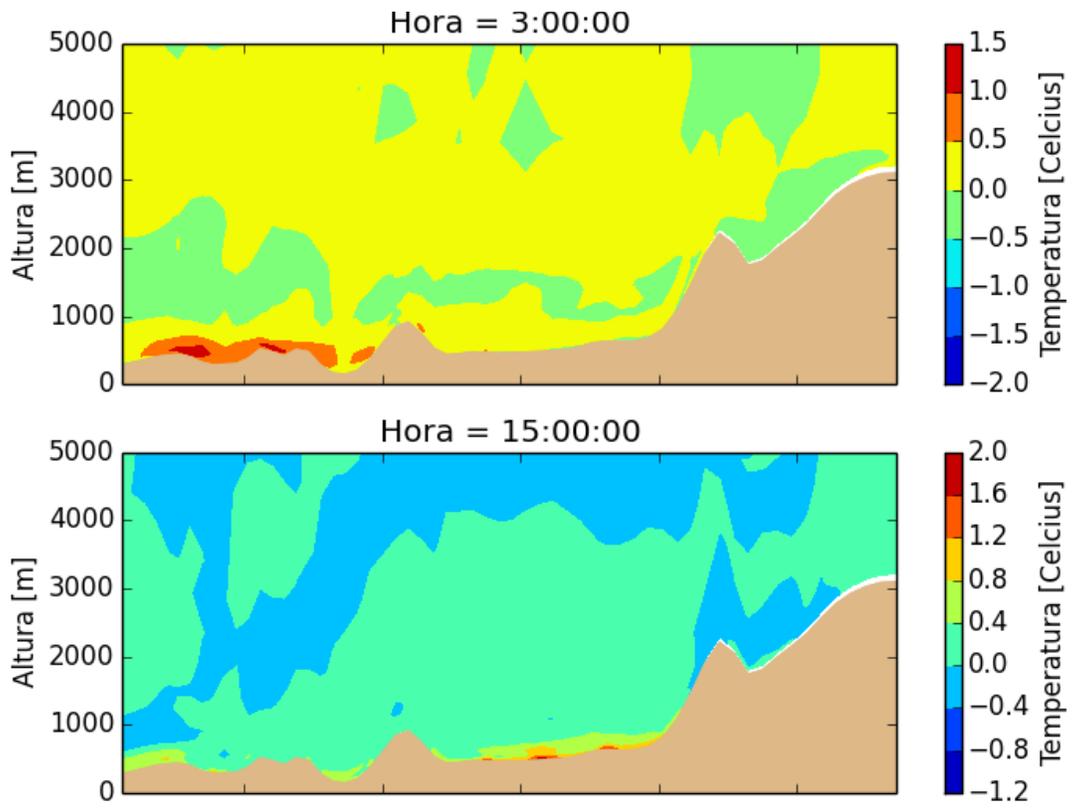


Figura 5.11: Corte vertical de la diferencia entre valores promedio de temperatura de OSM-BEP y SLAB, a las 03:00 y 15:00 hrs.

Las Figuras 5.12, 5.13, 5.14, 5.15 muestran las series de tiempo para las estaciones dispuestas en Puente Alto, Quilicura, Santiago y Talagante. En azul se puede ver el valor observado para las fechas indicadas, mientras que en los distintos colores se representan los valores obtenidos por las simulaciones.

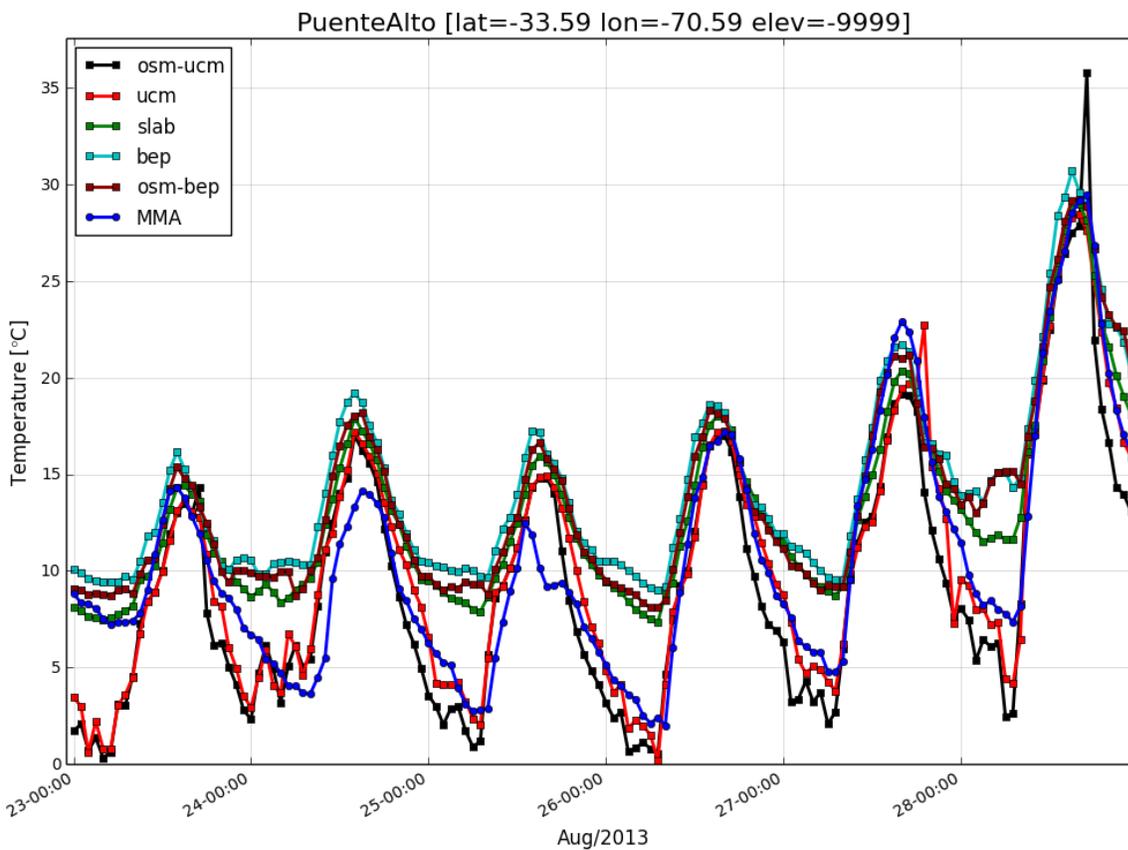


Figura 5.12: Serie de tiempo de variable Temperatura para estación Puente Alto en el periodo entre 23-08-2013 y 29-08-2013.

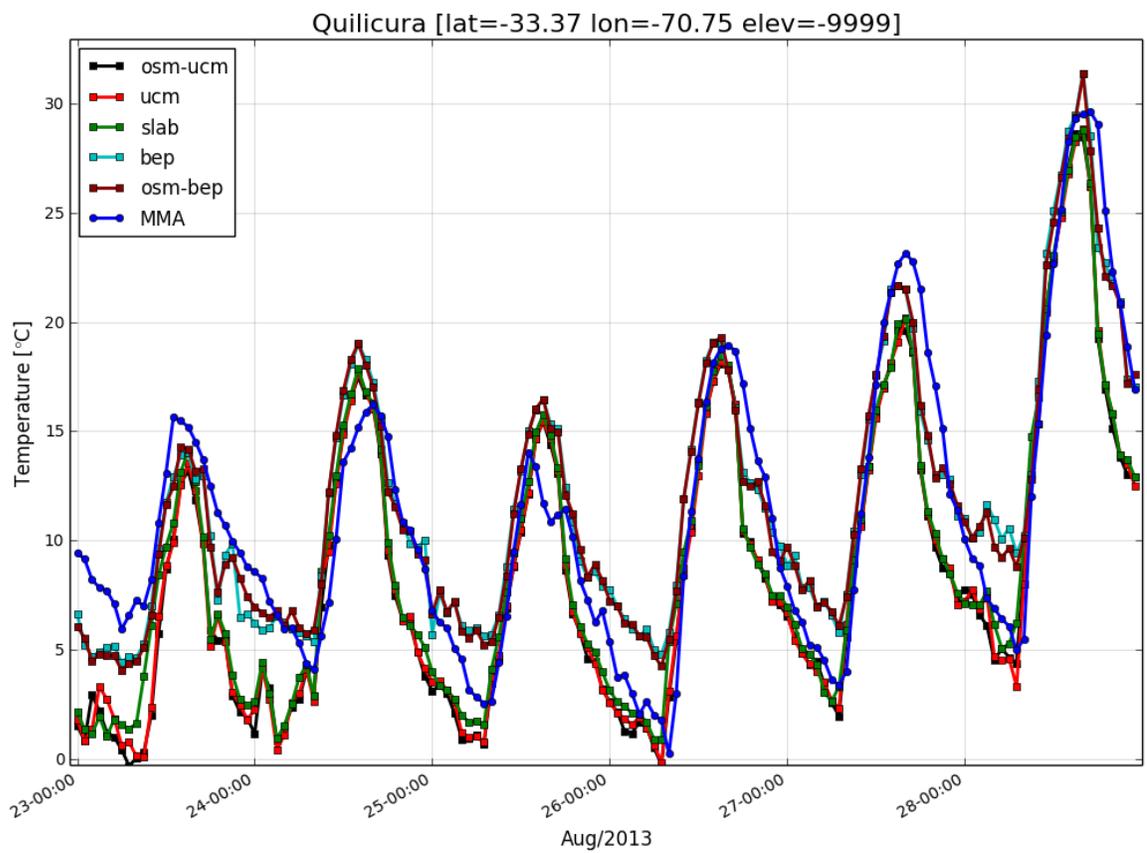


Figura 5.13: Serie de tiempo de variable Temperatura para estación Quilicura en el periodo entre 23-08-2013 y 29-08-2013.

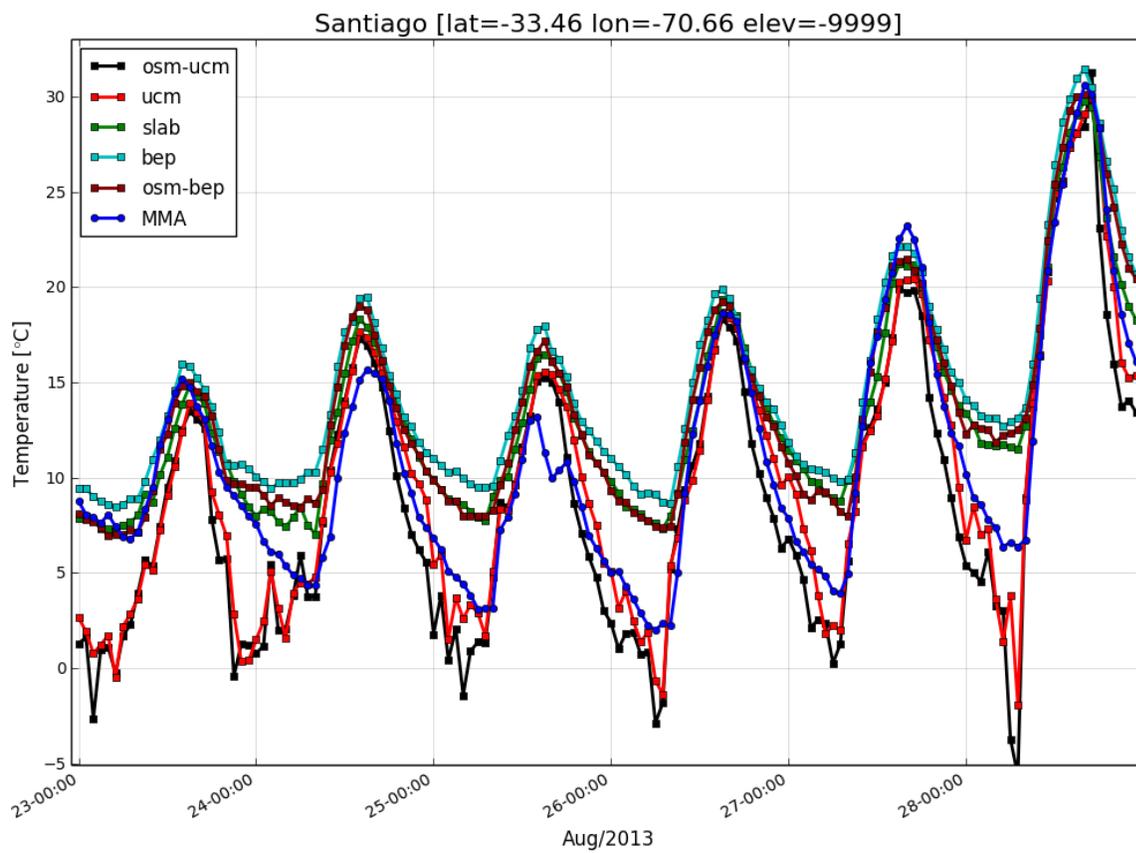


Figura 5.14: Serie de tiempo de variable Temperatura para estación Santiago en el periodo entre 23-08-2013 y 29-08-2013.

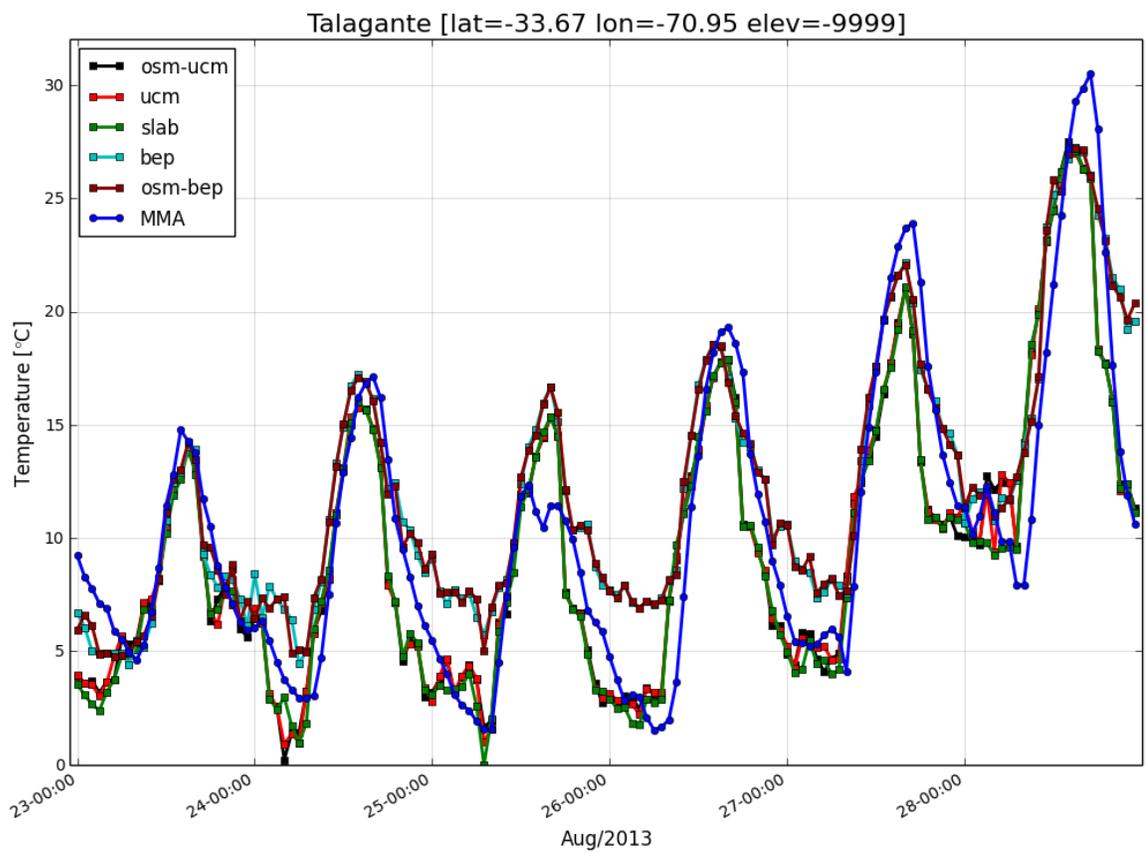


Figura 5.15: Serie de tiempo de variable Temperatura para estación Talagante en el periodo entre 23-08-2013 y 29-08-2013.

Las Tablas 5.2, 5.3, 5.4, 5.5 muestran las estadísticas representativas de los resultados obtenidos con respecto a los datos observacionales separadas por estación.

Stats/Models	osm-ucm	ucm	slab	bep	osm-bep
ObsMean	9.39	9.39	9.39	9.39	9.39
ModelMean	8.14	9.11	11.82	13.01	12.33
BIAS	-1.26	-0.28	2.43	3.61	2.94
RMSE	2.91	2.57	3.29	4.26	3.63

Tabla 5.2: Tabla de estadísticas para Puente Alto para Temperatura

Stats/Models	osm-ucm	ucm	slab	bep	osm-bep
ObsMean	9.99	9.99	9.99	9.99	9.99
ModelMean	7.29	7.41	7.82	10.49	10.48
BIAS	-2.69	-2.58	-2.16	0.50	0.49
RMSE	3.92	3.82	3.58	2.47	2.44

Tabla 5.3: Tabla de estadísticas para Quilicura para Temperatura

Stats/Models	osm-ucm	ucm	slab	bep	osm-bep
ObsMean	9.82	9.82	9.82	9.82	9.82
ModelMean	7.72	8.83	11.97	13.21	12.17
BIAS	-2.10	-0.99	2.15	3.39	2.35
RMSE	3.44	2.91	3.03	3.98	3.07

Tabla 5.4: Tabla de estadísticas para Santiago para Temperatura

Stats/Models	osm-ucm	ucm	slab	bep	osm-bep
ObsMean	9.25	9.25	9.25	9.25	9.25
ModelMean	8.29	8.29	8.13	10.74	10.77
BIAS	-0.97	-0.96	-1.12	1.49	1.51
RMSE	2.61	2.61	2.65	2.76	2.76

Tabla 5.5: Tabla de estadísticas para Talagante para Temperatura

5.3.2. Magnitud de Viento

Las Figuras 5.16, 5.17, 5.18, 5.19 muestran la diferencia existente entre las distintas corridas y la corrida control, utilizando los promedios de la magnitud de viento a 10m de elevación dentro del periodo completo de simulación a las 3:00 horas y a las 15:00 horas.

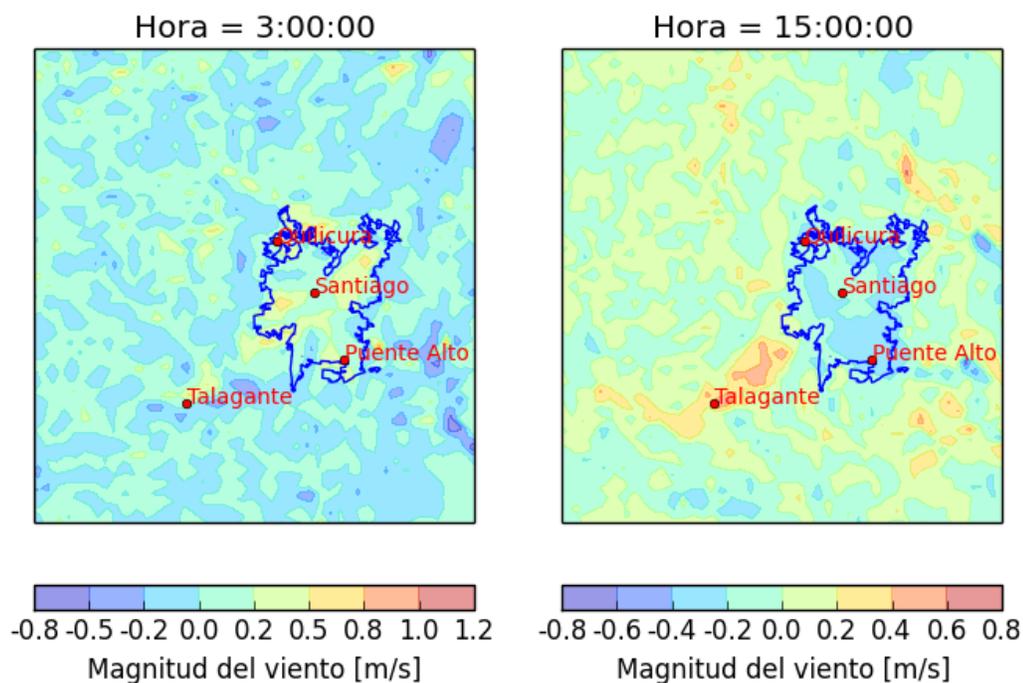


Figura 5.16: Diferencia entre valores promedio de magnitud de viento de UCM y SLAB, a las 03:00 y 15:00 hrs, a 10m de elevación.

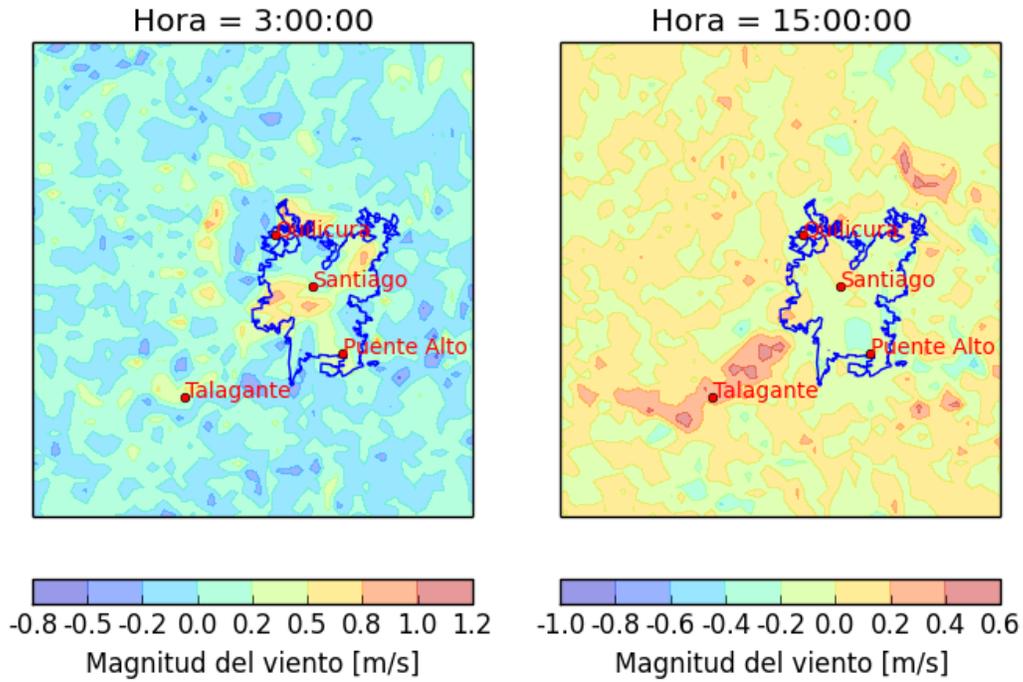


Figura 5.17: Diferencia entre valores promedio de magnitud de viento de OSMUCM y SLAB, a las 03:00 y 15:00 hrs, a 10m de elevación.

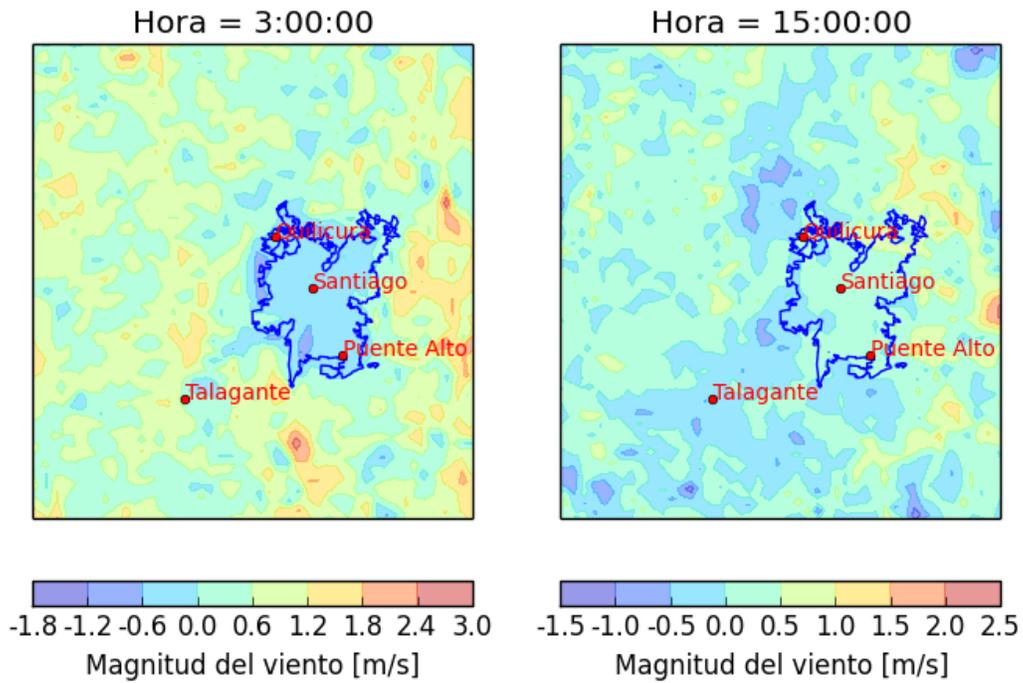


Figura 5.18: Diferencia entre valores promedio de magnitud de viento de BEP y SLAB, a las 03:00 y 15:00 hrs, a 10m de elevación.

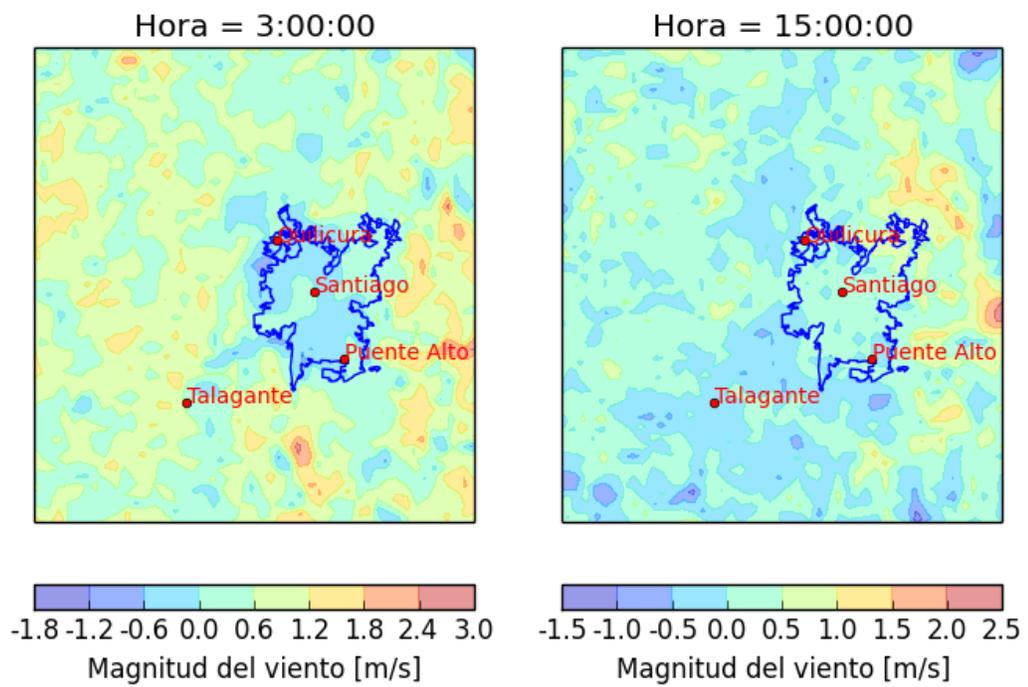


Figura 5.19: Diferencia entre valores promedio de magnitud de viento de OSMBEP y SLAB, a las 03:00 y 15:00 hrs, a 10m de elevación.

Las Figuras 5.20, 5.21, 5.22, 5.23 muestran un corte vertical de oeste a este en la zona media del dominio, utilizando la diferencia de promedios de magnitud de viento a las 3:00 horas y a las 15:00 horas, entre las distintas corridas y la corrida control.

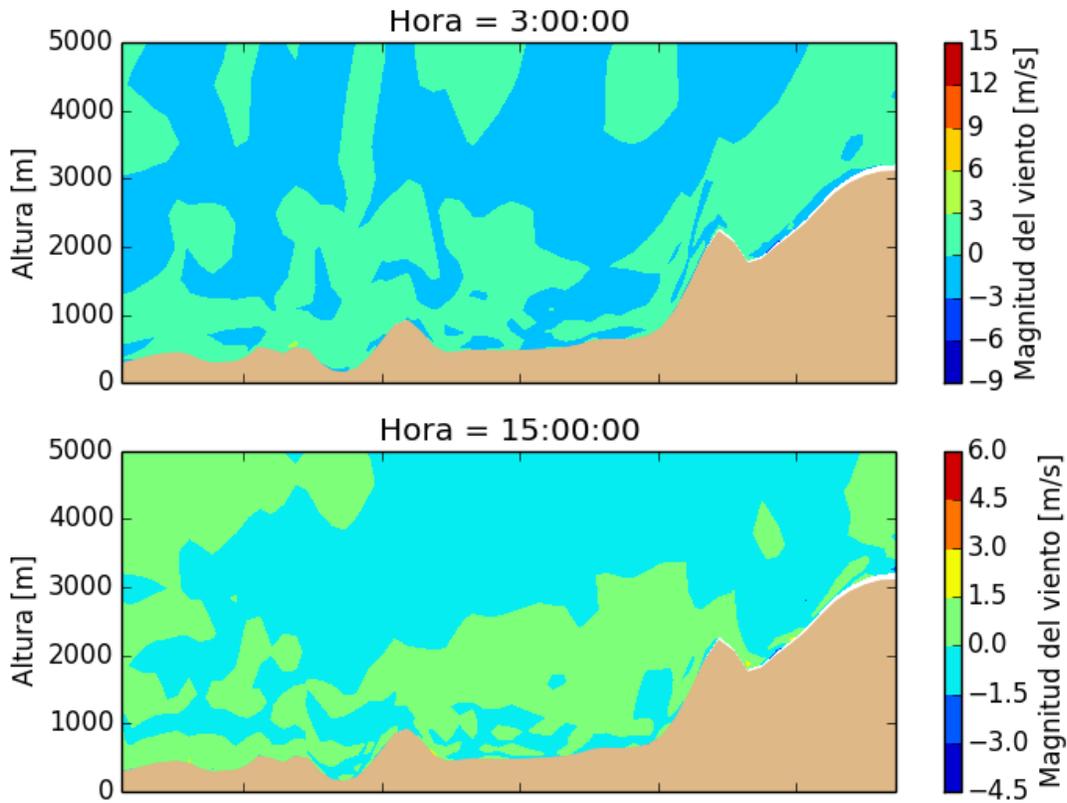


Figura 5.20: Corte vertical de la diferencia entre valores promedio de magnitud de viento de UCM y SLAB, a las 03:00 y 15:00 hrs.

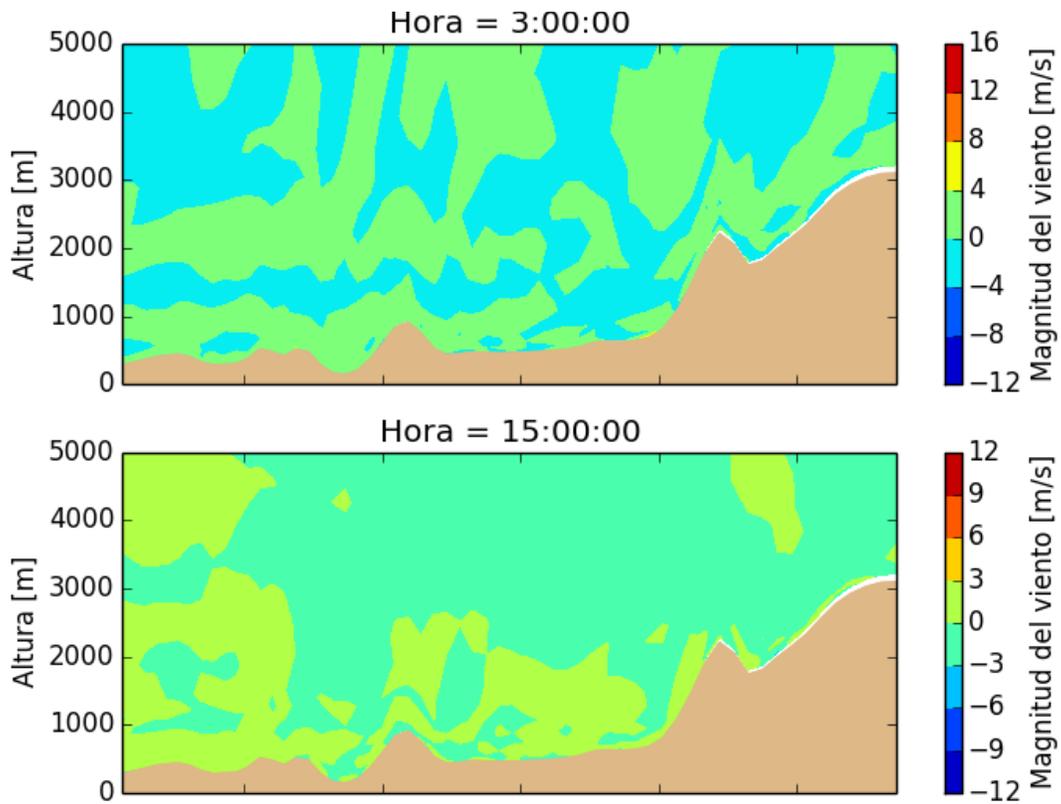


Figura 5.21: Corte vertical de la diferencia entre valores promedio de magnitud de viento de OSMUCM y SLAB, a las 03:00 y 15:00 hrs.

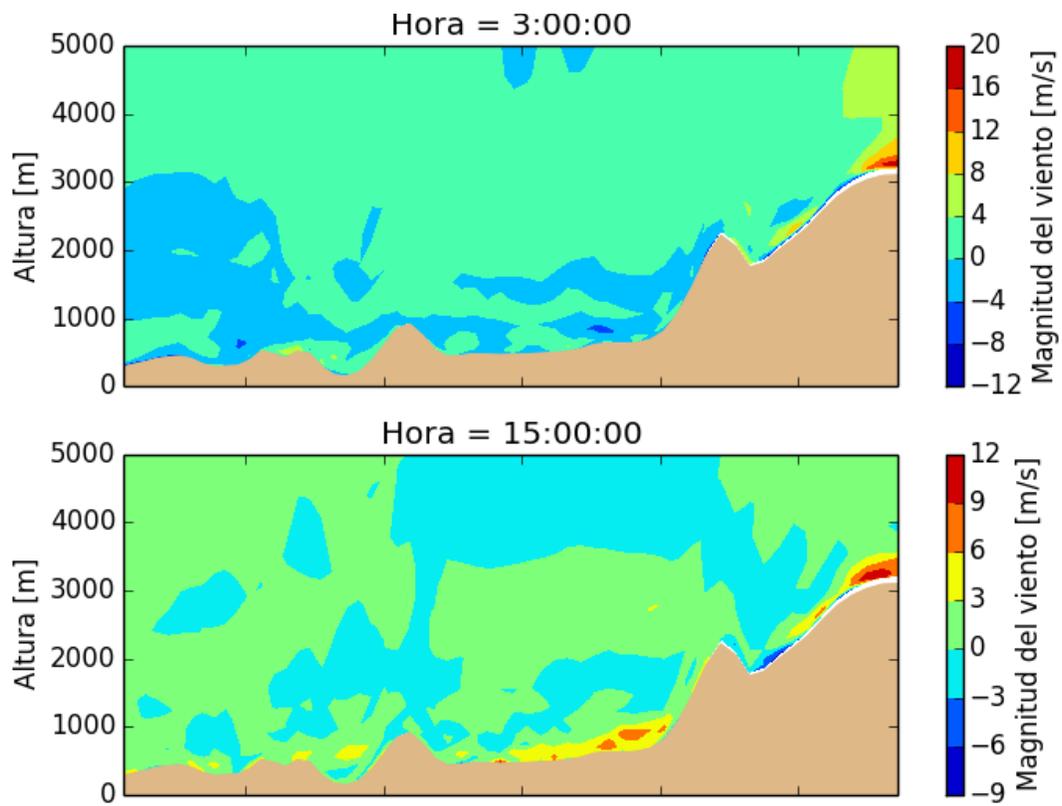


Figura 5.22: Corte vertical de la diferencia entre valores promedio de magnitud de viento de BEP y SLAB, a las 03:00 y 15:00 hrs.

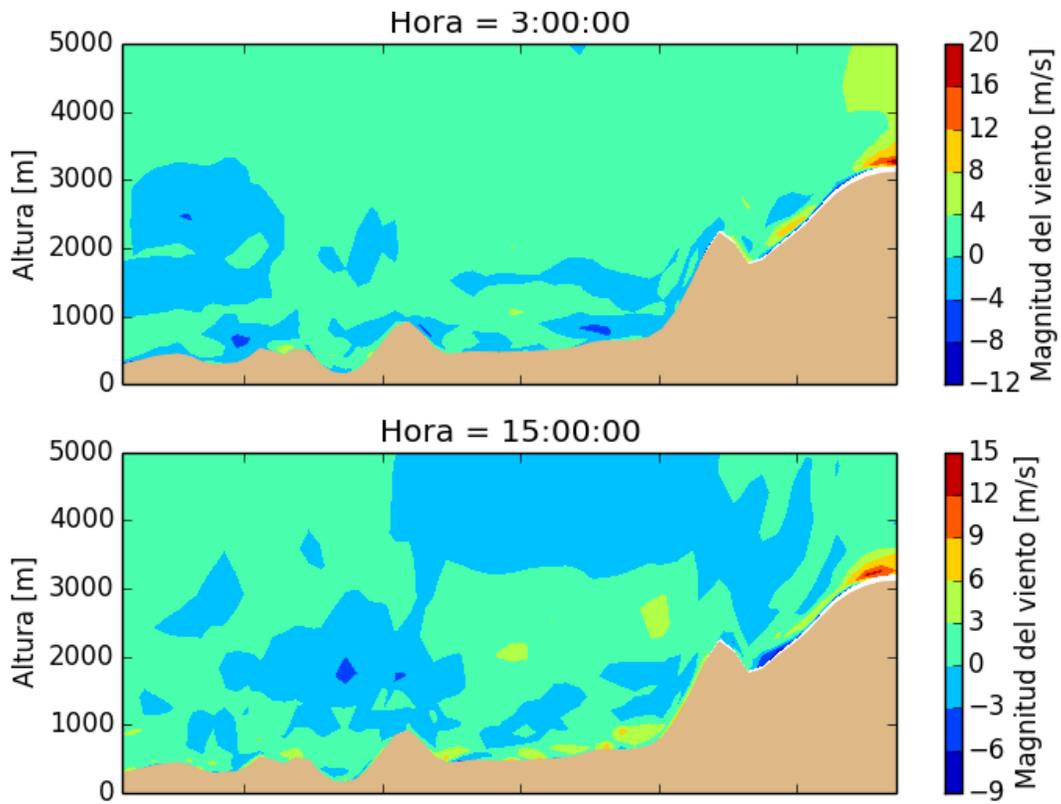


Figura 5.23: Corte vertical de la diferencia entre valores promedio de magnitud de viento de OSMBEP y SLAB, a las 03:00 y 15:00 hrs.

Las Figuras 5.24, 5.25, 5.26, 5.27 muestran las series de tiempo para las estaciones dispuestas en Puente Alto, Quilicura, Santiago y Talagante. En azul se puede ver el valor observado para las fechas indicadas, mientras que en los distintos colores se representan los valores obtenidos por las simulaciones.

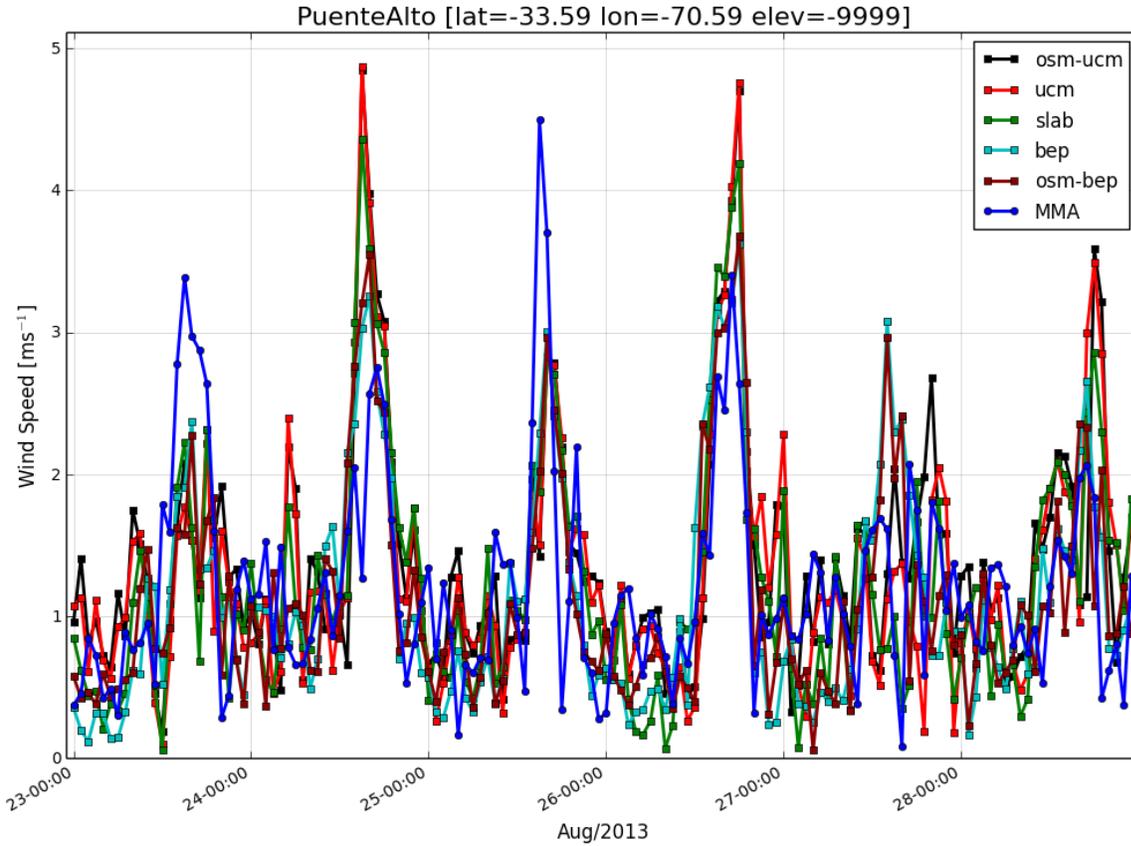


Figura 5.24: Serie de tiempo de variable Magnitud de Viento para estación Puente Alto en el periodo entre 23-08-2013 y 29-08-2013.

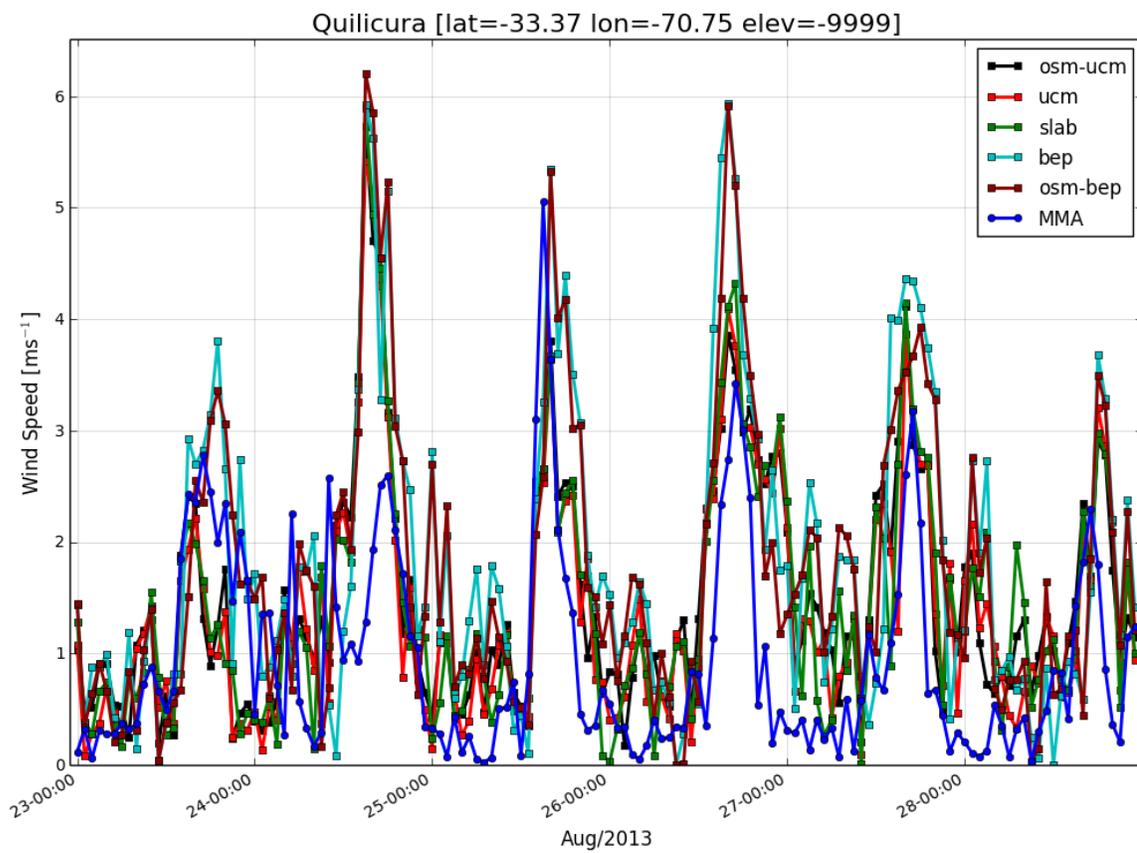


Figura 5.25: Serie de tiempo de variable Magnitud de Viento para estación Quilicura en el periodo entre 23-08-2013 y 29-08-2013.

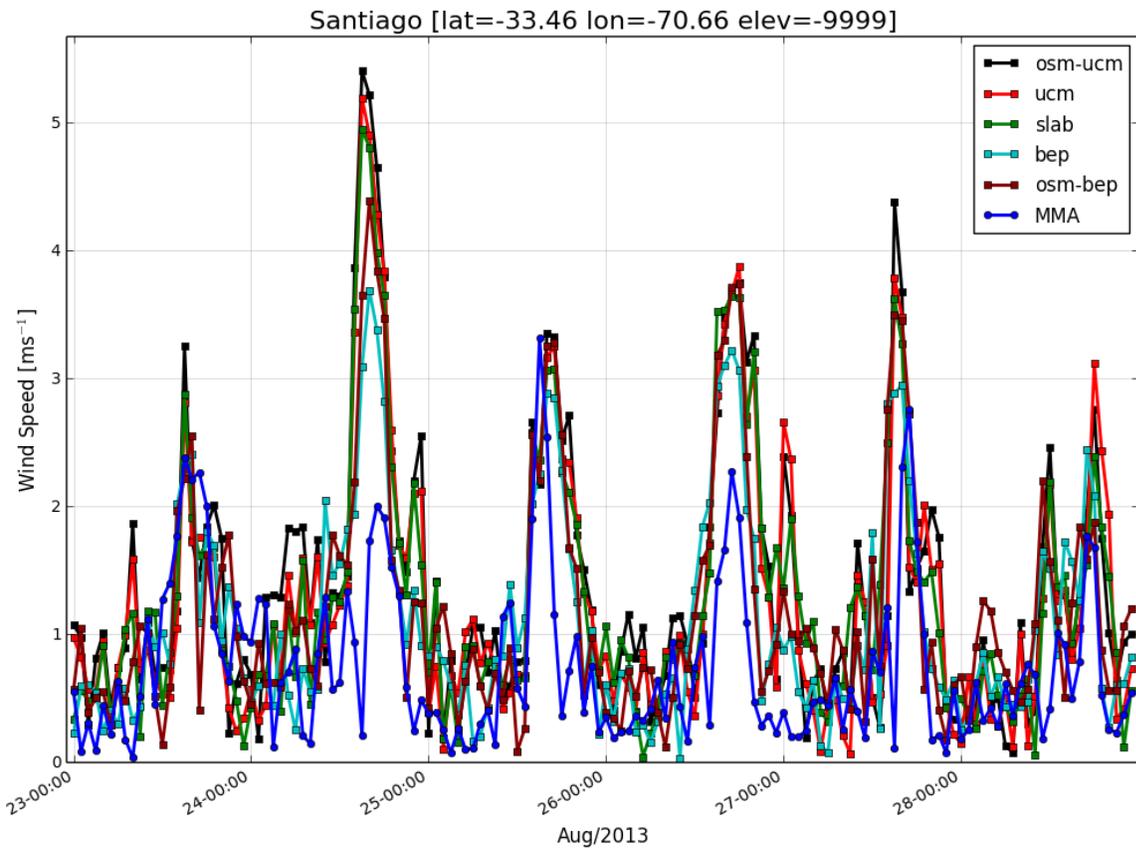


Figura 5.26: Serie de tiempo de variable Magnitud de Viento para estación Santiago en el periodo entre 23-08-2013 y 29-08-2013.

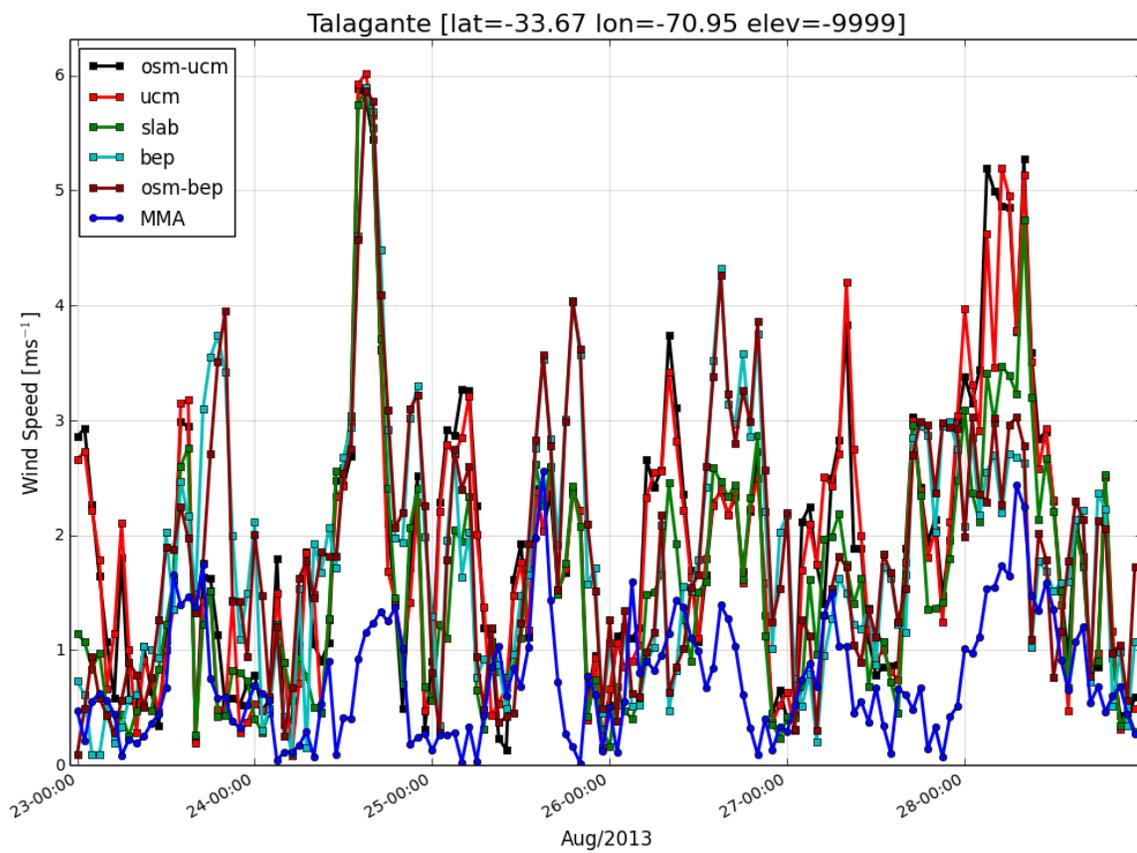


Figura 5.27: Serie de tiempo de variable Magnitud de Viento para estación Talagante en el periodo entre 23-08-2013 y 29-08-2013.

Las Tablas 5.6, 5.7, 5.8, 5.9 muestran las estadísticas representativas de los resultados obtenidos con respecto a los datos observacionales separadas por estación.

Stats/Models	osm-ucm	ucm	slab	bep	osm-bep
ObsMean	1.25	1.25	1.25	1.25	1.25
ModelMean	1.39	1.30	1.22	1.13	1.15
BIAS	0.14	0.05	-0.03	-0.12	-0.10
RMSE	0.84	0.85	0.76	0.71	0.72

Tabla 5.6: Tabla de estadísticas para Puente Alto para Magnitud de viento

Stats/Models	osm-ucm	ucm	slab	bep	osm-bep
ObsMean	1.01	1.01	1.01	1.01	1.01
ModelMean	1.52	1.44	1.47	1.94	1.89
BIAS	0.51	0.43	0.46	0.94	0.88
RMSE	1.10	1.08	1.11	1.47	1.40

Tabla 5.7: Tabla de estadísticas para Quilicura para Magnitud de viento

Stats/Models	osm-ucm	ucm	slab	bep	osm-bep
ObsMean	0.79	0.79	0.79	0.79	0.79
ModelMean	1.48	1.35	1.37	1.15	1.24
BIAS	0.69	0.56	0.58	0.36	0.45
RMSE	1.24	1.14	1.08	0.75	0.91

Tabla 5.8: Tabla de estadísticas para Santiago para Magnitud de viento

Stats/Models	osm-ucm	ucm	slab	bep	osm-bep
ObsMean	0.66	0.66	0.66	0.66	0.66
ModelMean	1.74	1.72	1.43	1.82	1.85
BIAS	1.08	1.06	0.77	1.16	1.19
RMSE	1.53	1.52	1.26	1.66	1.67

Tabla 5.9: Tabla de estadísticas para Talagante para Magnitud de viento

5.4. Análisis de resultados

A partir de los resultados de las Figuras 5.4, 5.5, 5.6, 5.7, 5.16, 5.17, 5.18, 5.19, se construyeron las Tablas 5.10 y 5.11, con los valores estimados de variación diurna y vespertina de temperatura y magnitud de viento, tanto al interior como al exterior de la zona urbana de Santiago de Chile.

Corrida	Variación Interior Ciudad		Variación Exterior Ciudad	
	Diurna [°C]	Vespertina [°C]	Diurna [°C]	Vespertina [°C]
ucm	[-4.5 , -1.5]	[-2.5 , -1.0]	[-1.5 , +1.5]	[-2.5 , -1.0]
osm-ucm	[-6.0 , -1.5]	[-2.0 , -1.0]	[-1.5 , +1.5]	[-1.0 , +0.5]
bep	[0.0 , +1.5]	[+1.0 , +3.0]	[0.0 , +6.0]	[-1.0 , +4.0]
osm-bep	[-1.5 , -1.5]	[0.0 , +2.0]	[0.0 , +6.0]	[0.0 , +4.0]

Tabla 5.10: Tabla variación de valores diurnos y vespertinos de **Temperatura** entre las distintas corridas y slab, tanto al interior como exterior de la zona urbana.

Corrida	Variación Interior Ciudad		Variación Exterior Ciudad	
	Diurna [m/s]	Vespertina [m/s]	Diurna [m/s]	Vespertina [m/s]
ucm	[-0.2 , +0.8]	[-0.4 , 0.0]	[-0.5 , +0.5]	[-0.4 , +0.6]
osm-ucm	[-0.2 , +1.0]	[-0.4 , +0.2]	[-0.8 , +0.8]	[-0.4 , +0.6]
bep	[-1.8 , 0.0]	[-0.5 , +0.5]	[-0.6 , +2.4]	[-1.5 , +2.5]
osm-bep	[-0.6 , +0.6]	[-0.5 , +1.0]	[0.0 , +2.4]	[-1.5 , +2.0]

Tabla 5.11: Tabla variación de valores diurnos y vespertinos de **Magnitud de Viento** entre las distintas corridas y slab, tanto al interior como exterior de la zona urbana.

Utilizando estas tablas se pueden hacer ciertas observaciones a las corridas realizadas. Con respecto a la **temperatura**, hay que notar que en la Tabla 5.10, los valores en **UCM** y **OSM-UCM** dentro de la zona de la ciudad tienden a disminuir, tanto para los diurnos como los vespertinos, lo que puede ser contradictorio considerando que la incorporación de una zona urbana produce un efecto de calentamiento y no de enfriamiento (UHI expuesto en la sección 1). Sin embargo, este extraño comportamiento se vuelve a repetir en los experimentos expuestos en [Liao et al., 2014], por lo que puede ser un efecto producido por el modelo por sobre las zonas urbana, subestimando la temperatura con respecto a un modelo sin efectos urbanos. Para las corridas **BEP**, se puede ver que existe una sobreestimación de la temperatura, por el contrario a **UCM**, pero que al momento de agregar datos de zonas urbanas en **OSM-BEP** comienza a mostrarse una reducción de alrededor de -0.5°C en la zona centro. Con respecto a la **magnitud de viento**, se puede ver en la Tabla 5.11 cómo la magnitud del viento se ve reducida en el modelo **UCM** para la zona interior de la ciudad, mientras que en **OSM-UCM** el efecto de resistencia al viento es menor, pudiéndose comparar valores que logran ser mayores a 0.0m/s en términos de magnitud de viento. Por otro lado, los valores fluctúan para los modelos **BEP** y **OSM-BEP**, en donde, si bien se obtienen reducciones en la magnitud de viento para los valores diurnos, también se aprecian aumentos en los valores vespertinos.

En los **cortes verticales**, utilizando los valores de **temperatura** de las Figuras 5.8, 5.9, 5.10, 5.11 se puede apreciar que las fluctuaciones se acentúan mayormente en las capas

cercanas a la altura del terreno, y que este comportamiento es mucho más notorio en la zona del valle que contiene a la ciudad. Se puede observar también que el efecto de cambio de temperatura en las corridas con modelos urbanos alcanzan un límite de altura, y que pasado este límite las temperaturas no tienen mayor variación con respecto al modelo sin efectos urbanos. Se puede ver como la inclusión del dosel urbano provoca en **OSM-UCM** un aumento del efecto capas mayores, provocando una reducción mayor en ellas, mientras que en el modelo **OSM-BEP** disminuye los efectos y los reduce a las capas inferiores más cercanas a la altura del terreno. Por el contrario, los cambios en la **magnitud de viento** en las Figuras 5.20, 5.21, 5.22, 5.23 no son observables a simple vista como los vistos en los cortes verticales de temperatura. No se puede apreciar ciertamente los efectos que se producen en la zona interior de la ciudad, y tampoco se ve un límite en la capa vertical pronunciado para poder estimar hasta qué punto existe un efecto en los flujos. Esto puede deberse a la diferencia de alturas entre el primer nivel vertical y la altura representada en el dosel generado. Es decir, dado que el primero tiene alturas que fluctúan entre los 15m y 20m, y que el dosel posee su mayor parte zonas residenciales bajas que mantienen alturas entre 5m y 7m, es posible que el efecto de rugosidad no logre ser visto en la capa de 10m de magnitud de viento. Lamentablemente no existe una observación de nivel inferior, por lo que una solución viable para lograr observar este límite vertical podría ser aumentar la altura del dosel a niveles mayores de 10m y de esta forma observar si existe o no un efecto real.

Dentro de las **series de tiempo de temperatura**, se puede observar que los valores de las corridas **UCM** y **OSM-UCM** mantienen el mismo comportamiento visto en los cortes a 2m de altura, en donde se produce un enfriamiento en la variable, pero comparando con respecto a los **valores observados**. Esto se puede corroborar en las Tablas 5.2, 5.3, 5.4, 5.5, en donde el *BIAS* indica una reducción del valor promedio esperado para el periodo en todas las estaciones observadas. El caso de **BEP** y **OSM-BEP**, es muy distinto, dado que los valores promedio de temperatura presentan un comportamiento de sobreestimación de la variable temperatura, lo cual se puede corroborar con el *BIAS* de las Tablas utilizadas anteriormente para todas las estaciones observadas. Con respecto a las series de tiempo de **magnitud de viento**, se puede ver que el comportamiento de las corridas es variable. Para el caso de Puente Alto, se observa en la Tabla 5.6 que las corridas presentan variaciones de entre -0.12m/s y +0.14 con respecto a los valores observado, lo cual indica un buen ajuste en los resultados. En Quilicura se observa en la Tabla 5.7 que las simulaciones **UCM** y **OSM-UCM** presentan una mayor resistencia al viento que las simulaciones **BEP** y **OSM-BEP**, lo cual se traduce a un menor aumento en los valores promedio de magnitud de viento. Sin embargo en Santiago la Tabla 5.8 indica que **BEP** y **OSM-BEP** presentan mayor resistencia al viento que **UCM** y **OSM-UCM**. Por último en Talagante las simulaciones mantienen un comportamiento similar, en donde el **BIAS** indica variaciones entre +1.8m/s y +1.19m/s según la Tabla 5.9.

Finalmente, hay que recordar que el objetivo de este experimento es observar el efecto de la inclusión del dosel urbano generado por **CityCanopy** dentro de los modelos urbanos del simulador WRF, por lo que no se busca definir cuál modelo es "mejor" que el otro, si no más bien ver los efectos que produce la inclusión del dosel urbano en la simulación, y ver si realmente existe una mejora o no. Concluyendo, el uso del dosel urbano produce cambios en los resultados, y dependiendo de las variables, se observan mejoras en las adaptaciones con respecto a los datos observados. De esta forma, se esperaría que la adición de mayor

información al dosel urbano puedan mejorar la aproximación de los resultados.

Capítulo 6

Conclusión

En el presente trabajo se logró desarrollar el sistema de generación de dosel urbano **CityCanopy**, el cual es un sistema de libre acceso que obtiene datos de zonas urbanas provenientes de **OpenStreetMap**, y luego produce los archivos necesarios de manera automática para integrarlos al simulador **WRF**. Se implementó una versión inicial del programa, la cual genera información de uso de suelo urbano y la dispone en forma de grilla para su visualización. En el contexto computacional, se logró el objetivo de ser un software extensible, por medio del diseño de un modelo de datos apropiado y la adición de patrones de diseño que permiten agregar nuevas funcionalidades a futuro.

Se puso a prueba el programa utilizando las zonas urbanas de Santiago de Chile y Buenos Aires. En primera instancia se observaron los distintos resultados obtenidos al aplicar dos variaciones del algoritmo para generar los datos, lo que provoca resultados distintos dependiendo ciertamente de la cantidad de datos disponibles. Luego se observó el comportamiento que tiene el sistema al ir variando el ancho de grilla, en donde se hizo un enfoque al tiempo computacional requerido para obtener la información. Para todo programa computacional, es importante que sea capaz de generar los datos en un tiempo razonable para el usuario. La complejidad de orden constante para grillas con una cantidad de puntos menor a 10^6 , y orden n^2 para cantidades mayores, lo hace tener un tiempo de cómputo bastante razonable para ser utilizado. Para el caso de Santiago de Chile, se realizó un pequeño análisis de los resultados obtenidos comparándolo con respecto a un dosel de referencia construido. Se definieron ciertas estadísticas para poder ver de cierto modo que tan semejante es el resultado obtenido por el sistema implementado con respecto a información más aproximada. Un 53% de acierto de cobertura de los datos muestran que aún no existe completitud de los datos para la zona, y esto es una de las falencias que posee **CityCanopy**. Sin embargo el 78% de acierto a las categorías y cobertura coincidente, y un 17% de innovación en los datos, son cifras que permiten exhibir lo positivo del sistema dado del procesamiento y generación automática de estos datos.

Finalmente se experimentó utilizando el simulador **WRF** y dos de sus modelos urbanos, con el objetivo de ver las variaciones que provoca el acoplamiento del dosel urbano generado. La zona de estudio contenía la región de Santiago de Chile, y se realizaron observaciones a las variables de temperatura y magnitud de viento. Acá se pudo ver el comportamiento que

tiene el hacer o no uso del dosel urbano dentro del modelo, en donde se concluyó que el ajuste que provoca mejoras en las aproximaciones dependiendo del caso o variable observada.

Se espera que a futuro *OpenStreetMap* adicione más datos a los que actualmente se tienen para de este modo tener un dosel urbano más robusto. De igual forma, se deja como trabajo propuesto la adición de nuevas variables y fuentes al sistema **CityCanopy**.

Bibliografía

- [pho,] Phong light reflection. https://upload.wikimedia.org/wikipedia/commons/thumb/8/8f/Phong_shading_model.svg/2000px-Phong_shading_model.svg.png. Accessed: 2016-11-06.
- [tri,] Triangle rasterization. https://upload.wikimedia.org/wikipedia/commons/thumb/b/bc/Rasterisation-triangle_example.svg/450px-Rasterisation-triangle_example.svg.png. Accessed: 2016-11-06.
- [wps, 2010] (2010). Wps advanced. http://www2.mmm.ucar.edu/wrf/users/tutorial/200909/15_WPS-advanced.pdf. Accessed: 2016-11-06.
- [wrf, 2010a] (2010a). Wrfv3 user guide v3 - chapter 1. http://www2.mmm.ucar.edu/wrf/users/docs/user_guide_V3/users_guide_chap1.htm.
- [wrf, 2010b] (2010b). Wrfv3 user guide v3 - chapter 1. http://www2.mmm.ucar.edu/wrf/users/docs/user_guide_V3/users_guide_chap5.htm.
- [wrf, 2010c] (2010c). Wrfv3 user guide v3 - chapter 3. http://www2.mmm.ucar.edu/wrf/users/docs/user_guide/users_guide_chap3.html.
- [red, 2015] (2015). Red de monitoreo de calidad del aire para la región metropolitana de santiago. <http://macam.mma.gob.cl/>. Accessed: 2015-08-31.
- [goo, 2016a] (2016a). Google streetview. <https://www.google.com/maps>. Accessed: 2016-11-06.
- [goo, 2016b] (2016b). Google streetview car device. <https://www.google.com/streetview/understand/>. Accessed: 2016-11-06.
- [ope, 2016a] (2016a). Openstreetmap. <https://www.openstreetmap.org>. Accessed: 2016-11-06.
- [ope, 2016b] (2016b). Openstreetmap wiki. <http://wiki.openstreetmap.org/wiki>. Accessed: 2016-11-06.
- [cgi, 2016a] (2016a). Portal cgiar-csi. <http://srtm.csi.cgiar.org/>. Accessed: 2016-11-06.
- [cgi, 2016b] (2016b). Portal cgiar-csi datos 90m. <http://srtm.csi.cgiar.org/SELECTION/>

inputCoord.asp. Accessed: 2016-11-06.

- [Baklanov et al., 2008] Baklanov, A., Korsholm, U., Mahura, A., Petersen, C., and Gross, A. (2008). Enviro-hirham: on-line coupled modelling of urban meteorology and air pollution. *Advances in Science and Research*, 2:41–46.
- [Burian et al., 2004] Burian, S. J., Stetson, S. W., Han, W., Ching, J., and Byun, D. (2004). High-resolution dataset of urban canopy parameters for houston, texas. In *Preprint proceedings, Fifth Symposium on the Urban Environment, AMS*, pages 23–26.
- [Chen et al., 2011] Chen, F., Kusaka, H., Bornstein, R., Ching, J., Grimmond, C., Grossman-Clarke, S., Loridan, T., Manning, K. W., Martilli, A., Miao, S., et al. (2011). The integrated wrf/urban modelling system: development, evaluation, and applications to urban environmental problems. *International Journal of Climatology*, 31(2):273–288.
- [Chen et al., 2004] Chen, F., Kusaka, H., Tewari, M., Bao, J., and Hirakuchi, H. (2004). Utilizing the coupled wrf/lsm/urban modeling system with detailed urban classification to simulate the urban heat island phenomena over the greater houston area. In *Fifth Symposium on the Urban Environment*, pages 9–11.
- [de Estadísticas INE, 2012] de Estadísticas INE, I. N. (2012). País y regiones por área urbana-rural: Actualización población 2002-2012 y proyecciones 2013-2020.
- [Gao et al., 2012] Gao, Z., Nocera, L., and Neumann, U. (2012). Visually-complete aerial lidar point cloud rendering. In *Proceedings of the 20th International Conference on Advances in Geographic Information Systems*, pages 289–298. ACM.
- [Glotfelty et al., 2013] Glotfelty, T., Tewari, M., Sampson, K., Duda, M., Chen, F., and Ching, J. (2013). Nudapt 44 documentation.
- [González-Aparicio et al., 2010] González-Aparicio, I., Nuterman, R., Korsholm, U., Mahura, A., Ángel, J., Acero, J. H., and Baklanov, A. (2010). Land-use database processing approach for meso-scale urban nwp model initialization. *Scientific Report 10-02*.
- [Gutiérrez et al., 2013] Gutiérrez, E., González, J. E., Bornstein, R., Arend, M., and Martilli, A. (2013). A new modeling approach to forecast building energy demands during extreme heat events in complex cities. *Journal of Solar Energy Engineering*, 135(4):040906.
- [Haklay and Weber, 2008] Haklay, M. and Weber, P. (2008). Openstreetmap: User-generated street maps. *IEEE Pervasive Computing*, 7(4):12–18.
- [Hidalgo et al., 2008] Hidalgo, J., Masson, V., Baklanov, A., Pigeon, G., and Gimeno, L. (2008). Advances in urban climate modeling. *Annals of the New York Academy of Sciences*, 1146(1):354–374.
- [Kalnay and Cai, 2003] Kalnay, E. and Cai, M. (2003). Impact of urbanization and land-use change on climate. *Nature*, 423(6939):528–531.
- [Kim et al., 2008] Kim, Y., Byon, J., and Choi, Y. (2008). The impact of high resolution

- surface properties retrieved from satellite in the urban canopy model. In *IOP Conference Series: Earth and Environmental Science*, volume 1, page 012062. IOP Publishing.
- [Lafarge et al., 2010] Lafarge, F., Descombes, X., Zerubia, J., and Pierrot-Deseilligny, M. (2010). Structural approach for building reconstruction from a single dsm. *IEEE Transactions on pattern analysis and machine intelligence*, 32(1):135–147.
- [Latif et al., 2011] Latif, S., Islam, K. R., Khan, M. M. I., and Ahmed, S. I. (2011). Opens-treetmap for the disaster management in bangladesh. In *Open Systems (ICOS), 2011 IEEE Conference on*, pages 429–433. IEEE.
- [Lee et al., 2011] Lee, S.-H., Kim, S.-W., Angevine, W., Bianco, L., McKeen, S., Senff, C., Trainer, M., Tucker, S., and Zamora, R. (2011). Evaluation of urban surface parameterizations in the wrf model using measurements during the texas air quality study 2006 field campaign. *Atmospheric Chemistry and Physics*, 11(5):2127–2143.
- [Liao et al., 2014] Liao, J., Wang, T., Wang, X., Xie, M., Jiang, Z., Huang, X., and Zhu, J. (2014). Impacts of different urban canopy schemes in wrf/chem on regional climate and air quality in yangtze river delta, china. *Atmospheric Research*, 145:226–243.
- [Luebke, 2003] Luebke, D. P. (2003). *Level of detail for 3D graphics*. Morgan Kaufmann.
- [Morabito et al., 2016] Morabito, M., Crisci, A., Messeri, A., Orlandini, S., Raschi, A., Maracchi, G., and Munafò, M. (2016). The impact of built-up surfaces on land surface temperatures in italian urban areas. *Science of The Total Environment*, 551:317–326.
- [Nations, 2014] Nations, U. (2014). World urbanization prospects: The 2014 revision, highlights. department of economic and social affairs. *Population Division, United Nations*.
- [Orfanoz, 2015] Orfanoz, A. P. (2015). High resolution simulations of pollution vertical stratification over santiago and its transport to the chilean andes.
- [Ortner et al., 2007] Ortner, M., Descombes, X., and Zerubia, J. (2007). Building outline extraction from digital elevation models using marked point processes. *International Journal of Computer Vision*, 72(2):107–132.
- [Sievinen et al., 2010] Sievinen, P., Hellsten, A., Praks, J., Koskinen, J., and Kukkonen, J. (2010). Megapoli scientific report 10-02 urban morphological database for paris, france.
- [Skamarock et al., 2008] Skamarock, W., Klemp, J., Dudhia, J., Gill, D., Barker, D., Duda, M., Huang, X., Wang, W., and Powers, J. (2008). A description of the advanced research wrf version 3, near tech note near/tn 475 str, 125 pp. Available from: UCAR Communications, PO Box, 3000.
- [Takahashi et al., 2011] Takahashi, K., Azami, A., Tochiyama, Y., Kubo, Y., Itakura, K., Goto, K., Kataumi, K., Takahara, H., Isobe, Y., Okura, S., et al. (2011). World-highest resolution global atmospheric model and its performance on the earth simulator. *State of the Practice Reports*, page 21.

- [Tewari et al., 2006] Tewari, M., Chen, F., and Kusaka, H. (2006). Implementation and evaluation of a single-layer urban canopy model in wrf/noah. In *WRF users workshop 2006*.
- [Tewari et al., 2005] Tewari, M., Chen, F., Warner, T. T., Coirier, W. J., and Kim, S. (2005). Numerical modeling study of wind flow over the salt lake city region using integrated wrf-noah-ucm model at meso-gamma scale. In *UCAR, www.mmm.ucar.edu/wrf/users/workshops/WS2005/presentations/session4/3-Tewari.pdf*.
- [Tursilowati et al., 2012] Tursilowati, L., Sumantyo, J. T. S., Kuze, H., and Adiningsih, E. S. (2012). The integrated wrf/urban modeling system and its application to monitoring urban heat island in jakarta, indonesia. *Journal of Urban & Environmental Engineering*, 6(1).
- [von Glasow et al., 2013] von Glasow, R., Jickells, T. D., Baklanov, A., Carmichael, G. R., Church, T. M., Gallardo, L., Hughes, C., Kanakidou, M., Liss, P. S., Mee, L., et al. (2013). Megacities and large urban agglomerations in the coastal zone: interactions between atmosphere, land, and marine ecosystems. *Ambio*, 42(1):13–28.
- [Yu et al., 2011] Yu, Z., Xu, C., Liu, J., Au, O. C., and Tang, X. (2011). Automatic object segmentation from large scale 3d urban point clouds through manifold embedded mode seeking. In *Proceedings of the 19th ACM international conference on Multimedia*, pages 1297–1300. ACM.

Apéndice A

Archivos de configuración de WRF

A.1. Archivo namelist.wps

```
1 &share
2 wrf_core = 'ARW',
3 max_dom = 3,
4 start_date = '2013-08-20_00:00:00', '2013-08-20_00:00:00', '2013-08-20_00:00:00'
5 end_date   = '2013-08-30_00:00:00', '2013-08-30_00:00:00', '2013-08-30_00:00:00'
6 interval_seconds = 21600,
7 io_form_geogrid = 2,
8 /
9 &geogrid
10 parent_id      = 0, 1, 2,
11 parent_grid_ratio = 1, 3, 3,
12 i_parent_start  = 1, 80, 27,
13 j_parent_start  = 1, 57, 22,
14 e_we           = 130, 61, 61,
15 e_sn           = 128, 61, 73,
16 geog_data_res  = 'modis_30s+10m', 'modis_30s+2m', 'modis_30s+30s',
17 dx = 18000,
18 dy = 18000,
19 map_proj = 'lambert',
20 ref_lat  = -34.066,
21 ref_lon  = -76.042,
22 truelat1 = -30.0,
23 truelat2 = -60.0,
24 stand_lon = -75.0,
25 geog_data_path = './geog_data',
26 /
27 &ungrib
28 out_format = 'WPS',
29 prefix = 'FILE',
30 /
31 &metgrid
32 fg_name = 'FILE'
33 io_form_metgrid = 2,
```

A.2. Archivo namelist.input

```

1 &time_control
2 run_days = 10,
3 run_hours = 0,
4 run_minutes = 0,
5 run_seconds = 0,
6 start_year = 2013, 2013, 2013,
7 start_month = 08, 08, 08,
8 start_day = 20, 20, 20,
9 start_hour = 00, 00, 00,
10 start_minute = 00, 00, 00,
11 start_second = 00, 00, 00,
12 end_year = 2013, 2013, 2013,
13 end_month = 08, 08, 08,
14 end_day = 30, 30, 30,
15 end_hour = 00, 00, 00,
16 end_minute = 00, 00, 00,
17 end_second = 00, 00, 00,
18 interval_seconds = 21600,
19 input_from_file = .true., .true., .true.,
20 fine_input_stream = 0,0,0,
21 history_interval = 1440, 60, 60,
22 frames_per_outfile = 1, 24, 24,
23 restart = .false.,
24 restart_interval = 1440,
25 io_form_history = 2,
26 io_form_restart = 2,
27 io_form_input = 2,
28 io_form_boundary = 2,
29 io_form_auxinput4 = 2,
30 auxinput11_interval_s = 72, 72, 72, 72,
31 auxinput11_end_h = 99999, 99999, 99999, 99999,
32 auxinput1_inname = "met_em.d<domain>.<date>",
33 debug_level = 0,
34 /
35
36 &domains
37 time_step = 24,
38 time_step_fract_num = 0,
39 time_step_fract_den = 1,
40 max_dom = 3,
41 s_we = 1, 1, 1,
42 e_we = 130, 61, 61,
43 s_sn = 1, 1, 1,
44 e_sn = 128, 61, 73,
45 s_vert = 1, 1, 1,
46 e_vert = 30, 30, 30,
47 num_metgrid_levels = 27,
48 num_metgrid_soil_levels = 4,
49 dx = 18000, 6000, 2000,
50 dy = 18000, 6000, 2000,
51 grid_id = 1, 2, 3,

```

```

52 parent_id = 1, 1, 2,
53 i_parent_start = 1, 80, 27,
54 j_parent_start = 1, 57, 22,
55 parent_grid_ratio = 1, 3, 3,
56 parent_time_step_ratio = 1, 3, 3,
57 feedback = 1,
58 smooth_option = 1,
59 use_adaptive_time_step = .false.,
60 /
61
62 &physics
63 mp_physics = 3, 3, 3,
64 progn = 0,
65 ra_lw_physics = 1, 1, 1,
66 ra_sw_physics = 1, 1, 1,
67 radt = 15, 15, 15,
68 sf_sfclay_physics = 5, 5, 5,
69 sf_surface_physics = 2, 2, 2,
70 bl_pbl_physics = 5, 5, 5,
71 bldt = 0, 0, 0,
72 cu_physics = 5, 5, 5,
73 cudt = 5, 5, 5,
74 isfflx = 1,
75 ifsnow = 0,
76 icloud = 1,
77 surface_input_source = 1,
78 num_soil_layers = 4,
79 mp_zero_out = 2,
80 mp_zero_out_thresh = 1.e-8,
81 sf_urban_physics = 0, 0, 0,
82 num_land_cat = 20,
83 maxiens = 1,
84 maxens = 3,
85 maxens2 = 3,
86 maxens3 = 16,
87 ensdim = 144,
88 cu_rad_feedback = .false.,
89 slope_rad = 0, 0, 0,
90 /
91
92 &fdda
93 obs_nudge_opt = 0, 0, 0, 0,
94 max_obs = 150000,
95 fdda_start = 0., 0., 0.,
96 fdda_end = 99999., 99999., 99999.,
97 obs_nudge_wind = 1,1,1,
98 obs_coef_wind = 6.E-3,6.E-3,6.E-3,
99 obs_nudge_temp = 1,1,1,
100 obs_coef_temp = 6.E-4,6.E-4,6.E-4,
101 obs_nudge_mois = 0,0,0,
102 obs_coef_mois = 6.E-4,6.E-4,6.E-4,
103 obs_rinxy = 20.,20.,10.,
104 obs_rinsig = 0.1,
105 obs_twindo = 0.6666667,0.6666667,0.6666667,
106 obs_npfi = 5,
107 obs_ionf = 1, 1, 1,

```

```

108  obs_idynin           = 0,
109  obs_dtramp           = 99999.,
110  obs_prt_freq        = 1, 1, 1,
111  obs_prt_max         = 10000
112  obs_ipf_errob       = .true.,
113  obs_ipf_nudob       = .true.,
114  obs_ipf_in4dob      = .true.,
115  obs_ipf_init        = .true.,
116  /
117
118  &dynamics
119  w_damping            = 0,
120  diff_opt             = 0,
121  mix_full_fields     = .false.,
122  km_opt              = 4,
123  diff_6th_opt        = 0,      0,      0,
124  diff_6th_factor     = 0.12,   0.12,   0.12,
125  base_temp           = 290.,
126  damp_opt            = 0,
127  zdamp               = 5000.,  5000.,  5000.,
128  dampcoef            = 0.2,     0.2,     0.2,
129  khdif               = 0,      0,      0,
130  kvdif               = 0,      0,      0,
131  non_hydrostatic     = .true.,  .true.,  .true.,
132  tke_adv_opt         = 1,      1,      1,
133  chem_adv_opt        = 1,      1,      1,
134  moist_adv_opt       = 1,      1,      1,
135  scalar_adv_opt      = 1,      1,      1,
136  /
137
138  &bdy_control
139  spec_bdy_width      = 5,
140  spec_zone           = 1,
141  relax_zone         = 4,
142  specified           = .true.,  .false., .false.,
143  nested              = .false., .true.,  .true.,
144  /
145
146  &fire
147  ifire               = 0, 0, 0, 0,
148  /
149
150  &namelist_quilt
151  nio_tasks_per_group = 0,
152  nio_groups = 1,
153  /

```

Apéndice B

Series de tiempo

B.1. Temperatura

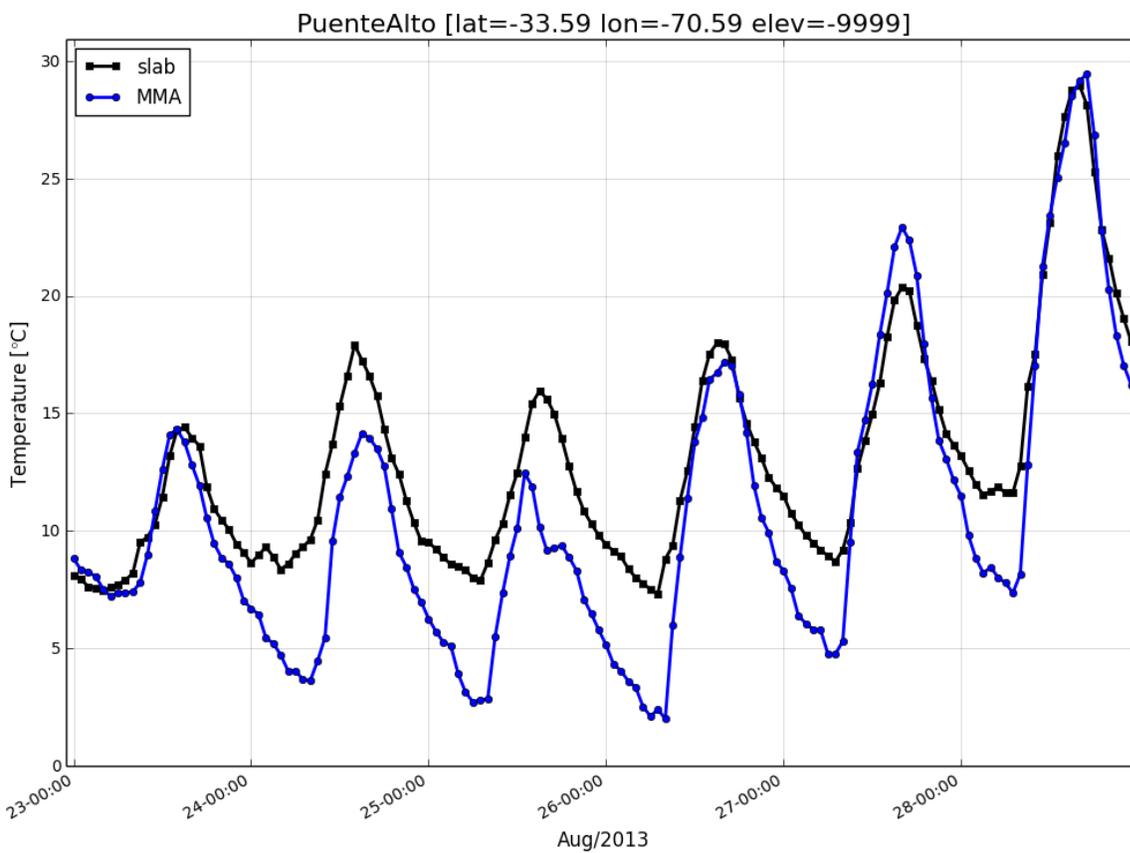
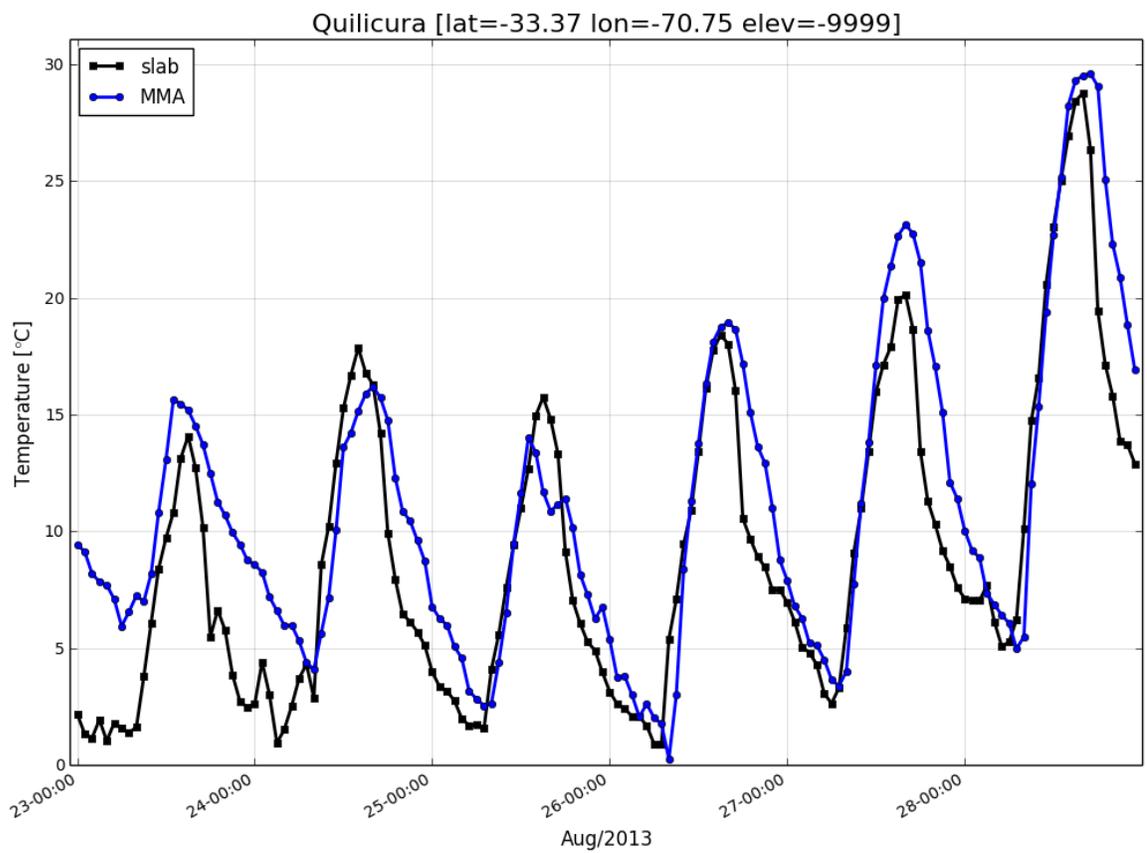


Figura B.1: Serie de tiempo para SLAB de variable Temperatura para estación Puente Alto en el periodo entre 23-08-2013 y 29-08-2013.



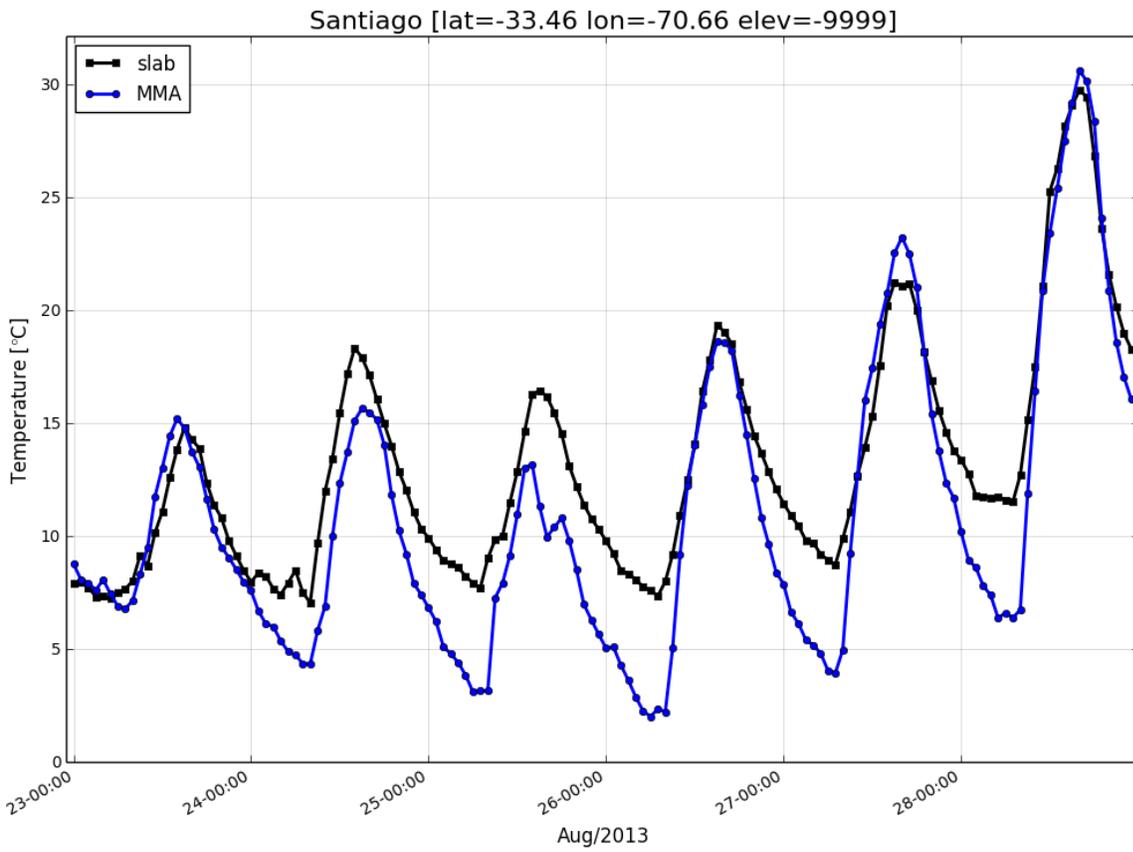


Figura B.3: Serie de tiempo para SLAB de variable Temperatura para estación Santiago en el periodo entre 23-08-2013 y 29-08-2013.

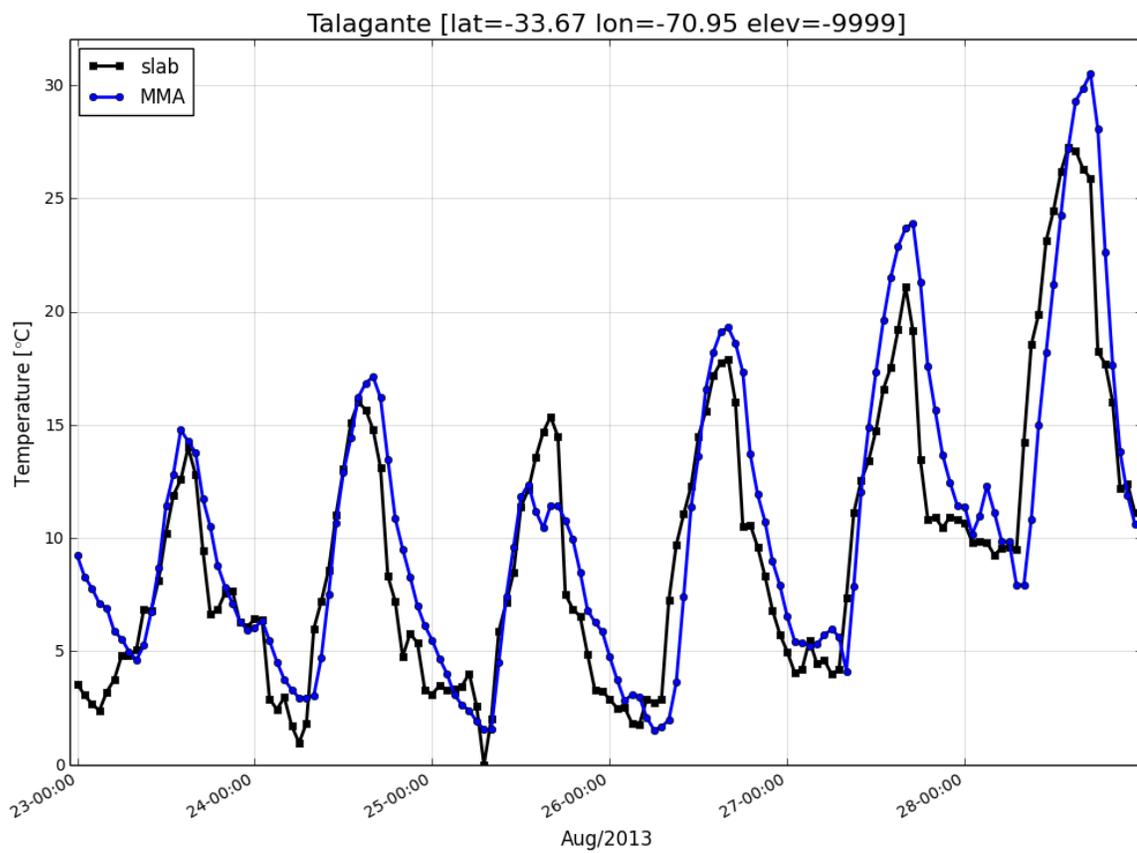


Figura B.4: Serie de tiempo para SLAB de variable Temperatura para estación Talagante en el periodo entre 23-08-2013 y 29-08-2013.

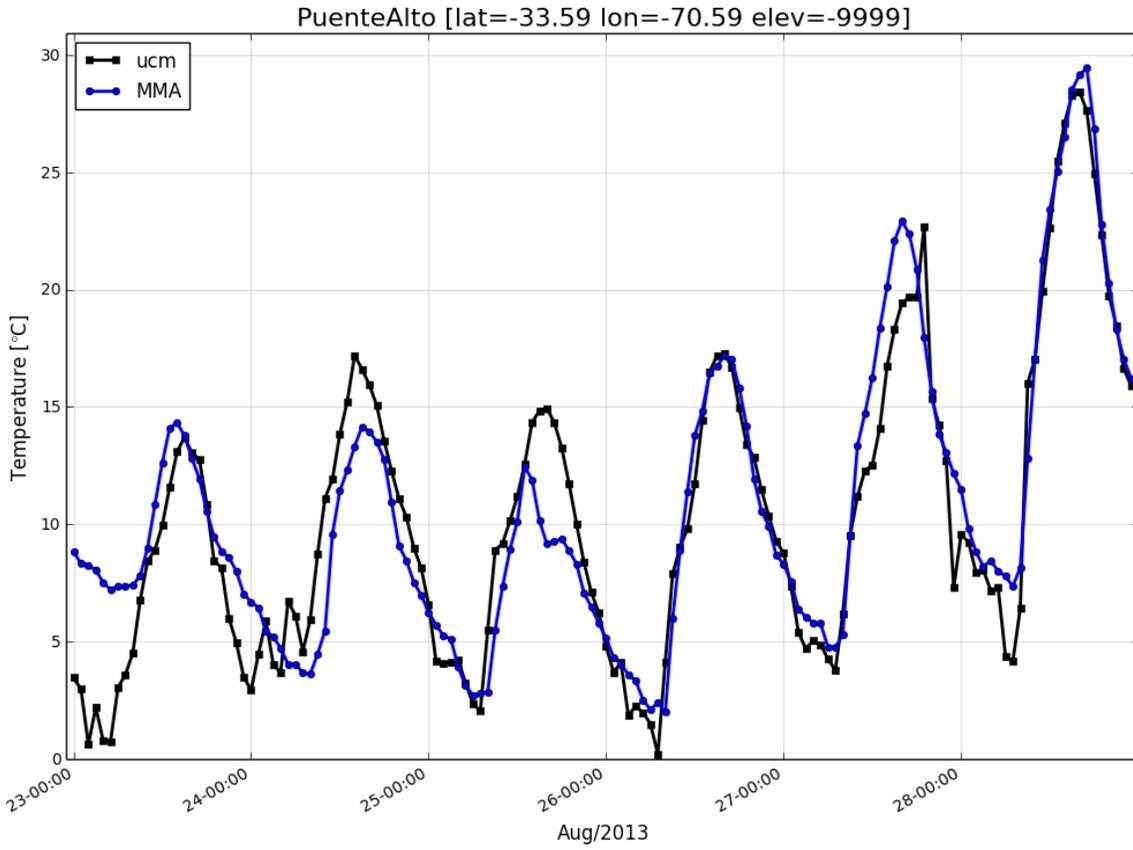


Figura B.5: Serie de tiempo para UCM de variable Temperatura para estación Puente Alto en el periodo entre 23-08-2013 y 29-08-2013.

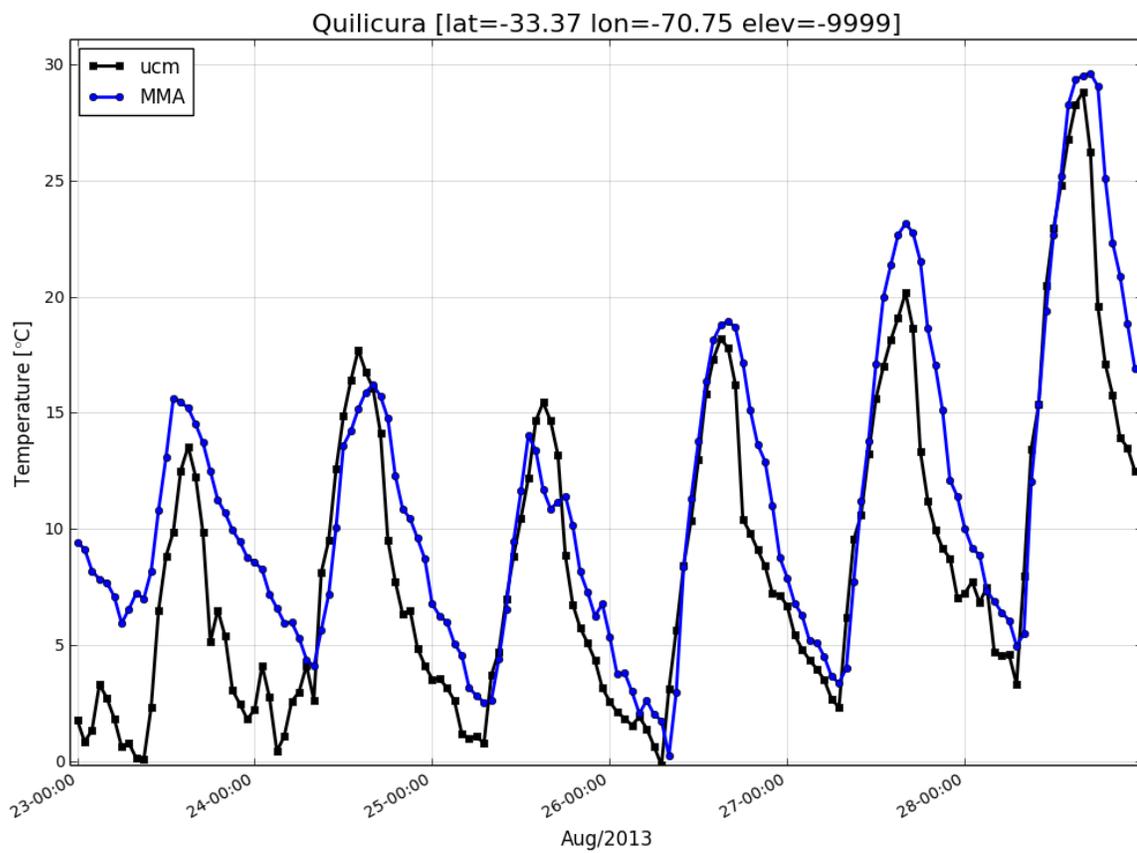


Figura B.6: Serie de tiempo para UCM de variable Temperatura para estación Quilicura en el periodo entre 23-08-2013 y 29-08-2013.

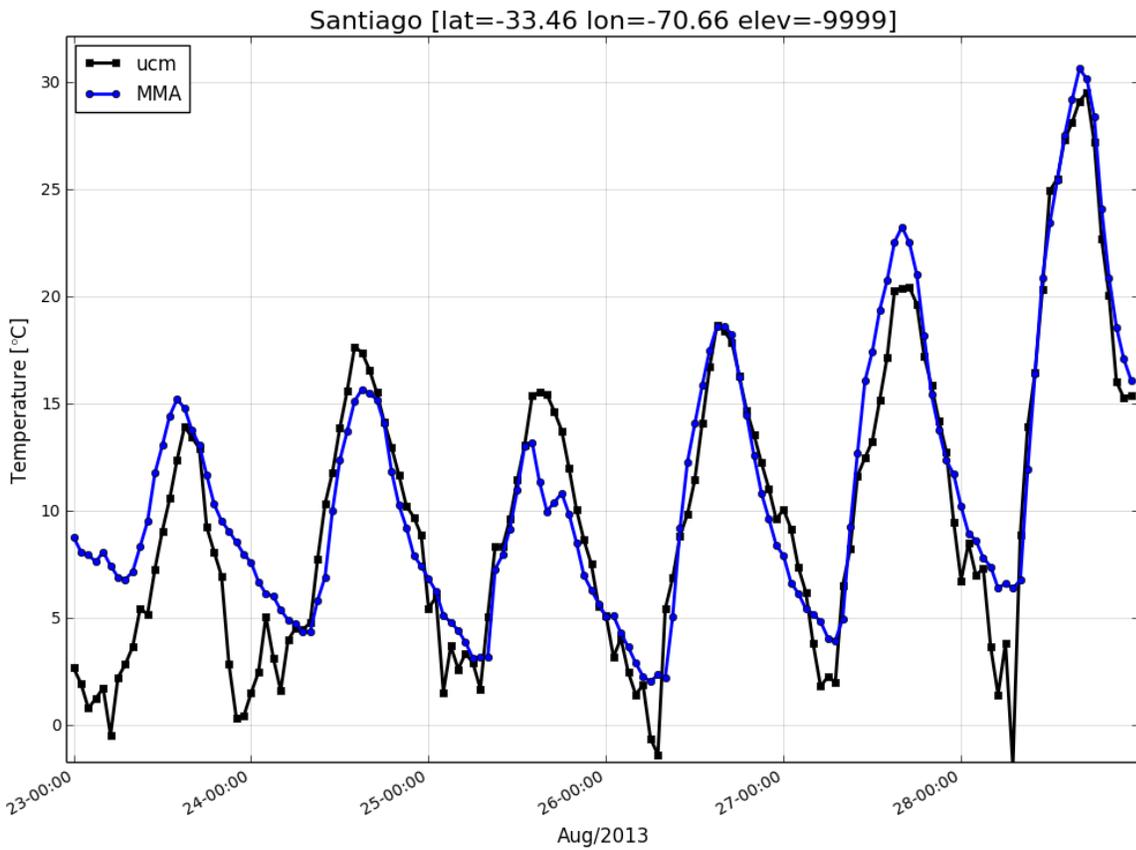


Figura B.7: Serie de tiempo para UCM de variable Temperatura para estación Santiago en el periodo entre 23-08-2013 y 29-08-2013.

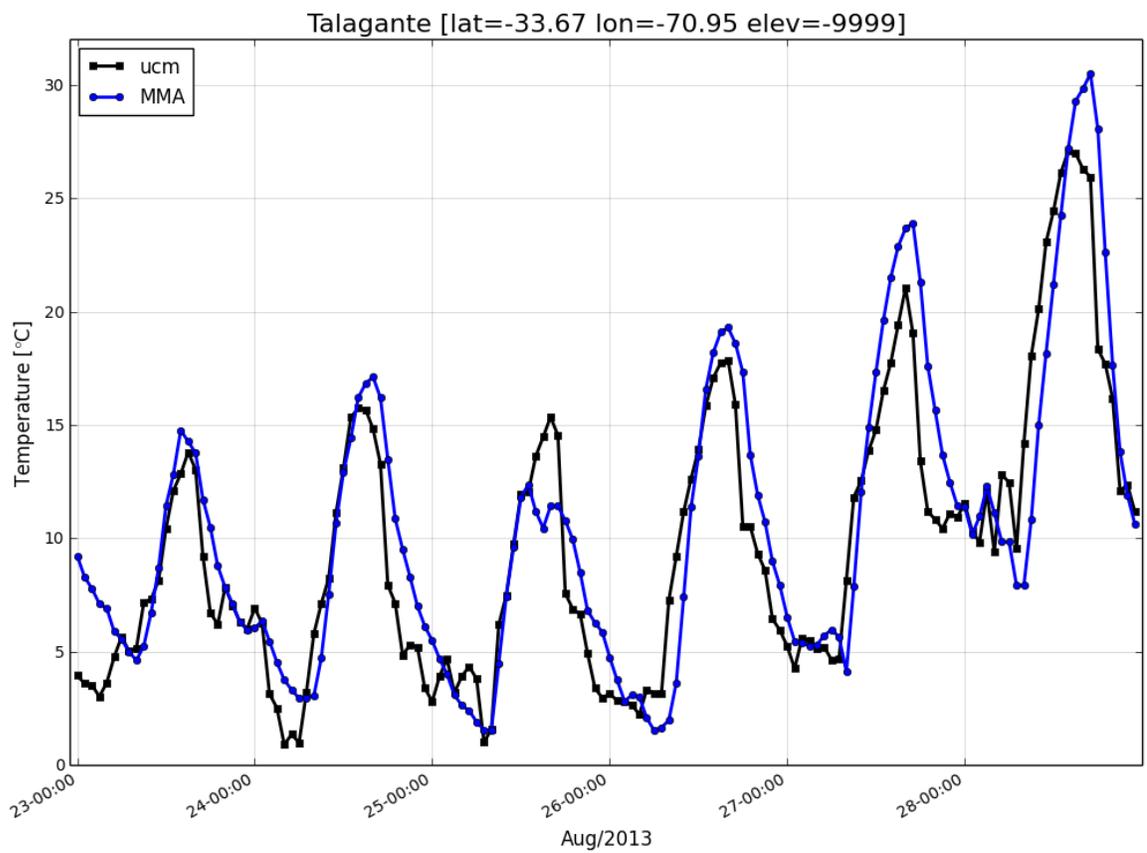


Figura B.8: Serie de tiempo para UCM de variable Temperatura para estación Talagante en el periodo entre 23-08-2013 y 29-08-2013.

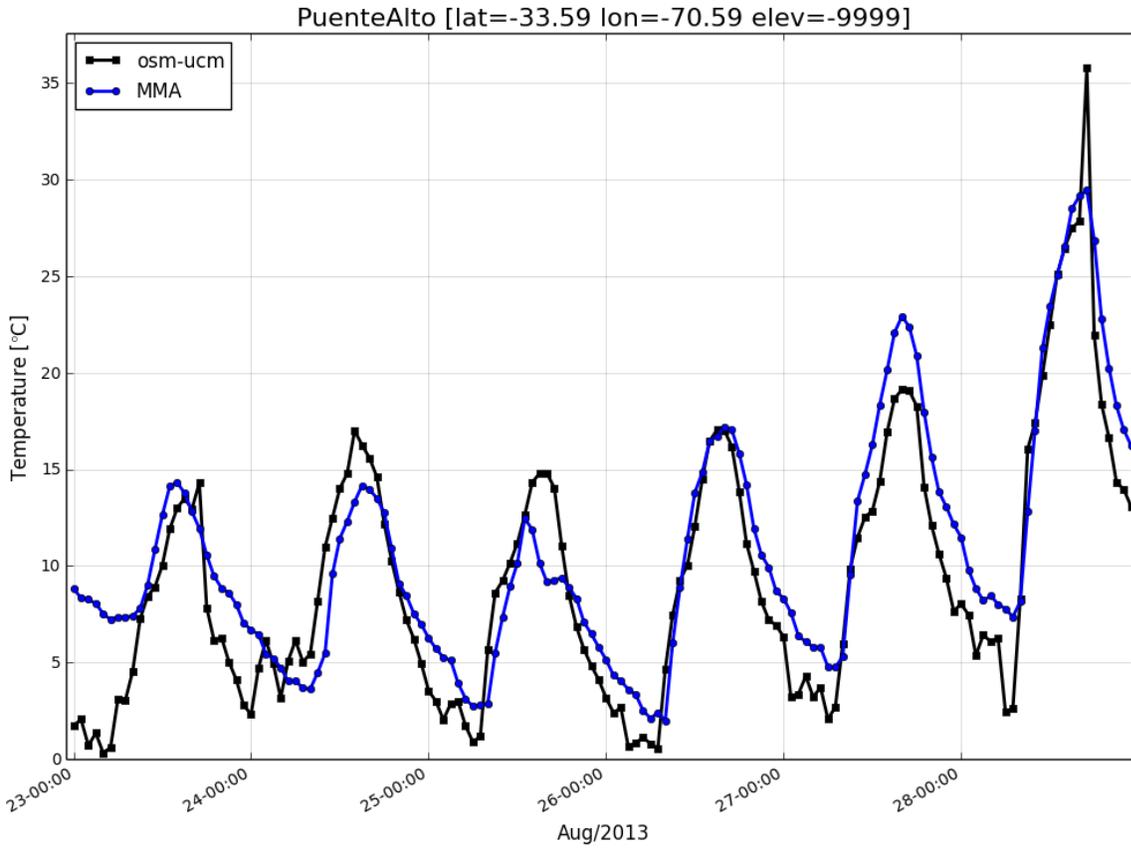


Figura B.9: Serie de tiempo para OSMUCM de variable Temperatura para estación Puente Alto en el periodo entre 23-08-2013 y 29-08-2013.

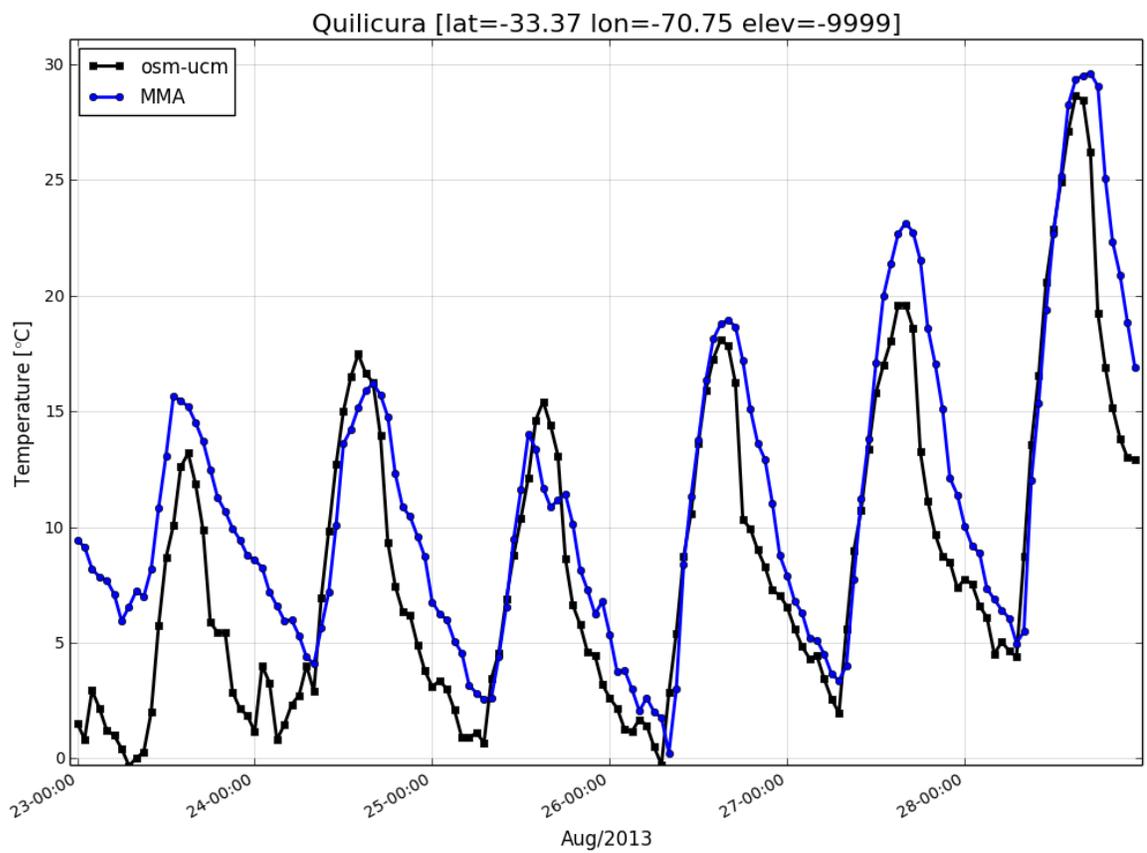


Figura B.10: Serie de tiempo para OSMUCM de variable Temperatura para estación Quilicura en el periodo entre 23-08-2013 y 29-08-2013.

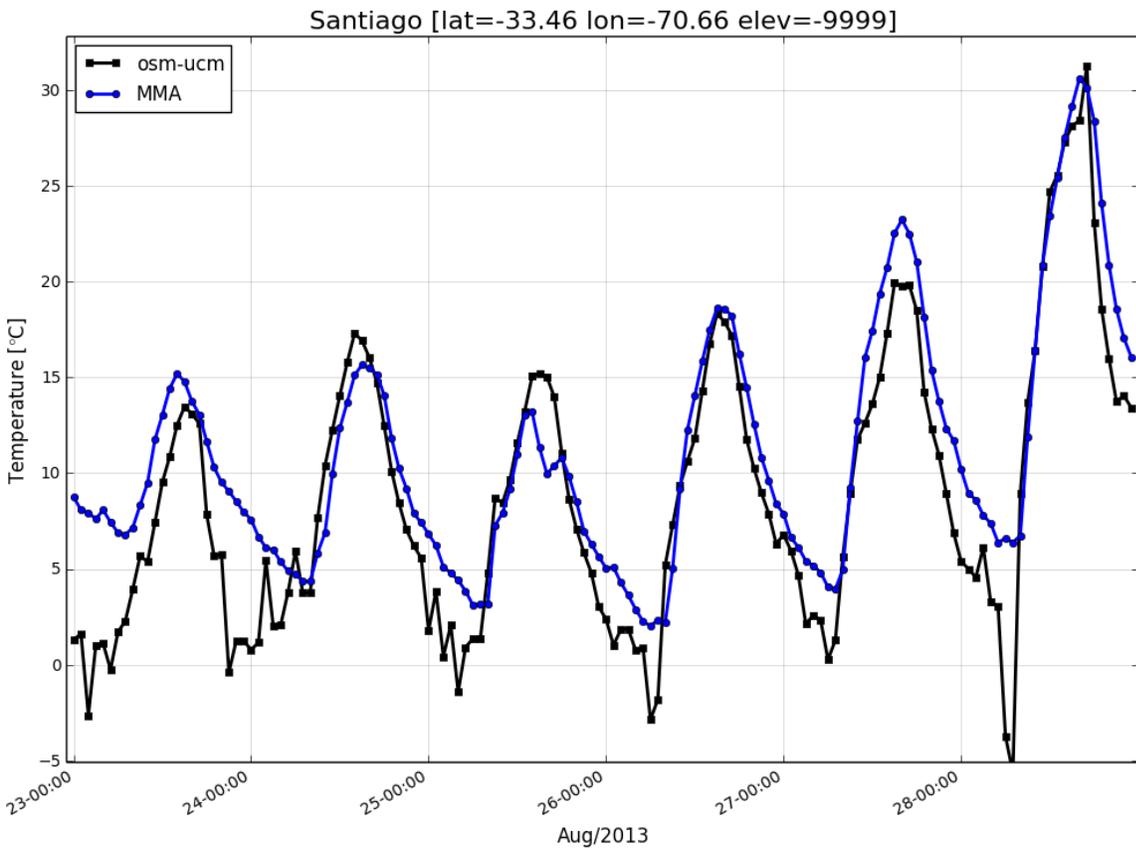


Figura B.11: Serie de tiempo para OSMUCM de variable Temperatura para estación Santiago en el periodo entre 23-08-2013 y 29-08-2013.

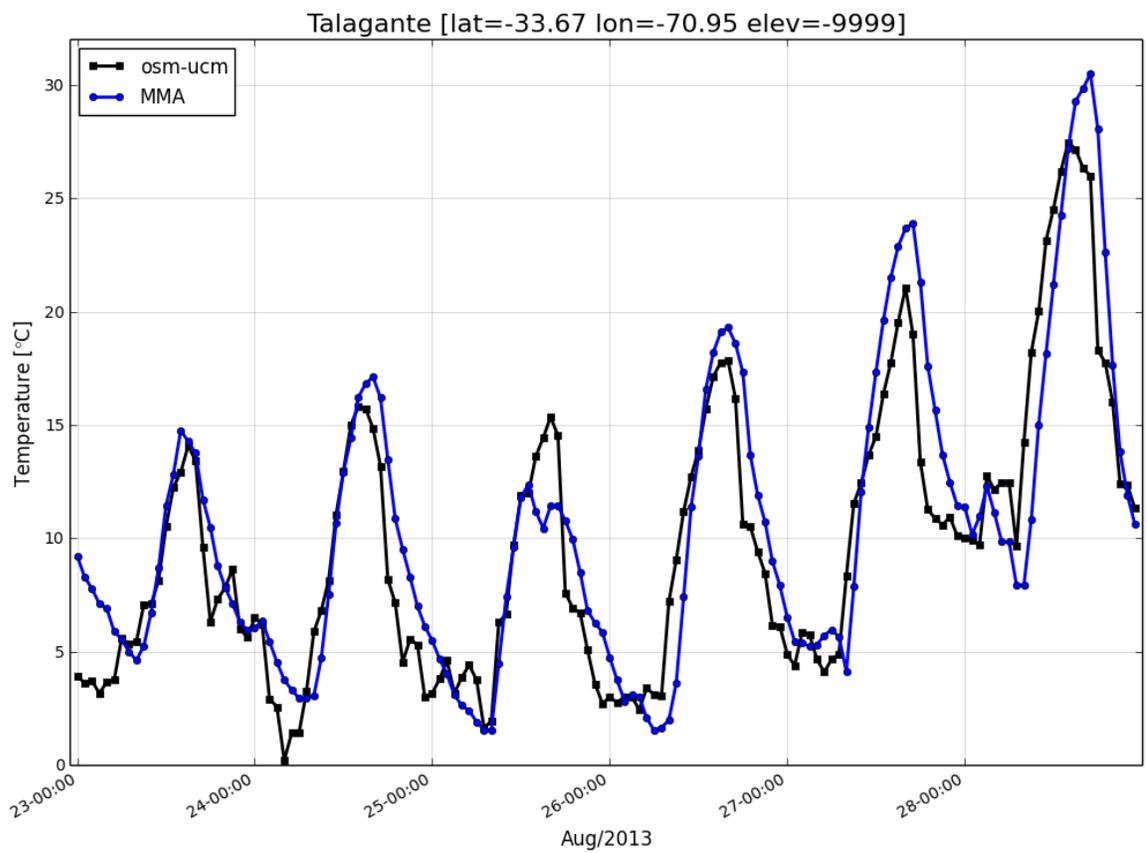


Figura B.12: Serie de tiempo para OSMUCM de variable Temperatura para estación Talagante en el periodo entre 23-08-2013 y 29-08-2013.

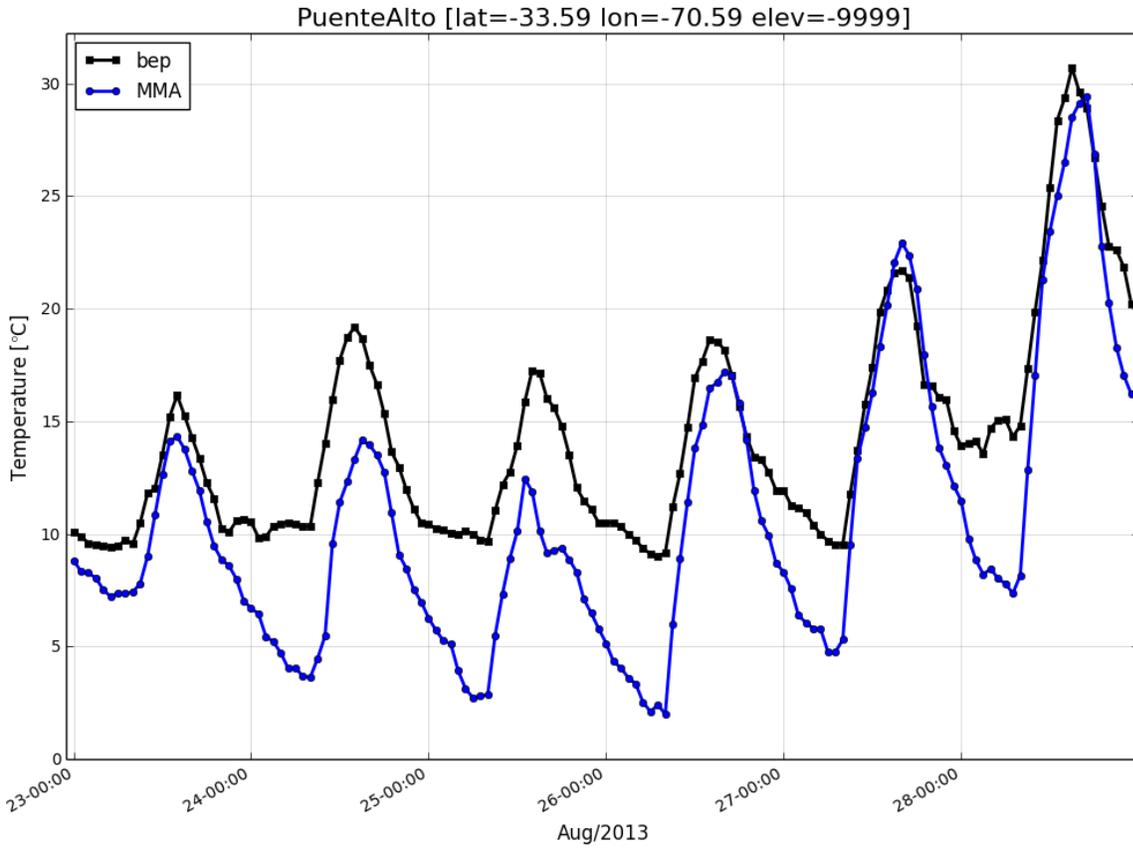


Figura B.13: Serie de tiempo para BEP de variable Temperatura para estación Puente Alto en el periodo entre 23-08-2013 y 29-08-2013.

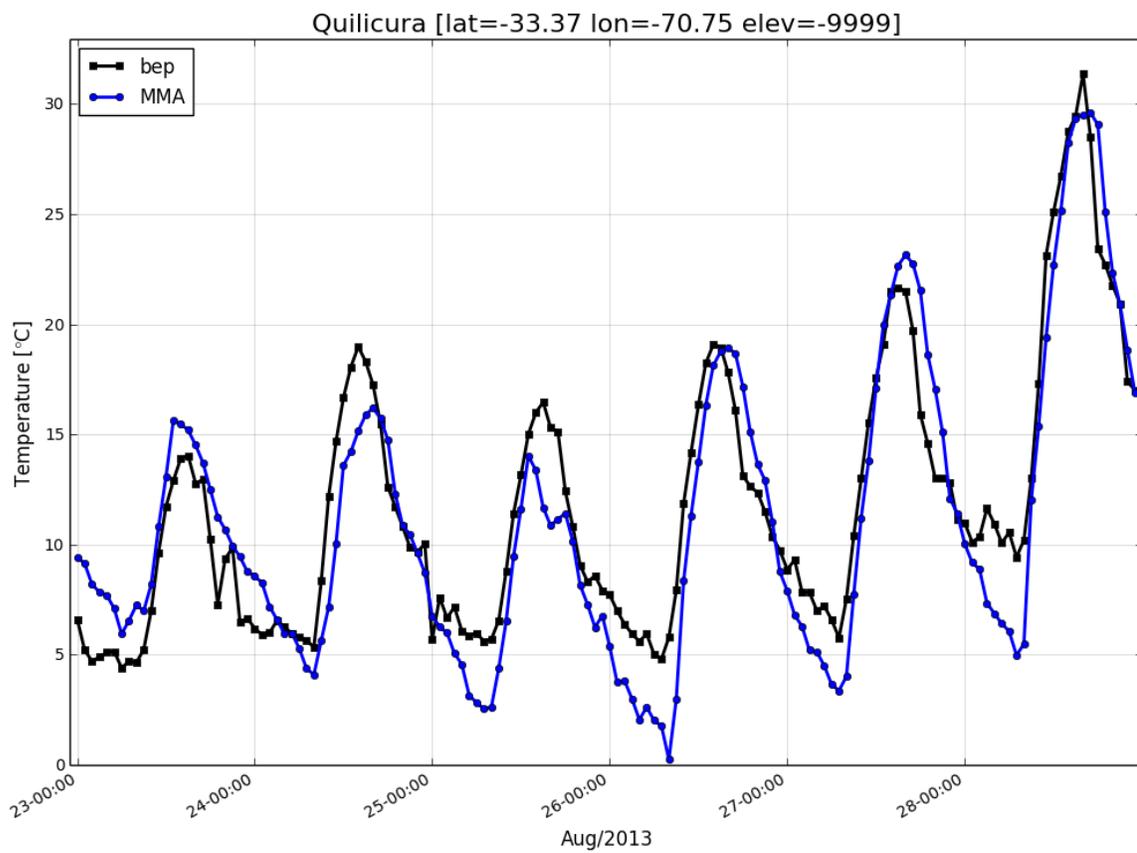


Figura B.14: Serie de tiempo para BEP de variable Temperatura para estación Quilicura en el periodo entre 23-08-2013 y 29-08-2013.

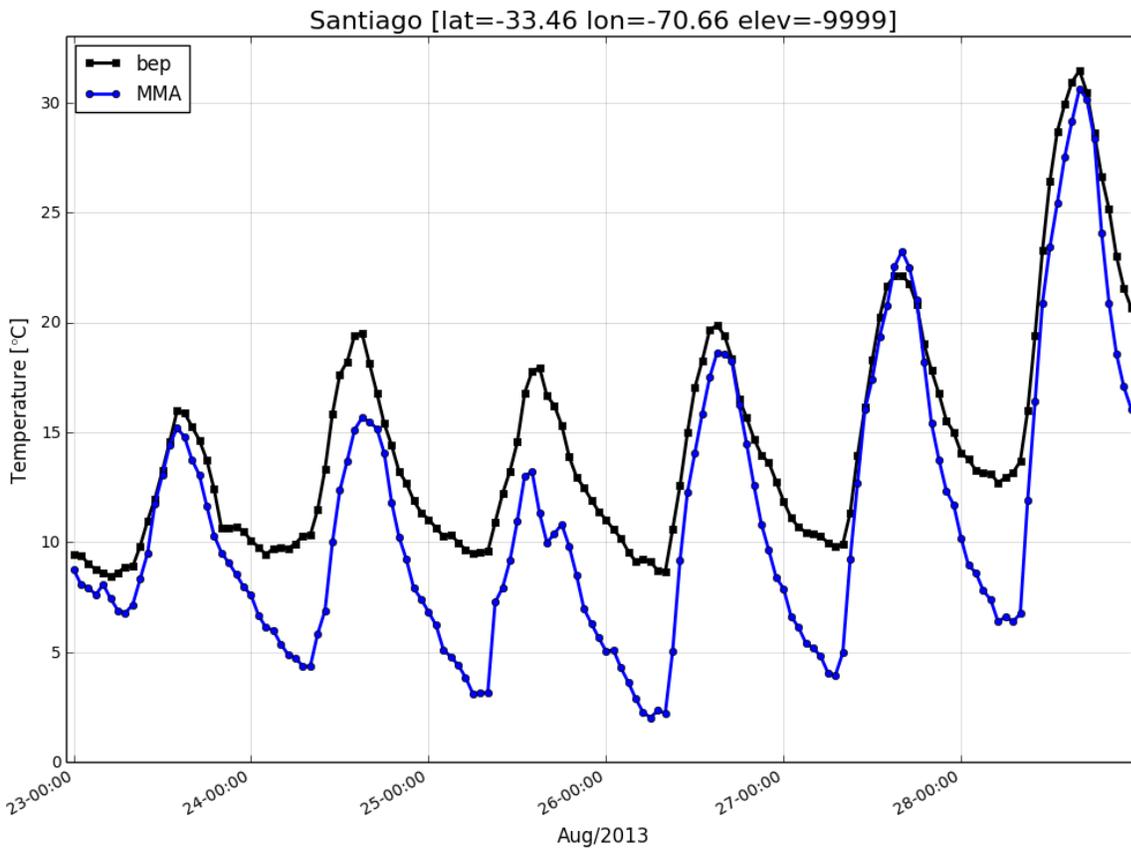


Figura B.15: Serie de tiempo para BEP de variable Temperatura para estación Santiago en el periodo entre 23-08-2013 y 29-08-2013.

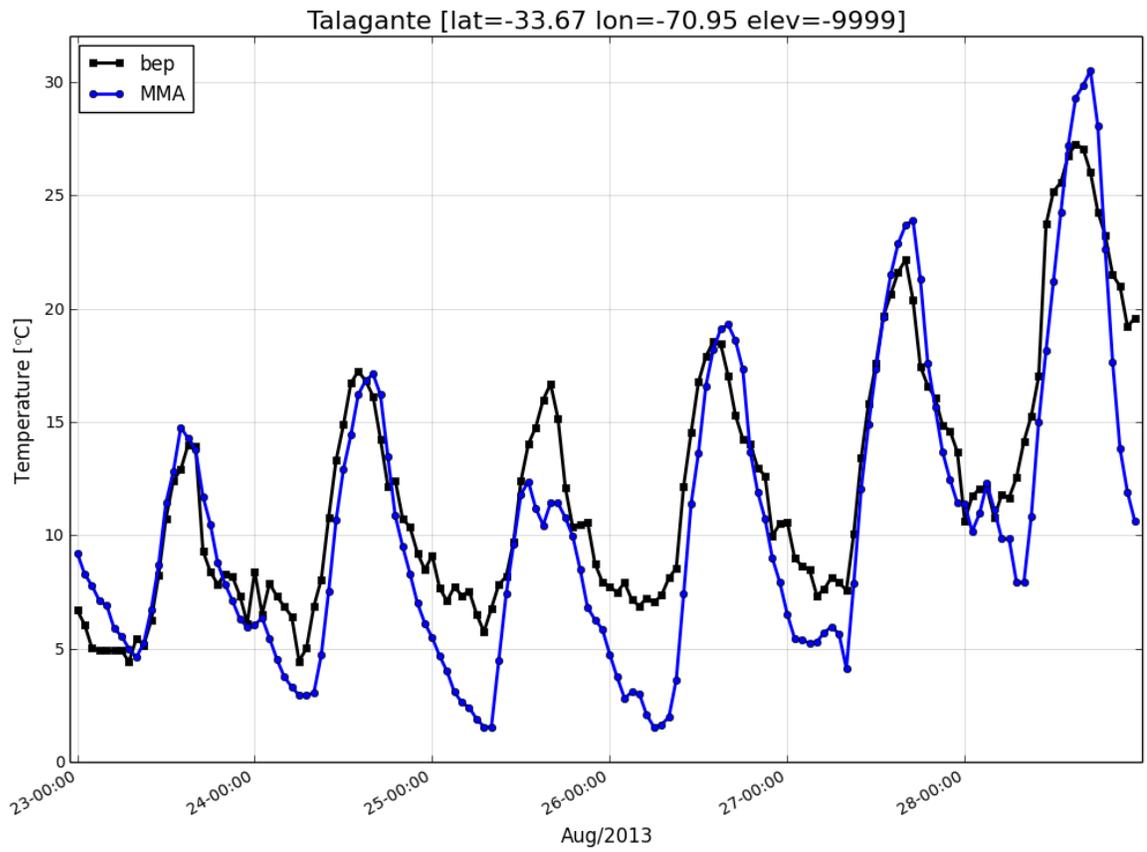


Figura B.16: Serie de tiempo para BEP de variable Temperatura para estación Talagante en el periodo entre 23-08-2013 y 29-08-2013.

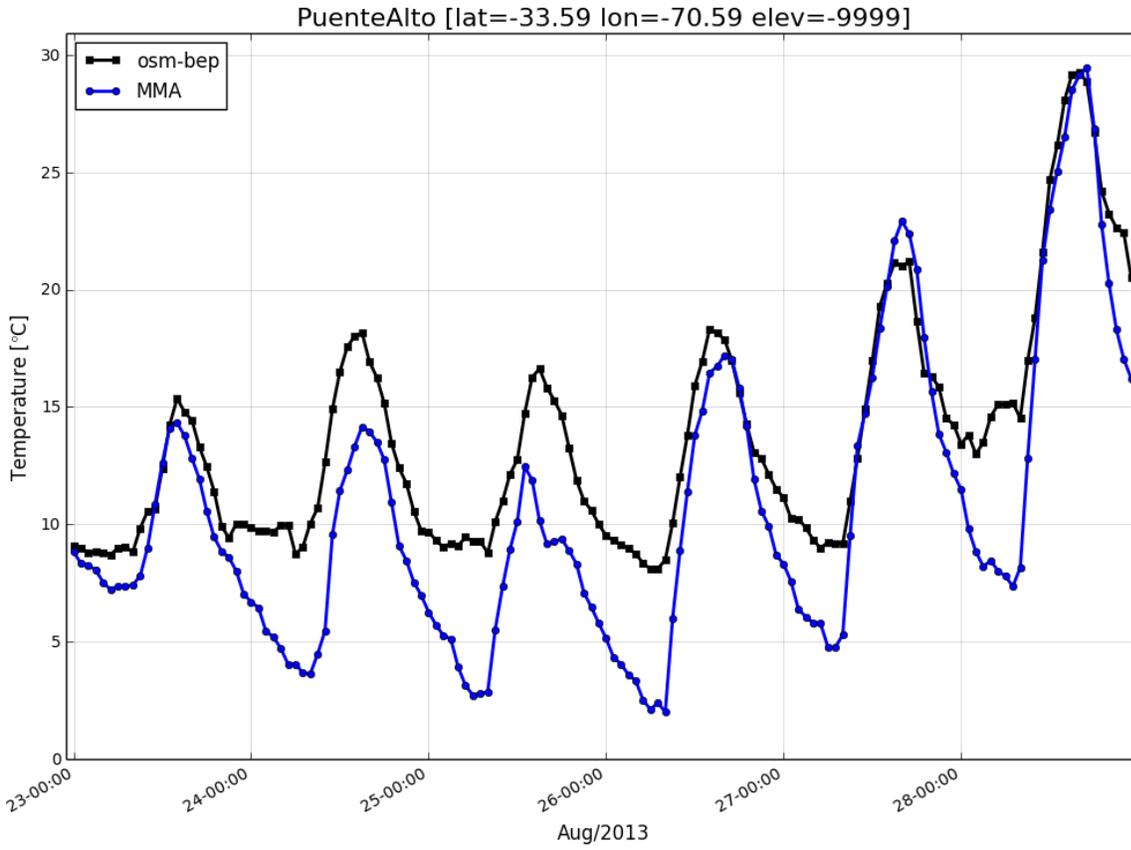


Figura B.17: Serie de tiempo para OSMBEP de variable Temperatura para estación Puente Alto en el periodo entre 23-08-2013 y 29-08-2013.

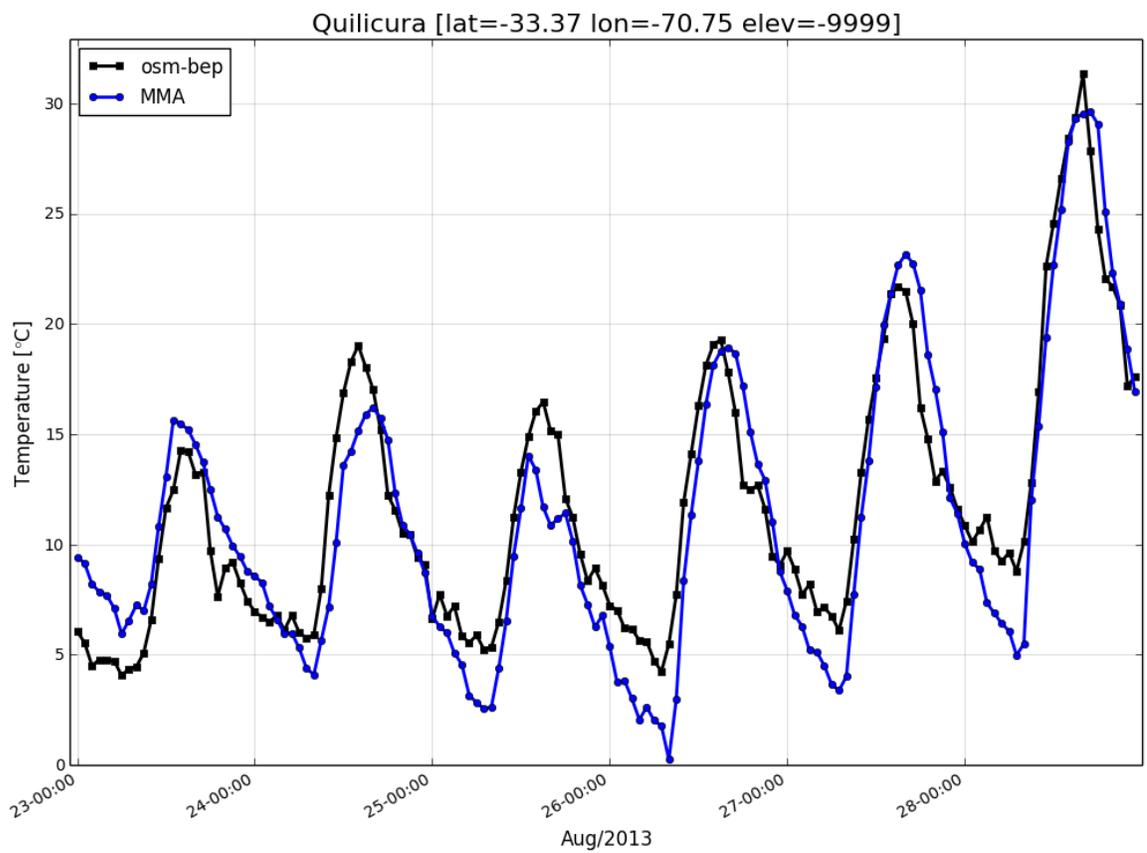


Figura B.18: Serie de tiempo para OSMBEP de variable Temperatura para estación Quilicura en el periodo entre 23-08-2013 y 29-08-2013.

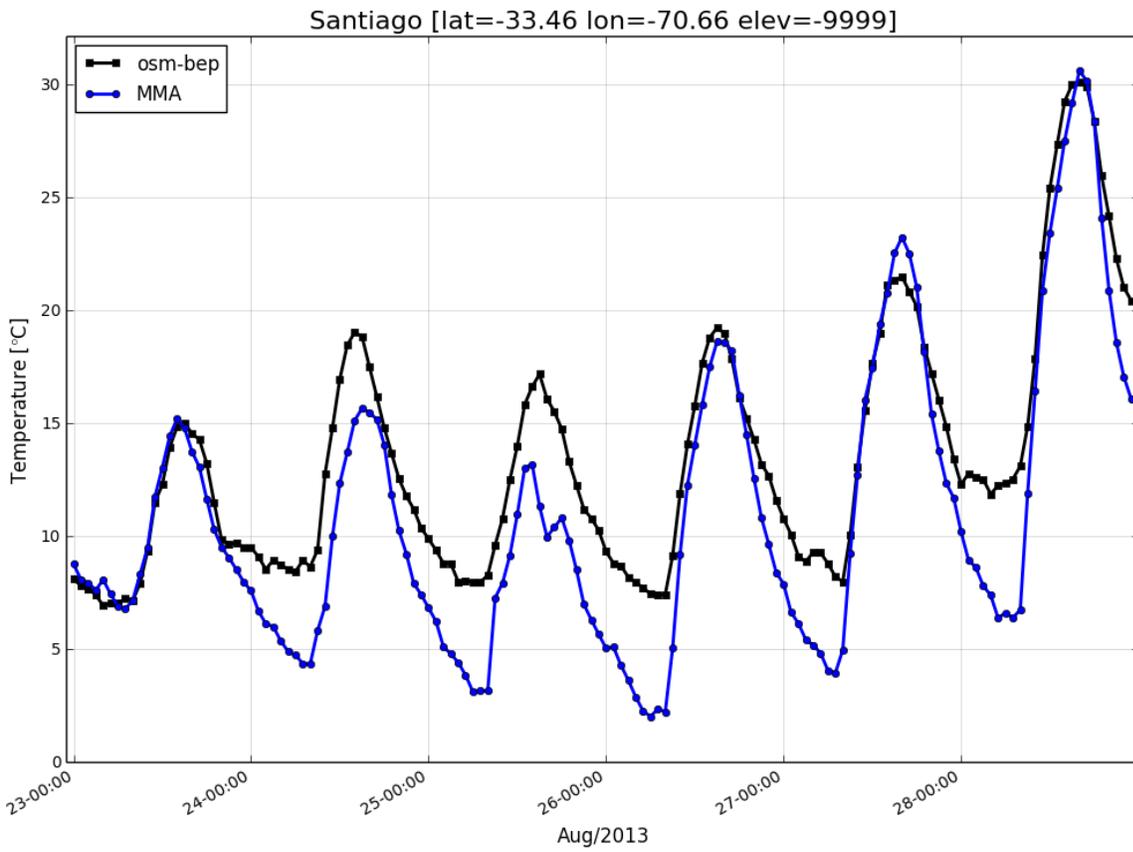


Figura B.19: Serie de tiempo para OSMBEP de variable Temperatura para estación Santiago en el periodo entre 23-08-2013 y 29-08-2013.

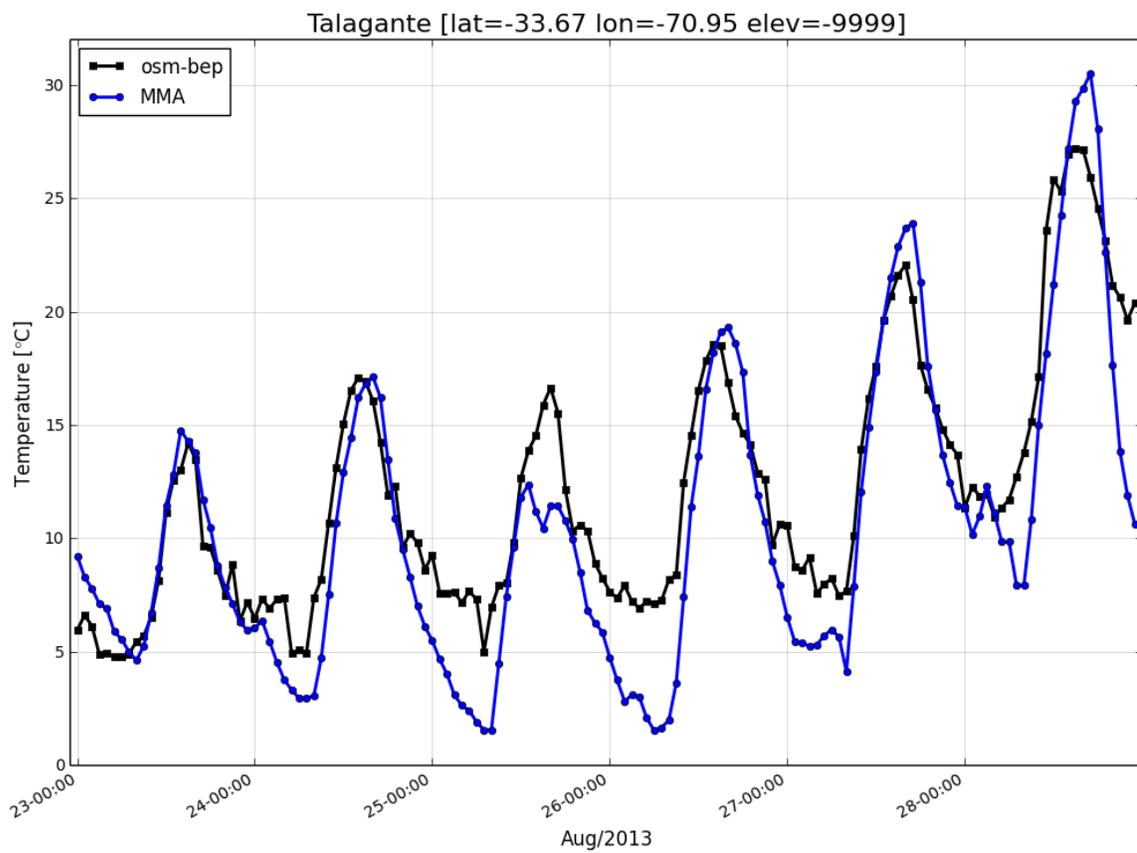


Figura B.20: Serie de tiempo para OSMBEP de variable Temperatura para estación Talagante en el periodo entre 23-08-2013 y 29-08-2013.

B.2. Magnitud de viento

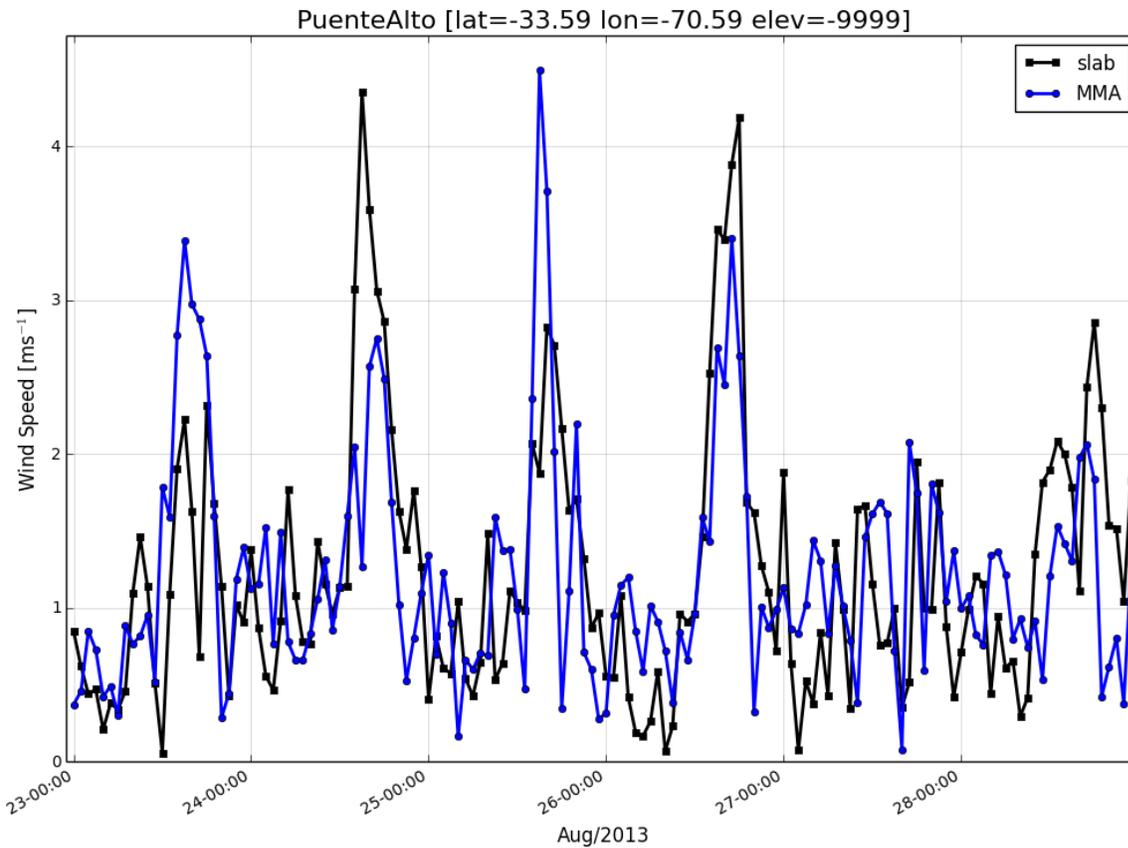


Figura B.21: Serie de tiempo para SLAB de variable Magnitud de Viento para estación Puente Alto en el periodo entre 23-08-2013 y 29-08-2013.

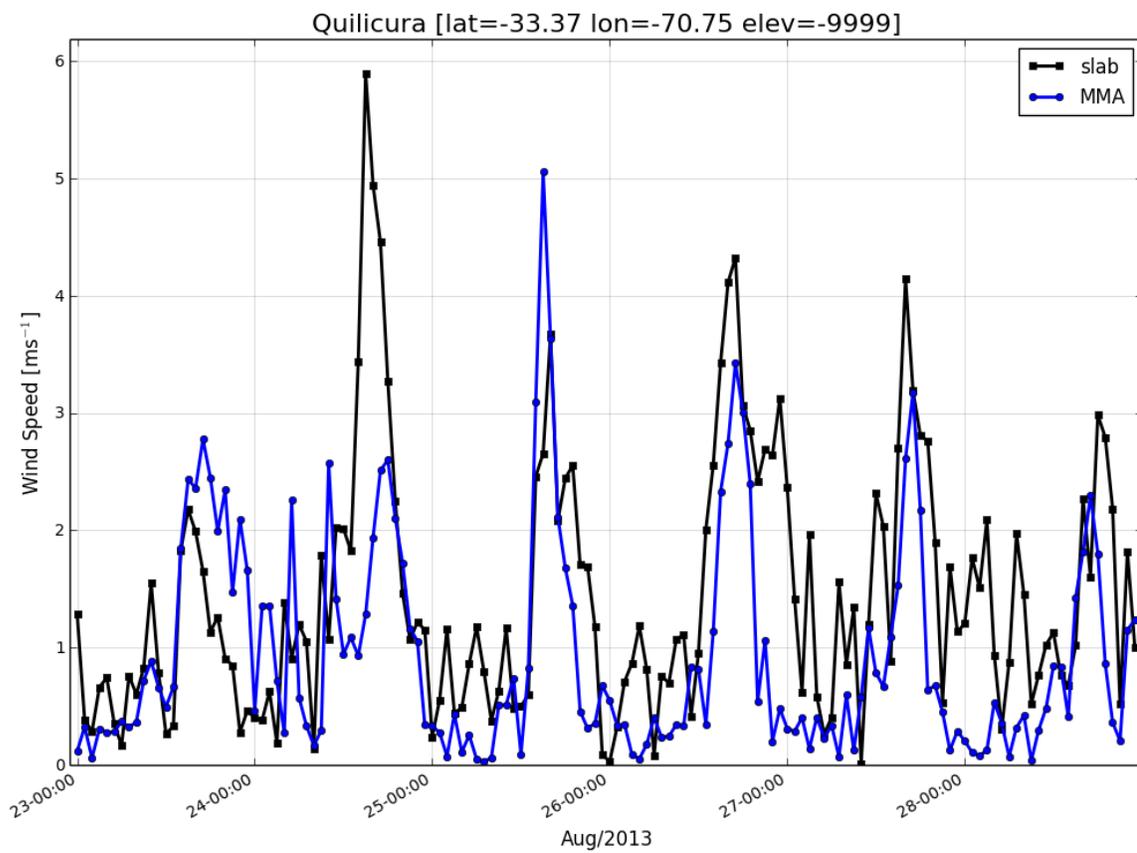


Figura B.22: Serie de tiempo para SLAB de variable Magnitud de Viento para estación Quilicura en el periodo entre 23-08-2013 y 29-08-2013.

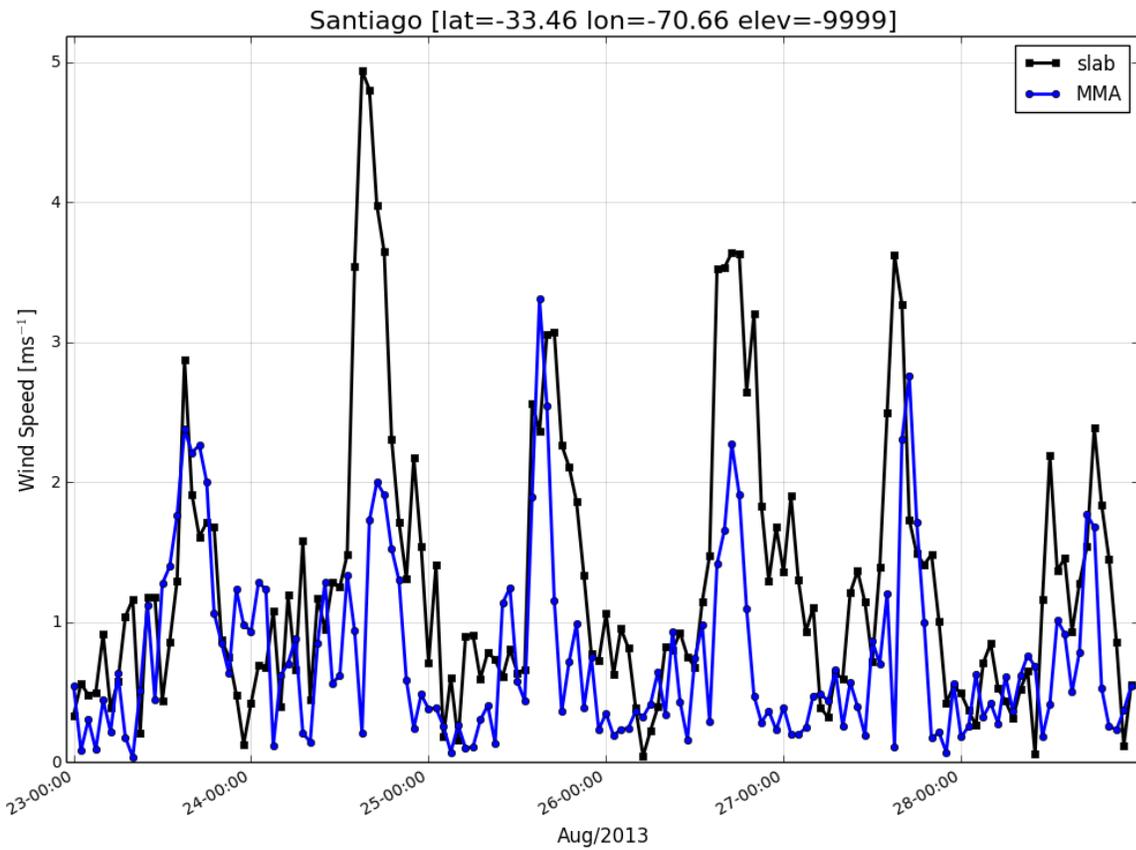


Figura B.23: Serie de tiempo para SLAB de variable Magnitud de Viento para estación Santiago en el periodo entre 23-08-2013 y 29-08-2013.

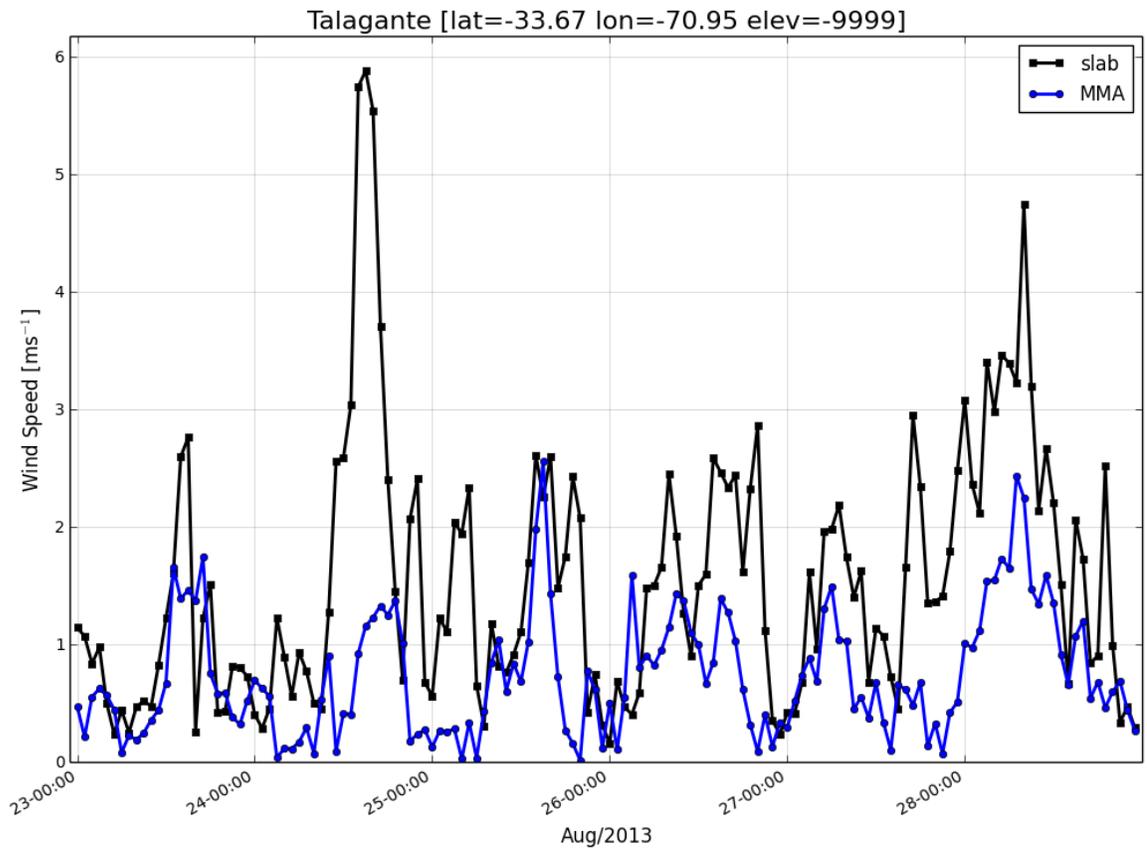


Figura B.24: Serie de tiempo para SLAB de variable Magnitud de Viento para estación Talagante en el periodo entre 23-08-2013 y 29-08-2013.

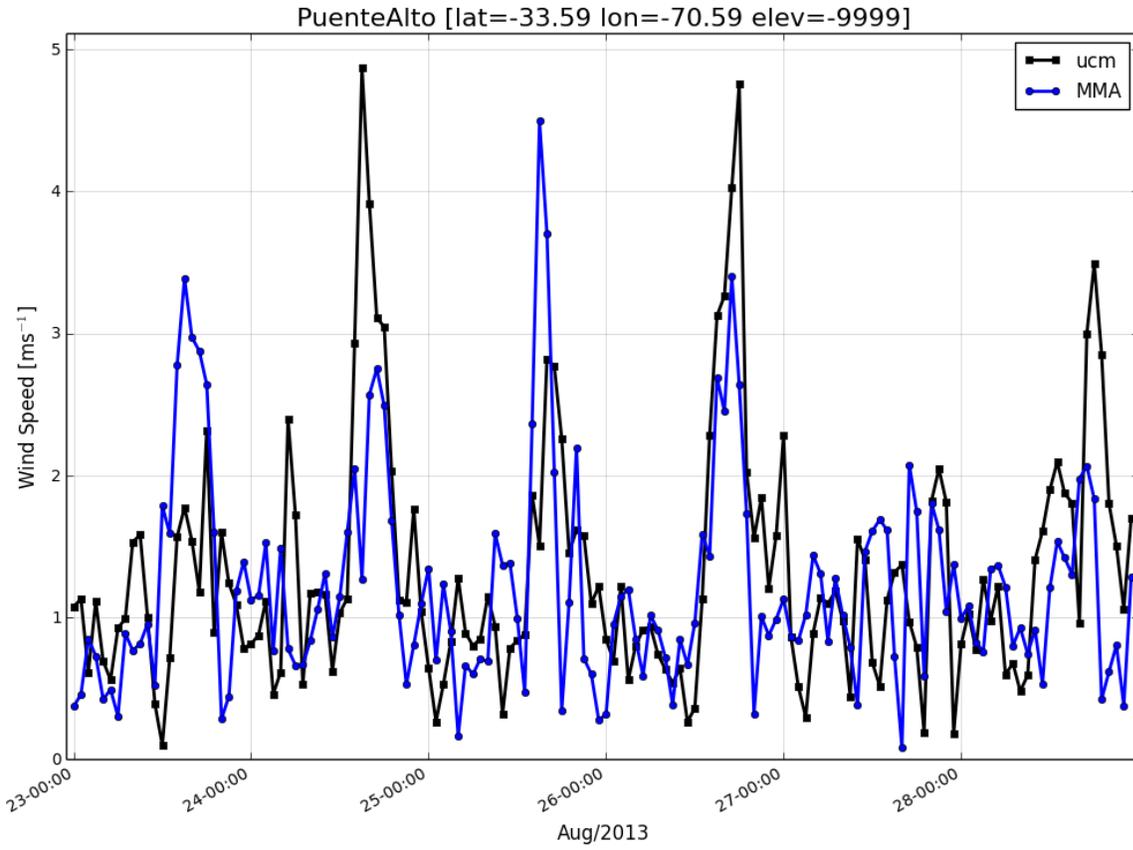


Figura B.25: Serie de tiempo para UCM de variable Magnitud de Viento para estación Puente Alto en el periodo entre 23-08-2013 y 29-08-2013.

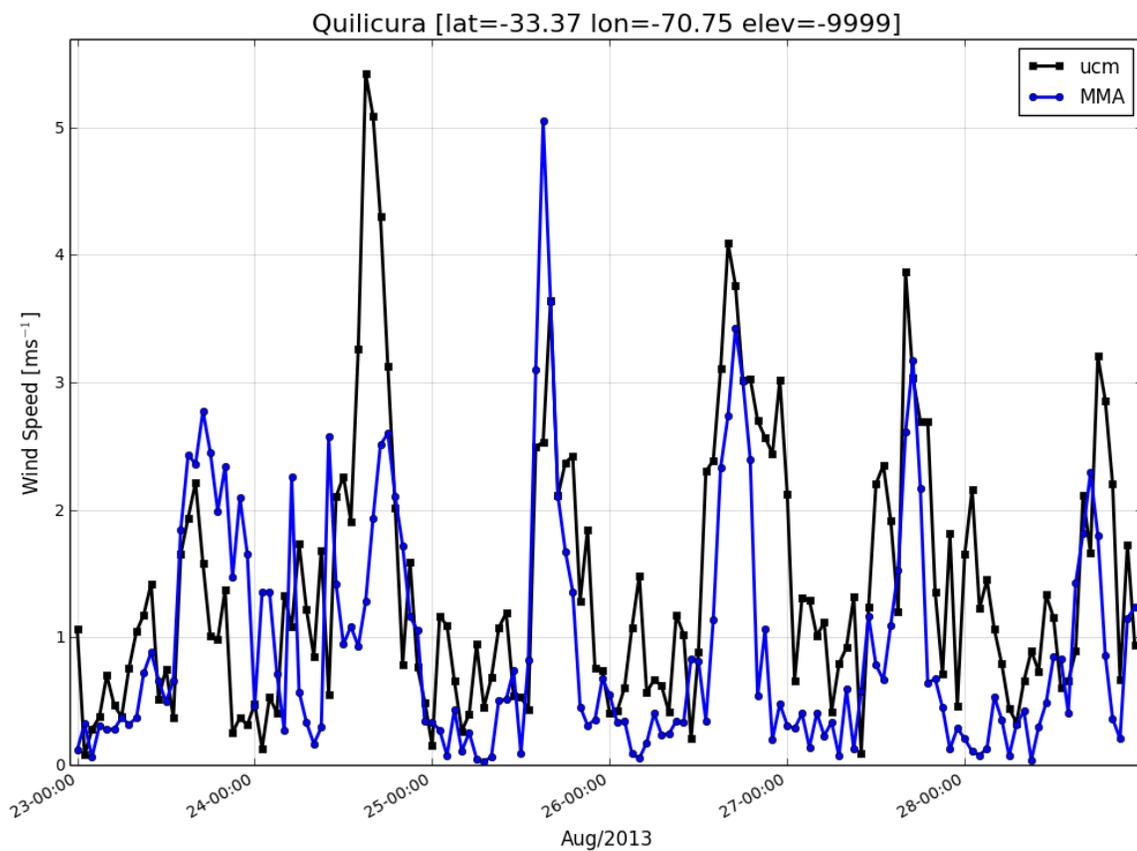


Figura B.26: Serie de tiempo para UCM de variable Magnitud de Viento para estación Quilicura en el periodo entre 23-08-2013 y 29-08-2013.

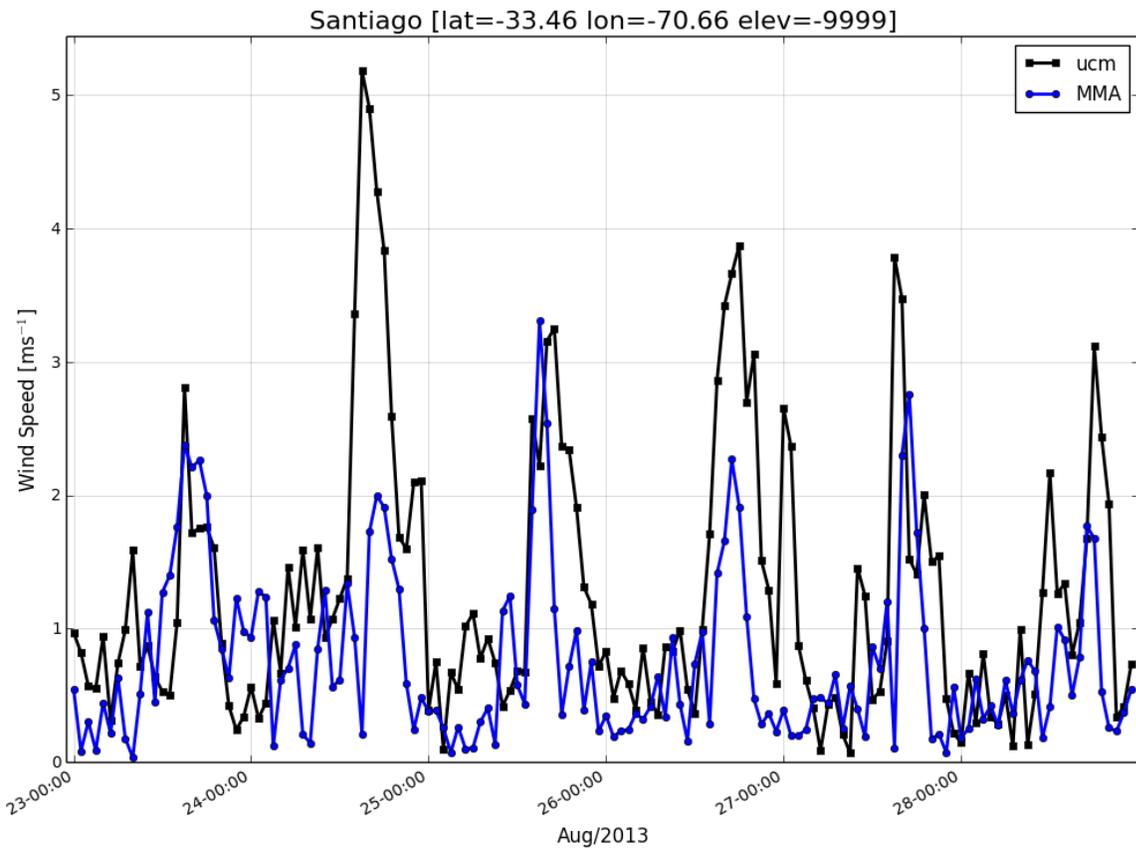


Figura B.27: Serie de tiempo para UCM de variable Magnitud de Viento para estación Santiago en el periodo entre 23-08-2013 y 29-08-2013.

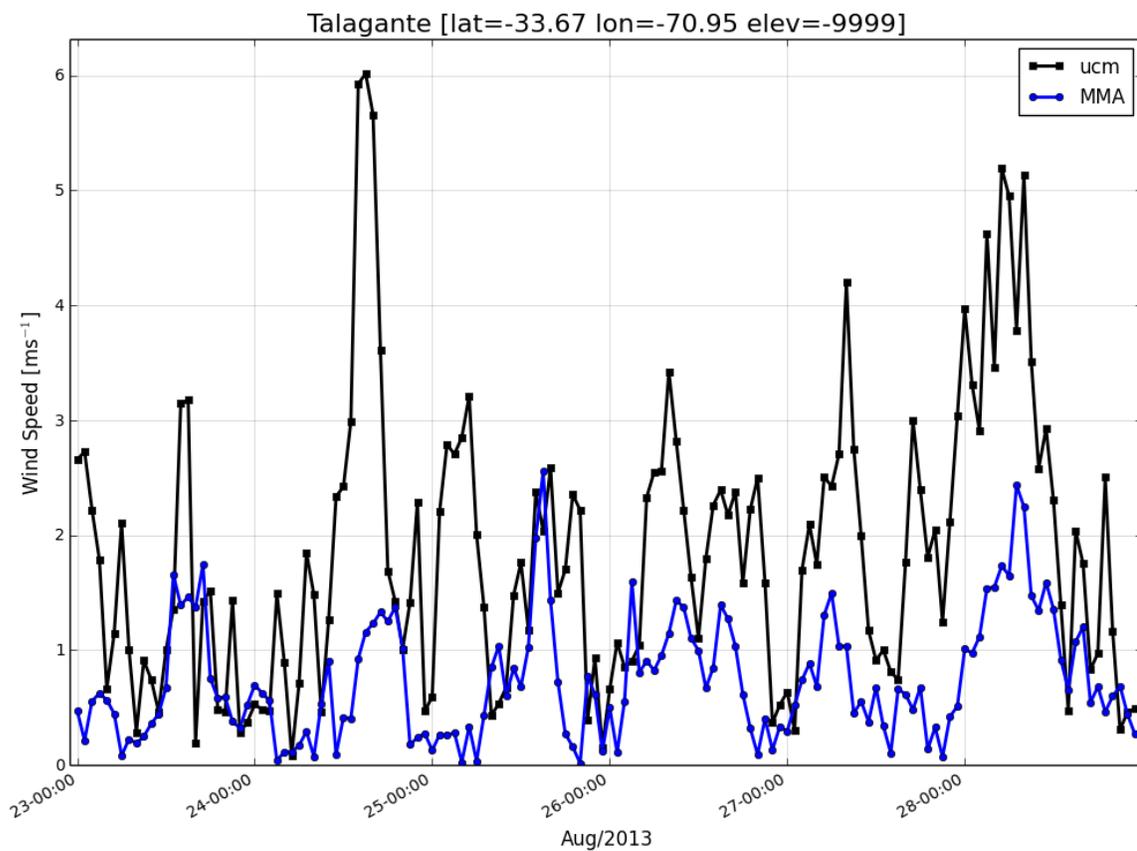


Figura B.28: Serie de tiempo para UCM de variable Magnitud de Viento para estación Talagante en el periodo entre 23-08-2013 y 29-08-2013.

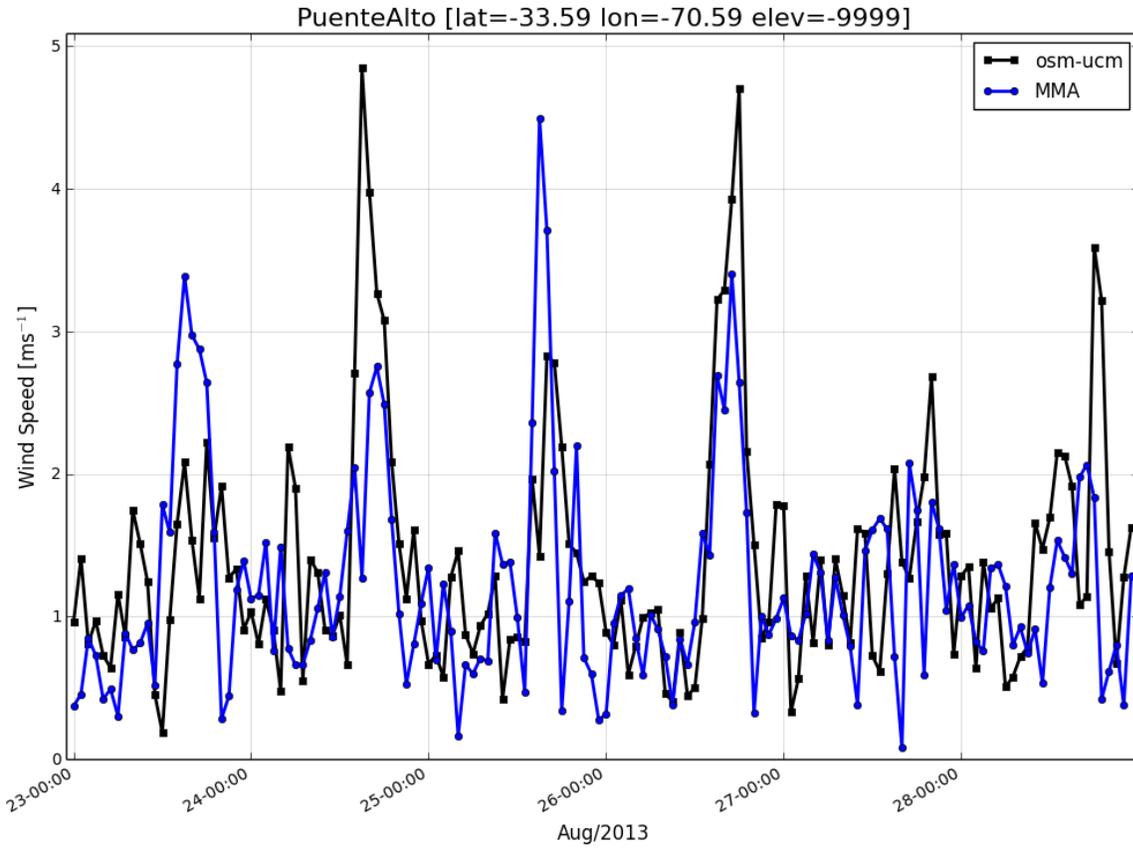


Figura B.29: Serie de tiempo para OSMUCM de variable Magnitud de Viento para estación Puente Alto en el periodo entre 23-08-2013 y 29-08-2013.

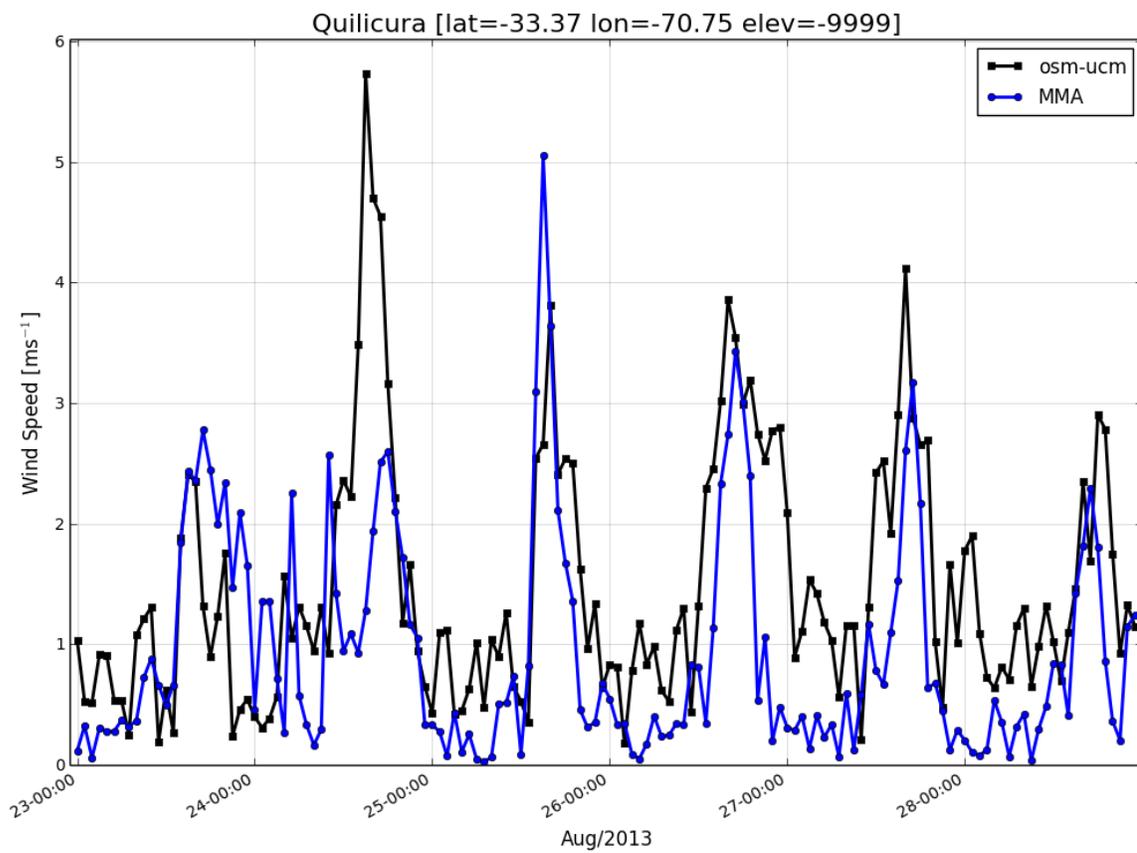


Figura B.30: Serie de tiempo para OSMUCM de variable Magnitud de Viento para estación Quilicura en el periodo entre 23-08-2013 y 29-08-2013.

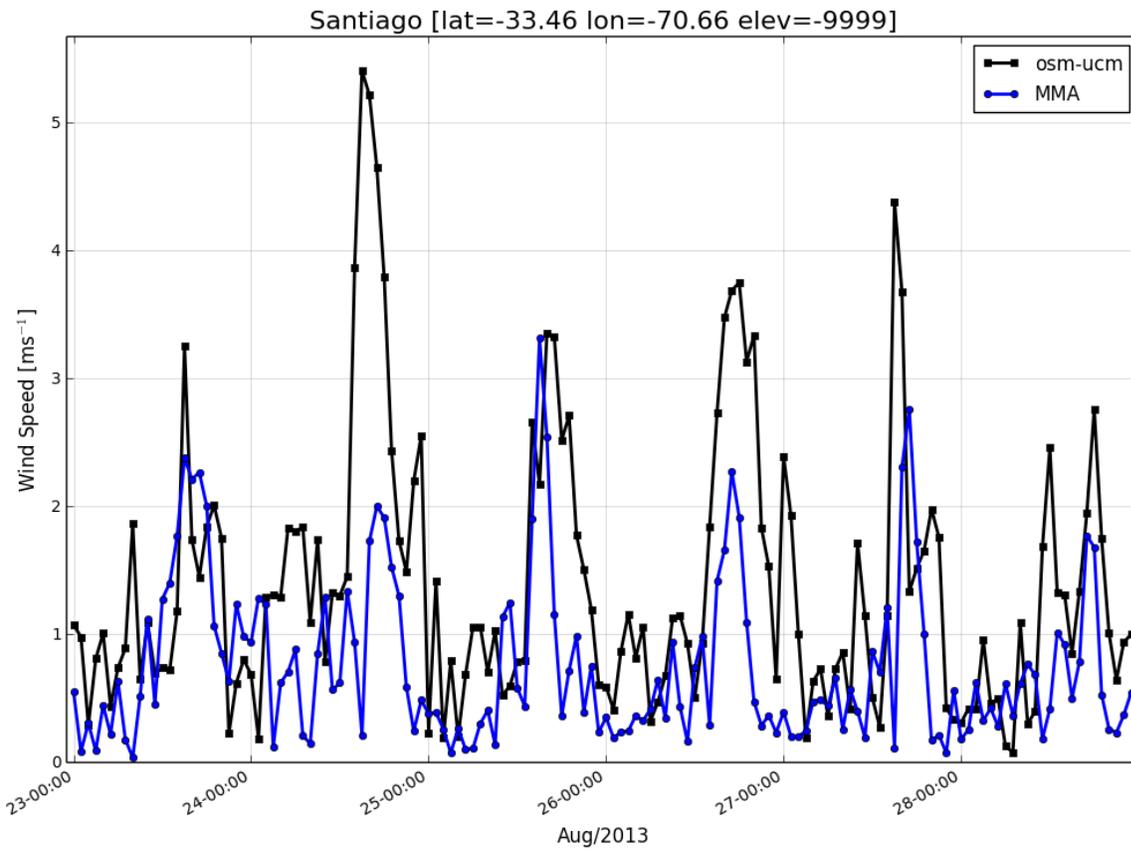


Figura B.31: Serie de tiempo para OSMUCM de variable Magnitud de Viento para estación Santiago en el periodo entre 23-08-2013 y 29-08-2013.

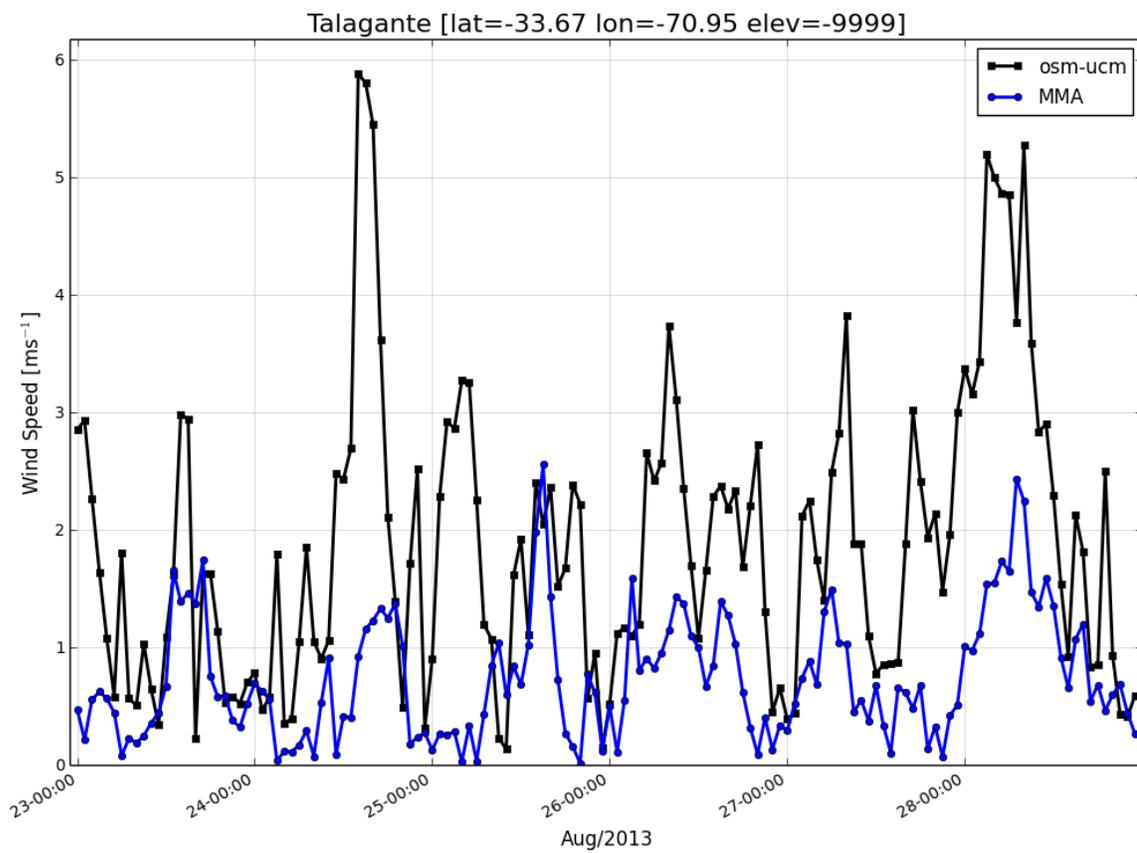


Figura B.32: Serie de tiempo para OSMUCM de variable Magnitud de Viento para estación Talagante en el periodo entre 23-08-2013 y 29-08-2013.

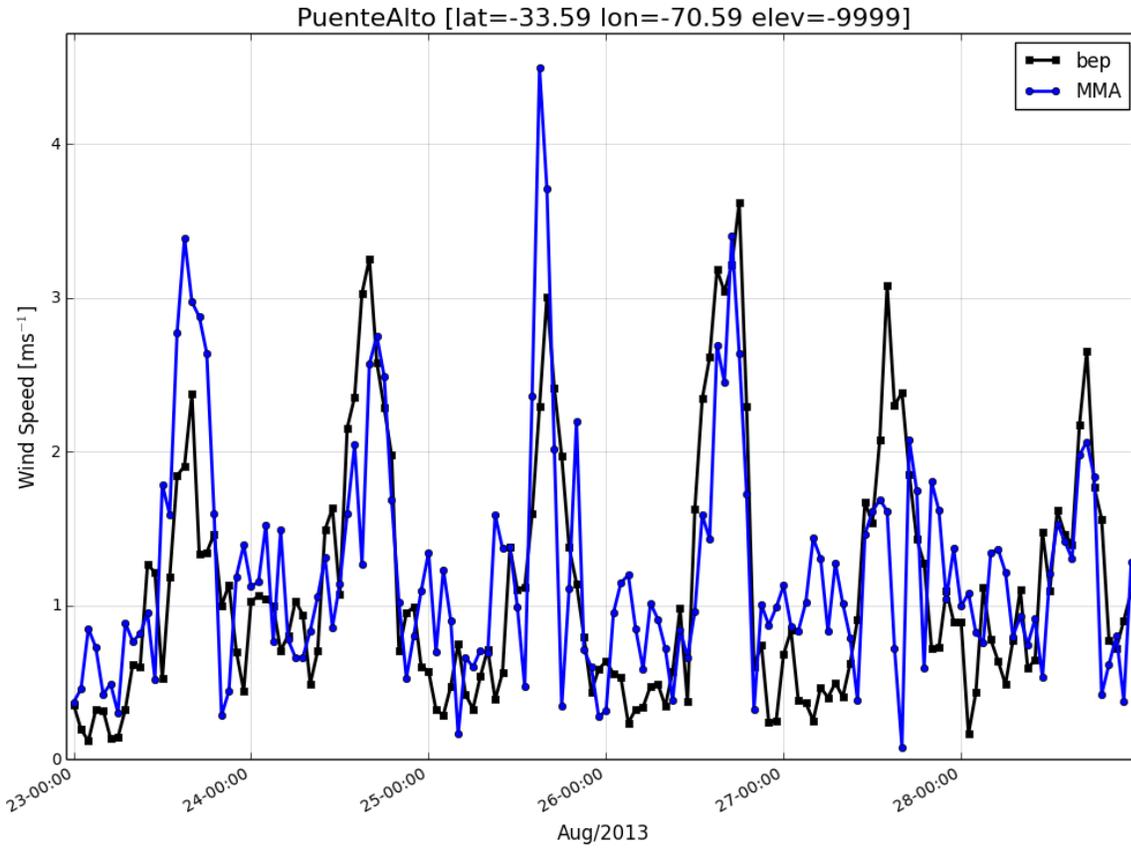


Figura B.33: Serie de tiempo para BEP de variable Magnitud de Viento para estación Puente Alto en el periodo entre 23-08-2013 y 29-08-2013.

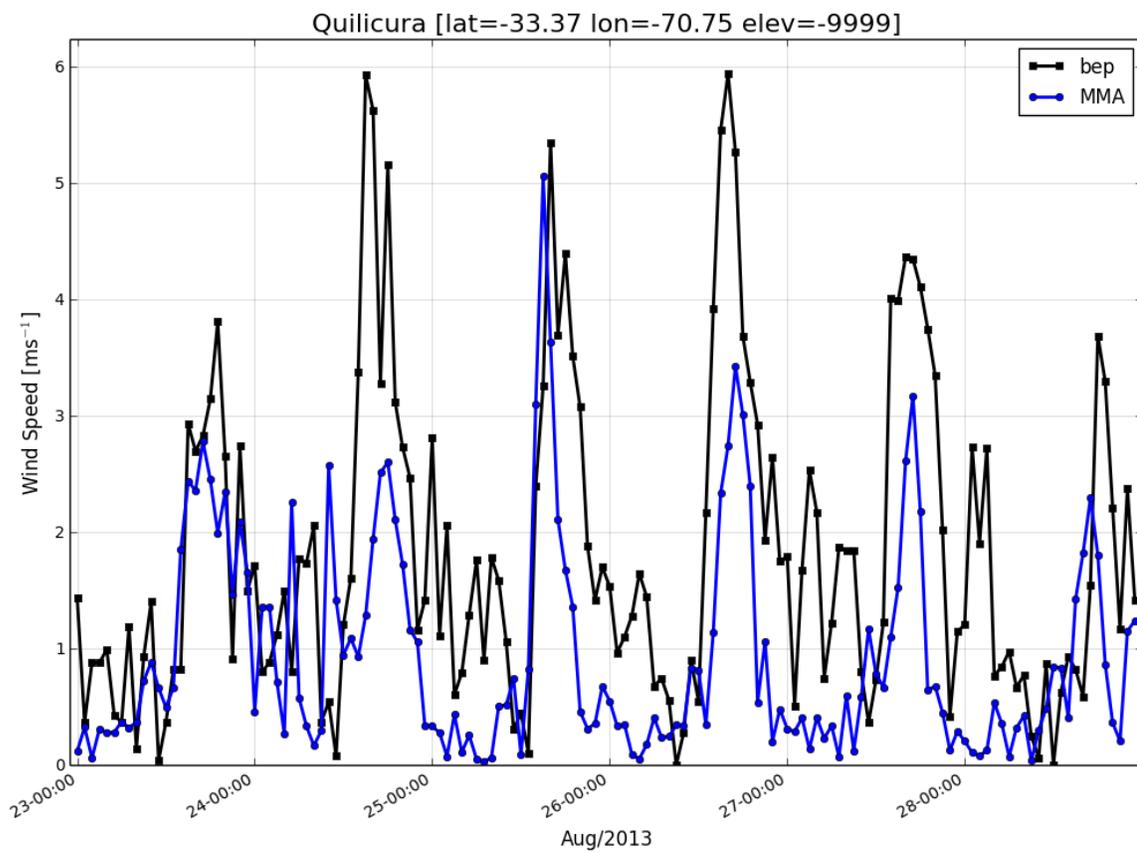


Figura B.34: Serie de tiempo para BEP de variable Magnitud de Viento para estación Quilicura en el periodo entre 23-08-2013 y 29-08-2013.

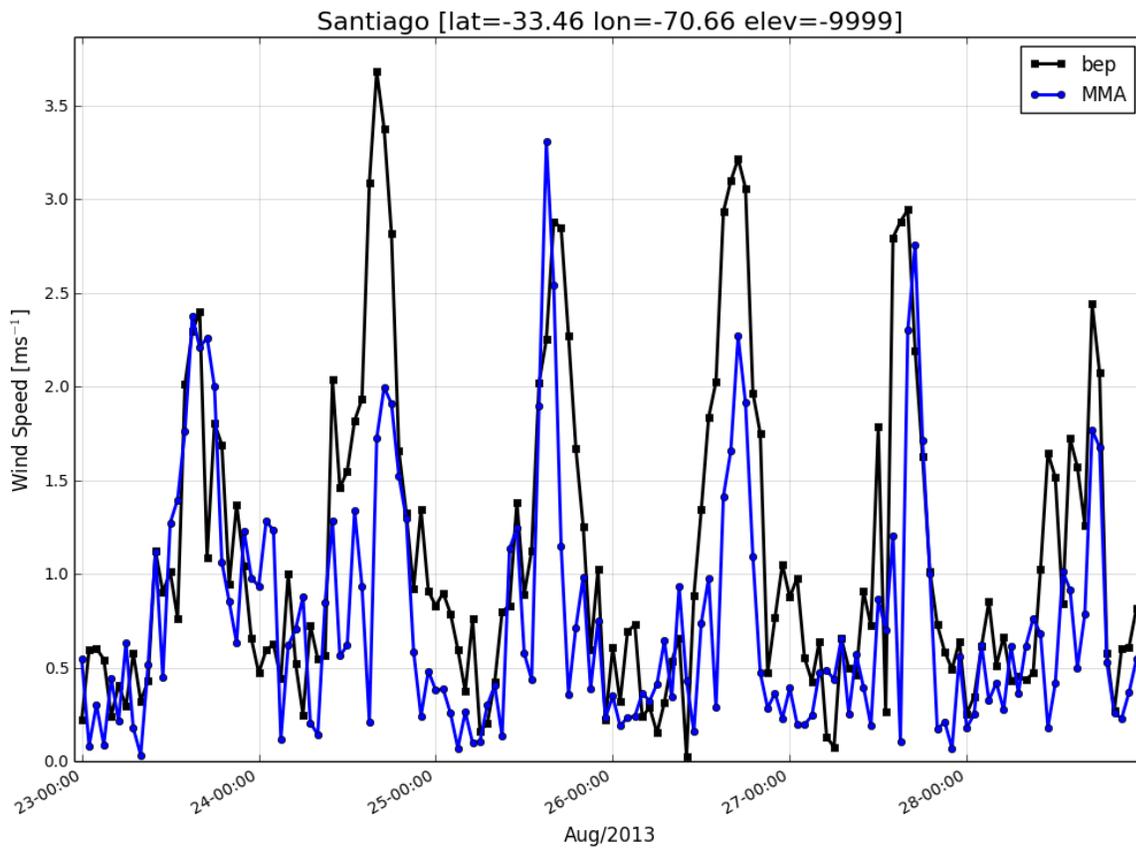


Figura B.35: Serie de tiempo para BEP de variable Magnitud de Viento para estación Santiago en el periodo entre 23-08-2013 y 29-08-2013.

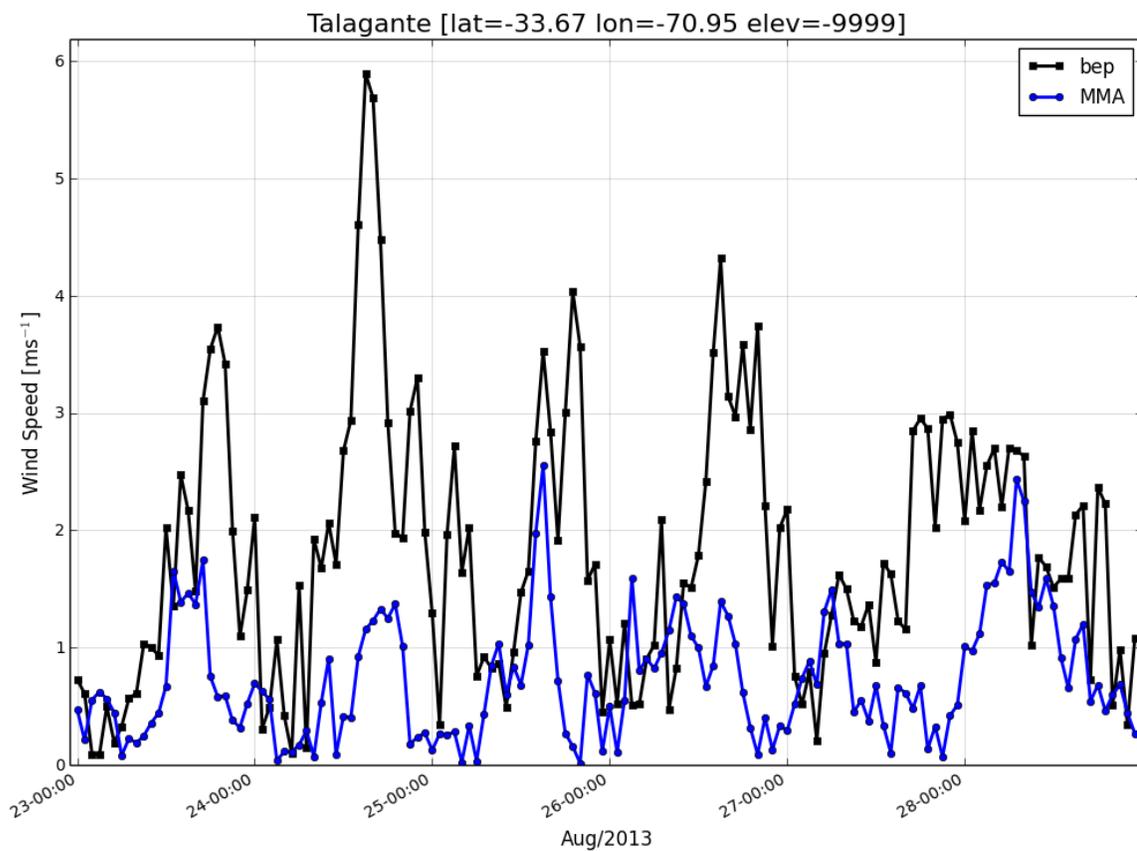


Figura B.36: Serie de tiempo para BEP de variable Magnitud de Viento para estación Talagante en el periodo entre 23-08-2013 y 29-08-2013.

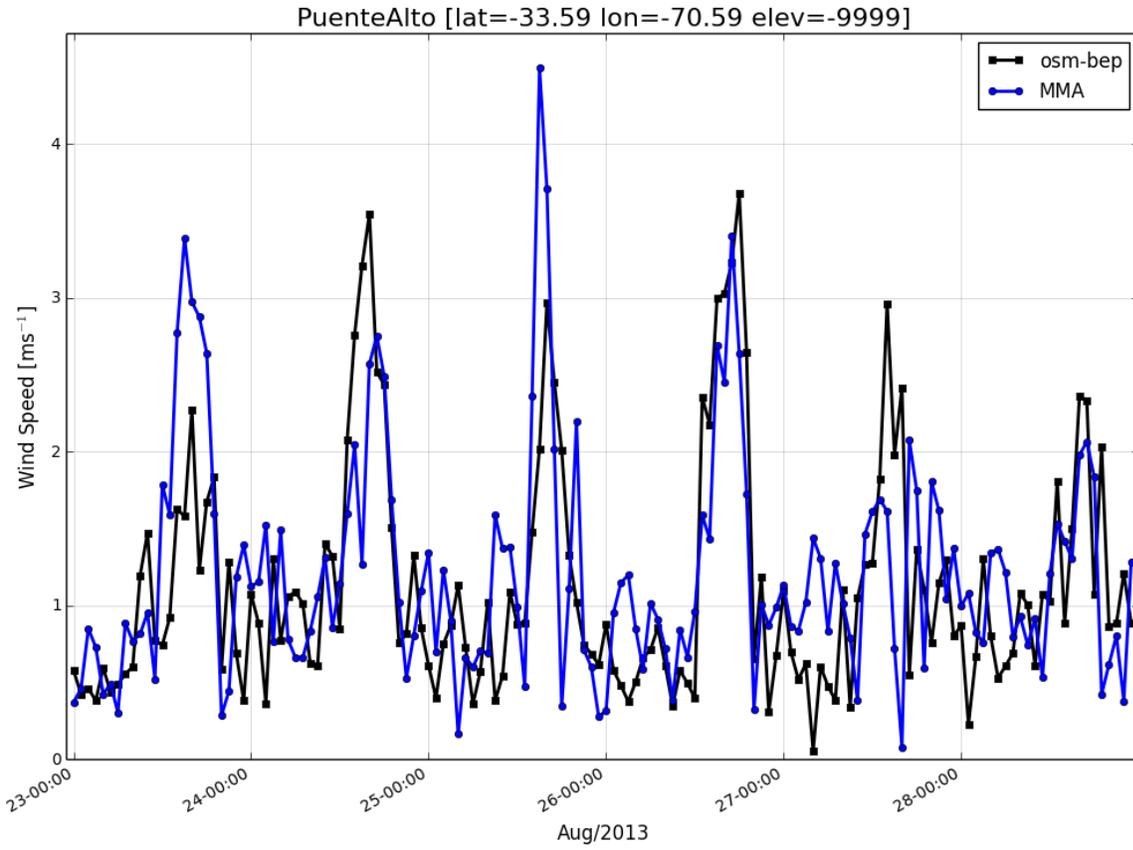


Figura B.37: Serie de tiempo para OSMBEP de variable Magnitud de Viento para estación Puente Alto en el periodo entre 23-08-2013 y 29-08-2013.

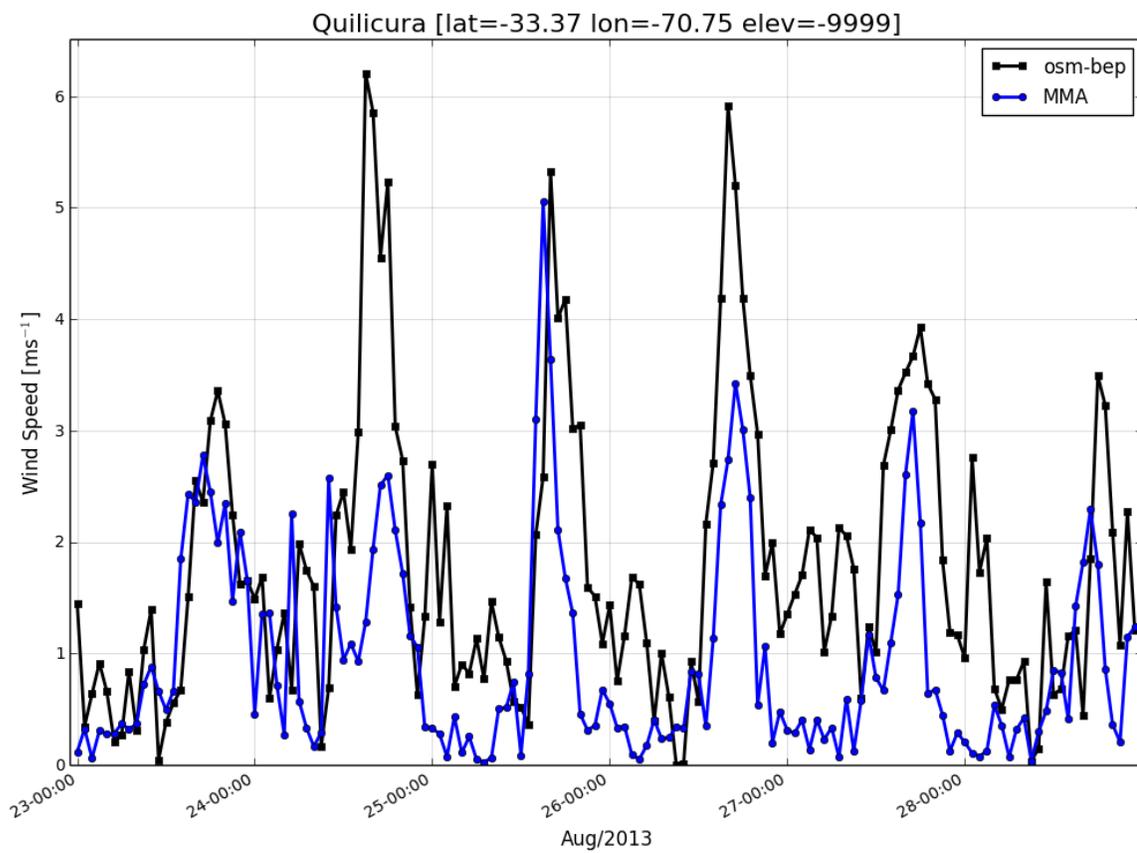


Figura B.38: Serie de tiempo para OSMBEP de variable Magnitud de Viento para estación Quilicura en el periodo entre 23-08-2013 y 29-08-2013.

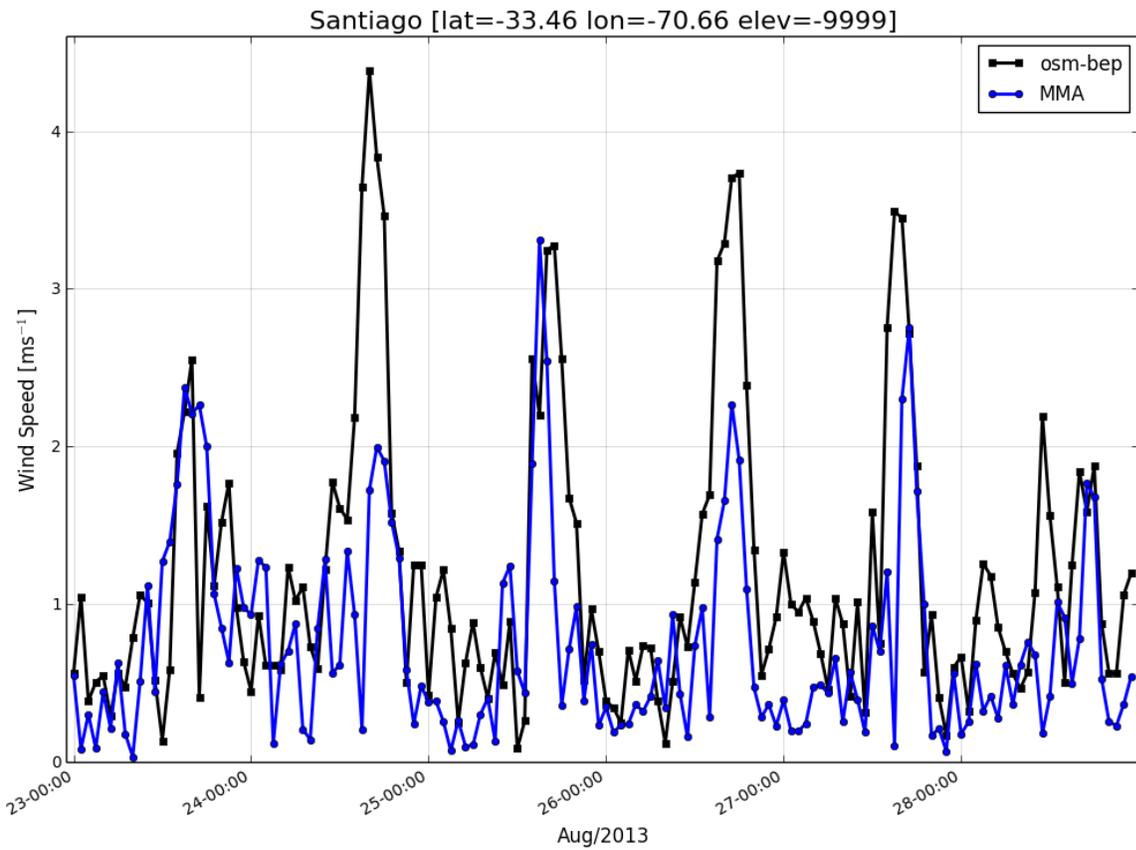


Figura B.39: Serie de tiempo para OSMBEP de variable Magnitud de Viento para estación Santiago en el periodo entre 23-08-2013 y 29-08-2013.

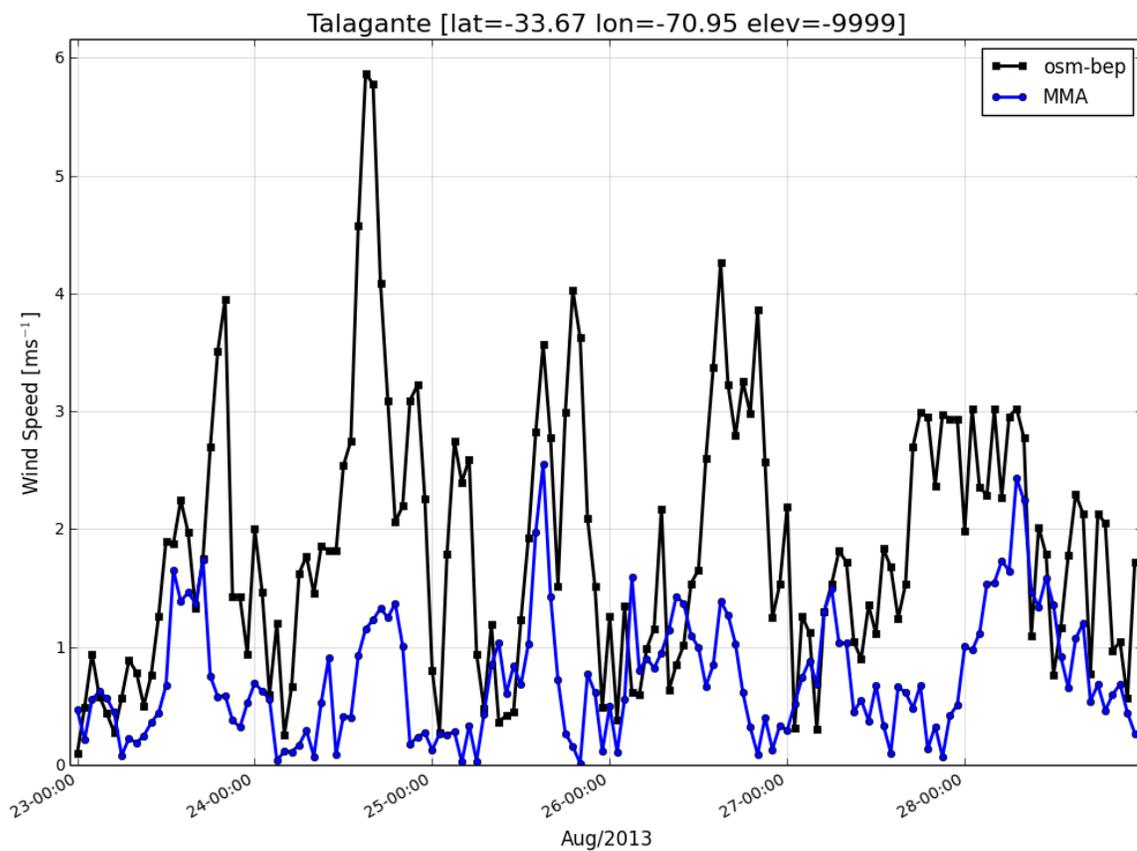


Figura B.40: Serie de tiempo para OSMBEP de variable Magnitud de Viento para estación Talagante en el periodo entre 23-08-2013 y 29-08-2013.