



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

APLICACIÓN DE ALGORITMOS EVOLUTIVOS MULTI-OBJETIVO AL DISEÑO DE EMPAQUETAMIENTOS DE
BATERÍAS

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL ELÉCTRICO

FELIPE GANA ORTEGA

PROFESOR GUÍA:
PABLO ESTÉVEZ VALENCIA

MIEMBROS DE LA COMISIÓN:
HÉCTOR AGUSTO ALEGRÍA
RODRIGO PALMA BEHNKE

SANTIAGO DE CHILE
MARZO 2013

RESUMEN DE LA MEMORIA
PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL ELÉCTRICO
POR: FELIPE GANA ORTEGA
FECHA: MARZO 2013
PROF. GUÍA: PABLO ESTÉVEZ VALENCIA

APLICACIÓN DE ALGORITMOS EVOLUTIVOS MULTI-OBJETIVO AL DISEÑO DE EMPAQUETAMIENTOS DE BATERÍAS

Las baterías de ion-litio son las más usadas actualmente para los vehículos eléctricos y para respaldo en sistemas eléctricos de potencia. Sin embargo, aún presentan altos costos y una baja vida útil con respecto a lo esperado por el mercado y los límites prácticos del material. Esto representa una importante barrera que ha dificultado la masificación de los vehículos eléctricos y los sistemas de respaldo en base a baterías. La temperatura de las celdas es uno de los aspectos más relevantes en la vida útil de un banco de baterías. En esta memoria se desarrolla una aplicación computacional que permite diseñar empaquetamientos de baterías con un comportamiento térmico óptimo, esto es con bajas temperaturas máximas y poca dispersión entre celdas de un mismo banco.

En primer lugar, se aborda la modelación del problema de optimización multi-objetivo de un banco de baterías. Se diseñaron tres funciones objetivo: Temperatura máxima de las celdas de un banco, diferencia máxima de temperatura entre celdas en un banco y volumen. Estas métricas se utilizan para comparar los diferentes diseños de empaquetamientos a ser evaluados en la optimización. Las variables corresponden a la posición y orientación de las celdas de baterías en un módulo o banco. Se implementó un algoritmo evolutivo multi-objetivo para la optimización de bancos de baterías utilizando las funciones objetivo anteriores y que permite comunicarse con un software comercial para evaluar de las funciones objetivo. El algoritmo implementado corresponde a Optimización Multi-objetivo por Enjambre de Partículas (Multi Objective Particle Swarm Optimization, MOPSO). Para la evaluación de las funciones objetivo se realizaron simulaciones en el software comercial COMSOL.

Se desarrollaron tres casos de estudio que muestran las ventajas de utilizar la metodología implementada a través de optimización multi-objetivo y simulaciones térmicas. Los casos de estudio corresponden al posicionamiento óptimo de celdas para un módulo de seis celdas cilíndricas del tipo 18650, un banco de seis módulos de celdas prismáticas donde varía la orientación de los módulos y el banco de baterías del vehículo solar eléctrico Eolian III. En estos tres casos se presentan las mejoras obtenidas en disminución de temperatura máxima y diferencia máxima de temperatura con respecto a otras metodologías de optimización y diseños iniciales. Para el caso de Eolian III, la reducción en temperatura máxima alcanza $8,22^{\circ}\text{C}$ y la reducción en diferencia de temperatura máxima es de $7,61^{\circ}\text{C}$. Esto, al igual que las mejoras en los otros casos de estudio, permitirían aumentar la vida útil y rendimiento de los bancos de baterías. Se estima que un aumento de 15°C en la temperatura de operación de una celda hace caer a la mitad su vida útil.

Agradecimientos

Se agradece el apoyo de los proyectos FONDECYT 1107101, Innova-CORFO 11IDL1-10466 “Perfil de I+D Aplicada: Empaquetamiento Óptimo de Baterías de Litio” y “Empaquetamiento óptimo de baterías de litio para uso en vehículos eléctricos” del Centro de Innovación del Litio en el desarrollo de este trabajo.

Tabla de contenido

1. Introducción	12
1.1. Objetivos	13
1.2. Estructura de la memoria	14
2. Antecedentes	15
2.1. Empaquetamiento de baterías de ion-litio	15
2.1.1. Celdas de baterías de ion-litio	15
2.1.2. Medidas de condición de una batería	17
2.1.3. Modelos eléctricos de baterías de ion-litio	18
2.1.4. Mecanismos de envejecimiento de una celda de batería de ion-litio	22
2.1.5. Generación de calor de una celda de batería	23
2.1.6. Aspectos básicos del empaquetamiento de baterías	24
2.2. Simulación computacional de mecánica de fluidos	26
2.2.1. Modelo fluidodinámico	26
2.2.2. Mecanismos de transferencia de calor	27
2.2.3. Simulación computacional de mecánica de fluidos	28
2.2.4. Uso de simulación computacional en modelamiento térmico de bancos de baterías	29
2.3. Optimización Multi-Objetivo	30
2.3.1. Definición del Problema de Optimización Multi-Objetivo	30
2.3.2. Métodos para Búsqueda de Soluciones a Problemas Multi-Objetivo	31

2.4.	Algoritmos Evolutivos aplicados a Problemas Multi-Objetivo	33
2.4.1.	Aspectos Básicos	33
2.4.2.	Estructura Genérica de un Algoritmo Evolutivo Multi-Objetivo	35
2.4.3.	Diseño de un Algoritmo Evolutivo Multi-Objetivo	35
2.4.4.	Etapas del desarrollo de un algoritmo evolutivo multi-objetivo	40
2.4.5.	Aplicaciones de Algoritmos Evolutivos en Problemas Multi-Objetivo	41
2.4.6.	Aplicaciones de Algoritmos Evolutivos Multi-Objetivo en empaquetamiento de baterías	43
3.	Metodología e Implementación	45
3.1.	Modelación del Problema	45
3.2.	Implementación de Algoritmos Evolutivos	48
3.2.1.	Implementación de MOPSO	48
3.3.	Diagrama de bloques de la implementación	51
3.4.	Modelación térmica de empaquetamientos de baterías en COMSOL Multiphysics .	52
3.5.	Evaluación del modelo térmico para empaquetamientos de baterías	54
3.6.	Datos de las Simulaciones	56
3.6.1.	Celdas de baterías	56
3.6.2.	Estimación de la resistencia interna de las baterías	58
3.6.3.	Velocidad del aire	59
3.6.4.	Generación de Calor	60
3.6.5.	Tiempo de Simulación	63
3.6.6.	Efecto del uso de la perfil US06 en generación de calor	63
4.	Resultados	68
4.1.	Caso 1: Posicionamiento óptimo de seis celdas cilíndricas 18650 en un módulo . .	68
4.1.1.	Descripción del caso de estudio	68
4.1.2.	Determinación de parámetros del algoritmo evolutivo	70
4.1.3.	Resultados	71

4.1.4. Comparación de la metodología	81
4.2. Caso N°2: Seis módulos de celdas prismáticas	87
4.2.1. Descripción del caso de estudio	87
4.2.2. Resultados	89
4.2.3. Comparación de resultados	93
4.3. Caso N°3: Eolian III	95
4.3.1. Descripción del caso de estudio	95
4.3.2. Resultados	97
4.3.3. Mejoras al sistema de ventilación	102
5. Conclusiones	107
5.1. Conclusiones generales	107
5.2. Trabajos Futuros	109
Bibliografía	111
A. Proceso de Fabricación de celdas de baterías de ion-litio	I
A.1. Fabricación de Ánodos y Cátodos	I
A.2. Fabricación de la Celda	I
B. Criterios de Decisión en Problemas Multi-Objetivo	VII
B.1. Criterios A Priori	VII
B.2. Criterios Progresivos	IX
B.3. Criterios A Posteriori	X
C. Imágenes banco de baterías Eolian III	XII
D. Resultados completos caso celdas aleatorias	XV
D.1. Perfil de velocidad	XVII
D.2. Temperatura celdas en función del tiempo	XIX

Índice de figuras

2.1. Partes de una celda de batería de ion-litio.	16
2.2. Difusión de iones de litio y electrones en la carga y descarga de una celda de ion-litio.	17
2.3. Celdas Cilíndricas [1]	17
2.4. Celdas Prismáticas [2]	17
2.5. Modelo de Thevenin para una batería.	19
2.6. Modelo de Randles para una batería.	19
2.7. Modelo RC de SAFT para una batería.	20
2.8. Modelo de doble polarización para una batería.	20
2.9. Modelo de Copetti para una batería.	21
2.10. Pasos del diseño de un banco de baterías.	24
2.11. Elementos típicos de un banco de baterías.	25
2.12. Estructura básica de un algoritmo evolutivo de un objetivo	36
2.13. Estructura básica de un algoritmo evolutivo multi-objetivo.	36
3.1. Esquema de representación del cambio de posición de las celdas dentro de un módulo o un banco de baterías. Las baterías pueden cambiar su posición libremente dentro del mismo plano.	47
3.2. Esquema de representación del cambio de orientación de una celda o un módulo. Las baterías pueden cambiar su orientación en ángulos a favor o en contra de las manillas del reloj.	47
3.3. Etapas del algoritmo MOPSO implementado en MATLAB.	50
3.4. Etapas de la herramienta implementada en este trabajo usando en COMSOL y MATLAB para la optimización multi-objetivo de módulos y bancos de baterías. . .	51

3.5. Esquema de representación de condiciones de borde y fuentes de calor en COMSOL	53
3.6. Malla utilizada para la simulación en tres dimensiones en COMSOL de una celda 18650.	56
3.7. Malla escogida para celdas prismáticas, siguiendo las reglas presentadas en esta sección para celdas cilíndricas.	56
3.8. Respuesta en temperatura para la simulación de una celda 18650, con calor constante $140.000 \frac{W}{m^3}$ y una velocidad de entrada del aire de $0,5 \frac{m}{s}$. Captura en instante final de tiempo. Captura en instante final de tiempo.	56
3.9. Respuesta en velocidad para la simulación de una celda 18650, con calor constante $140.000 \frac{W}{m^3}$ y una velocidad de entrada del aire de $0,5 \frac{m}{s}$.	57
3.10. Celda cilíndrica de litio 18650	57
3.11. Celda prismática de litio SPB636395	58
3.12. Curva de descarga utilizada para la estimación de parámetros de la celda 18650.	60
3.13. Resistencia interna en función del Estado de Carga (SOC) de la batería 18650 de ion-litio.	60
3.14. Perfil de conducción US06 (Velocidad en $\frac{m}{s}$ en función del tiempo)	61
3.15. Perfil de conducción US06 (Aceleración en $\frac{m^2}{s}$ en función del tiempo)	62
3.16. Corriente en función del tiempo para el perfil de conducción US06, tomando como corriente máxima 20 Amperes.	62
3.17. Geometría N°1 utilizada en la prueba para el perfil US06.	63
3.18. Geometría N°2 utilizada en la prueba para el perfil US06.	64
3.19. Resultados de la prueba para diferentes combinaciones de tramos del perfil de uso US06 para la diferencia de temperatura máxima. En rojo se muestran los resultados para la Geometría N°1 y en azul para la Geometría N°2.	66
3.20. Resultados de la prueba para diferentes combinaciones de tramos del perfil de uso US06 para la diferencia de temperatura máxima. En rojo se muestran los resultados para la Geometría N°1 y en azul para la Geometría N°2.	67
4.1. Geometría utilizada para el caso de 6 celdas cilíndricas aleatorias.	69
4.2. Frente de Pareto encontrado tras 200 iteraciones del algoritmo MOPSO.	72
4.3. Frente de Pareto encontrado, proyectado en dos dimensiones para las funciones objetivo temperatura máxima y diferencia de temperatura máxima.	74

4.4. Frente de Pareto encontrado, proyectado en dos dimensiones para las funciones objetivo temperatura máxima y volumen.	74
4.5. Frente de Pareto encontrado, proyectado en dos dimensiones para las funciones objetivo diferencia de temperatura máxima y volumen.	75
4.6. Perfil de temperatura del módulo diseñado en la Solución N°1 en el Frente de Pareto. En esta figura se observa la grilla de dos filas y tres columnas encontrada por el algoritmo evolutivo.	79
4.7. Perfil de velocidad del aire de la Solución N°1 del Frente de Pareto. En esta figura se observa la grilla de dos filas y tres columnas encontrada por el algoritmo evolutivo.	79
4.8. Perfil de temperatura del diseño N°18 en el Frente de Pareto. Esta es la que obtiene la menor temperatura máxima de todas las configuraciones exploradas, con 26,441°C. En los perfiles de temperatura, los colores van del azul al rojo y significan mayor temperatura.	80
4.9. Perfil de velocidad del aire de la Solución N°18 del Frente de Pareto.	80
4.10. Perfil de temperatura del diseño N°21 en el Frente de Pareto. Esta es la solución que obtiene la menor diferencia de temperatura máxima de todas las configuraciones exploradas, con 0,627°C.	81
4.11. Perfil de velocidad del aire de la Solución N°21 del Frente de Pareto. Se observan los <i>jets</i> de alta velocidad de paso del aire en torno a las celdas, con formas similares, que permiten que la diferencia de temperatura sea baja.	81
4.12. Diseño resultante tras operar PSO con un objetivo, minimización de la temperatura máxima.	82
4.13. Distribución de las soluciones en la última generación del algoritmo PSO con un objetivo, minimización de temperatura máxima.	83
4.14. Distribución de las soluciones en la última generación del algoritmo MOPSO con tres objetivos: Minimización de temperatura máxima, volumen y diferencia de temperatura máxima.	83
4.15. Diseño resultante tras operar PSO con un objetivo, minimización del volumen	84
4.16. Diseño resultante tras operar PSO con un objetivo, minimización de la diferencia de temperatura.	85
4.17. Caso base, configuración de grilla con separación de 27 milímetros.	86
4.18. Caso base, configuración de grilla con separación de 36 milímetros en la horizontal y 27 milímetros en la vertical.	86
4.19. Caso de ejemplo, configuración de grilla escalonada con separación de 18 milímetros en la horizontal y 27 milímetros en la vertical.	87

4.20. Geometría utilizada para el caso de estudio de seis módulos. La presión en Pascal aparece junto a la entrada y las salidas de aire.	88
4.21. Frente de Pareto encontrado para el Caso de Estudio N°2 después de 100 generaciones del algoritmo MOPSO	90
4.22. Perfil de velocidad del aire del diseño N°13 en el Frente de Pareto.	93
4.23. Perfil de temperatura de la Solución N°13 del Frente de Pareto.	93
4.24. Evolución del mínimo de la temperatura máxima a lo largo de las generaciones para el caso de estudio N°2.	94
4.25. Evolución del mínimo de la diferencia de temperatura máxima a lo largo de las generaciones para el caso de estudio N°2.	95
4.26. Perfil de velocidad del aire para la simulación del caso base con ángulo de rotación de cero grados para todos los módulos.	95
4.27. Geometría utilizada para la optimización del banco de baterías del vehículo solar Eolian III. Se muestran las variables que se utilizarán en la optimización con los nombres d1, d2, d3, d4 y d5 además de las entradas y salidas de aire.	97
4.28. Caso inicial del banco de baterías de Eolian III.	98
4.29. Perfil de temperatura del diseño inicial del banco de baterías de Eolian III.	98
4.30. Perfil de velocidad del diseño inicial del banco de baterías de Eolian III.	99
4.31. Geometría del diseño óptimo del banco de baterías de Eolian III encontrado por el algoritmo.	100
4.32. Perfil de temperatura del diseño óptimo del banco de baterías de Eolian III encontrado por el algoritmo.	100
4.33. Perfil de velocidad del diseño óptimo del banco de baterías de Eolian III.	101
4.34. Evolución del mínimo de la temperatura máxima para el banco de baterías de Eolian III.	101
4.35. Evolución del mínimo de la diferencia de temperatura máxima para el banco de baterías de Eolian III.	102
4.36. Geometría utilizada para la optimización del banco de baterías del vehículo solar Eolian III. Se muestran las variables que se utilizarán en la optimización con los nombres d1, d2, d3, d4, d5 y d6, además de las entradas y salidas de aire.. . . .	103
4.37. Geometría del diseño óptimo del banco de baterías de Eolian III, considerando el cambio de posición de entrada del aire, encontrado por el algoritmo evolutivo. . . .	105

4.38. Perfil de temperatura del diseño óptimo del banco de baterías de Eolian III con cambio en la entrada de aire.	105
4.39. Perfil de velocidad del diseño óptimo del banco de baterías de Eolian III con cambio en la entrada del aire.	106
A.1. Proceso de fabricación de las tiras de ánodo.	III
A.2. Proceso de fabricación de las tiras de cátodo.	IV
A.3. Proceso de fabricación de las celdas prismáticas.	V
A.4. Proceso de fabricación de las celdas cilíndricas.	VI
C.1. Módulo de baterías Eolian III.	XII
C.2. Módulo de baterías Eolian III.	XIII
C.3. Banco de baterías Eolian III.	XIII
C.4. Banco de baterías Eolian III.	XIV

Índice de tablas

2.1. Fenómenos que generan disminución de la potencia y capacidad, reduciendo la vida útil de las baterías de ion-litio. [3]	22
2.2. Técnicas más utilizadas en Algoritmos Evolutivos Multi-Objetivo	34
3.1. Tamaños y tipos de mallas evaluados para las simulaciones en COMSOL	54
3.2. Mallas evaluadas para las simulaciones en COMSOL	55
3.3. Características de la celda 18650 utilizada en este trabajo.	57
3.4. Características celda SPB636395	58
3.5. Valores estimados de los parámetros del Modelo de Copetti para la celda 18650. . .	59
3.6. Valores estimados de los parámetros del Modelo de Copetti para la celda SPB636395	59
3.7. Orden de los tramos del perfil US06 utilizados en las 10 pruebas realizadas para determinar posible sesgo en los resultados de la optimización, utilizando esta curva. Los tramos son numerados como 1 para 0 a 100 segundos, 2 entre 100 y 200 segundos y así sucesivamente.	65
3.8. Resultados de la prueba para diferentes combinaciones de tramos del perfil de uso US06 para la temperatura máxima en las celdas. Nueve de los diez casos presentan el mismo resultado, favoreciendo a la Geometría N°2.	65
3.9. Resultados de la prueba para diferentes combinaciones de tramos del perfil de uso US06 para la diferencia de temperatura máxima. Ocho de los diez casos presentan la misma información, favoreciendo a la Geometría N°2.	66
4.1. Resultados obtenidos después de 10 generaciones del algoritmo MOPSO para el problema de seis celdas en posiciones aleatorias con diferentes tamaños de población. La tabla incluye el tiempo que tomó en promedio cada generación y la desviación estándar para las variables que indican la posición horizontal y vertical de las celdas.	70

4.2.	Resultados obtenidos después de 10 generaciones del algoritmo MOPSO para el problema de seis celdas en posiciones aleatorias con diferentes tamaños de población. La tabla incluye los rangos cubiertos en el Frente de Pareto y la cantidad de elementos en este. Un mayor rango permite un frente de Pareto más diverso.	71
4.3.	Soluciones óptimas obtenidas para el caso de 6 celdas 18650 en ubicaciones aleatorias. Se destacan los mínimos para cada función objetivo.	73
4.4.	Soluciones óptimas obtenidas para el caso de 6 celdas 18650 en ubicaciones aleatorias.	75
4.5.	Frente de Pareto encontrado para el Caso de Estudio N°2 después de 100 generaciones del algoritmo MOPSO.	89
4.6.	Soluciones óptimas obtenidas para el caso de orientación de 6 módulos de celdas prismáticas. Se muestra el perfil de velocidad del aire para cada solución.	90
4.7.	Valores de los ángulos para cada solución en el Frente de Pareto.	94
4.8.	Valores máximos y mínimos para las variables del problema de diseño del banco de baterías del vehículo solar Eolian III.	97
4.9.	Valores de las variables del diseño del banco de baterías de Eolian III para el caso base entregado por el equipo para ser mejorado.	98
4.10.	Valores de las variables para el caso optimizado del banco de baterías de Eolian III.	99
4.11.	Valores máximos y mínimos para las variables del problema de diseño del banco de baterías del vehículo solar Eolian III considerando la opción de modificar la posición de la entrada de aire.	103
4.12.	Valores de las variables para el caso optimizado del banco de baterías de Eolian III sin cambiar la entrada de aire. Este diseño es parte de la población inicial de la optimización considerando la posibilidad de cambios en la entrada de aire.	103
4.13.	Valores de funciones objetivo de las soluciones que se encuentran en el Frente de Pareto para el caso de la optimización del banco de baterías del Eolian III, considerando la opción de cambiar la posición de la entrada de aire.	104
4.14.	Valores de las variables de los diseños encontrados en el Frente de Pareto para el caso de la optimización del banco del Eolian III con cambio en la posición de la entrada de aire.	104
D.1.	Valores de la posición de las celdas 1, 2, 3, 4, 5 y 6 en la horizontal.	XV
D.2.	Valores de la posición de las celdas 1, 2, 3, 4, 5 y 6 en la vertical.	XVI
D.3.	Soluciones óptimas obtenidas para el caso de 6 celdas 18650 en ubicaciones aleatorias.	XVII

D.4. Soluciones óptimas obtenidas para el caso de 6 celdas 18650 en ubicaciones aleatorias. XIX

Capítulo 1

Introducción

El trabajo que se presenta en esta memoria tiene como eje central la mejora del comportamiento térmico de bancos de baterías de ion-litio, con la búsqueda de mejores diseños a través de optimización.

Las baterías de ion-litio son las más utilizadas actualmente para los vehículos eléctricos y los BESS (Battery Energy Storage System) para respaldo en sistemas eléctricos de potencia. Si bien las baterías de ion-litio son eficientes para resolver el problema de almacenamiento de energía debido a la alta densidad energética que logran en comparación con baterías en base a otros compuestos como plomo-acido y níquel-cadmio, los bancos de baterías de ion-litio aún presentan altos costos, baja vida útil y baja densidad energética, con respecto a sus límites teóricos. Esto representa una importante barrera que ha dificultado la masificación de los vehículos híbridos, eléctricos y los BESS para respaldo. Sin embargo, cabe mencionar que esta industria ha crecido considerablemente en los últimos años.

Aunque el alto costo de las baterías de ion-litio depende en gran parte del precio de los materiales y de la fabricación de las celdas que lo componen, el problema no radica simplemente en la adquisición y desarrollo de mejores y más eficientes celdas de baterías, sino también en optimizar su integración en módulos y bancos. Uno de los aspectos sobre el cual influye mayormente la integración de las celdas es el comportamiento térmico, esto es, la temperatura de operación del banco ante diferentes condiciones de uso. La temperatura tiene un directo efecto sobre el desempeño, vida útil y seguridad del banco de baterías. Si la temperatura de las celdas se encuentra lejos de su punto de operación óptimo o normal, se produce pérdida de capacidad y degradación de la celda. En función de lo anterior, el integrar las celdas en un banco de baterías es un problema relevante ya que ni aún las mejores celdas y formas de empaquetamiento se comportarán de la mejor manera debido a las condiciones en que funcionan los vehículos eléctricos [4] ((velocidad, aceleraciones, temperatura exterior, entre otras). Por lo tanto, una mala integración de celdas, producirá efectos que terminan en aumentar el costo de los bancos de baterías, ya sea por sobredimensionamiento para cumplir con las restricciones de vida útil o por la mayor probabilidad de fallas que no permiten que el equipo cumpla con su vida útil.

Para enfrentar los costos de las baterías de ion-litio en la industria se han tomado dos caminos.

La primera alternativa consiste en disminuir la cantidad de horas de ingeniería necesarias para diseñar un sistema de baterías a través de la creación de mejores herramientas, las que permiten encontrar diseños más robustos, de mejor comportamiento y más ajustados a los parámetros entregados por los fabricantes. La segunda alternativa, ha sido analizar con más detalle la posibilidad de darle una segunda vida a las baterías de ion-litio, de modo de reducir el costo para el usuario final (comprador del vehículo o BESS) ya que se podrá re-vender el pack de baterías cuando este ya no sea apropiado para ser utilizado con sus requerimientos.

El trabajo que se presenta en esta memoria se enmarca en la primera de las ramas: En la actualidad no existen herramientas en el mercado que aporten al diseño óptimo de bancos de baterías de forma automatizada y que ofrezcan varias alternativas de diseño para la etapa del empaquetamiento de baterías considerando su comportamiento térmico. Lo anterior implica que el desarrollo de una herramienta que aporte a resolver este problema serían un aporte importante para hacer que la tecnología de las baterías de ion-litio sea más accesible y duradera para la industria, tanto de los vehículos eléctricos como de los sistemas eléctricos de potencia.

El problema del empaquetamiento de baterías es complejo, ya que involucra una gran cantidad de variables, alternativas y existen muchas medidas y objetivos que deben cubrirse en un diseño. Es por esta razón que el diseño y desarrollo de un banco en la actualidad toma varios meses y requiere de un trabajo de ingeniería inter-disciplinario. Una forma de representar el problema de empaquetamiento para encontrar soluciones adecuadas es a través de la modelación multi-objetivo. De este modo pueden representarse las diferentes características que se desean para un diseño de un banco de baterías, que no necesariamente son mejoradas en la misma dirección y sobre las cuales no existe una claridad evidente de cual es más importante. Una de las técnicas con mejores resultados para resolver problemas multi-objetivo es el uso de algoritmos evolutivos, por lo que la propuesta del trabajo de título es realizar una aplicación que utilice esta técnica para resolver el problema de empaquetamiento de baterías de ión-litio desde el punto de vista térmico. La obtención de los valores de las funciones objetivo se realiza a través de la modelación en un software de simulación computacional que permite representar los fenómenos de generación de calor en las celdas de baterías.

1.1. Objetivos

El objetivo general de la memoria consiste en la aplicación de algoritmos evolutivos al problema de optimización multi-objetivo de empaquetamiento de baterías de ion-litio.

Los objetivos específicos del trabajo son los siguientes:

- Conocer el estado del arte de los algoritmos de optimización multi-objetivo basadas en algoritmos evolutivos, en particular aplicados a problemas de diseño similares al de las baterías.
- Modelar el problema de optimización de empaquetamiento de bancos de baterías de ion-litio como un problema multi-objetivo.

- Disponer de un esquema de simulación en un software comercial. Esto con el fin de representar el comportamiento térmico de celdas y bancos de baterías para obtener los valores de las funciones objetivo del problema planteado.
- Aplicar algoritmos evolutivos multi-objetivo para el problema de optimización de empaquetamiento de bancos de baterías de ion-litio a tres casos de estudio.

1.2. Estructura de la memoria

La estructura utilizada en este documento para exponer el trabajo realizado es la siguiente:

- **Capítulo 1. Introducción:** Corresponde a la descripción del tema, la motivación de éste y los objetivos del trabajo.
- **Capítulo 2. Antecedentes:** En este capítulo se explican los conceptos necesarios para la contextualización del trabajo. Se detallan los principales aspectos y estado del arte en baterías de ion-litio, empaquetamiento de baterías y algoritmos evolutivos multi-objetivo. Además se presentan trabajos relacionados, en empaquetamiento de baterías y en diseño con algoritmos multi-objetivo.
- **Capítulo 3. Metodología e Implementación:** En este capítulo se entregan los detalles y justificaciones de los métodos y supuestos utilizados en este trabajo. Se describen las características de las simulaciones computacionales utilizadas, sus parámetros y definiciones además de la configuración del algoritmo evolutivo utilizado para la optimización.
- **Capítulo 4. Resultados:** En este capítulo se detallan las características específicas y los resultados obtenidos en los tres casos de estudio en los cuales se implementó la metodología descrita en el capítulo 3. Los casos de estudio corresponden al posicionamiento óptimo de seis celdas cilíndricas, a la orientación óptima de seis módulos de baterías prismáticas y a la mejora del banco de baterías del vehículo eléctrico Eolian.
- **Capítulo 5. Conclusiones:** En este capítulo se señalan las conclusiones del trabajo y se describe el posible trabajo futuro de acuerdo a lo investigado y encontrado en este trabajo.

Capítulo 2

Antecedentes

2.1. Empaquetamiento de baterías de ion-litio

2.1.1. Celdas de baterías de ion-litio

Un banco de baterías corresponde al elemento que se utiliza para almacenar energía en aplicaciones para vehículos eléctricos y para sistemas eléctricos de potencia (BESS, Battery Energy Storage System). Un banco se compone de módulos de baterías, que corresponden a arreglos de celdas de baterías, los elementos esenciales de éstas. Las celdas tienen características eléctricas de voltaje, corriente y capacidad que vienen determinadas principalmente por los balances de las reacciones electroquímicas que se producen dentro de ellas.

Existen celdas primarias y secundarias. Las primarias son aquellas que no son recargables, en tanto que las secundarias sí pueden ser recargadas luego de entregar parte de su energía.

Las celdas secundarias de ion-litio se conocen entonces como acumuladores reversibles, elementos capaces de entregar energía eléctrica a un circuito exterior gracias a la energía química que contienen las sustancias activas en su interior como también acumular energía química a partir de energía eléctrica suministrada desde un circuito externo [5].

Las celdas de ion-litio contienen un ánodo, un cátodo, un separador y un electrolito que permiten entregar corriente como resultado de una reacción electroquímica [5]. Un esquema de las partes de una celda de ion-litio colocado en una dimensión se puede observar en la figura 2.1. El separador es una membrana porosa que evita el contacto físico entre los electrodos. En el caso de las celdas de ion-litio, el ánodo normalmente está compuesto por carburo de litio (Li_4C) y el cátodo, se construye a partir de un óxido de litio metálico. Una sal de litio se utiliza como electrolito para entregar los iones necesarios para la reacción química. El electrolito actúa únicamente como medio conductor, a través del cual los iones de litio pueden fluir y no toma parte de la reacción química [6].

En una descarga, los iones de litio (Li^+) se separan del ánodo y migran a través del electrolito

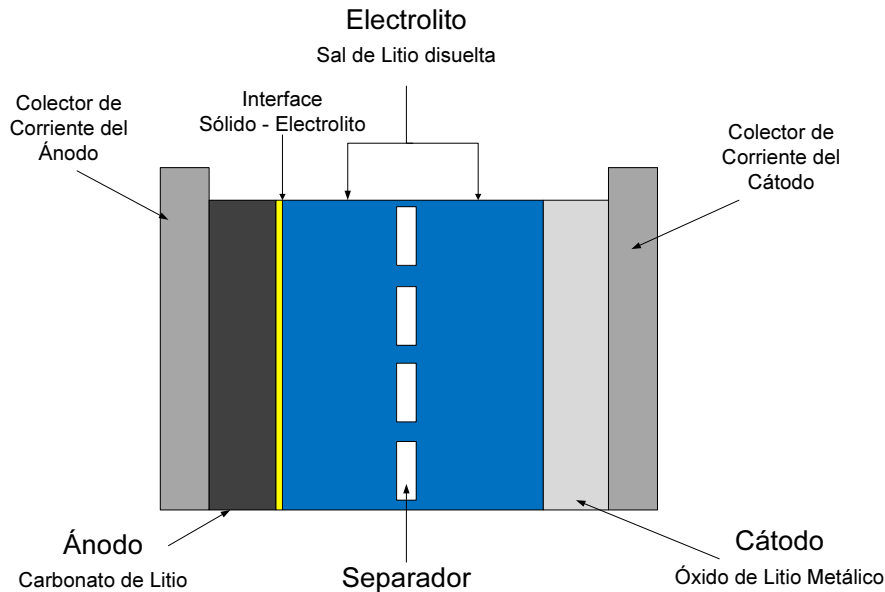


Figura 2.1: Partes de una celda de batería de ion-litio.

hasta la estructura del cátodo. Los electrones viajan a través del circuito externo en la dirección contraria. El proceso es reversible y la situación que se da para la descarga se da en la dirección inversa para la carga del acumulador. El proceso de carga y descarga descrito anteriormente se muestra en la figura 2.2.

Las baterías de ion-litio son el sistema de baterías más utilizado para los vehículos eléctricos (EV) y los BESS (Battery Energy Storage System) para respaldo en sistemas eléctricos de potencia. El litio tiene alta densidad energética lo que permite que sea un material viable para su incorporación a las baterías para aplicaciones ya sea en movilidad eléctrica como en soporte en BESS [4]. Otras ventajas de las celdas de ion-litio son su baja tasa de auto-descarga, su alta eficiencia y el hecho de que el voltaje en la celda se mantiene casi constante durante su descarga [7].

Las celdas de ion-litio pueden encontrarse en diferentes configuraciones geométricas y de materiales que se utilizan para ánodos, cátodos y separadores. La familia de las celdas de ion-litio generalmente se divide en tres grupos, dependiendo del material utilizado como cátodo: litio-ion cobalto, litio-ion-manganeso y litio-ion-fosfato. Se considera como otro grupo las celdas de litio-ion polímero, que tienen diferencias en la forma del electrolito utilizado [7].

Los dos tipos de geometría más comunes en las celdas de litio son las prismáticas, o rectangulares (similares a la que se muestra en la Figura 2.4), y las cilíndricas (ver Figura 2.3). Las configuraciones químicas y constructivas determinan los diferentes valores de resistencia interna, densidad energética y densidad de potencia para cada tipo de celda de ion-litio.

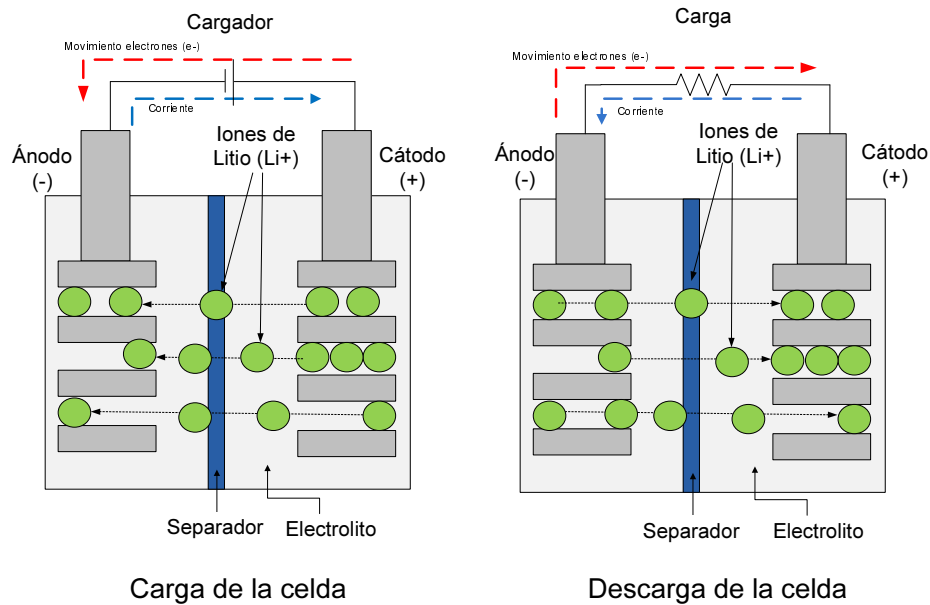


Figura 2.2: Difusión de iones de litio y electrones en la carga y descarga de una celda de ion-litio.



Figura 2.3: Celdas Cilíndricas [1]



Figura 2.4: Celdas Prismáticas [2]

2.1.2. Medidas de condición de una batería

Existen medidas que definen el estado y características de una celda o un banco de baterías. Estas medidas y terminologías permiten clasificar, describir y comparar diferentes tipos de baterías, además de entregar a los fabricantes una forma de caracterizar los valores nominales de sus productos.

- **Estado de Carga (State of Charge, SOC):** Corresponde a una expresión porcentual de la capacidad actual de la batería con respecto a su capacidad máxima. Un SOC de 100 % (o 1) representa a una batería completamente cargada en tanto que 0 % (o 0) representa a una batería completamente descargada. El SOC es calculado a través de la integral de la corriente en el tiempo, lo que permite determinar el cambio de la capacidad de la batería, como se puede observar en la ecuación 2.1, donde $C(t)$ representa la capacidad, $I(w)$ a la corriente y $\eta(w)$ a la eficiencia de la descarga o carga.

$$(2.1) \quad SoC(t) = SoC(0) + \frac{1}{C(t)} \int_0^t \eta(w) \cdot I(w) \cdot dw$$

- **Profundidad de Descarga (Depth of Discharge, DOD):** Corresponde al porcentaje de la capacidad de la batería que se ha descargado, expresado como porcentaje de la capacidad máxima.
- **Resistencia Interna:** Corresponde a la resistencia a través de la batería. Depende de la conductividad del electrolito de la batería, que a su vez depende de la concentración de iones, de la movilidad de los iones, la temperatura y del estado de carga del acumulador. La resistencia interna permite medir la eficiencia de la batería, ya que un aumento de esta tiene relación con el desprendimiento del material activo, lo que al mismo tiempo disminuye la capacidad del acumulador y hace que disipe más energía como calor. Por lo anterior, la resistencia interna es una buena medida del envejecimiento de la batería [5].

Por otra parte, las especificaciones de una batería y que son entregadas por los fabricantes corresponden al voltaje nominal, el voltaje de corte (en Volt), la capacidad nominal (en Ah), energía nominal (en Wh), energía específica (energía nominal por unidad de masa, en Wh/kg), densidad de energía (energía por unidad de volumen en Wh/L), potencia específica (potencia máxima por unidad de masa en W/kg) y densidad de potencia (en W/L).

2.1.3. Modelos eléctricos de baterías de ion-litio

En la literatura se han desarrollado una serie de modelos para poder representar el comportamiento eléctrico de las baterías de ion-litio (y de otras químicas como níquel-cadmio y plomo-ácido). La batería de ion-litio es un sistema no lineal y si bien puede ser representada casi completamente por modelos basados en la teoría electro-química, éstos son complejos y computacionalmente costosos, por lo que generalmente se utilizan abstracciones para conseguir circuitos equivalentes de la celda, que permitan representar los comportamientos dinámicos y principios de operación de la batería en base a resistencia, capacitores, inductancias y fuentes de voltaje. Normalmente, un capacitor de alta capacidad o una fuente ideal de voltaje se utiliza para modelar el voltaje de circuito abierto, en tanto que el resto del circuito modela la impedancia interna de la batería [8].

Modelo de Thévenin

El modelo de Thévenin tiene una fuente ideal de voltaje V_{oc} , una resistencia interna R_i , una capacitancia C_0 y una resistencia de sobrevoltaje R_0 . En la figura 2.5 se muestra el circuito descrito. La fuente V_{oc} modela el voltaje de circuito abierto, la resistencia R_0 modela la resistencia de contacto de las placas del electrolito en tanto que la resistencia interna modela las pérdidas en el electrolito y los circuitos conductores. En el modelo convencional de Thévenin los parámetros son fijos y se determinan por pruebas experimentales de carga y descarga.

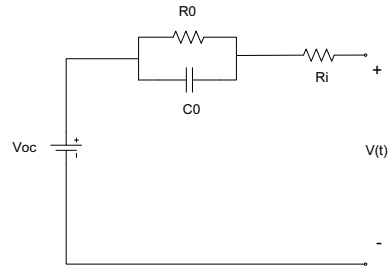


Figura 2.5: Modelo de Thevenin para una batería.

Modelo de Randles

Corresponde a una modificación del modelo de Thevenin, en la cual la fuente ideal que representa al voltaje de circuito abierto [8]. Es modelado con un circuito de una capacitancia C_1 y una resistencia R_1 en paralelo. La resistencia R_1 modela la auto descarga de la batería. El equivalente se observa en la figura 2.6. Al igual que en el Modelo de Thevenin los parámetros son fijos y se determinan por pruebas experimentales de carga y descarga.

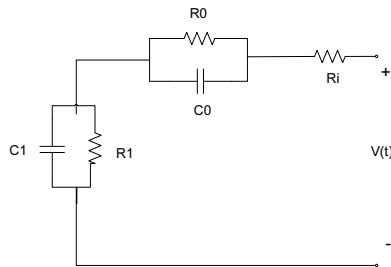


Figura 2.6: Modelo de Randles para una batería.

Modelo RC

El modelo RC fue desarrollado por la compañía de baterías SAFT [8]. Consiste en dos capacitores y tres resistencias. El capacitor C_c representa los efectos de la superficie de la batería, en tanto que el capacitor C_b , de gran magnitud, representa la capacidad de almacenar energía de la batería. Los resistores R_t , R_e y R_c representan la resistencia del terminal, del borne y del capacitor respectivamente. El circuito equivalente se observa en la figura 2.7.

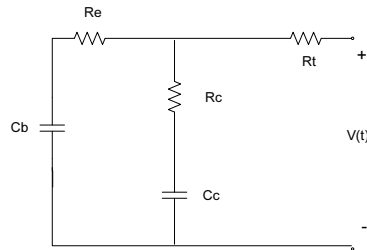


Figura 2.7: Modelo RC de SAFT para una batería.

Modelo de doble polarización

Este modelo se basa en el modelo de Thevenin y es similar al equivalente de Randles, pero incorpora una fuente de voltaje para representar los efectos de la polarización por concentración y por reacciones electroquímicas que se producen dentro de la batería [8]. El voltaje de circuito abierto está representado por U_{oc} , las resistencias R_{pa} y R_{pc} representan la resistencia de polarización electroquímica y de concentración, respectivamente, lo mismo que C_{pa} y C_{pc} . La resistencia R_0 representa la resistencia interna ohmica de la celda. El circuito equivalente se ilustra en la figura 2.8.

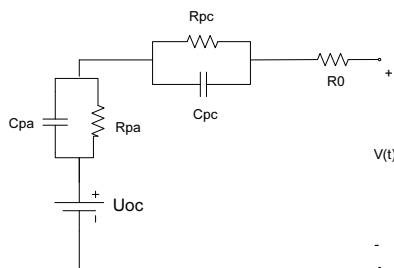


Figura 2.8: Modelo de doble polarización para una batería.

Modelo de Copetti

En [9] y [10] se modela a la batería como un circuito que contiene una fuente de voltaje $V_{oc}(t)$, representando la tensión en circuito abierto y una resistencia variable en serie $R_i(t)$. La resistencia variable modela la oposición al flujo de electrones dentro de la batería. Ambos elementos del circuito son funciones no lineales dependientes del tiempo. En el caso de la resistencia interna, esta depende de la temperatura de la celda, el estado de carga (SoC), el estado de salud (SoH) y la corriente eléctrica del circuito. El voltaje de circuito abierto depende exclusivamente del SoC. El circuito equivalente, también conocido como Modelo de Rint se muestra en la figura 2.9:

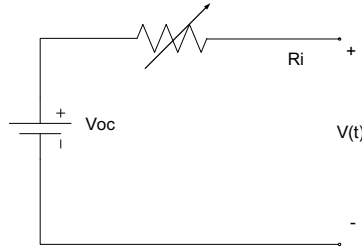


Figura 2.9: Modelo de Copetti para una batería.

Para la descarga, en [9] se presentan las siguientes definiciones, con $\eta=1$ para todo instante de tiempo:

$$(2.2) \quad C(t) = \frac{C_{nom} \cdot C_c}{1 + A_c} \cdot \left(\frac{I(t)}{I_{nom}}\right)^{B_c} \cdot (1 + q_{1c} \cdot \Delta T(t) + q_{2c} \cdot \Delta T(t)^2)$$

$$(2.3) \quad V(t) = (V_{0d} - K_{0d} \cdot (1 - SoC(t))) - \frac{|I(t)|}{C_{nom}} \cdot \left(\frac{P_{1d}}{1 + |I(t)|^{P_{2d}}} + \frac{P_{3d}}{SoC(t)^{P_{4d}}} + P_{5d}\right) \cdot (1 - q_d \cdot \Delta T(t))$$

$$(2.4) \quad V_{oc}(t) = V_{0d} - K_{0d} \cdot (1 - SoC(t))$$

$$(2.5) \quad R_i = \frac{1}{C_{nom}}$$

Para estimar los valores de los parámetros K_{0d} , P_{1d} , P_{2d} , P_{3d} , P_{4d} y P_{5d} se utilizan curvas de descarga de la batería.

El modelo de Copetti también tiene ecuaciones para la tensión en bornes para la carga, sobrecarga y zona de transición entre carga y descarga de las baterías, las cuales se presentan en [9].

2.1.4. Mecanismos de envejecimiento de una celda de batería de ion-litio

Dados los altos costos de las baterías de ion-litio, existen altas exigencias sobre la vida útil de estas en su utilización para vehículos eléctricos (EV) o híbridos (HEV) y sistemas eléctricos de potencia. La vida útil se entiende como el período de tiempo en que la batería opera correctamente en la función para la cual fue creada. Como referencia, la organización estadounidense United States Advanced Battery Council (USABC) a través de su proyecto FreedomCAR [11], definió en 2002, los objetivos de vida útil para las baterías desarrolladas para un EV o un HEV. En cuanto a los EV, se espera una duración de 10 años o al menos 1000 ciclos alcanzando un 80 % de profundidad de descarga. Esta exigencia ha hecho que uno de los aspectos que se ha analizado con mayor profundidad en la investigación de celdas de ion-litio sea su envejecimiento. La complejidad de las celdas como sistema hace que existan muchos procesos y factores involucrados en la degradación de las celdas, que puede ser observada a través de una caída en capacidad y potencia.

En [3] se hace una completa revisión de las causas del envejecimiento de la celda, para las diferentes partes de esta, ánodo, cátodo y electrolito. Los efectos descritos en [3] se detallan en la tabla 2.1.

Tabla 2.1: Fenómenos que generan disminución de la potencia y capacidad, reduciendo la vida útil de las baterías de ion-litio. [3]

Fenómeno	Causa
Descomposición de Electrolito, crecimiento de interface con ánodo	Altas temperaturas y alto SOC
Intercalamiento de solvente en el ánodo	Sobrecarga
Disminución de superficie accesible en ánodo	Altas temperaturas y alto SOC
Cambios en la porosidad del ánodo	Alta tasa de ciclado y alto SOC
Pérdida de material activo por cambio de volumen	Alta tasa de ciclado y alta profundidad de descarga
Descomposición de uniones	Altas temperaturas y alto SOC
Corrosión del colector del ánodo	Sobredescarga o bajo SOC
Descomposición del electrólito por litio metálico	Baja temperatura, alta tasa de ciclado, desbalance de celda o fallas constructivas
Disolución de material activo del cátodo en el electrolito	Altas temperaturas y Bajo SOC
Descomposición de material activo del cátodo en el electrolito	Bajo SOC
Crecimiento de interface cátodo - electrólito	Altas temperaturas y alto SOC

Como puede observarse en la tabla 2.1, la temperatura de la celda juega un papel muy relevante en el envejecimiento prematuro de ésta, ya que participa en la mayoría de los efectos

que causan envejecimiento, lo que hace fundamental el manejo adecuado térmico de una batería. En [3] se indica que un aumento de 15°C permanente en la temperatura de operación (en general en torno a los 25°C) de una celda de ion-litio puede hacer que el número de ciclos de vida útil de una celda caiga a la mitad.

2.1.5. Generación de calor de una celda de batería

Uno de los aspectos más relevantes en el uso de una batería de ion-litio es el manejo de su temperatura. Las celdas sufrirán degradación y disminución en su vida útil como se señaló en la sección 2.1.4 si es que operan a temperaturas muy altas o muy bajas, en relación a su estado óptimo de operación, definido por el fabricante. Este rango es usualmente entre 25°C y 40°C. La temperatura de las celdas de ion-litio queda determinada por el balance entre el calor generado por la celda y el disipado fuera de ella. El calor generado dentro de una celda se da por descomposición o reacción de los materiales dentro de la celda por las siguientes causas [12]:

- Reacción entre electrolito y ánodo.
- Descomposición térmica del electrolito
- Reacción entre electrolito y cátodo.
- Descomposición térmica del ánodo.
- Descomposición térmica del cátodo.
- Cambio de entropía en el material activo del cátodo o el ánodo.
- Corriente fluyendo a través de una celda con resistencia interna.

De todos estos fenómenos, a una temperatura de operación cercana a la óptima, el efecto dominante es el calor generado por el paso de la corriente a través de la celda. El calor generado en este caso está dado por el efecto Joule, esto es si la resistencia interna tiene un valor R_{in} y la corriente un valor i , el calor generado queda representado por la siguiente ecuación:

$$(2.6) \quad Q = i^2 \cdot R_{in} [W]$$

Si la temperatura supera ciertos umbrales, entre los 70°C y los 180°C, según la literatura y dependiendo del tipo de celda, los fenómenos de descomposición comienzan a ser más relevantes en la generación de calor y la celda puede entrar en un proceso de escape de temperatura (*thermal runaway*) y que la puede llevar en un proceso realimentado positivamente, a explotar o fallar violentamente [6] [12].

2.1.6. Aspectos básicos del empaquetamiento de baterías

Un banco se compone de módulos, que corresponden a arreglos de celdas de baterías, los elementos esenciales de estos. El empaquetamiento de baterías consiste en la distribución de las celdas y de los equipos necesarios para que el banco de baterías funcione, por ejemplo, ventiladores para control de temperatura, sensores de temperatura y corriente eléctrica, sensores de impacto en el caso de vehículos eléctricos, computadores para control (Battery Management System, BMS), elementos de electrónica de potencia y contactos eléctricos. Los pasos del diseño de un banco de baterías pueden observarse en la Ilustración 2.10, estos inician con la selección de una celda dependiendo de los requerimientos eléctricos y económicos para la aplicación para la cual se desea el banco y luego se continua con la agrupación de celdas en módulos y luego en un banco de baterías. Los elementos que conforman un banco de baterías pueden observarse en la Ilustración 2.11

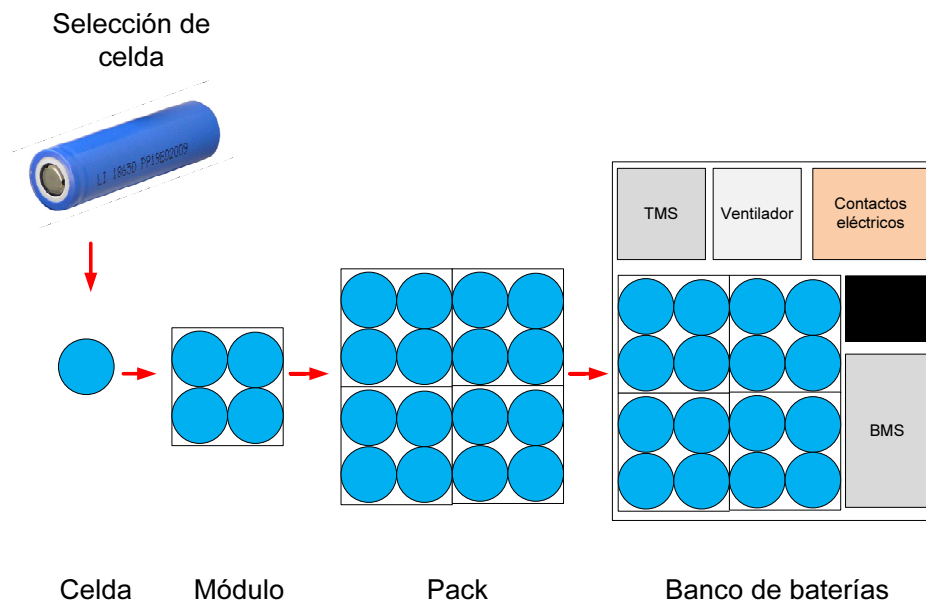


Figura 2.10: Pasos del diseño de un banco de baterías.

El problema del empaquetamiento no es trivial. La definición de cada uno de los elementos necesarios para construir un banco de baterías debe considerar los *trade-off* entre priorizar los diferentes objetivos de diseño que se tienen, los que se enumeran a continuación [4]:

- **Desempeño de la Batería:** Seguridad. Tolerancia al Abuso. Costo de Empaquetamiento. Protección contra Impactos y Accidentes. Durabilidad. Control de la Vida Útil. Reciclaje y re-uso. Construcción del Banco de Baterías. Mantenimiento y Consideraciones para Reparaciones. Administración y Control Térmico, Electrónica de Control y Medidores. Ejemplos de Parámetros: Capacidad (Ah), Potencia Específica (W/kg), Energía (Wh),

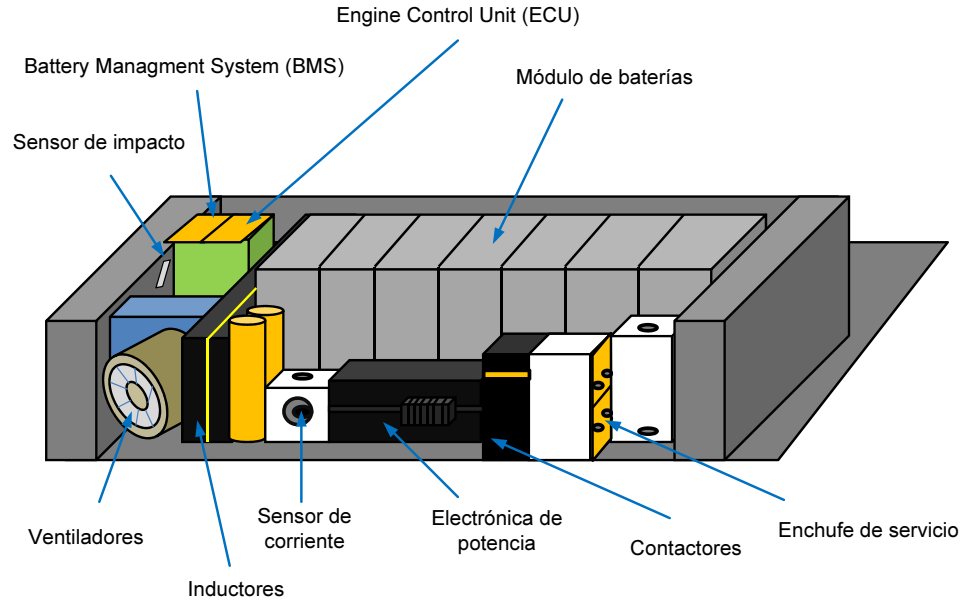


Figura 2.11: Elementos típicos de un banco de baterías.

Densidad Energética (Wh/L), Densidad de Potencia (W/L), Máxima Corriente de Descarga (A), Máximo pulso de corriente de descarga (A), Vida Útil (número de ciclos), Temperatura de Operación (°C).

- **Aspectos Mecánicos:** Interconexión entre celdas. Ensamblaje de celdas, módulos y electrónica. Condiciones Estructurales para resistir golpes y vibración. Protección contra accidentes. Puntos en encaje y ensamblaje al vehículo. Interfaces mecánicas con otros dispositivos del vehículo.
- **Aspectos Eléctricos:** Interfaces eléctricas con el vehículo, a través de inversores y controladores. Monitoreo del voltaje, corriente y temperatura en el vehículo. Control de Estado de Carga (SOC). Capacidad. Energía Mínima Disponible. Potencia. Control de Estado de Salud (SOH). Impedancia. Medición de Degradación.
- **Elección de Celdas:** Uso de arreglos de muchas celdas pequeñas o pocas celdas grandes (rango de 2 a 40 Ampere-hora (Ah)). Uso de celdas cilíndricas o prismáticas.

Los diferentes objetivos de diseño generalmente compiten entre sí, lo que lleva a que el problema de empaquetamiento de baterías de ion-litio sea modelado como un problema de optimización multi-objetivo. El problema se resuelve con herramientas computacionales, de modo de reducir el tiempo que se utiliza en generar un diseño que cumpla con todas las condiciones y además tenga desempeño y vida útil adecuada para las aplicaciones para las cuales se desea.

2.2. Simulación computacional de mecánica de fluidos

2.2.1. Modelo fluidodinámico

Las ecuaciones que gobiernan los sistema de fluidodinámica son conocidas como las ecuaciones de Navier-Stokes y sirven para representar el comportamiento de un flujo newtoniano. Para resolver el sistema se utilizan las ecuaciones diferenciales parciales no lineales de continuidad, moméntum y energía que explican los fenómenos de flujo de fluido, transferencia de calor y turbulencia. Las ecuaciones de continuidad, moméntum y energía se basan en tres relaciones físicas fundamentales: Conservación de masa, la segunda ley de Newton (Fuerza es igual al producto de la aceleración y la masa) y la conservación de la energía, respectivamente [13].

La ecuación de conservación de masa depende de la densidad del fluído ρ y del vector de velocidad \vec{u} :

$$(2.7) \quad \frac{\partial(\rho Vecu)}{\partial t} + \nabla \cdot (\rho \vec{u}) = 0$$

Las ecuaciones de conservación de momentum se calculan en los ejes x , y y z , y dependen de las magnitudes de la velocidad $Vecu$ en cada eje u , v y w respectivamente, además de la densidad del fluído ρ , la presión p y las fuentes de momentum por eje, denotadas S_m :

$$(2.8) \quad \frac{\partial(\rho u)}{\partial t} + \nabla \cdot (\rho u \vec{u}) = \frac{\partial p}{\partial x} + \nabla \cdot (u \nabla u) + S_{m_x}$$

$$(2.9) \quad \frac{\partial(\rho v)}{\partial t} + \nabla \cdot (\rho v \vec{v}) = \frac{\partial p}{\partial y} + \nabla \cdot (v \nabla v) + S_{m_y}$$

$$(2.10) \quad \frac{\partial(\rho w)}{\partial t} + \nabla \cdot (\rho w \vec{w}) = \frac{\partial p}{\partial z} + \nabla \cdot (w \nabla w) + S_{m_z}$$

La ecuación de conservación de la energía depende de e_i , la energía interna del fluído, la conductividad térmica k , la temperatura T , la presión, la velocidad, la generación interna de energía S_i y la disipación Φ .

$$(2.11) \quad \frac{\partial(\rho E_i)}{\partial t} + \nabla \cdot (\rho E_i \vec{u}) = -p \nabla \cdot \vec{u} + \nabla \cdot (k \nabla T) + \Phi + S_i$$

Para completar el sistema, se utilizan las ecuaciones de estado para gases ideales compresibles:

$$(2.12) \quad p = \rho RT$$

$$(2.13) \quad E_i = C_v T$$

En las ecuaciones anteriores, R representa la constante universal de los gases ideales y C_v corresponde a la capacidad calorífica a volumen constante [13].

2.2.2. Mecanismos de transferencia de calor

Existen tres mecanismos de transferencia de calor, descritos por la literatura: conducción, convección y radiación [14]. Durante el funcionamiento de los bancos de baterías de vehículos eléctricos y sistemas para redes eléctricas, predominan la conducción de calor dentro de la celda y la convección forzada del fluido hacia las celdas por medio de la ventilación.

Conducción

La conducción se produce por la actividad molecular en un cuerpo. Se debe a la transmisión de energía de las partículas más excitadas a aquellas que están menos excitadas por interacción y choques entre ellas. Las partículas más excitadas o con alta energía molecular están asociadas a mayores temperaturas y transfieren su energía a aquellas con menor temperatura. La conducción queda determinada por la Ley de Fourier, en la cual se señala que el flujo de calor es proporcional a la magnitud del gradiente de temperatura y con el signo contrario. La constante de proporcionalidad k es conocida como conductividad térmica y es una característica del material en el cual se produce la conducción de calor. La Ley de Fourier se presenta en la ecuación 2.14

$$(2.14) \quad \dot{q} = -k \frac{\partial T}{\partial x}$$

Convección

Este mecanismo de transferencia de calor se produce por el movimiento macroscópico de un fluido. La convección se produce entre un fluido en movimiento y una superficie, mientras ambas tengan diferentes temperaturas. Cuando existe esta diferencia, existe una región del fluido que tiene una temperatura que va desde T_s , asociada a la superficie, hasta una temperatura T_∞ que está asociada al fluido en un sector lejano. Esta región se conoce como capa límite térmica y determina la ecuación 2.15, que rige este flujo de calor.

$$(2.15) \quad \dot{q} = h(T_s - T_\infty)$$

En la ecuación anterior, la constante h representa el coeficiente de convección y depende de la geometría de la superficie y del momentum y las propiedades termodinámicas del fluido.

Radiación

Corresponde a la transferencia de calor por la emisión de energía a través de radiación electromagnética. Todos los cuerpos emiten radiación, dependiendo de su temperatura y características de la superficie radiativa. El flujo de energía radiativa de un cuerpo queda determinado por la Ley de Stefan-Boltzmann (ver ecuación 2.16) y es proporcional a una potencia de la temperatura y a una constante σ llamada la constante de Stefan-Boltzmann. Basado en la ecuación anterior, la transferencia de calor entre dos cuerpos negros por radiación corresponde a la diferencia de calor generado por ambos cuerpos, por un factor de eficiencia F_1 y el área de transferencia de los objetos A_1 , lo que se muestra en la ecuación 2.17.

$$(2.16) \quad e(T) = \sigma T^4$$

$$(2.17) \quad \dot{q} = A_1 F_1 \sigma (T_1^4 - T_2^4)$$

2.2.3. Simulación computacional de mecánica de fluidos

Tal como se señala en la sección 2.1.5, uno de los aspectos más importantes que influyen en el comportamiento de un banco de baterías es la temperatura [15]. Para poder representar adecuadamente la temperatura de una celda en un banco de baterías sin necesidad de hacer costosos experimentos reales, se utiliza software de simulación computacional [16].

Este enfoque se conoce como Ingeniería asistida por computador o CAE, por sus siglas en el inglés *Computer Aided Engineering*. Dentro de los software CAE se encuentran los programas de mecánica de fluidos computacional (CFD, *Computer Fluid Dynamics*) que utilizan métodos numéricos y algoritmos computacionales para analizar problemas de flujo de sustancias.

Los programas CFD resuelven las ecuaciones de Navier-Stokes. Con las expresiones presentadas en la sección 2.2.1, se define un sistema de siete ecuaciones. Por la complejidad de la cómputo de dichas ecuaciones, el sistema se resuelve a través de técnicas como análisis de elementos finitos y grillas computacionales. Las grillas discretizan el espacio a simular en elementos computacionales imaginarios, pequeños y no traslapables que permiten evaluar los problemas de forma finita. Cada iteración de un programa de simulación de este tipo resuelve

millones de operaciones matemáticas por segundo, luego su utilización es solo posible en máquinas de alta capacidad de cómputo [13].

El análisis computacional a través de un CAE permite predecir el valor de la presión, temperatura y velocidad de fluidos y sólidos en cada celda de la grilla y de ese modo, predecir el comportamiento de un sistema sin necesidad de construirlo y hacer un costoso experimento real [13].

La aplicación de CFD para resolver un problema de flujo normalmente requiere de tres etapas [17]: Pre-Procesamiento, Resolución y Post-Procesamiento. La etapa de pre-procesamiento es la que requiere mayor esfuerzo por parte del operador, ya que en ésta se debe definir el modelo a simular, la malla para subdividir a este dominio en volúmenes o elementos finitos. Además se definen las condiciones de borde y se escogen los modelos físicos a utilizar y sus ajustes para el problema particular a evaluar, de modo que los resultados sean consistentes con la realidad. En esta etapa, se dedica especial énfasis a la construcción de la malla, ya que es un aspecto crítico de la simulación, porque influye tanto en la velocidad de la obtención de las soluciones como en su precisión: una malla más fina permite resolver con mayor precisión, pero también requiere mayor utilización de recursos computacionales, tanto en memoria como en número de iteraciones que le tomará al programa converger a una solución. Por lo mismo, encontrar una malla que represente la situación de forma suficientemente adecuada, pero que no sea demasiado fina, permite ahorrar alto costo computacional.

Cuando el modelo CFD está definido, se puede comenzar la etapa de resolución. Las ecuaciones gobernantes de los modelos físicos son discretizadas y resueltas dentro de cada una de las celdas definidas por la malla y operadas en conjunto con las condiciones de borde. Para alcanzar una estabilidad en procesos altamente no lineales, se utilizan métodos numéricos iterativos para la resolución de estos sistemas de ecuaciones.

Finalmente, la etapa de post-procesamiento corresponde al análisis del modelo y sus resultados por parte del usuario.

2.2.4. Uso de simulación computacional en modelamiento térmico de bancos de baterías

El importante efecto que tiene la temperatura sobre la vida útil de las baterías de ion-litio ha motivado ampliamente el uso de software de simulación computacional para representar la generación de calor y flujos de fluido en bancos ventilados con convección natural, convección forzada o refrigerados a través de circuitos de líquido.

En [17] se estudia y optimiza la distribución de los canales de refrigeración para un banco de baterías equivalente al que monta el Chevrolet Volt.

En [18] , el investigador Xiao Hu describe los diferentes niveles de detalle y aspectos característicos de la simulación térmica de un banco de baterías, dentro de la plataforma ANSYS. En [19], Hu utiliza la simulación en ANSYS para obtener un modelo en base a redes Foster para poder representar el comportamiento térmico de un banco de baterías de vehículo eléctrico a largo

plazo.

En [20], Muratori realiza la modelación térmica de una celda de batería para ser representada a través de un modelo computacional de una dimensión, que es comparado con el obtenido en el software de simulación COMSOL y validado a través de pruebas en laboratorio a través de una cámara termografica .

En [21] se utiliza software CAE para modelar un banco de baterías prismáticas para un vehículo híbrido.

Finalmente, es particularmente relevante el trabajo realizado por NREL (National Renewable Energy Laboratory) en Estados Unidos acerca de la simulación de empaquetamiento de baterías. NREL ha desarrollado herramientas propietarias como el software ADVISOR y el programa de investigación CAEBAT para representar térmicamente las celdas en los bancos. En NREL se ha estudiado a través de simulación con CAE y en laboratorio los bancos de varios de los vehículos eléctricos e híbridos que se venden en Estados Unidos, como el Honda Isight y el Toyota Prius. El foco principal de su investigación ha estado enfocado en encontrar mejores formas de diseñar sistemas de manejo de temperatura (Thermal Management System) para los bancos de baterías [22] [23] [24] [25] [26] [27] [28].

En aplicaciones comerciales, los software de simulación ANSYS, COMSOL y STAR-CCM+ de CD-Adapco ofrecen documentación acerca de como modelar de forma eléctrica y térmica a las celdas e incluso desde el punto de vista de la electroquímica.

2.3. Optimización Multi-Objetivo

2.3.1. Definición del Problema de Optimización Multi-Objetivo

Los problemas de Optimización Multi-Objetivo son aquellos en los cuales existe una serie de objetivos que no necesariamente van en el mismo sentido y en general, se contraponen entre sí. Además, pueden ser difíciles o imposibles de comparar por estar medidos en unidades diferentes.

La meta de la resolución del problema de Optimización Multi-Objetivo es determinar todos los vectores que satisfacen las restricciones y entregan un valor óptimo para todas las funciones. Ese caso evidentemente es ideal, ya que en los problemas reales, es imposible que un solo punto en el espacio logre optimizar todas las funciones objetivo. Por lo anterior, tomar una decisión no es trivial como en el caso de la optimización convencional con un solo objetivo, ya que cuando se encuentra una solución, siempre se estará priorizando uno o más objetivos sobre el resto.

Los problemas de la optimización Multi-objetivo para una minimización se definen de la siguiente forma [29]:

$$\text{Minimizar: } [f_1(x), f_2(x), \dots, f_k(x)]$$

Sujeto a las siguientes condiciones o restricciones de desigualdad g_i con $i = 1 \dots m$:

$$g_i(x) \leq 0$$

Y a otras tantas restricciones de igualdad h_j con $j = 1 \dots p$:

$$h_j(x) = 0$$

Dónde:

k : Número de funciones objetivo, luego $f_l(x)$ con $l = 1 \dots k$

$x = [x_1, x_2, x_3, \dots, x_n]$: Vector de variables de decisión n -dimensional

En general se toma que $f(x)$ se encuentra en el espacio de los objetivos y x en el espacio de las decisiones.

Dado que las soluciones en general no serán óptimas para todos los objetivos, que normalmente se busca tener un grupo de soluciones entre las cuales sea sencillo determinar el *trade-off* entre las diferentes opciones que se tiene en diseño (cuanto se pierde por priorizar una función objetivo u otra). El contar con un conjunto de soluciones con “trade-off” claros, permite a quién toma la decisión final, contar con mayor información y hacer su elección de forma fundamentada.

La Optimalidad de Pareto se define como sigue:

Para $x^* \in F$, esta solución será Pareto Óptima si no existe ninguna otra solución $x \in F$ tal que cumpla que:

$$f_i(x) \leq f_i(x^*) \quad \forall i = 1 \dots k$$

$$f_j(x) < f_j(x^*) \quad \text{para al menos un } j$$

En otras palabras, x^* es óptima ya que ningún otro vector de decisiones que cumpla las restricciones (factible) puede hacer que uno de los criterios baje, sin hacer que otro criterio aumente su valor en una dirección. El conjunto de soluciones Pareto Óptimo generalmente recibe el nombre de “No Dominado”. Se define también que las soluciones que dominan el mismo conjunto de elementos se encuentran en el mismo “Frente de Pareto” [29] [30], donde el primer frente, que no es dominado, es el Frente de Pareto óptimo.

2.3.2. Métodos para Búsqueda de Soluciones a Problemas Multi-Objetivo

En general, se tienen tres posibles formas de encontrar soluciones a un problema Multi-Objetivo y siguen visiones particulares sobre la forma del problema [31]:

- Suponer que hay solo un objetivo mucho más importante que el resto (optimizar ésta).
- Aplicar una suma ponderada de los objetivos (una función objetivo).
- Definir que el problema es multi-objetivo y que se requiere búsqueda del Frente de Pareto.

Las primeras dos opciones consideran reducen el problema de optimización a un problema de un solo objetivo, con lo cual las técnicas a utilizar para encontrar las soluciones son generalmente más sencillas, simplificando el problema. Esta visión hace que se encuentren soluciones que no son enteramente adecuadas o no entregan toda la información que se puede extraer del problema, si es que todos los objetivos son importantes o bien si no se pueden medir con la misma unidad.

Por el hecho de que en un problema multi-objetivo la función objetivo es un vector, en general aumenta aún más la complejidad del problema de optimización, más allá del hecho de que la existencia de un óptimo global para todas las funciones no sea factible de encontrar. En general, los algoritmos clásicos y técnicas de optimización numéricas tienen limitaciones para resolver problemas Multi-Objetivo. En primer lugar, estos métodos son sensibles a la forma, continuidad y concavidad del Frente de Pareto, por lo tanto no existe seguridad de que puedan encontrar una solución óptima. También son poco robustos a las inicializaciones, lo cual indica que habría que conocer una vecindad de alguna solución para comenzar a iterar y buscar una solución mejor u óptima, lo cual es evidentemente insuficiente ya que muchas veces no se conocen vecindades de soluciones que sean buenas. Finalmente, cada iteración entregará soluciones únicas y no el Frente de Pareto completo, por lo tanto para encontrar el Frente deben hacerse búsquedas individuales, lo que redundará en un alto costo computacional [29] [30].

Lo anterior lleva a que se definan métodos diferentes a los habituales para solucionar problemas de optimización multi-objetivo. En general, estos métodos pueden separarse en tres tipos dependiendo de la interacción que tiene el operador que toma la decisión final (elección de la solución óptima dentro del Frente de Pareto) en la solución del problema [31] [32]:

- **A Priori:** En este tipo de métodos, se toma una decisión antes de realizar la búsqueda de soluciones. El diseñador indica explícitamente sus preferencias antes de comenzar a resolver el problema multi-objetivo y la búsqueda se realiza en base a estas decisiones. Como ejemplo, el operador puede indicar que el costo de operación es el doble de importante que el de implementación de una solución antes de comenzar a buscar posibles resultados óptimos [31]. El mayor esfuerzo del operador se realiza antes de que un método computacional busque el óptimo, luego se requiere información amplia del problema a enfrentar antes de programar un algoritmo que encuentre las soluciones óptimas.
- **Progresivo o Interactivo:** En este tipo de métodos la decisión se toma de forma paralela a la búsqueda. El operador continuamente recibe soluciones del problema, las evalúa y modifica el problema en base a sus aspiraciones o a que tan buenas son las soluciones obtenidas hasta el momento. Luego, recibe otro conjunto de soluciones y sigue el ciclo. Las decisiones que toma el operador son en general teniendo información sobre las soluciones deseadas: valores máximos que podría obtener cada función objetivo, aspiraciones y costo en los otros objetivos tras aumentar uno de los objetivos en particular. Por otra parte, esta estrategia

obliga a que el mayor esfuerzo o trabajo del operador se haga durante la optimización, por lo tanto en general es la que mayor tiempo consume.

- **A Posteriori:** En esta última estrategia, la decisión se toma después de haber realizado la búsqueda. El investigador recibe un conjunto de soluciones y elige la mejor en base a alguna regla dependiendo de la prioridad que le dé a cada objetivo en particular. Estas reglas de decisión pueden estar pre-definidas, pero siempre se elige después de haber obtenido el conjunto completo de soluciones. Lo anterior implica que el programa computacional opera antes de que el investigador haga su mayor trabajo.

2.4. Algoritmos Evolutivos aplicados a Problemas Multi-Objetivo

2.4.1. Aspectos Básicos

Los algoritmos evolutivos son métodos de optimización y búsqueda basados en los mecanismos de la evolución biológica. Se considera que los algoritmos evolutivos son adecuados para resolver problemas multi-objetivo y por ello a la fecha existe una veintena de estas técnicas en etapa de investigación y desarrollo. El inicio de estas investigaciones comenzó en la década de los 80' [29]. Los Algoritmos Evolutivos han sido utilizados ampliamente para resolver problemas de un solo objetivo con éxito y sus implementaciones han sido la base para construir herramientas que permitan obtener los Frentes de Pareto para problemas Multi-Objetivo. Las ventajas de los Algoritmos Evolutivos en problemas con múltiples funciones que se desea optimizar, son en primer lugar que trabajan con poblaciones de soluciones y no con individuos. Lo anterior es positivo por varias razones. En primer lugar, debido a que las soluciones de un problema multi-objetivo en general se entregan en forma de "trade-off" a través de un Frente de Pareto, por lo tanto, tener una población tiene sentido, ya que siempre se deberá entregar más de una solución. En segundo lugar, el contar con varios individuos es más robusto ante la forma y continuidad del Frente de Pareto y no requiere información adicional acerca de las derivadas y restricciones del problema [29] [30]. En el caso particular de los Algoritmos Genéticos esta robustez viene dada por los operadores genéticos, tales como la recombinación (cross-over) y la mutación. Finalmente, al evolucionar continuamente cada uno de los individuos de la población, un Algoritmo Evolutivo no requiere información acerca de las derivadas de los objetivos o restricciones, por lo tanto, no es sensible a inicializaciones pobres como los métodos clásicos ya que la recombinación y mutación permiten la exploración continua de nuevas soluciones no dominadas. Si se implementa de la forma correcta, el costo computacional del funcionamiento de un Algoritmo Evolutivo aplicado a Problemas Multi-Objetivo (MOEA, Multi-Objective Evolutionary Algorithm) es mucho menor que el de un método convencional. Finalmente, se pueden usar soluciones híbridas que permitan sacar lo mejor de ambos métodos [29] [30].

En base a lo anterior, se ha escogido a los algoritmos evolutivos como herramienta fundamental para resolver el problema de empaquetamiento de baterías, que como ha sido descrito en secciones anteriores, es un problema multi-objetivo particularmente complejo con muchas variables, varias finalidades que satisfacer y que se contraponen mutuamente.

En los prácticamente 30 años de historia de los MOEA, los que se inician en los 80 (estos se inician con David Schaffer con la creación del algoritmo VEGA (Vector Evaluated Genetic Algorithm) [29]), se han desarrollado en la academia y en la industria más de 20 algoritmos diferentes que permiten aprovechar las ventajas de los Algoritmos Evolutivos aplicados a problemas Multi-Objetivo. La herramienta que se seleccionará para resolver el problema de Empaquetamiento Óptimo de Baterías resultará de una combinación o modificación de algunas de estas técnicas. Particularmente en los estudios más recientes, la utilización del método NSGA II (Fast Nondominated Sorting Genetic Algorithm) es muy popular y ha entregado resultados adecuados en problemas de diseño de circuitos electrónicos, piezas en aerodinámica e ingeniería y micro-redes eléctricas, además, de otras aplicaciones industriales [29]. Otras técnicas con desarrollo acelerado en los últimos años han sido aquellas técnicas que no se basan en la arquitectura de algoritmos genéticos, como Optimización por enjambre de partículas (PSO, Particle Swarm Optimization), Evolución Diferencial (DE) y Sistema Inmune Artificial (AIS, Artificial Immune System).

Las técnicas más populares y efectivas según las diferentes investigaciones en MOEA se muestran en la Tabla 2.2.

Tabla 2.2: Técnicas más utilizadas en Algoritmos Evolutivos Multi-Objetivo

Algoritmo	Nombre Completo
VEGA	Vector Evaluated Genetic Algorithm
MOGA	Multi Objective Genetic Algorithm
WBGA	Weight Based Genetic Algorithm
NPGA	Niched Pareto Genetic Algorithm
RWGA	Random Weighted Genetic Algorithm
PESA	Pareto Envelope based Selection Algorithm
PAES	Pareto Archived Genetic Algorithm
NSGA	Nondominated Sorting Genetic Algorithm
NSGA II	Fast Nondominated Sorting Genetic Algorithm
SPEA	Strength Pareto Evolutionary Algorithm
SPEA2	Improved Strength Pareto Evolutionary Algorithm
RDGA	Rank Density based Genetic Algorithm
DMOEA	Dynamic Multi-Objective Evolutionary Algorithm
MEA	Multi-Objective Evolutionary Algorithm
Micro-GA	Micro Genetic Algorithm
MOPSO	Multi Objective Particle Swarm Optimization
AIS	Artificial Immune Systems
ACO	Ant Colony Optimization
SS	Scatter Search
DE	Differential Evolution

2.4.2. Estructura Genérica de un Algoritmo Evolutivo Multi-Objetivo

Un algoritmo evolutivo genérico para resolver problemas multi-objetivo cuenta con las etapas que se describen en esta sección en forma de pseudo-código. Algunos de estos pasos se pueden evitar en caso de utilizarse arquitecturas más sencillas:

- Inicializar la población (aplicado a un problema de diseño, los individuos de la población son prototipos factibles).
- Evaluar la función objetivo vectorial en la población.
- Asignar un Ranking a cada individuo en base a su dominancia en el sentido de Pareto.
- Calcular la densidad o nichos en fenotipo o genotipo.
- Si aún no se ha alcanzado la condición de término (N° de generaciones o mínimo avance), ejecutar:
 - Selección de los individuos aptos para formar una nueva población.
 - Operación Genética entre los individuos seleccionados (Recombinación, Mutación).
 - Evaluar los valores de la función objetivo vectorial en la población.
 - Asignar Ranking a cada individuo de la población en base a su dominancia Pareto.
 - Calcular la densidad o nichos en fenotipo o genotipo.
 - Aplicar métodos de compartimiento de *fitness sharing* según densidad.
 - Guardar los individuos dominantes en conjuntos externos, volver a la selección.

La diferencia entre un algoritmo genético estándar (mono-objetivo) y uno multi-objetivo está en los métodos de cálculo y evaluación de *fitness* (capacidad reproductiva o aptitud). En el caso de tener sólo una función objetivo, el cálculo del *fitness* es directo ya que usualmente es proporcional al valor de la función (en el caso de un problema de maximización, inversamente proporcional en el caso de un problema de minimización). Como en el caso de multi-objetivo existen varias funciones objetivos y el *fitness* es siempre un valor numérico, debe hacerse una transformación. En general se considera la dominancia sobre otros elementos como una buena medida de *fitness* y es por eso que se utilizan diversas técnicas para medirla. Los métodos de Ranking, Densidad y *fitness sharing* pueden considerarse como etapas dentro del cálculo y transformación del *fitness*. Las figuras 2.12 y 2.13 muestran un diagrama de bloques para un algoritmo evolutivo mono-objetivo y multi-objetivo, respectivamente.

2.4.3. Diseño de un Algoritmo Evolutivo Multi-Objetivo

La elección de la técnica de algoritmo evolutivo es importante pero no necesariamente fundamental para la solución de un problema. En general, la modelación del problema resulta ser más importante que la elección de la herramienta de optimización, pero existen diferencias

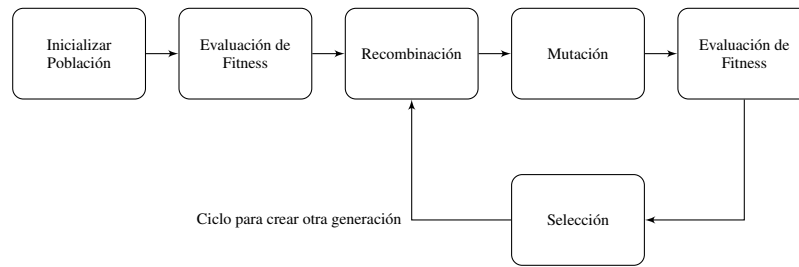


Figura 2.12: Estructura básica de un algoritmo evolutivo de un objetivo

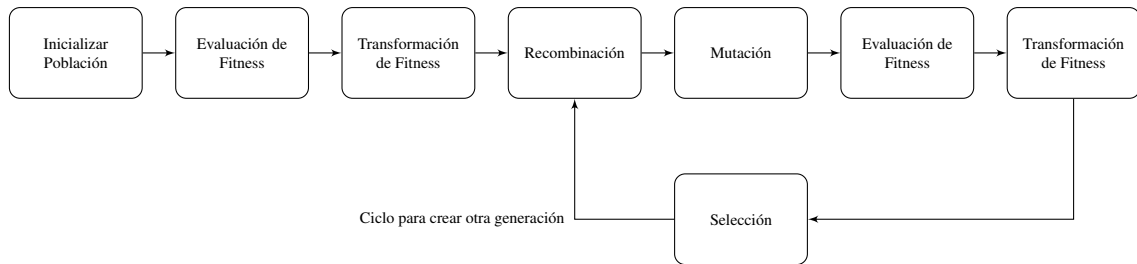


Figura 2.13: Estructura básica de un algoritmo evolutivo multi-objetivo.

que podrían llevar a tomar mayor tiempo, costo computacional en encontrar las mismas o peores soluciones que en el caso de elegir una estrategia de MOEA adecuada al problema [31].

La elección de un MOEA adecuado se basa en la priorización de los diferentes objetivos de diseño que se han definido para este tipo de algoritmos y que son los siguientes [31]:

- **Conservación de Soluciones:** Se desea preservar los puntos no dominados en el espacio objetivo y sus puntos asociados en el espacio de decisiones. Estos puntos no dominados no necesariamente son el Frente de Pareto, pero son las mejores soluciones encontradas hasta el momento según las reglas de la Optimalidad de Pareto, por lo tanto debieran ser guardadas de alguna forma o tener mayor capacidad de reproducción en el caso de algoritmos genéticos, al menos hasta que sean superadas por nuevas soluciones no dominadas en siguientes generaciones.
- **Progreso:** El algoritmo evolutivo debe mantener un avance continuo hacia el Frente de Pareto en el espacio de los objetivos. Esto significa que mientras el algoritmo esté operando, se irá mejorando progresivamente las soluciones no dominadas idealmente hasta llegar al verdadero Frente de Pareto del problema.
- **Diversidad:** El algoritmo debe ser capaz de mantener la diversidad de los puntos que actualmente tiene en el Frente de Pareto, tanto en el espacio de los objetivos (fenotipos) como en el espacio de las decisiones (genotipos). La diversidad es deseada para evitar evolucionar exclusivamente algunos grupos de especies y por lo mismo, no encontrar todos los “trade-off” representativos existentes. Mientras más separados estén los puntos tanto en fenotipos como en genotipos, el Frente de Pareto encontrado en la última generación, aún cuando no sea el “real” o “teórico”, permitirá al investigador tomar una decisión más certera.

- **Soluciones Acotadas:** El algoritmo evolutivo debe entregar suficientes soluciones a quién tomará la decisión, pero en un número limitado. De nada sirve entregar una cantidad enorme de soluciones, ya que el operador deberá hacer mucho trabajo para escoger y no podrá ver con claridad los diferentes “trade-off”.

Los objetivos anteriormente señalados se podrán lograr dependiendo de la arquitectura de algoritmo evolutivo escogido para el problema en particular. Dependiendo de este, en algunos casos se preferirá alguno u otro. Las modificaciones a los algoritmos que permiten lograr los cuatro objetivos anteriores son los siguientes [31]:

Cálculo del Fitness

En general el *fitness* representa el número que permite determinar la probabilidad de un individuo de reproducirse. Por lo tanto, aporta a buscar mejores soluciones, al progreso del algoritmo y también a la mantención de las soluciones no dominadas ya encontradas. Básicamente es un puntaje, y quién tenga el mayor puntaje es el individuo más fuerte de la población y por tanto el más apto para reproducirse. En el caso de un problema de un solo objetivo, la determinación del *fitness* de un individuo de la población es relativamente sencilla, directa y lógica. Sin embargo, esto cambia en el caso de los MOEA porque al haber múltiples objetivos, no es tan claro como asignar el *fitness* [31]. A continuación se describen algunas estrategias básicas que se han utilizado en los MOEA en los últimos 30 años de estudios y trabajos:

- **Suma ponderada de las funciones objetivo normalizadas:** Se agrega un peso por cada función objetivo y se hace la suma ponderada de las variables previamente normalizadas para evitar que haya diferencias simplemente por cuestiones numéricas. El vector de pesos puede agregarse como dado a priori o bien puede ir evolucionándolo al agregarlo al vector de características. El método de sumas ponderadas fue presentado en el algoritmo RWGA [31].
- **Alteración de Función Multi-Objetivo a Función Mono-Objetivo:** Esto puede hacerse ya sea haciendo una partición de la población y evaluando cada partición con una función objetivo particular para determinar el *fitness* y hacer la selección dentro de cada partición, o bien elegir una función objetivo aleatoriamente en cada iteración y aplicar esa para determinar el *fitness*. La ventaja de este método es que convierte el problema multi-objetivo en un problema que puede resolver un Algoritmo Genético simple. Este método se utilizó en el algoritmo VEGA.
- **Ranking de Pareto:** Se utiliza la dominancia de Pareto para generar primero un ranking (en el cual quién tenga el número más bajo es el que tendrá mayor *fitness*). Existen varias alternativas, como por ejemplo, determinar en qué Frente de Pareto se encuentra la solución (el primer frente es el de las soluciones no dominadas, el segundo es el de las soluciones no dominadas excluyendo al primer frente y así hasta el final), a través del número de soluciones que son dominadas por el individuo o aquellas que dominan al individuo que se está evaluando. Este método se utiliza en NSGA II.

Generación de Soluciones Diversas

Como se describe anteriormente, el objetivo de encontrar el frente óptimo de Pareto es tener una serie de *trade-off* entre las mejores soluciones posibles, de modo que permitan al diseñador (en este caso de un banco de baterías), tomar la decisión de priorizar una función objetivo sobre otra dependiendo de los valores obtenidos por su solución. Luego, es natural que mientras más diversas sean las soluciones, mejor será para el investigador, ya que tendrá más facilidad para ver los *trade-off* y así tomar la decisión con mayor información. Si las medidas preventivas no se toman, por ejemplo se puede observar el fenómeno de *drift Genético* que corresponde a que la población tiende a formarse en torno a ciertos centros, generando clusters y no diversidad. Esto sucede incluso en el caso de los Algoritmos Evolutivos aplicados en funciones de un solo objetivo y por lo mismo es necesario evitarlo en el caso de MOEA. La generación de soluciones diversas aporta al objetivo de diversidad y también en algunos casos a controlar el tamaño de las poblaciones de soluciones no dominadas (antes de almacenarlas o entregarles mayor *fitness*). Algunas estrategias existentes son:

- **Compartición de Fitness (Fitness Sharing):** Premia la búsqueda de soluciones en secciones que no han sido exploradas. Esto se hace a través de la reducción artificial del *fitness* de las soluciones que se encuentran en secciones con densidades de población muy altas (Método de penalización en el algoritmo MOGA) ya sea en el espacio de características o en el espacio de los objetivos. Se definen umbrales para las vecindades aceptadas.
- **Distancia de Población:** Busca la separación de las soluciones encontradas, a través del cálculo de una medida de distancia entre los individuos, ponderada por la resta de la máxima distancia y la mínima distancia encontrada en toda la población. De este modo es dinámica y no requiere la definición de umbrales. En algunos algoritmos se utiliza como condición de desempate para el método de reproducción por torneo (caso de NSGA II).
- **Densidad basada en Celdas:** Consiste en dividir el espacio de objetivos en celdas (o hipercubos) k-dimensionales. Luego, se calcula el número de individuos presentes en cada una de las celdas y se evalúa la dominancia con respecto a los individuos que comparten la misma celda, además de la cantidad de elementos. Si se supera un umbral de tamaño de la población, se escoge algunos individuos dominantes en cada celda y el resto es eliminado. Este mismo enfoque puede utilizarse para transformar el problema multi-objetivo en un problema bi-objetivo (caso de RDGA) en el cual se pretende minimizar el ranking individual de cada elemento de la población y por otro lado, la densidad de población. Luego, las funciones objetivo se vuelven una suerte de características. El concepto de densidad también se utiliza para escoger las partículas líderes en el algoritmo MOPSO.

Elitismo

El elitismo consiste en conservar las mejores soluciones encontradas en una cierta iteración y permitir que estas “sobrevivan” directamente sin reproducirse. Todos los métodos modernos de MOEA utilizan el elitismo, por la dificultad de los problemas multi-objetivo que enfrentan y

que pueden permitir que las soluciones correctas sean encontradas en cualquier momento de la evolución de la población. Todas las soluciones no dominadas son consideradas de elite [31].

- **Mantener Soluciones Elitistas en la Población:** Simplemente consiste en copiar las soluciones no dominadas hasta que se alcance un cierto umbral. Esto es porque no se quiere tampoco que las soluciones encontradas hasta ahora sean las únicas que estén dentro de la población. Si se pasa el umbral, deben eliminarse algunas soluciones no dominadas. Esto puede hacerse por ejemplo, eliminando una cierta cantidad parametrizada de soluciones a través de un campeonato utilizando la densidad (como es el caso de NSGA II).
- **Elitismo con Población Externa:** Este método es bastante popular en las implementaciones de MOEA. Consiste en almacenar un número acotado de soluciones no dominadas dentro de una nueva población que se almacena en un archivo externo. En el caso de SPEA, simplemente se mantienen ambas poblaciones por separado (siempre controlando el tamaño para mantener la diversidad) en tanto que en SPEA2 y MOPSO, las listas de la población y el archivo externo se combinan para generar la nueva generación de individuos, para luego volver a guardar el archivo externo.

Manejo de Restricciones

Que todas las soluciones generadas tengan que cumplir las restricciones del problema de optimización parece natural, sin embargo, en algunas implementaciones de MOEA no se eliminan las soluciones infactibles de inmediato, ya que podrían contener información relevante y solo tener una o dos restricciones saltadas, las que podrían ser modificadas con mutación o recombinación. El manejo de restricciones tiene influencia en el progreso del algoritmo evolutivo. Esencialmente, existen cuatro alternativas para manejar las soluciones infactibles en una generación para un MOEA [31]:

- Eliminar las soluciones infactibles inmediatamente. (Pena de Muerte)
- Reducir artificialmente el *fitness* de las soluciones infactibles (complejo en Multi-Objetivo por el carácter de dominancia)
- Imponer que el Algoritmo Genético o Evolutivo genere siempre soluciones factibles
- Transformar, corregir o reparar las soluciones infactibles

Distribución Paralela o Híbrida

Dado que los algoritmos evolutivos también pueden caer en problemas como posibles convergencias a mínimos locales, dependiendo de la implementación, sobre los cuales no caen los métodos clásicos de búsqueda del frente de Pareto, entonces las soluciones híbridas se han popularizado entre los desarrolladores. Luego, las soluciones híbridas tienen que ver con el progreso del algoritmo hacia las soluciones óptimas. Particularmente, el elitismo y la

conservación de diversidad son muy costosas computacionalmente y por lo anterior, en algunas implementaciones se han reemplazado por métodos menos eficaces pero mucho más eficientes. Las dos alternativas más utilizadas hasta ahora en problemas de diseño son la Distribución Paralela (que consiste en generar “poblaciones vecinas” a una población a través de perturbaciones, y escoger los mejores individuos padres en base a todas las poblaciones) y la Búsqueda Local de soluciones [31].

2.4.4. Etapas del desarrollo de un algoritmo evolutivo multi-objetivo

En general, la vida de un algoritmo evolutivo para resolver problema multi-objetivo como herramienta tiene seis etapas [31], las cuales se describen a continuación:

- **Modelación (Etapa 0):** Esta etapa es fundamental en lo que viene más adelante, ya que determina las características del problema que se intentará resolver utilizando las herramientas de algoritmos evolutivos, por lo tanto es muy importante dedicarle gran énfasis ya que determina el éxito de lo que sigue en el proyecto. En esta sección, hay que definir el problema multi-objetivo, lo cual significa construir funciones objetivo y una representación cromosómica (las variables de decisión). Una vez definido lo anterior, deben agregarse las restricciones de igualdad y desigualdad.

En una segunda etapa de la sección de modelación, debe integrarse el modelo anterior a una estrategia de optimización de un algoritmo evolutivo. Esto significa definir una plataforma para obtener matemáticamente los valores de las funciones objetivo y también una plataforma para realizar la integración con el algoritmo evolutivo. No necesariamente se encuentran en el mismo lenguaje de programación o basadas en el mismo software.

- **Ejecución (Etapa 1):** Esta etapa corresponde a la utilización del algoritmo evolutivo para encontrar las soluciones óptimas en el sentido de Pareto. Comienza construyendo una primera generación de individuos y rescatando los elementos no dominados en cada generación. Este avance del algoritmo llevará a converger hacia el Frente de Pareto real del problema (que en general es desconocido). El algoritmo se ejecuta hasta que se alcance un cierto número máximo de generaciones o bien hasta que se alcance alguna métrica que indique que el algoritmo no está avanzando lo suficiente como para justificar una nueva generación.
- **Distribución (Etapa 2):** Aunque se ha obtenido un conjunto que es óptimo en el sentido de Pareto, puede que no esté equilibrado o distribuido uniformemente tanto en fenotipo como en genotipo. Lo anterior es importante para intentar que se representen la mayor cantidad de *trade-off* existentes en el espacio de objetivos y ante todas las restricciones que impone el espacio de decisiones.
- **Selección (Etapa 3):** En esta etapa, se deben elegir algunos (o todos) los puntos óptimos del Frente de Pareto conocido para que el investigador u operador del software escoja la solución adecuada, ya sea a través de una regla de decisión predeterminada o numérica, o bien en base a sus necesidades o intereses para resolver el problema. La decisión se toma en base a los valores de las funciones objetivo, sin considerar el espacio de decisiones.

- **Implementación (Etapa 4):** En base a la decisión del operador, recuperar las variables de decisión que permiten lograr el valor óptimo escogido para realizar su posterior implementación en un sistema, o en un diseño como se hará en este caso.
- **Experimentación (Etapa 5):** Hacer pruebas reales con el sistema diseñado en base al algoritmo evolutivo, visualizar resultados y definir como mejorar, ya sea dentro o fuera del optimizador.

2.4.5. Aplicaciones de Algoritmos Evolutivos en Problemas Multi-Objetivo

Las aplicaciones de los MOEA son amplias y abarcan una gran cantidad de industrias, entre las que se puede destacar aplicaciones en Aerodinámica, Ingeniería Química, Electrónica, Electricidad y Redes de Potencia y Termodinámica [31]. Varias de estas aplicaciones son similares al problema de diseño que se propone para un banco de baterías, que es el que se pretende abordar en este trabajo, ya sea por las variables, restricciones o funciones objetivo que se utilizan, por la complejidad de los problemas o por la modelación que se requiere para el problema.

Dentro de las anteriores, se pueden destacar las siguientes aplicaciones:

- **Termodinámica:** Las aplicaciones en sistemas de control térmico y temperaturas están relacionadas con el problema del empaquetamiento de las baterías de ion-litio porque en estas, uno de los criterios más importantes para determinar performance y vida útil de un banco, es precisamente el empaquetamiento. En termodinámica se ha utilizado la modelación multi-objetivo para resolver problemas en los siguientes ámbitos:
 - Diseño de sistemas de ventilación / refrigeración / calentamiento para edificios [33] [34].
 - Diseño de sistemas de refrigeración para artículos electrónicos [35] [36] [37].
 - Diseño de sistemas de control térmico para ambientes [38] [39].
 - Optimización de muros multi-capa según características térmicas [39].
 - Diseño térmico de motores [40].
- **Electrónica:** En general el diseño de micro-chips y circuitos integrados es un proceso complejo que requiere de un trabajo intensivo por parte de ingenieros y software especializado. Además, puede interpretarse como un problema multi-objetivo, dadas las necesidades térmicas, eléctricas, electrónicas y de costo que existen para cada circuito [41] [42]. En general se puede modificar la posición y cantidad de celdas (o transistores visto desde un nivel más bajo). Dentro de las aplicaciones multi-objetivo resueltas con algoritmos evolutivos más relevantes por su similitud al problema de empaquetamiento de baterías destacan las siguientes:
 - Diseño de Circuitos Electrónicos Analógicos [43].
 - Diseño térmico de Procesadores [41] [44].

- Posicionamiento óptimo de celdas en circuitos VLSI (Very Large System Integration) [43] [45] [46] [47] [48].
- **Aerodinámica:** En [49] y en [50] se entregan aplicaciones de mejora de formas de objetos a través de la utilización de problemas multi-objetivo. En el caso particular de [50] se diseña una aleta para un avión en base a tres objetivos opuestos y a ciertas restricciones físicas para los coeficientes de roce y la forma. El problema cuenta con 13 variables y se utiliza un algoritmo llamado SMS-MOEA que es similar a SPEA 2. En [49] se utiliza un híbrido de NSGA II, búsqueda local y clustering para diseñar diferentes objetos.
- **Diseño de Micro-Redes:** En [51] se utiliza la Optimización Multi-Objetivo para diseñar una Micro-Red que podría contar con un sistema de paneles fotovoltaicos (PV), un generador eólico, un generador diesel, un generador a hidrógeno y sistemas BESS. Existen tres objetivos que se oponen y que se desea optimizar simultáneamente, que corresponden a los costos totales de la instalación (en Euros), la emisión de dióxido de carbono como contaminante (en kilogramos por año) y la energía que no se entrega para uso en carga (Unmet Load, en kilowattthora/año). El problema cuenta con 11 variables, que determinan el tipo de tecnología para cada uno de los tipos de generadores y baterías además de la cantidad de estos considerados. Los costos dependen tanto de la adquisición de los equipos como del mantenimiento y reparaciones programadas. Del mismo modo, se tienen estimaciones para la energía que se requerirá y para las emisiones de CO₂ que produce cada una de las micro-generadoras. Para resolver el problema se utiliza una combinación de SPEA2 y un Algoritmo Genético mono-objetivo, que tiene por objetivo hacer un refinamiento de la solución original (para los tres objetivos) exclusivamente en el aspecto de costo, dándole mayor prioridad. Las restricciones tienen que ver con las potencias máximas y mínimas, regulaciones medioambientales y gubernamentales, además de otros aspectos propios del problema. El diseño de micro-redes de generación también en otras aplicaciones con diferentes objetivos, variables y restricciones en [52], [53], [54] y [55].
- **Otras Aplicaciones:** En [56] se modela como problema multi-objetivo la configuración óptima para un vehículo híbrido (combustible – eléctrico, HEV). Los objetivos que se superponen son la mayor economía de combustible, además de la reducción de emisiones de hidrocarburos, dióxido y monóxido de carbono además de NO_x. Las especificaciones que pueden modificarse están en las baterías y en el motor. Entre las variables que se aceptaron para el problema de optimización están la velocidad máxima a la cual el vehículo opera exclusivamente con el motor eléctrico (Modo ZEV, Zero Emission Vehicle), características de la envolvente del torque (Nm), Tamaño del Motor, Tamaño del Motor Eléctrico y la Capacidad de las Baterías. Las restricciones impuestas son propias de las variables (cilindrada, potencia) así como de performance, correspondiendo a tiempos máximos para alcanzar ciertas velocidades. En [57] se utiliza la optimización multi-objetivo para diseñar un motor sincrónico.

2.4.6. Aplicaciones de Algoritmos Evolutivos Multi-Objetivo en empaquetamiento de baterías

En la literatura no se conoce información disponible acerca de trabajos realizados con algoritmos multi-objetivo en el ámbito de los empaquetamientos de baterías, sin embargo si existen trabajos relacionados. En [17] se hace un planteamiento multi-objetivo para la optimización de la forma de una placa de refrigeración para celdas de baterías prismáticas en un banco de un vehículo eléctrico. Se presentan como funciones objetivo la desviación estándar de la temperatura de las celdas, la temperatura promedio de las celdas y la caída de la presión del fluido que circula por cada placa, con estos valores obtenidos a través de simulaciones en un software CAE. Sin embargo, para su resolución se utiliza una suma ponderada de los objetivos y se construye un Frente de Pareto utilizando combinaciones de seis valores para los pesos de cada una de las funciones objetivo normalizadas. La búsqueda del óptimo para cada suma ponderada se realiza a través del método de búsqueda no lineal de MATLAB.

Otra aplicación relacionada fue realizada por NREL (National Renewable Energy Laboratory) en Estados Unidos en 2003 [58]. En este trabajo se utiliza un software de simulación CAE para buscar los valores de la separación entre las celdas, la resistencia interna y el flujo de aire que permiten obtener los valores de las funciones objetivo. Las funciones objetivo son la temperatura máxima, la diferencia de temperatura y la caída de presión dentro del banco y se las analiza a través de un enfoque de optimización “*six-sigma*”. Después de las simulaciones, el resultado son histogramas de los resultados en cada una de las funciones objetivo, que permiten definir las regiones de diseño que cumplen con el esquema six-sigma para los tres criterios y que por tanto representan los valores de las variables que podrían escogerse para lograr la mejor performance térmica del banco.

En [59] también se hace diseño óptimo de un sistema de refrigeración de baterías utilizando como primera aproximación una postura multi-objetivo. Los autores proponen un caso teórico de un banco de baterías de celdas prismáticas para un vehículo eléctrico y buscan optimizar la temperatura promedio del banco y la desviación estándar de la temperatura sobre las celdas. Sus variables son el ancho de las ranuras para colocar las celdas, la separación entre las baterías y la distancia entre las paredes de ventilación y la caja con las baterías. Debido a que sus primeras simulaciones, en los extremos de los valores de cada una de las variables de decisión generan diferencias en la temperatura promedio del banco de solo 0,3°C, deciden optimizar exclusivamente en base a la desviación estándar de la temperatura.

Por otra parte, si se han utilizado algoritmos evolutivos con un solo objetivo para resolver problemas de optimización relacionados con las baterías. En particular en [60], se diseña un sistema de refrigeración de un banco de baterías utilizando algoritmos evolutivos. En ese caso, se cuenta con el diámetro de los ductos de refrigeración del banco y la velocidad del fluido que circula por el ducto como variables y la función objetivo busca maximizar el coeficiente de transferencia de calor unitario para obtener la mejor respuesta térmica posible del *pack*.

En [61], se utiliza la optimización multi-objetivo con algoritmos genéticos para extraer los parámetros de tres baterías de ion-litio (resistencia interna, resistencia parásita y capacidad en un modelo de circuito equivalente) a través de una aproximación de 31 parámetros que se utilizan

para multiplicar funciones del estado de carga (SOC) y la tasa de descarga. Los dos objetivos corresponden a minimizar el error del modelo para los datos de una carga y una descarga de cada batería. Las variables son los 31 parámetros para cada batería.

Capítulo 3

Metodología e Implementación

La herramienta diseñada para la optimización de empaquetamientos de baterías corresponde a un lazo cerrado entre los software MATLAB y COMSOL Multiphysics. El software COMSOL es un programa de análisis de elementos finitos que permite realizar la simulación térmica de los bancos de baterías. En MATLAB se hacen los cálculos numéricos y operaciones propias de algoritmos de optimización multi-objetivo implementados, utilizando los valores de las variables obtenidas en la simulación en COMSOL como funciones objetivo.

3.1. Modelación del Problema

Las fallas que se presentan en un banco de baterías de ion-litio, al igual que en el caso de otros equipos y máquinas, siguen una forma similar a la “curva de la bañera” en el tiempo [6]. Esto significa que las fallas o bien se presentan principalmente en la primera parte de su uso o en la parte final de su vida útil. Las fallas que se presentan en el inicio de la vida útil de un equipo se dan como consecuencia de condiciones de funcionamiento inadecuadas (ya sea por diseño o por utilización por parte de un usuario). Como normalmente estos modos de uso se mantienen en el tiempo, si la falla se presenta, lo hará en los primeros usos del equipo. Por otro lado, las fallas que se presentan cuando el equipo está cerca de cumplir con su vida útil tienen que ver con el envejecimiento de los materiales y componentes del equipo.

En ese sentido, aplicando la situación propuesta a baterías de ion-litio, las fallas se presentan en las mismas dos situaciones: En los primeros ciclos de carga y descarga o bien, causados por la degradación, cerca del final de la vida útil de la batería. Uno de los aspectos más relevantes para determinar si el funcionamiento de un banco de baterías diseñado será adecuado, es el análisis de la temperatura del banco. Si existen celdas o puntos dentro del banco que superen ciertos umbrales de temperatura, existe una alta probabilidad de que exista una falla, que podría ser peligrosa. Particularmente en el caso de las baterías de ion-litio, un aumento brusco de temperatura, incluso localizado, puede llevar a esa celda o incluso a todo el banco a fallar violentamente [6].

Por otra parte, si las celdas se encuentran por sobre su temperatura óptima de operación

esto tiene como consecuencia una baja en la vida útil del banco como se detalló en la sección 2.1.4. Lo mismo sucede cuando se presenta una diferencia de temperatura importante entre las diferentes celdas de un mismo banco, por la aparición de corrientes parásitas por diferentes curvas de descarga. En definitiva, el mantener la temperatura lo más baja posible para todas las celdas y la homogeneidad de temperatura entre las celdas son dos elementos muy importantes que se debe tener en cuenta para el diseño de un banco de baterías si se busca maximizar su vida útil [15].

Luego, las simulaciones térmicas de los bancos de baterías se realizarán con el objetivo de obtener los valores de la temperatura en cada una de las celdas y así poder determinar que soluciones son mejores desde ese punto de vista, pensando en la minimización de las posibles fallas de funcionamiento y maximización de la vida útil. Para analizar el efecto de la degradación y estimar las fallas en la sección final de la vida útil de la batería, se requeriría una simulación a largo plazo, que es impracticable de realizar en un software de simulación térmica por restricciones de tiempo.

Considerando lo anterior, se proponen las siguientes funciones objetivo para la evaluación un banco de baterías en una simulación de corto plazo:

- Minimizar la temperatura máxima entre las N celdas del banco y en todo momento de tiempo:

$$(3.1) \quad \min \max(T_i(t)) \quad \forall i \in 1, 2 \dots N$$

- Minimizar la diferencia de temperatura máxima entre celdas conectadas en el banco:

$$(3.2) \quad \min (\max(T_i(t)) - \min(T_i(t))) \quad \forall i \in 1, 2 \dots N$$

- Por otra parte, para el mismo tipo de celdas de baterías se utiliza como función objetivo la minimización del volumen V de espacio requerido para construir el banco. Este resultado también puede utilizarse como una interpretación de la densidad energética del banco de baterías, la cual se busca maximizar disminuyendo el volumen. Todas las medidas presentadas anteriormente tienen por objetivo ser minimizadas para el funcionamiento óptimo del equipo.

$$(3.3) \quad \min V$$

En los casos que se estudian en este trabajo de título, se utilizarán combinaciones de dichas funciones objetivo para obtener un Frente de Pareto óptimo con las mejores soluciones para el diseño de un banco de baterías (o un módulo) con un número de celdas pre-determinado.

Como variables de diseño, en este trabajo se utilizará la posición y orientación de las celdas en un módulo. El concepto de variación de posición de una celda de batería en un módulo o banco de baterías se muestra en la figura 3.1, en tanto que el cambio de orientación se ilustra en la figura 3.2. La distancia entre las celdas tiene influencia sobre su comportamiento térmico. Del mismo modo, la orientación de los módulos o celdas también tiene influencia en la forma en la cual el flujo de aire se mueve a través del banco de baterías, permitiendo mejor refrigeración de las celdas. En los

casos a estudiar se asume que el tipo de celda ya fue escogido antes de diseñar el módulo, por lo que el número de celdas en el banco ya está definido por las necesidades eléctricas, es conocido y por lo tanto, no es variable del problema. Con lo anterior, el esfuerzo se limita a encontrar las mejores geometrías para lograr la distribución térmica adecuada.

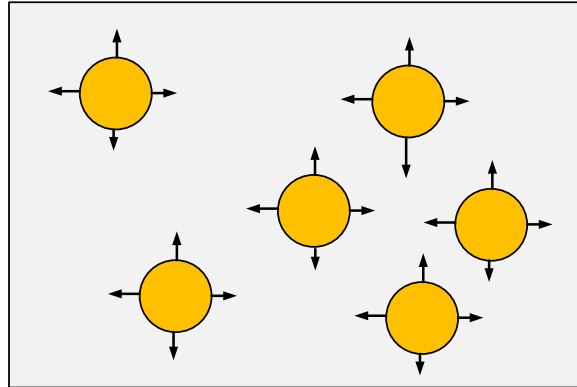


Figura 3.1: Esquema de representación del cambio de posición de las celdas dentro de un módulo o un banco de baterías. Las baterías pueden cambiar su posición libremente dentro del mismo plano.

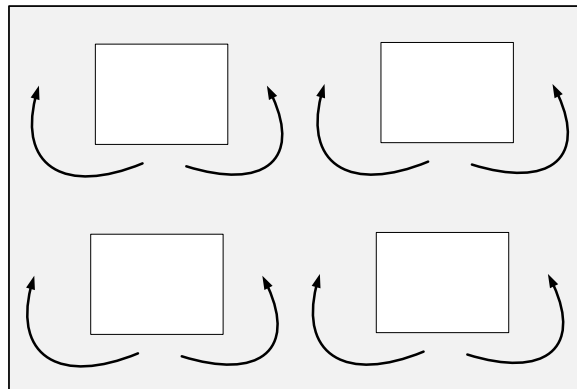


Figura 3.2: Esquema de representación del cambio de orientación de una celda o un módulo. Las baterías pueden cambiar su orientación en ángulos a favor o en contra de las manillas del reloj.

La celda de batería escogida entrega todos los parámetros necesarios para una simulación

térmica: Tipo de geometría, dimensiones físicas, resistencia interna, capacidad nominal, voltaje nominal, corrientes máximas de descarga y carga.

3.2. Implementación de Algoritmos Evolutivos

Para encontrar las mejores soluciones de diseño, se utilizan algoritmos evolutivos multi-objetivo. En este caso, se utilizarán implementaciones de MOPSO (Multi-Objective Particle Swarm Optimization) programadas para este caso en MATLAB. Dentro de la implementación de este algoritmo, se incorpora la comunicación con el software de simulación COMSOL, que permite obtener los valores de las funciones objetivo y asignar las variables correctamente. La elección de MOPSO se debe a su simplicidad en programación, al uso de una población externa (repositorio) que permite extraer diseños Pareto-óptimos en cualquier momento del proceso de simulación y el uso de líderes locales aleatorios que permite evitar caer en mínimos locales, que son probables en problemas de diseño.

3.2.1. Implementación de MOPSO

El algoritmo MOPSO implementado tiene las siguientes etapas:

1. **Crear Partículas:** Inicializar la población, colocando las partículas en posiciones aleatorias respetando las restricciones de espacio.
2. **Inicializar Velocidad:** La velocidad para todas las partículas y en todas las direcciones en el espacio de las variables es cero en la primera iteración.
3. **Evaluar Soluciones:** Se calculan los valores objetivo para cada partícula.
4. **Guardar Soluciones No Dominadas en el Repositorio:** El algoritmo cuenta con un archivo externo a la población llamado Repositorio, donde se almacenan las mejores soluciones encontradas hasta el momento. En el primer paso se guardan todas las alternativas no dominadas. En las siguientes iteraciones, se mezclan las soluciones no dominadas de la nueva población y se mezclan con las que se encontraban en el repositorio, eliminando todas aquellas que hayan dejado de ser óptimas. El repositorio normalmente tiene un tamaño acotado, luego se descartan soluciones en función de su ubicación en los hipercubos de distancia.
5. **Generar Hipercubos:** Para encontrar a los líderes que guían a las partículas, se separan las soluciones en hipercubos en el espacio de los objetivos. Si un hipercubo se encuentra muy poblado, sus elementos tendrán menor probabilidad de ser escogidos como líderes en la siguiente actualización de la velocidad. Los hipercubos se generan dividiendo el espacio de objetivos en partes iguales, en un número de secciones arbitrario.

6. **Actualizar Memoria:** Para cada partícula, se evalúa si la solución actual es mejor que la anterior mejor posición lograda hasta el momento. Si es así, se guarda la nueva posición de la partícula (variables) en la memoria.
7. **Calcular Velocidad:** La velocidad de cada partícula se calcula con una expresión que combina la velocidad anterior, la diferencia entre la mejor posición de la partícula y su posición actual y la diferencia entre una partícula líder del repositorio y la posición de la partícula. Dicha expresión puede ser observada en la ecuación 3.4 para cada partícula i , con la partícula líder representada por $L_i(t - 1)$, la mejor posición lograda representada por $M_i(t - 1)$ y la posición actual por $P_i(t - 1)$. La partícula líder se escoge aleatoriamente en base a los hipercubos, ambas deben ser del mismo hipercubo o se escoge aleatoriamente. Cada diferencia es ponderada por una variable aleatoria que va entre 0 y 1 (α y β en la ecuación 3.4, respectivamente). La velocidad anterior es ponderada por un parámetro de inercia (γ).

$$(3.4) \quad vel_i(t) = \gamma \cdot vel_i(t - 1) + \alpha \cdot (M_i(t - 1) - P_i(t - 1)) + \beta \cdot (L_i(t - 1) - P_i(t - 1))$$

8. **Actualizar Posición de las Partículas:** Se suma la velocidad y la posición actual de la partícula para obtener la nuevas variables. Con estas, se puede volver a iniciar el ciclo, calculando los valores objetivo. El algoritmo se detiene cuando se alcanza un número máximo de iteraciones o se alcanza una condición de avance mínima entre las soluciones del repositorio.

$$(3.5) \quad P_i(t) = P_i(t - 1) + vel_i(t)$$

Las etapas del algoritmo pueden verse también en la figura 3.3.

El algoritmo MOPSO tiene una serie de parámetros relevantes y cuya forma de elección se detalla a continuación:

- **Cantidad de partículas:** Este concepto es equivalente al tamaño de la población en un algoritmo genético. Ante mayor número de partículas, mayor capacidad de exploración tiene la población en cada generación y por lo mismo debe ser proporcional al espacio de búsqueda. La elección del tamaño de la población es crítica porque un número bajo podría llevar al algoritmo a converger hacia mínimos locales, por falta de varianza y un número muy alto hace que el costo computacional del algoritmo haga imposible su convergencia en tiempos razonables. La elección del tamaño de la población se hará dependiendo de cada problema a evaluar, considerando el tamaño del espacio de exploración.
- **Número de ciclos:** Este parámetro normalmente es inversamente proporcional al número de partículas y corresponde a la cantidad de generaciones diferentes que se evaluará. Al igual que la cantidad de partículas, se decidirá el número de ciclos o generaciones para cada problema a estudiar, considerando el tamaño del espacio de exploración.

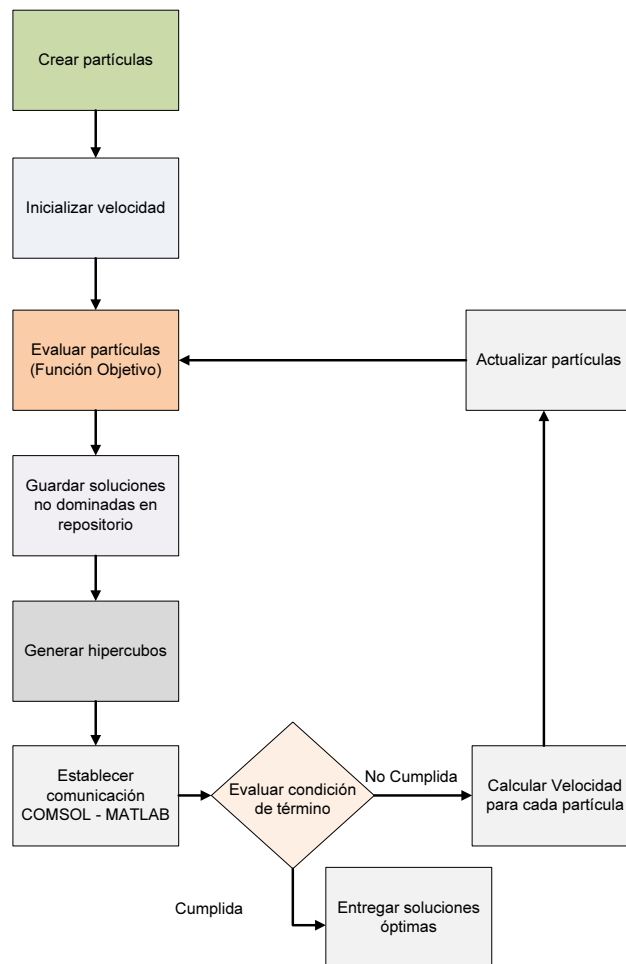


Figura 3.3: Etapas del algoritmo MOPSO implementado en MATLAB.

- **Número de divisiones para hipercubos:** Permite determinar el número de hipercubos que se utilizará para cada generación y que sirven para encontrar la partícula líder que determina la velocidad de cada partícula. A mayor cantidad de divisiones, se requiere mayor tiempo para generar los hipercubos y luego para asignar cada solución en el repositorio a una división. En [62] se recomiendan valores entre 30 y 50 divisiones por dimensión en el espacio de los objetivos. En este trabajo se utilizan 50 divisiones por dimensión.
- **Tamaño máximo del repositorio:** Este valor tiene por objetivo limitar el número de soluciones no dominadas que se almacena en el archivo. Mientras más alto el valor del tamaño máximo del repositorio, teóricamente es mejor es la calidad del Frente de Pareto producido ya que disminuye la posibilidad de tener que eliminar soluciones óptimas si es que el Frente llegara a crecer muy rápidamente. En este caso, se buscará en el diseño de empaquetamientos no tener un número excesivo de soluciones, por lo cual se escoge 50 como tamaño máximo del repositorio.
- **Inercia:** Corresponde al parámetro que multiplica a la velocidad en la generación anterior

para cada partícula. La inercia se puede programar de forma adaptativa o mantener un valor fijo. En el caso de este trabajo se utilizará el valor recomendado en [62], igual a 0,4.

3.3. Diagrama de bloques de la implementación

En la figura 3.4 se muestra un diagrama de bloques del sistema implementado y cuyo objetivo es obtener soluciones óptimas para un problema de empaquetamiento de baterías a través de un algoritmo evolutivo y utilizando un software de simulación física, en este caso COMSOL.

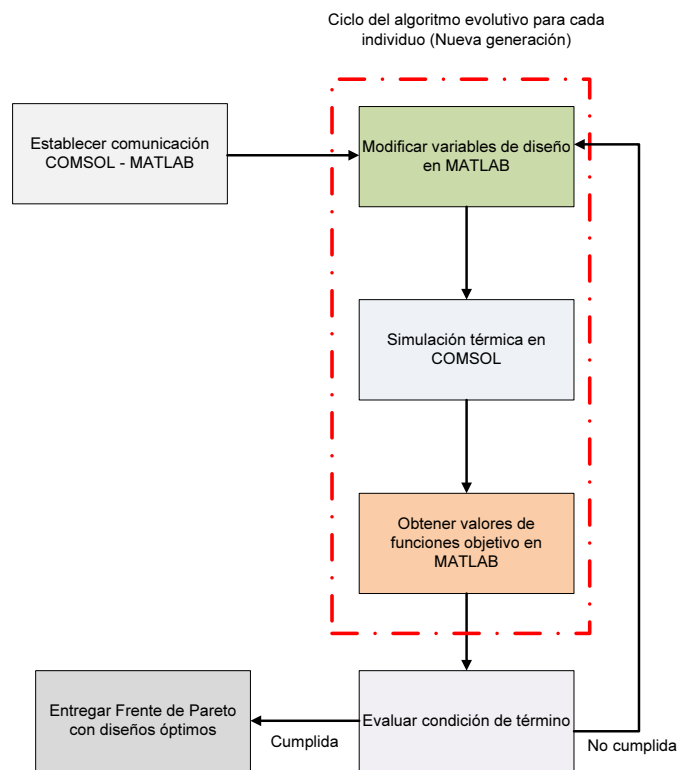


Figura 3.4: Etapas de la herramienta implementada en este trabajo usando en COMSOL y MATLAB para la optimización multi-objetivo de módulos y bancos de baterías.

El sistema implementado comienza su funcionamiento con la conexión de MATLAB con COMSOL a través de un diálogo tipo cliente-servidor, de forma que MATLAB tenga control de los parámetros de las simulaciones en COMSOL y pueda rescatar los resultados de estas. Luego,

se ingresa en el ciclo de optimización dentro del algoritmo evolutivo utilizando los pasos propios de las implementaciones indicadas en la sección 3.2.

En cada ciclo, se modifican las variables de diseño de cada individuo en MATLAB y luego, para este nuevo diseño se evalúa su comportamiento térmico con una simulación en COMSOL, que permite obtener los valores de las funciones objetivo a través de la temperatura de cada celda para el respectivo diseño. Estas funciones objetivo permiten determinar el Frente de Pareto óptimo dentro de los individuos en cada generación, hasta que se alcance una condición de término. Esta condición corresponde a un número máximo de iteraciones o a un criterio de detención por mínimo avance entre un número de generaciones diferentes.

Al término del proceso, se cuenta con la última generación de diseños y el Frente de Pareto óptimo, con los cuales se puede escoger los mejores diseños o patrones para construir los bancos de baterías.

3.4. Modelación térmica de empaquetamientos de baterías en COMSOL Multiphysics

La modelación corresponde a construir la geometría de un empaquetamiento de baterías, entregarle materiales a los elementos, definir las condiciones de borde y características de la generación y transmisión de calor entre los cuerpos participantes. También se define la forma en que se obtendrán los resultados de las simulaciones y los parámetros que podrían ser modificados en un proceso de optimización. COMSOL Multiphysics integra una serie de módulos para realizar simulaciones en múltiples físicas. En el caso de las simulaciones de módulos de baterías en un espacio que se requieren para este trabajo es necesario utilizar el módulo de transferencia de calor conjugada (*Conjugated Heat Transfer* dentro del programa). Este módulo tiene por objeto combinar ecuaciones de transferencia de calor con un flujo laminar o turbulento. El módulo de transferencia de calor conjugado está regido por relaciones termodinámicas y la ley de Fourier de transferencia de calor.

En COMSOL se simula la interacción entre sólidos, en este caso correspondientes a las celdas de baterías de ion-litio y el aire en el espacio en el cual se encuentran confinadas a través de la transferencia de calor, por convección, conducción y radiación. Para las simulaciones, se utiliza el equivalente de dos dimensiones de la situación real, asumiendo simetría en profundidad.

Las simulaciones se programan considerando los siguientes supuestos:

- Cada celda representa una fuente de calor independiente, con un calor generado dictado por su resistencia interna y una corriente de carga o descarga. El valor de la generación de calor queda determinado a través del Efecto Joule.
- Las celdas comienzan la simulación a una temperatura inicial determinada.
- Dado que se utiliza el equivalente de dos dimensiones, el confinamiento es rectangular, por lo que existen a lo menos cuatro fronteras con el exterior. Estas fronteras requieren

de condiciones de borde adecuadas para que la simulación pueda converger y entregar resultados consistentes: En una de las caras del rectángulo, se utiliza como condición de borde la presencia de ventilación forzada para llevar las celdas a un equilibrio térmico. Esta ventilación corresponde a una entrada de aire, que inserta el fluido a una temperatura determinada, a una velocidad constante en $\frac{m}{s}$. La entrada de ventilación puede ocupar toda la cara o una sección de ella. En al menos una de las caras se debe colocar una salida del aire caliente, con condición de borde equivalente a presión cero. Al igual que en el caso de las entradas de aire, la apertura puede ocupar una sección de una cara o la cara completa. En las caras o secciones que no corresponden a entradas o salidas de aire, se utiliza como condición de borde que son fronteras adiabáticas, equivalente al peor caso en el cual no hay transferencia de calor entre el interior y el exterior del módulo.

- La temperatura de la celda se calcula como el promedio de temperatura en toda la celda. Luego, la temperatura máxima del módulo o banco de baterías corresponde a la temperatura promedio de la celda con mayor temperatura y la diferencia de temperatura máxima se mide restando los valores máximos y mínimos de temperatura promedio de las celdas. Ambos valores se calculan para todo instante de tiempo.

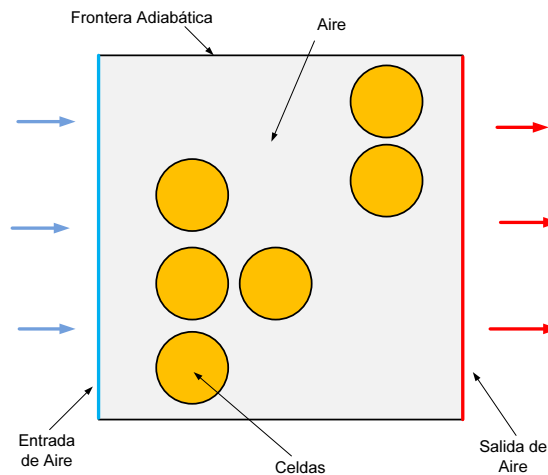


Figura 3.5: Esquema de representación de condiciones de borde y fuentes de calor en COMSOL

Las simulaciones térmicas en COMSOL pueden realizarse tanto dependientes del tiempo (“*Time Dependent*” dentro del programa) como en el modo estacionario (“*Stationary*”). En las simulaciones dependientes del tiempo, se registra un valor de las variables de salida cada cierto periodo de tiempo fijo y determinado por el usuario. Los pasos de tiempo del programa de elementos finitos no necesariamente coinciden con los definidos por el usuario porque son adaptativos y dependen del error que se comete en cada iteración del programa. Los datos son interpolados por el software de simulación, revisando que el error esté acotado. La simulación termina cuando se ha cumplido un tiempo total definido por el operador.

En el caso de las simulaciones estacionarias, COMSOL, al igual que otros software de simulación de elementos finitos utiliza una unidad de pseudo-tiempo por lo tanto no entrega ninguna información temporal durante la corrida. El resultado de la simulación es solamente el valor de las variables de salida, una vez que el sistema se ha estabilizado completamente, esto es, que la diferencia entre las variables entre un paso y otro sea menor que un cierto margen definido por el usuario. Al mismo tiempo, las variables del sistema no varían en función del tiempo. Las simulaciones estacionarias se utilizarán para determinar la situación de equilibrio térmico para un cierto banco de baterías.

3.5. Evaluación del modelo térmico para empaquetamientos de baterías

Para determinar la consistencia de las condiciones de borde y la validez de las simulaciones se estudió la convergencia para casos de transferencia de calor en COMSOL y se realizó un análisis de mado para geometrías similares a las que se desea evaluar para evitar errores de consistencia numéricos. En estos casos se utilizaron iguales fuentes de calor (constante e igual $140.000 \frac{W}{m^3}$) y una velocidad de entrada del aire de $0,5 \frac{m}{s}$.

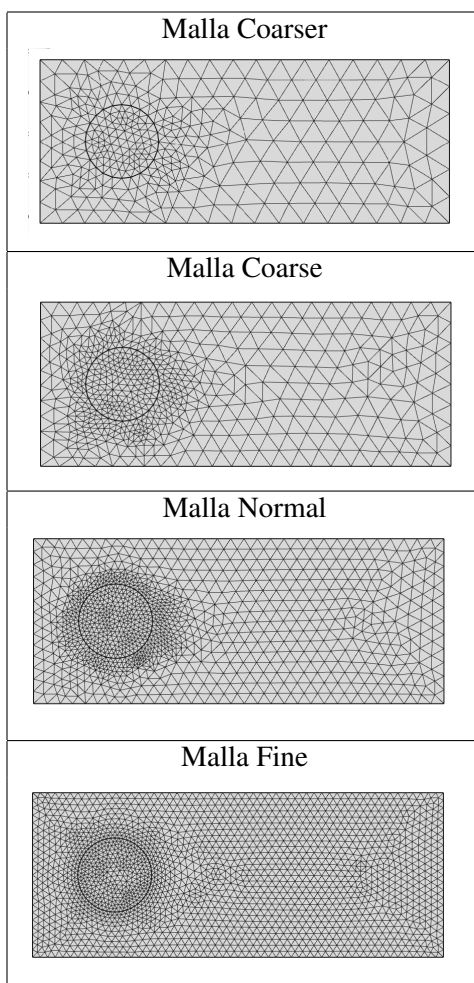
Las simulaciones de comprobación y de análisis de malla se realizaron para celdas de ion-litio 18650. El análisis de malla consiste en evaluar diferentes tamaños de los elementos de la malla para construir los puntos donde se evaluarán las ecuaciones de transferencia de calor en cada instante de tiempo. Una malla muy gruesa lleva a que los resultados de la simulación se alejen de la realidad, pero por otra parte, una malla muy fina (con elementos muy cerca) requerirá de mayor tiempo de cómputo. Las mallas construídas utilizan una configuración por defecto para el aire alrededor de una celda 18650 con un nivel de prioridad extra para mayor detalle a la frontera de la celda, donde existe mayor diferencia de temperatura y por lo mismo se requiere mayor precisión en los cálculos. Las configuraciones probadas corresponden a cuatro de los niveles entregados por COMSOL, Coarser (Muy Gruesa), Coarse (Gruesa), Normal y Fine (Fina). En base a las diferencias y tiempo utilizado, se opta por usar la malla Normal. Los resultados y tamaños de las mallas se muestran en la tabla 3.1. Las mallas se muestran graficamente en la tabla 3.2.

Tabla 3.1: Tamaños y tipos de mallas evaluados para las simulaciones en COMSOL

Malla	Nº de elementos	Temperatura media de la celda	Tiempo de Simulación
Coarser	340	36,95 °C	17 segundos
Coarse	1124	36,21 °C	18 segundos
Normal	2196	35,84 °C	22 segundos
Fine	3840	35,79 °C	34 segundos

los resultados fueron comparados con aquellos obtenidos experimentalmente en [63], encontrando que el aumento de temperatura tiene una diferencia menor a 1°C entre el caso simulado ($35,84^{\circ}\text{C}$) y el experimental ($34,92^{\circ}\text{C}$) para condiciones similares, con una generación de calor equivalente y una celda 18650 del mismo tipo. Al mismo tiempo, se realizaron comparaciones con simulaciones realizadas en ANSYS Workbench, en las cuales se pudo comprobar la similitud

Tabla 3.2: Mallas evaluadas para las simulaciones en COMSOL



de las respuestas en temperatura y velocidad del aire para las mismas condiciones de borde. Los resultados de la simulación se muestran en la Figura 3.8 para la temperatura de la celda y en la Figura 3.9 para la velocidad del aire.

Otro de los objetivos correspondía a evaluar la diferencia entre una simulación en tres dimensiones y una en dos dimensiones, ya que estas últimas toman menor tiempo en procesarse. La diferencia encontrada en la simulación corresponde a $0,13^{\circ}\text{C}$ más en el caso de la simulación 3D transcurridos 600 segundos con la misma fuente de generación de calor para el caso de una celda. La diferencia anterior se considera despreciable con respecto al aumento de temperatura de $10,8^{\circ}\text{C}$ de la celda en ambos casos. En cuanto a tiempo de simulación, en 2 dimensiones la simulación tarda 28 segundos contra 440 segundos en el caso 3D. Esto se debe a que la malla para el mismo caso evaluado en dos dimensiones requiere de 101594 elementos. Parte de esta malla se muestra en la figura 3.6. Por los resultados de este análisis se escoge la configuración de dos dimensiones con malla "Normal" de 2196 elementos por celda para las simulaciones de los casos de estudio de este trabajo. Para celdas de geometría prismática se utiliza la misma regla con malla "Normal" usando

un nivel extra de prioridad para los bordes e interior de las celdas o módulos. La malla se muestra en la figura 3.7.

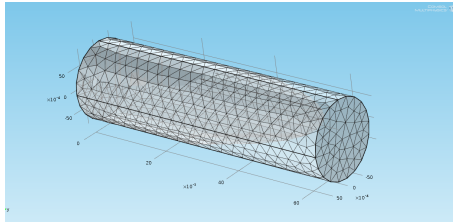


Figura 3.6: Malla utilizada para la simulación en tres dimensiones en COMSOL de una celda 18650.

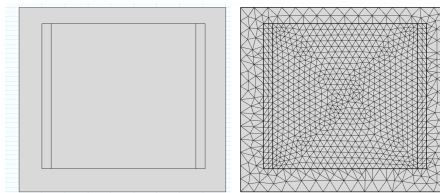


Figura 3.7: Malla escogida para celdas prismáticas, siguiendo las reglas presentadas en esta sección para celdas cilíndricas.

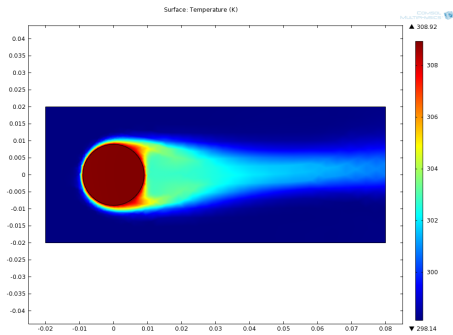


Figura 3.8: Respuesta en temperatura para la simulación de una celda 18650, con calor constante $140.000 \frac{W}{m^3}$ y una velocidad de entrada del aire de $0,5 \frac{m}{s}$. Captura en instante final de tiempo. Captura en instante final de tiempo.

3.6. Datos de las Simulaciones

3.6.1. Celdas de baterías

Como se indicó en la sección 3.1, se asume que las necesidades eléctricas de los bancos y módulos de baterías han determinado el número de celdas requeridas, por lo tanto esto no es variable del problema. En el caso de este trabajo, para los estudios de celdas cilíndricas se

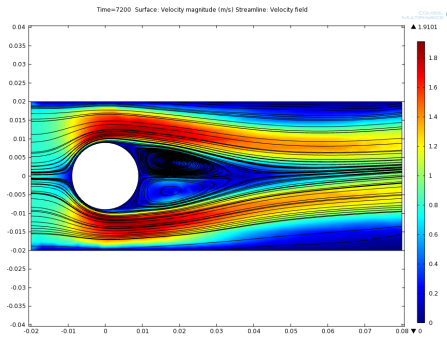


Figura 3.9: Respuesta en velocidad para la simulación de una celda 18650, con calor constante $140.000 \frac{W}{m^3}$ y una velocidad de entrada del aire de $0,5 \frac{m}{s}$.

utilizará la celda de ion-litio 18650 fabricada por Molicel, similar a la que se muestra en la figura 3.10 y cuyas características [64] se encuentran en la tabla 3.3:



Figura 3.10: Celda cilíndrica de litio 18650

Tabla 3.3: Características de la celda 18650 utilizada en este trabajo.

Material	Litio Ion Manganeso
Voltaje Nominal	3,8 V
Capacidad Nominal	1400 mAh
Largo	65 mm
Diámetro	18,24 mm
Resistencia Interna Máxima	85 m Ω
Resistencia Interna Promedio	32 m Ω
Corriente Máxima de Descarga	20 A

La celda 18650 es una de las más utilizadas en la industria de los computadores portátiles, además ha sido utilizada para empaquetamientos de baterías en sistemas eléctricos de potencia y en vehículos eléctricos, particularmente en el modelo Roadster de Tesla [65]. La celda define parámetros relevantes de la simulación, como son el tamaño de los cuerpos a simular térmicamente y la generación de calor de cada uno de ellos.

Otra de las geometrías que se estudiarán para otro tipo de empaquetamiento de baterías, corresponde a celdas prismáticas. En este caso, se utilizará la celda SPB636395 fabricada por Leo Industries cuyas características [66] se muestran en la tabla 3.4. Esta celda fue utilizada en el vehículo solar Eolian desarrollado dentro de la Universidad [67]. La celda SPB636395 se muestra

en la figura 3.11.

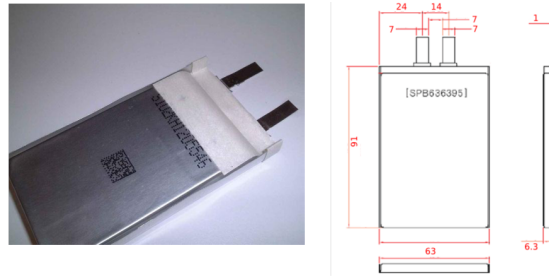


Figura 3.11: Celda prismática de litio SPB636395

Tabla 3.4: Características celda SPB636395

Material	Litio Ion Polímero
Voltaje	3,7 V
Capacidad Nominal	4500 mAh
Largo	63 mm
Ancho	6.3 mm
Alto	95 mm
Resistencia Interna Máxima	50 $m\Omega$
Resistencia Interna Promedio	15 $m\Omega$

3.6.2. Estimación de la resistencia interna de las baterías

La resistencia interna de una batería de ion-litio, en general no es constante durante la carga y descarga de la batería, ya que varía con el estado de carga de la celda (SoC, state of charge), con el estado de salud (SoH, state of health) y con la temperatura. Para evaluar estos efectos se utiliza el Modelo de Copetti [9] [10].

Este modelo detallado en la sección 2.1.3 permite estimar el voltaje de una celda o banco de baterías a partir de una expresión dependiente del SoC, la corriente, la temperatura y la capacidad nominal para las diferentes operaciones que se realizan con una batería. En particular, en este caso de estudio es relevante estudiar la descarga de la batería, para la cual la expresión del modelo de Copetti para la Resistencia Interna se deduce de la ecuación 2.3 [9], asumiendo capacidad constante:

$$(3.6) \quad R_{in} = \frac{1}{C_{nom}} \cdot \left(\frac{P_{1d}}{1 + |I(t)|^{P_{2d}}} + \frac{P_{3d}}{SoC(t)^{P_{4d}}} + P_{5d} \right) \cdot (1 - q_d \cdot \Delta T(t))$$

La resistencia interna corresponde al factor que multiplica a la corriente $I(t)$ en la ecuación anterior. La obtención de los parámetros de la resistencia interna según el Modelo de Copetti para

una celda en particular se realiza utilizando curvas de descarga y aproximando los parámetros con un método de regresión [9]. En este caso se utilizó el método de mínimos cuadrados, implementado a través de MATLAB.

Los parámetros obtenidos para las celdas se muestran en la tabla 3.5 para la celda 18650 y en la tabla 3.6 para la celda SPB636395. Con lo anterior, la resistencia interna de la celda 18650 se encuentra entre $26,9 \text{ m}\Omega$ y $82,4 \text{ m}\Omega$ y la resistencia para la celda SPB636395 se encuentra entre $14,5 \text{ m}\Omega$ y $52 \text{ m}\Omega$. Los valores son consistentes con lo entregado en las diferentes hojas de datos entregadas por los fabricantes de estas celdas.

Tabla 3.5: Valores estimados de los parámetros del Modelo de Copetti para la celda 18650.

Parámetro	Valor
P_{1d}	0,003
P_{2d}	1,5
P_{3d}	0,0048
P_{4d}	1,1
P_{5d}	0,021
q_d	0,005

Tabla 3.6: Valores estimados de los parámetros del Modelo de Copetti para la celda SPB636395

Parámetro	Valor
P_{1d}	0,002
P_{2d}	1,4
P_{3d}	0,0033
P_{4d}	1,5
P_{5d}	0,01126
q_d	0,005

Como referencia, en la figura 3.12 se muestra la curva de descarga utilizada para estimar los parámetros para la celda 18650 y en la figura 3.13 se muestra el valor de la resistencia interna en función del SOC para esta misma celda.

En las simulaciones se asumirá que el estado de salud (SoH) es constante e igual a 1 y solo se tomarán en cuenta las modificaciones de corriente, SOC y temperatura.

3.6.3. Velocidad del aire

Trabajos realizados en la industria automotriz que han determinado el valor del flujo de aire que entregan ventiladores de convección forzada en los bancos de baterías para vehículos eléctricos e híbridos. Estos han determinado que el valor normalmente llega hasta los $48,4 \text{ SCFM}$ (*Standard Cubic Feet per Minute*, que equivalen a $0,02284 \frac{\text{m}^3}{\text{s}}$ de flujo volumétrico de aire) [27]. Si se divide

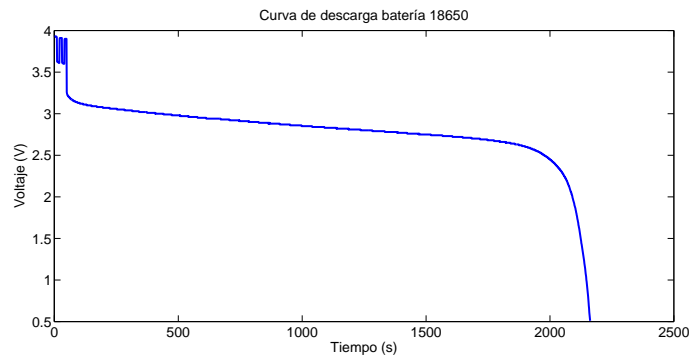


Figura 3.12: Curva de descarga utilizada para la estimación de parámetros de la celda 18650.

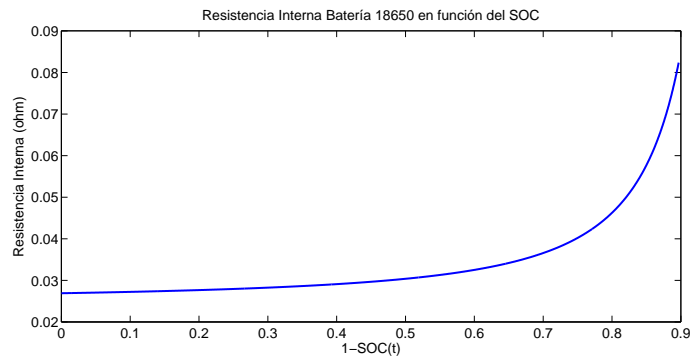


Figura 3.13: Resistencia interna en función del Estado de Carga (SOC) de la batería 18650 de ion-litio.

dicho valor por el área de entrada del ventilador, entonces se puede obtener el valor de la velocidad del aire.

Otra forma de definir la condición de borde para el flujo de entrada de aire corresponde a utilizar diferencias de presión. En ese caso se utilizan presiones de entrada y de salida del aire, que permiten el flujo de aire a través de un banco de baterías. En base a evaluaciones hechas en el diseño del vehículo solar Eolian II, se utilizarán presiones de entrada de 47 Pascal y presiones de salida de 4,9 Pascal.

3.6.4. Generación de Calor

El calor generado por una batería suele modelarse con el Efecto Joule que depende directamente de la corriente y la resistencia interna de la batería, de la siguiente forma:

$$Q = i^2 \cdot R_{in} [W]$$

Conocida la resistencia interna de la celda, con un valor de corriente, puede obtenerse el valor del calor generado Q en Watt. La magnitud de calor generado por una fuente se expresa en la unidad $\frac{W}{m^3}$ para una generación volumétrica de calor. Para obtener dicho valor, se debe utilizar el volumen de la fuente, dividiendo el calor total generado por Efecto Joule por el volumen obtenido. Para la celda 18650, el volumen corresponde a $1,654 \cdot 10^{-5} m^3$.

En general, una batería de un vehículo eléctrico o un sistema eléctrico de potencia se verán controlados por necesidades variables de corriente dependiendo de las exigencias de la ruta o del sistema al cual proveen energía, luego es más realista utilizar una generación de calor variable, que un caso fijo, aunque este sea el peor caso posible.

En el caso de los vehículos, existen perfiles de uso desarrollados por entidades internacionales para probar la autonomía y emisiones de estos. Los perfiles corresponden a valores de velocidad en función del tiempo, durante un intervalo acotado. Estas pruebas fueron desarrolladas para vehículos de combustión interna, pero representan formas de usar un automóvil cualquiera en una ciudad o carretera, por lo que las mismas pruebas también se han utilizado para comprobar la respuesta de las baterías ante exigencias de aceleración y velocidad. Para este estudio en los casos en que se hagan simulaciones dependientes del tiempo se utilizará la prueba US06 o *Supplemental Federal Test Procedure* (SFTP) diseñada por la *Environmental Protection Agency* (EPA) [68] para representar un estilo de conducción agresivo, que utiliza aceleraciones bruscas y velocidades altas de circulación. El ciclo representa una ruta de 12,8 kilómetros recorrida en 600 segundos y puede observarse en las figuras 3.14 y 3.15. La utilidad de un perfil como el US06 para probar el comportamiento térmico de bancos de baterías corresponde a que durante los 600 segundos se presentan períodos de exigente uso de las baterías, por lo tanto, existe un calentamiento por Efecto Joule, como también hay períodos de desaceleración, en los cuales se puede medir el efecto de la ventilación sobre el banco. El ciclo US06 se ha utilizado frecuentemente en trabajos similares de comportamiento térmico de baterías para vehículos eléctricos [4] [23] [25].

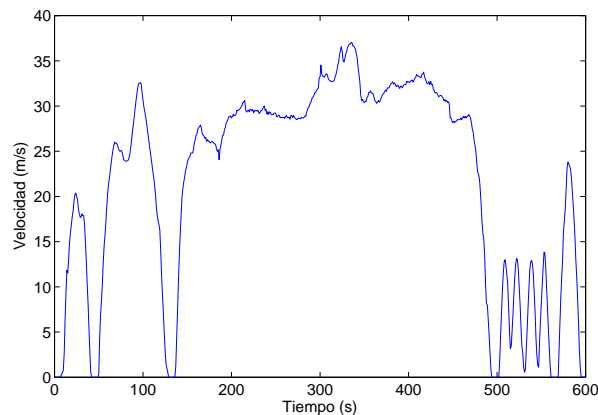


Figura 3.14: Perfil de conducción US06 (Velocidad en $\frac{m}{s}$ en función del tiempo)

La potencia eléctrica exigida al banco de baterías y la aceleración del vehículo son proporcionales. Calculando el voltaje de una celda en función de los datos entregados por los fabricantes de la celda para el estado de carga, entonces se puede hacer la división de potencia

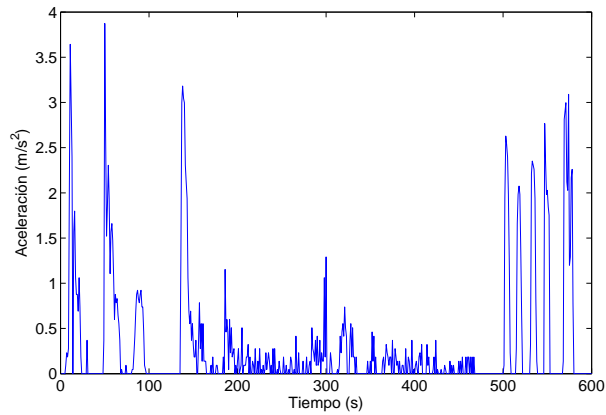


Figura 3.15: Perfil de conducción US06 (Aceleración en $\frac{m}{s^2}$ en función del tiempo)

y voltaje, para obtener que la corriente es proporcional a la aceleración. Si bien el valor de la constante de proporcionalidad exacta dependerá de la configuración del banco y de la masa y el motor eléctrico que utilice el vehículo se hace el supuesto de que la corriente máxima de descarga corresponderá a la máxima aceleración del automóvil. Dado que la corriente máxima de descarga está determinada por el tipo de celda utilizado, entonces puede hacerse la transformación directa de aceleración a corriente y así a calor generado por cada celda. El perfil de corriente entonces queda representado por lo que se muestra en la figura 3.16 para una celda con corriente máxima igual a 20 Amperes.

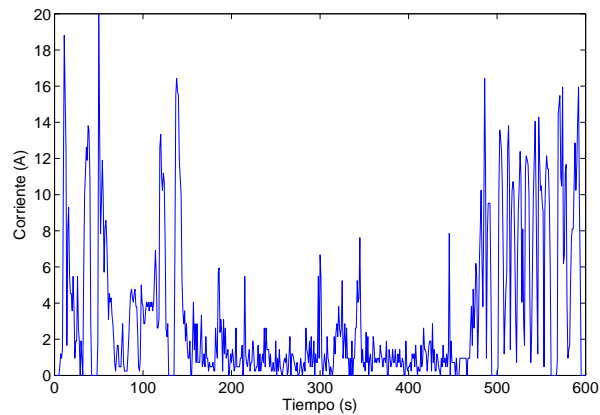


Figura 3.16: Corriente en función del tiempo para el perfil de conducción US06, tomando como corriente máxima 20 Amperes.

3.6.5. Tiempo de Simulación

Dado que el ciclo US06 tiene datos para 600 segundos, las simulaciones dependientes del tiempo se realizarán durante ese período. Los resultados se extraen cada un segundo de simulación para el posterior análisis de la temperatura en los bancos.

Las simulaciones estacionarias se realizan hasta que el software alcance un nivel de error inferior a una cierta cota para las variables que está estimando a través de la resolución de los sistemas de ecuaciones en derivadas parciales que utilizan los métodos de elementos finitos. Para los casos a simular en COMSOL, este valor se ajusta en 10^{-3} para presión, velocidad y temperatura, en la unidad correspondiente a cada magnitud.

3.6.6. Efecto del uso de la perfil US06 en generación de calor

Si bien el perfil de conducción US06 es utilizado en pruebas reales para automóviles con motores de combustión interna y eléctricos, el hecho de incluirla como parte del fitness de un algoritmo evolutivo podría llevar a un efecto de sobre-entrenamiento, en el cual los diseños escogidos sean aptos para responder ante esta curva de generación de calor, pero que realmente no sean buenos diseños para otro tipo forzamientos en potencia. Para comprobar si esta situación es posible utilizando el perfil US06, se diseña una prueba para evaluar el efecto de diferentes perfiles aleatorios, derivados del perfil US06. Se separa el perfil US06 en seis segmentos de 100 segundos y se generan aleatoriamente 9 perfiles con diferentes órdenes para estos trozos. Luego, se realiza una simulación con cada uno de estos dos perfiles para dos geometrías de seis celdas de batería 18650 con el objetivo de observar si se mantiene la dominancia de una sobre otra, en temperatura máxima y diferencia de temperatura, con el cambio del forzamiento eléctrico. Las geometrías se muestran en las figuras 3.17 y 3.18, respectivamente.

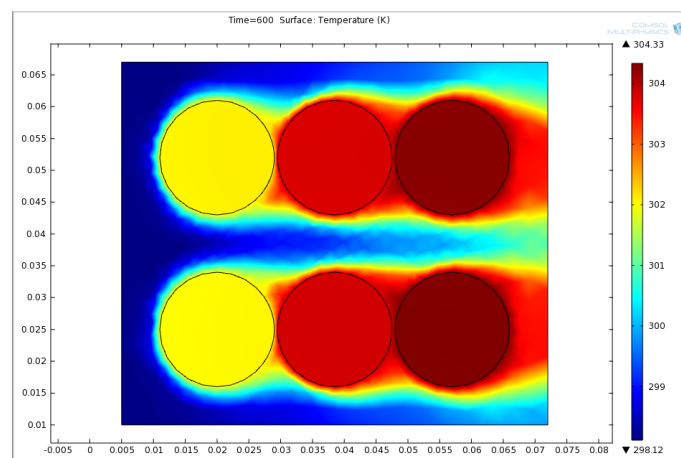


Figura 3.17: Geometría N° 1 utilizada en la prueba para el perfil US06.

Los resultados de la prueba se muestran en la tabla 3.8 para el valor de la temperatura máxima y en la tabla 3.9 para la diferencia de temperatura, además de gráficamente en las figuras 3.19 e

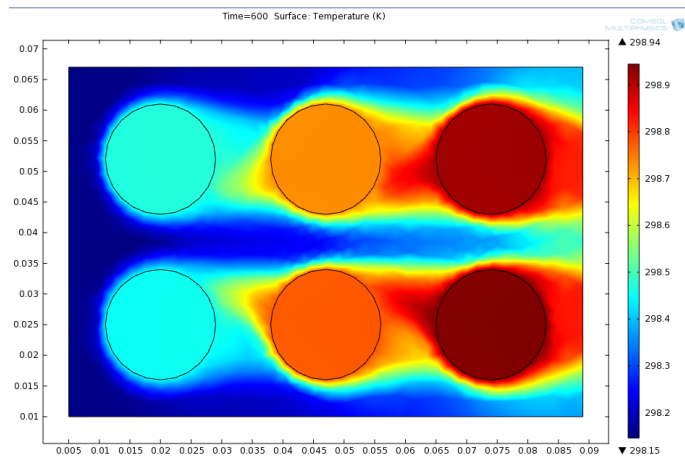


Figura 3.18: Geometría N°2 utilizada en la prueba para el perfil US06.

3.20. La evaluación N°1 corresponde al uso de la curva US06, en tanto que las nueve siguientes representan las combinaciones aleatorias de segmentos de esta. El orden de los segmentos de 100 segundos se encuentra en la tabla 3.7. Como puede observarse en las tablas de resultados, siete de las nueve veces, el orden se mantiene, esto es la temperatura máxima y la diferencia de temperatura son menores en el caso de la geometría N°1 con respecto a la geometría N°2, respetando el resultado que se obtiene usando el perfil US06. Existen dos excepciones, en el caso de la prueba N°8 ambas funciones objetivo favorecen a la geometría N°2 y en la prueba N°9, la diferencia de temperatura es menor en el caso de la geometría N°2. Los resultados muestran que si bien existe la posibilidad de que el ordenamiento cambie si se utiliza un perfil diferente al US06, el ciclo real, equivalente al que se usa en pruebas reales de vehículos eléctricos, es consistente en la mayoría de las ocasiones, aún haciendo cambios importantes en su forma. Luego, el perfil US06 permite hacer una comparación entre dos diseños de módulos o bancos de baterías pensados para su uso en un vehículo eléctrico, sin forzar necesariamente sobre-entrenamiento o sesgo sobre los resultados.

Tabla 3.7: Orden de los tramos del perfil US06 utilizados en las 10 pruebas realizadas para determinar posible sesgo en los resultados de la optimización, utilizando esta curva. Los tramos son numerados como 1 para 0 a 100 segundos, 2 entre 100 y 200 segundos y así sucesivamente.

Combinaciones	1º Tramo	2º Tramo	3º Tramo	4º Tramo	5º Tramo	6º Tramo
1	1	2	3	4	5	6
2	4	3	5	1	6	2
3	4	3	2	1	6	5
4	2	6	1	3	4	5
5	4	2	3	1	6	5
6	3	5	4	2	1	6
7	5	3	4	6	1	2
8	4	5	1	2	6	3
9	5	2	6	1	4	3
10	1	2	4	6	5	3

Tabla 3.8: Resultados de la prueba para diferentes combinaciones de tramos del perfil de uso US06 para la temperatura máxima en las celdas. Nueve de los diez casos presentan el mismo resultado, favoreciendo a la Geometría N°2.

Prueba	Temperatura Máxima Geometría 1 (°C)	Temperatura Máxima Geometría 2 (°C)	Diferencia (°C)
1	39,137	28,928	10,209
2	33,690	33,445	0,246
3	34,441	34,157	0,284
4	32,541	30,296	2,245
5	34,714	34,406	0,308
6	31,927	29,155	2,772
7	29,401	26,412	2,990
8	35,865	38,221	-2,357
9	34,881	28,145	6,736
10	31,168	30,213	0,956

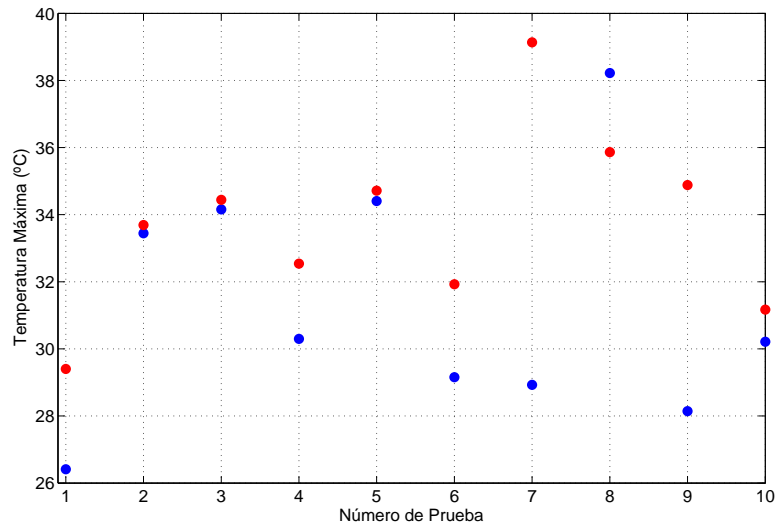


Figura 3.19: Resultados de la prueba para diferentes combinaciones de tramos del perfil de uso US06 para la diferencia de temperatura máxima. En rojo se muestran los resultados para la Geometría N°1 y en azul para la Geometría N°2.

Tabla 3.9: Resultados de la prueba para diferentes combinaciones de tramos del perfil de uso US06 para la diferencia de temperatura máxima. Ocho de los diez casos presentan la misma información, favoreciendo a la Geometría N°2.

Prueba	Diferencia de Temperatura Máxima Geometría 1 (°C)	Diferencia de Temperatura Máxima Geometría 2 (°C)	Diferencia (°C)
1	5,120	1,655	3,465
2	3,423	3,243	0,180
3	3,796	3,593	0,203
4	3,062	1,915	1,147
5	3,957	3,742	0,215
6	2,595	1,684	0,911
7	1,434	0,483	0,951
8	4,070	5,513	-1,443
9	1,218	3,467	-2,249
10	2,414	2,126	0,288

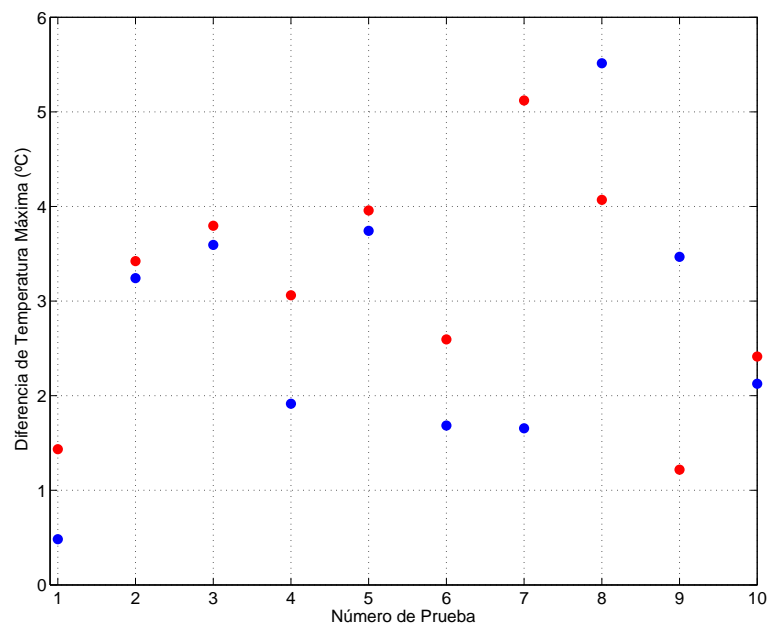


Figura 3.20: Resultados de la prueba para diferentes combinaciones de tramos del perfil de uso US06 para la diferencia de temperatura máxima. En rojo se muestran los resultados para la Geometría N°1 y en azul para la Geometría N°2.

Capítulo 4

Resultados

4.1. Caso 1: Posicionamiento óptimo de seis celdas cilíndricas 18650 en un módulo

4.1.1. Descripción del caso de estudio

El objetivo de este caso de estudio es encontrar patrones de seis celdas cilíndricas 18650 que permitan construir módulos de baterías, de forma que tengan el mejor comportamiento térmico ante un forzamiento similar al que tendrían al ser utilizadas para entregar energía a un vehículo eléctrico. La búsqueda de patrones térmicamente adecuados entrega información relevante sobre el efecto de la distancia entre las celdas en la respuesta en temperatura y permite encontrar geometrías nuevas, que no son necesariamente intuitivas o utilizadas en la actualidad.

Las variables son las posiciones de cada una de las seis celdas en el plano XY, con una restricción de tamaño, tal que el alto del módulo no pueda exceder 6 centímetros ni tampoco 12 centímetros de largo. Se asume que todas las celdas son ubicadas en la misma posición en profundidad, lo que permite simular los módulos en dos dimensiones, asumiendo simetría. Lo anteriormente descrito se muestra en la figura 4.1.

Los módulos son de forma rectangular. El tamaño del módulo queda determinado por las celdas que se encuentran en las posiciones más extremas, hacia la izquierda, la derecha, arriba y abajo. Como supuesto se asume que la separación entre la celda más extrema y el borde del módulo será fija y de 15 milímetros. La ventilación es por convección forzada y proviene desde el lado izquierdo con una temperatura de 20°C. En el lado derecho del rectángulo existe una abertura para permitir la salida del aire que está ingresando al módulo. La velocidad con la cual entra el aire queda determinada por el flujo señalado en la sección 3.6.3. La simulación es dependiente del tiempo y utiliza el ciclo US06.

En este caso de estudio, se ha dejado de lado el aspecto constructivo como una posible restricción del problema. El supuesto es que si los patrones son adecuados, se encontrará la forma

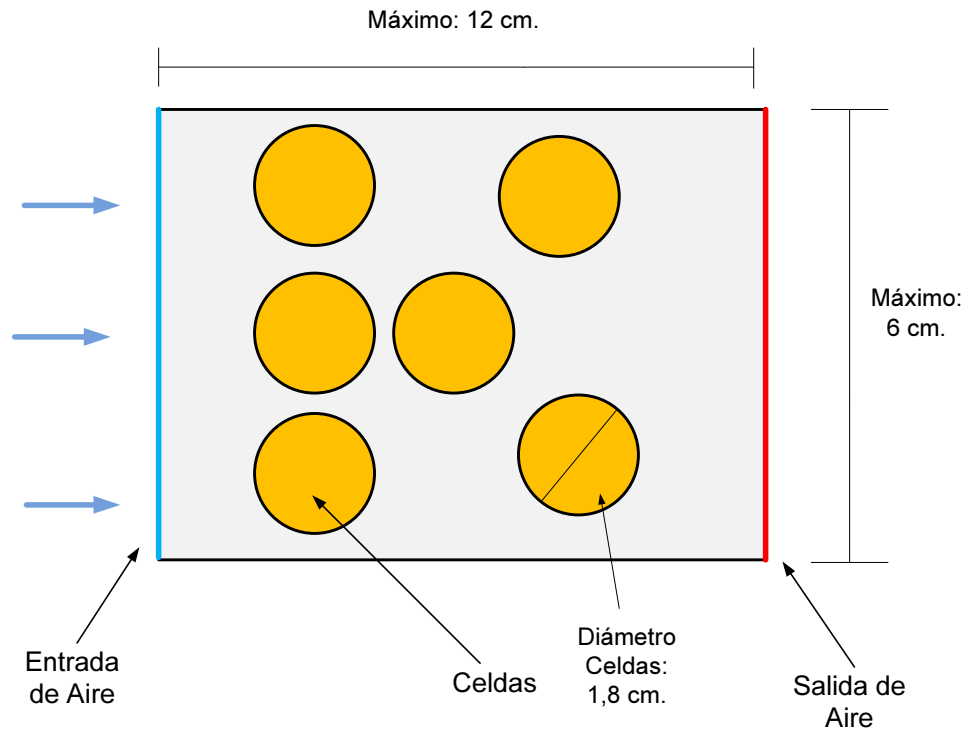


Figura 4.1: Geometría utilizada para el caso de 6 celdas cilíndricas aleatorias.

en que se pueda conectar eléctricamente las celdas más allá de que no sean formas convencionales de hacer los ensamblajes y cableados para cada módulo.

Para este caso de estudio se utilizó el algoritmo MOPSO iterando durante 200 generaciones de 200 individuos (lo que totaliza 40000 evaluaciones o simulaciones térmicas), con la temperatura máxima de una celda, la diferencia máxima de temperatura entre celdas y el volumen del módulo como las tres funciones objetivo. El volumen se calcula multiplicando el área del rectángulo que representa al módulo por la profundidad de las celdas 18650, en este caso 65 milímetros. La población inicial es aleatoria, no se entregan casos base.

Con un tiempo total de operación del algoritmo de 16 días, 16 horas y 25 minutos, el promedio del costo computacional de cada simulación térmica fue de 36,045 segundos. Dichos valores fueron obtenidos utilizando un computador con un cuatro núcleos de 3,2 Ghz (procesador AMD Phenom II X4 840) y una memoria RAM de 4 GB.

4.1.2. Determinación de parámetros del algoritmo evolutivo

Para la elección de los parámetros relevantes del programa multi-objetivo escogido, se hicieron pruebas para determinar el tamaño del espacio a explorar con diferentes tamaños de población. En general, para decidir el número adecuado de individuos en la población, se hacen corridas completas del algoritmo. En este caso, el tiempo requerido para completar la corrida del algoritmo hace muy difícil evaluar la convergencia de diferentes tamaños de población, por el alto costo en tiempo de una generación.

Por ello, las pruebas se realizaron haciendo 10 generaciones para 10 diferentes tamaños de población entre 20 y 400 individuos. Se evaluó el espacio cubierto y el promedio de velocidad para cada variable en cada caso, para elegir el tamaño de población adecuado para este problema. Se pusieron en consideración los siguientes aspectos para medir lo anterior: el tiempo que toma completar una generación para cada tamaño, la temperatura máxima más baja obtenida, la diferencia de temperatura más baja obtenida, el número de elementos en el Frente de Pareto y el rango de las funciones objetivo que se miden en el Frente de Pareto además de la desviación estándar de las posiciones de las celdas en la horizontal y en la vertical. Los resultados se muestran en la tabla 4.1 y en la tabla 4.2.

Tabla 4.1: Resultados obtenidos después de 10 generaciones del algoritmo MOPSO para el problema de seis celdas en posiciones aleatorias con diferentes tamaños de población. La tabla incluye el tiempo que tomó en promedio cada generación y la desviación estándar para las variables que indican la posición horizontal y vertical de las celdas.

Tamaño de la Población	Tiempo por Generación [min]	Desviación Estándar Posición Horizontal [cm]	Desviación Estándar Vertical [cm]
20	12,05	2,21	1,15
40	24,09	2,89	1,31
80	48,15	3,06	1,28
120	72,11	3,08	1,33
160	96,06	3,16	1,53
200	120,14	3,24	1,58
240	144,15	3,28	1,58
280	168,14	3,28	1,55
320	192,14	3,3	1,56
360	216,14	3,24	1,57
400	240,05	3,24	1,57

Considerando los valores de la prueba anterior, se escogió utilizar una población de 200 individuos, que al cabo de las 5 generaciones realizadas, permite alcanzar varianza y amplio rango en el Frente de Pareto tanto dentro del espacio de variables como de objetivos y evitar la posibilidad de convergencia hacia mínimos locales. Además con respecto al costo en CPU requiere de un tiempo de aproximadamente dos horas por generación, que es menor a los de tamaños más

Tabla 4.2: Resultados obtenidos después de 10 generaciones del algoritmo MOPSO para el problema de seis celdas en posiciones aleatorias con diferentes tamaños de población. La tabla incluye los rangos cubiertos en el Frente de Pareto y la cantidad de elementos en este. Un mayor rango permite un frente de Pareto más diverso.

Tamaño de la Población	Tiempo por Generación [min]	Rango Temperatura Máxima [°C]	Rango Dif. Temperatura [°C]	Rango Volumen [L]	Elementos en Frente de Pareto
20	12,13	9,52	9,61	0,258	5
40	24,13	5,45	6,19	0,305	6
80	48,12	3,2	9,06	0,198	6
120	72,11	9,57	12,64	0,185	8
160	96,08	9,09	13,28	0,249	10
200	120,09	17,635	11,369	0,224	11
240	144,08	17,28	11,88	0,267	9
280	168,13	13,825	14,24	0,324	11
320	192,06	15,633	20,63	0,25	11
360	216,06	12,627	6,84	0,236	12
400	240,13	5,175	16,64	0,175	9

considerables.

En todas las pruebas se tomaron medidas sobre la velocidad de movimiento de las partículas para cada una de sus variables. Este valor se mantuvo estable durante las simulaciones y en torno a un promedio de 1 [cm] para la horizontal y 0,5 [cm] en la vertical por cada avance por individuo que se encuentra lejos de su mejor posición o de su líder, determinado por un elemento del repositorio. Considerando lo anterior, para que una celda pueda moverse de un extremo del banco a otro requeriría en promedio 72 generaciones. Para asegurarse de que se presente la convergencia considerando dicha velocidad, se utiliza un total de 200 generaciones, al menos tres veces mayor al número anterior y que por tanto permite que el resultado no dependa de la población inicial de partículas. Ambos valores siguen siendo mucho menores que el espacio de exploración que contempla 72 centímetros cuadrados para seis celdas.

4.1.3. Resultados

Luego de que se completara el estudio, se obtuvo un frente de Pareto con 23 soluciones no dominadas en las funciones objetivo de temperatura máxima, diferencia de temperatura y volumen. Dicho frente de Pareto encontrado se muestra gráficamente en la figura 4.2.

Los valores de las funciones objetivo para las soluciones Pareto dominantes se presentan en la tabla 4.3, ordenadas de menor a mayor volumen.

En las figuras 4.3, 4.4 y 4.5 se muestra el Frente de Pareto separado por dos funciones objetivo,

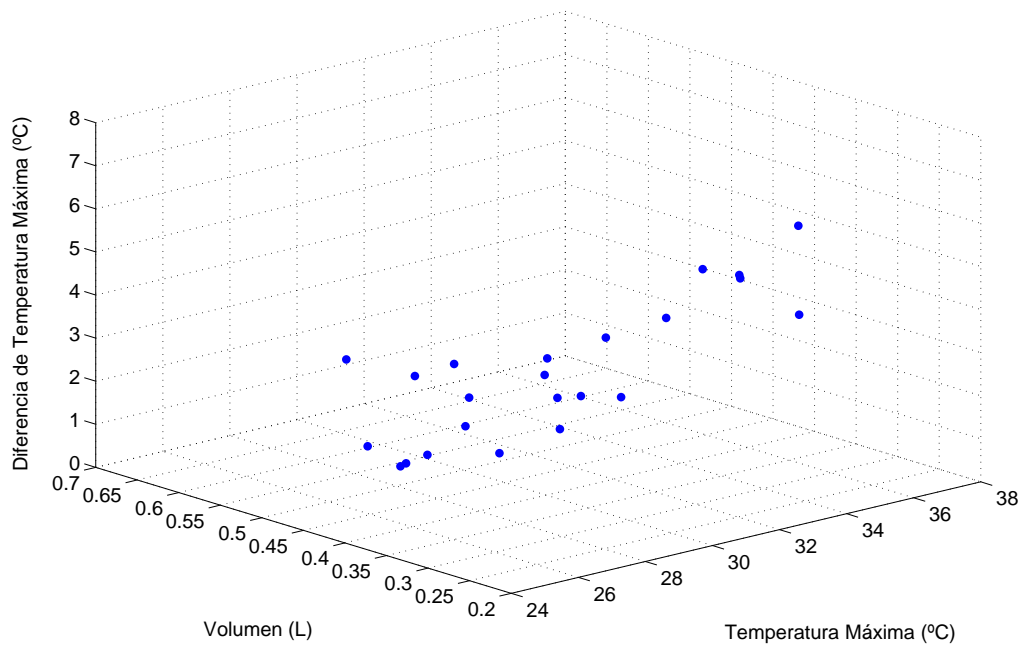


Figura 4.2: Frente de Pareto encontrado tras 200 iteraciones del algoritmo MOPSO.

para una mayor simplicidad en la visualización de los mejores resultados, en comparación con la figura 4.2.

A partir de las distancias entre las celdas, se puede rescatar valiosa información sobre la forma en que las celdas pueden agruparse para obtener mejor comportamiento térmico en función del volumen ocupado. Si bien las 23 soluciones son todas óptimas en el sentido de Pareto, puede notarse en el *trade-off* entre temperatura máxima y diferencia de temperatura (Figura 4.3) que estos dos objetivos, si bien no apuntan en la misma dirección para todas las configuraciones óptimas, si son más coincidentes que el caso de comparar con el volumen. Si bien correspondería realizar un nuevo experimento y otra búsqueda se puede señalar que si solamente se considerara la diferencia de temperatura máxima entre celdas y la temperatura máxima, entonces la cantidad de soluciones óptimas en el Frente de Pareto sería probablemente mucho menor que la encontrada, situación que no se daría para volumen y temperatura.

El *trade-off* entre temperatura máxima y volumen, así como el de diferencia de temperatura y volumen entrega información relevante con respecto a la separación entre celdas y su efecto térmico. Como se observa en la tabla 4.3, la diferencia de temperatura aumenta 5,7°C con una disminución de volumen del 52 % entre la mejor y la peor solución según diferencia de temperatura (equivalente a un cambio de 230 centímetros cúbicos). En el caso de la temperatura máxima hay un aumento de 7,4°C al reducir el volumen un 49 % entre la mejor y la peor solución encontrada en el Frente según temperatura máxima.

Esta diferencia entre dos módulos parece pequeña, entre 100 y 300 centímetros cúbicos, lo

Temperatura Máxima (°C)	Volumen (L)	Diferencia de Temperatura Máxima (°C)
33,887	0,254	6,391
31,635	0,278	5,657
32,77	0,28	5,304
32,798	0,28	5,218
28,957	0,286	4,513
34,822	0,29	3,939
31,207	0,304	4,448
29,889	0,306	2,849
28,816	0,351	3,68
30,118	0,363	2,494
29,424	0,363	2,578
29,222	0,371	3,106
29,873	0,378	1,683
28,287	0,387	1,364
27,326	0,403	3,52
27,935	0,41	2,591
27,987	0,416	1,877
26,441	0,423	3,255
27,176	0,429	1,287
26,752	0,438	1,118
27,66	0,481	0,627
26,69	0,507	3,136
26,587	0,514	0,935

Tabla 4.3: Soluciones óptimas obtenidas para el caso de 6 celdas 18650 en ubicaciones aleatorias. Se destacan los mínimos para cada función objetivo.

que indicaría que las soluciones de mayor volumen son considerablemente mejores que las de menor volumen, tanto desde el punto de vista de control y equalización (diferencia de temperatura) como de temperatura máxima. Sin embargo, hay que considerar que para un vehículo eléctrico o un sistema eléctrico de potencia, se requerirían cientos de estos módulos. Si el elemento diseñado se utilizara como módulo elemental para un vehículo eléctrico como el Nissan Leaf (que cuenta con un banco de baterías de 24 kWh y 360 volt nominales, con 48 módulos conectados en serie [69]) se requerirían 750 módulos de seis celdas cilíndricas 18650, conectados adecuadamente, luego la diferencia entre usar un volumen de 0,254 litros (solución N°1) en lugar de uno de 0,514 litros (solución N°23) significaría un ahorro en espacio de 195 litros, pero al costo de 7,4 °C de diferencia en temperatura máxima en el ciclo US06, lo cual en un sistema real se traduciría en mayor degradación o mayor gasto energético en refrigeración y control.

Las soluciones encontradas por el algoritmo se muestran en la tabla 4.4, de menor a mayor volumen. Las figuras dentro de la tabla 4.4 muestran la temperatura de cada celda en el último instante de tiempo. Si bien existe asimetría en casi todas las soluciones encontradas con el programa, ninguna de las opciones que se muestran pareciera ser impracticable en el sentido de

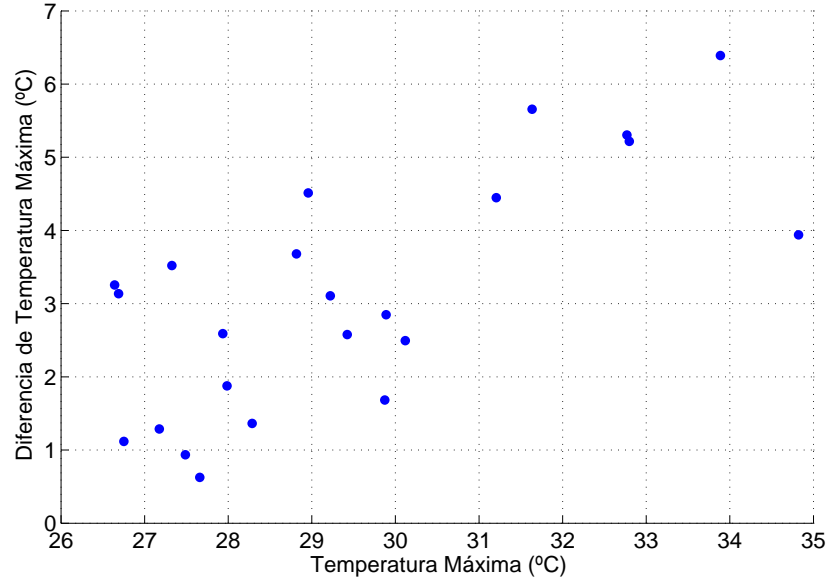


Figura 4.3: Frente de Pareto encontrado, proyectado en dos dimensiones para las funciones objetivo temperatura máxima y diferencia de temperatura máxima.

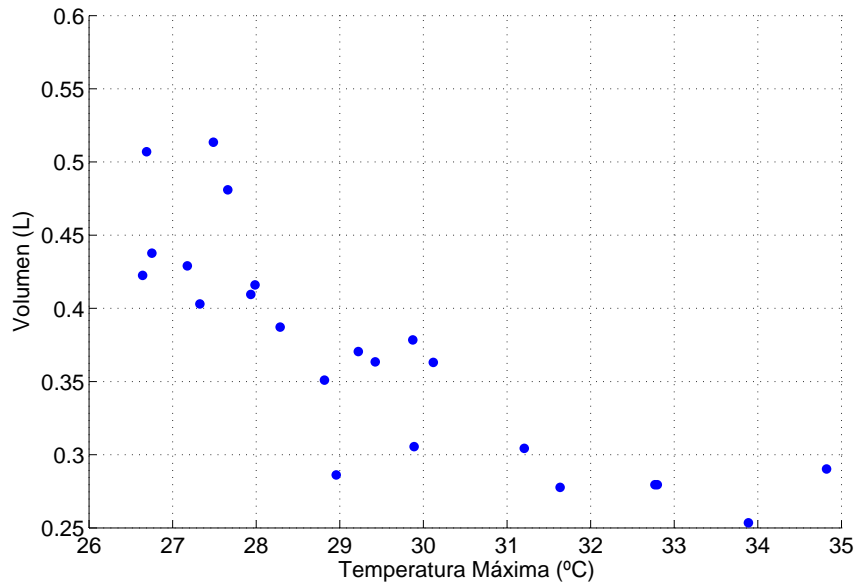


Figura 4.4: Frente de Pareto encontrado, proyectado en dos dimensiones para las funciones objetivo temperatura máxima y volumen.

su posibilidad de ser construída o conectada eléctricamente. Con las mejores soluciones obtenidas, se podría hacer un análisis de sensibilidad moviendo las diferentes celdas para hacer más simétrica

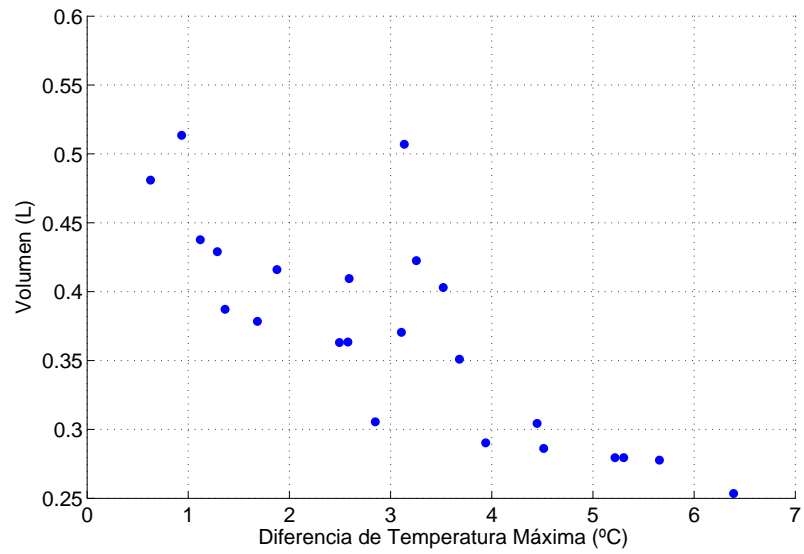
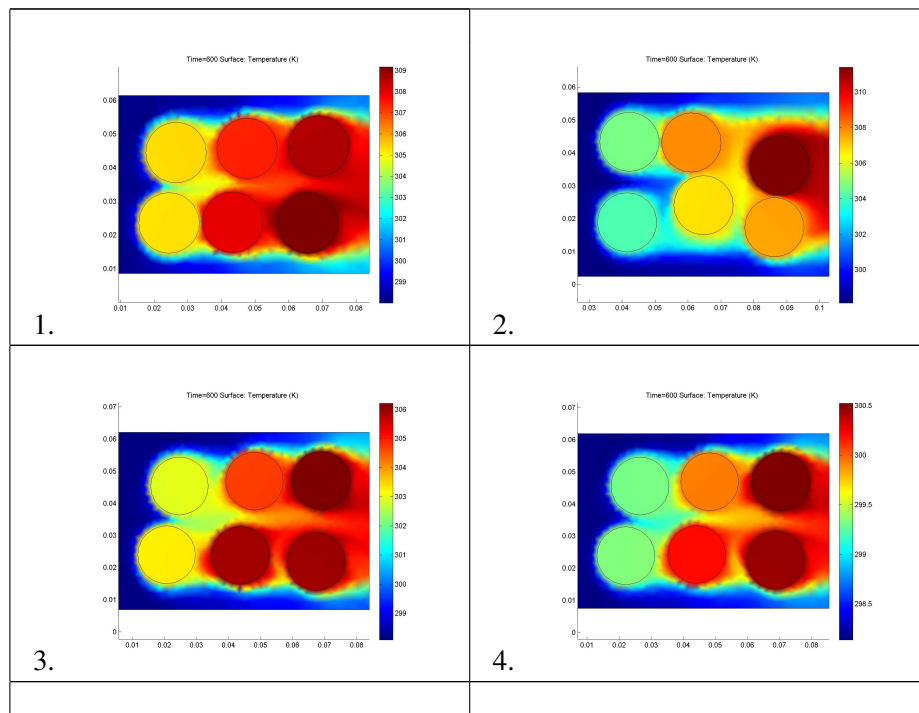
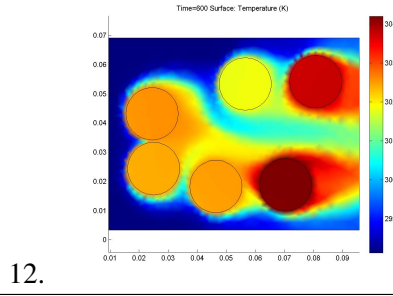
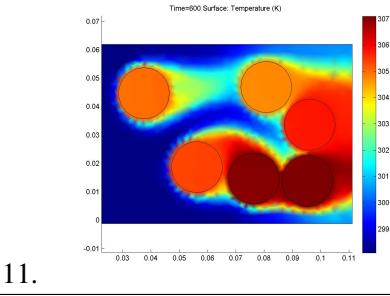
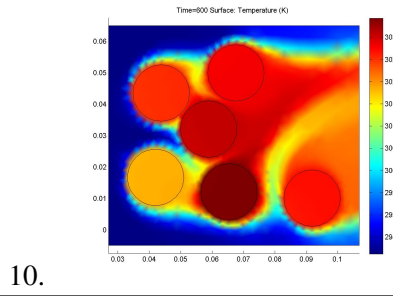
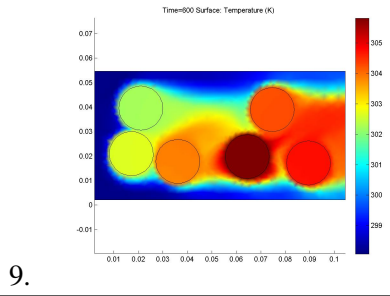
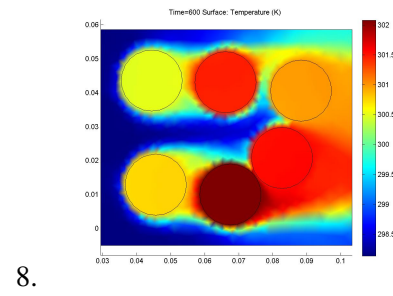
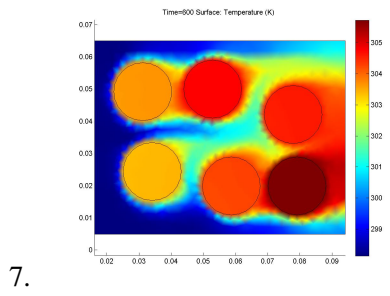
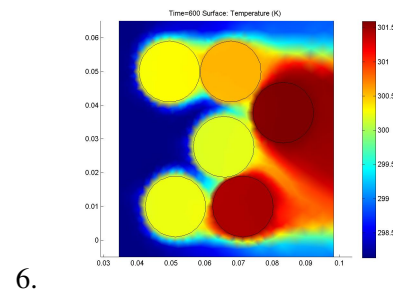
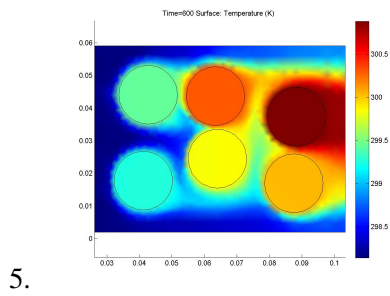
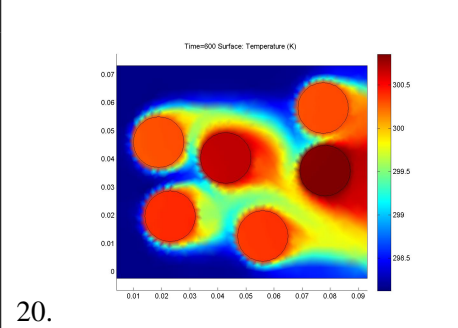
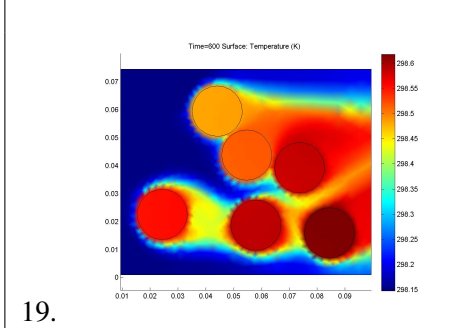
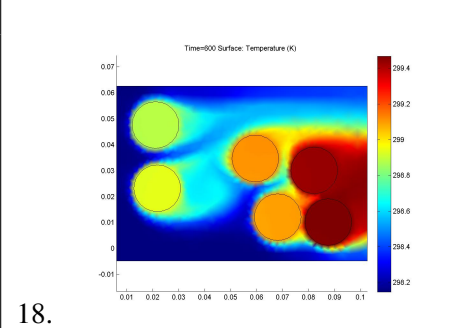
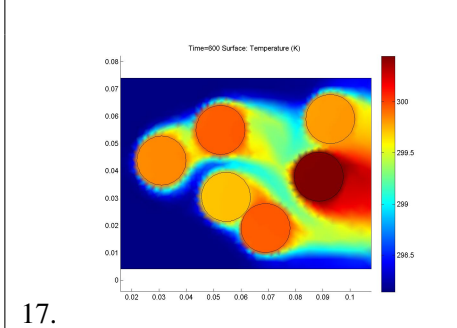
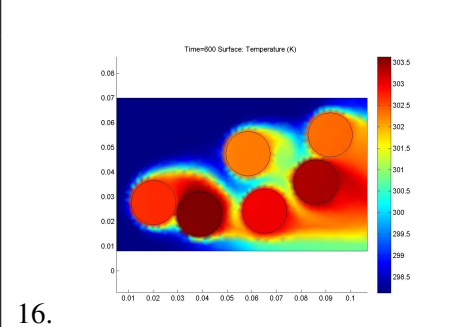
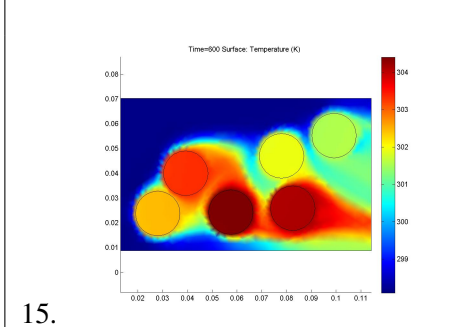
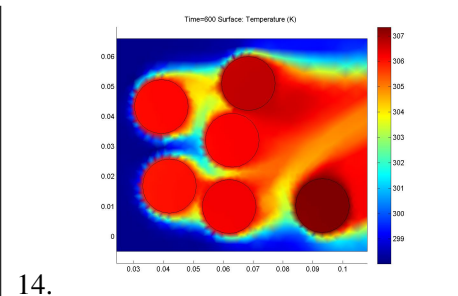
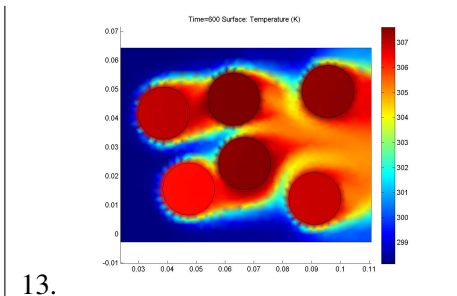


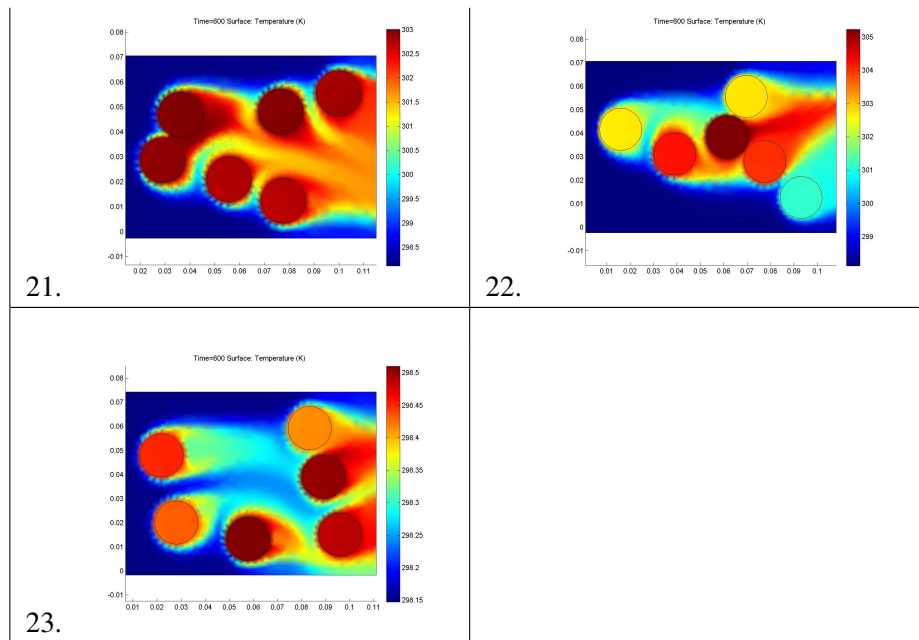
Figura 4.5: Frente de Pareto encontrado, proyectado en dos dimensiones para las funciones objetivo diferencia de temperatura máxima y volumen.

Tabla 4.4: Soluciones óptimas obtenidas para el caso de 6 celdas 18650 en ubicaciones aleatorias.









y sencilla de construir la solución desde el punto de vista del diseño industrial.

En las soluciones obtenidas pueden observarse una serie de patrones, que permitirían construir módulos de seis celdas para un vehículo eléctrico (u otra aplicación) en función del volumen disponible para cada uno de estos módulos. La principal ventaja de la aplicación del método multi-objetivo y con exploración basada en algoritmos evolutivos es que permite encontrar diseños no necesariamente intuitivos y que no respetan las simetrías y geometrías usuales de agrupación de celdas, las cuales podrían ser adaptadas para hacerlas constructivas.

Para los volúmenes más acotados, se puede ver que el algoritmo evolutivo, a través de su exploración, encuentra que las soluciones más adecuadas corresponden a grillas de celdas, con dos filas y tres columnas, casi alineadas, pero aprovechando ligeros acercamientos entre las celdas en algunas columnas para impulsar los “jets” que aumentan la velocidad del aire y permiten refrigerar más rápido las celdas de las últimas columnas. Esto se observa para las soluciones 1, 2, 3, 4, 5 y 7. La solución de diseño N° 1 se muestra en sus perfiles de temperatura y velocidad del aire en el último instante de la simulación en las figuras 4.6 y 4.7, respectivamente.

La mayoría de las soluciones ocupan valores muy cercanos al máximo valor permitido para el alto del módulo, luego el aumento en el volumen está acompañado por una mayor separación horizontal entre las celdas de baterías en el arreglo de seis celdas y en la búsqueda de patrones que permitan disminuir la temperatura máxima y la diferencia de temperatura máxima. La búsqueda de utilizar la mayor altura disponible se explica por la mayor ventilación directa que pueden recibir así las celdas en el módulo.

La menor temperatura máxima se obtiene en el caso del patrón 18, similar a una grilla escalonada, con tres columnas de celdas en grupos no alineados. Dicha configuración es asimétrica, separando las dos columnas que se encuentran más lejos de la ventilación de la primera columna, lo

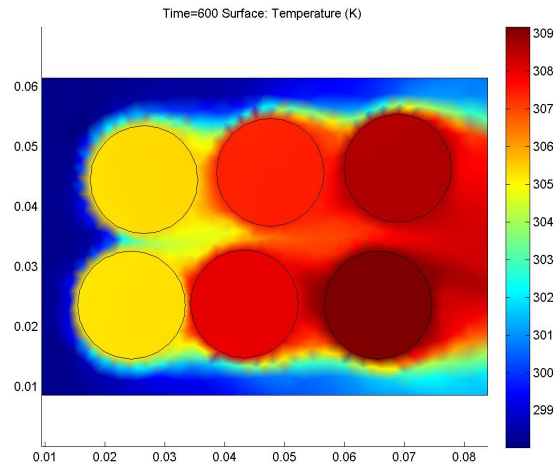


Figura 4.6: Perfil de temperatura del módulo diseñado en la Solución N° 1 en el Frente de Pareto. En esta figura se observa la grilla de dos filas y tres columnas encontrada por el algoritmo evolutivo.

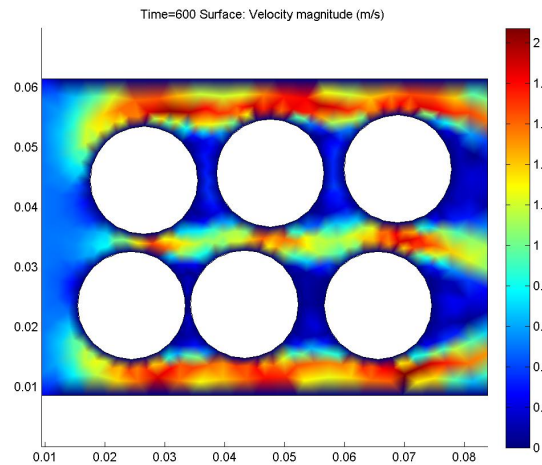


Figura 4.7: Perfil de velocidad del aire de la Solución N°1 del Frente de Pareto. En esta figura se observa la grilla de dos filas y tres columnas encontrada por el algoritmo evolutivo.

que hace que las celdas que se encuentran atrás reciban una temperatura de aire baja. Sin embargo, esta solución presenta una alta diferencia de temperatura en comparación con otros módulos, lo cual la haría más difícil de controlar en el caso de ser construida. La solución N°18 se muestra en las figuras 4.8 y 4.9, en sus respuestas en temperatura y velocidad del aire en el último instante de la simulación.

La menor diferencia de temperatura se obtiene en el caso de la solución 21, similar a una “V”

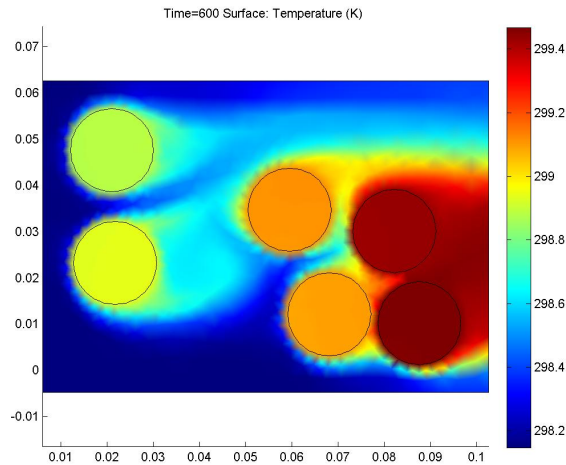


Figura 4.8: Perfil de temperatura del diseño N° 18 en el Frente de Pareto. Esta es la que obtiene la menor temperatura máxima de todas las configuraciones exploradas, con 26,441°C. En los perfiles de temperatura, los colores van del azul al rojo y significan mayor temperatura.

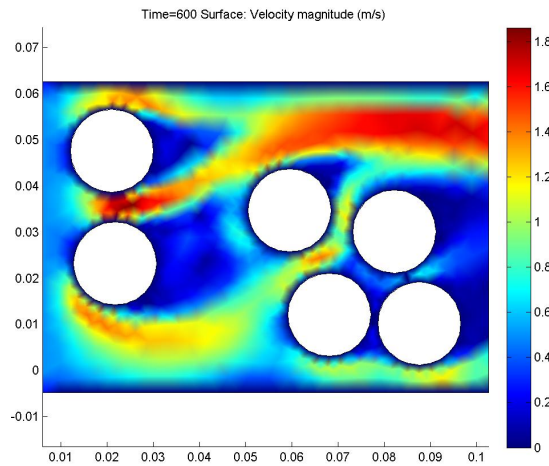


Figura 4.9: Perfil de velocidad del aire de la Solución N° 18 del Frente de Pareto.

que se abre hacia el extremo más lejano a la ventilación, permitiendo que todas las celdas reciban aire en alguna fracción de su superficie. La solución N°21 se muestra en las figura 4.10 para la respuesta en temperatura en el último instante de tiempo de la simulación y en la figura 4.11 para su perfil de velocidad del aire.

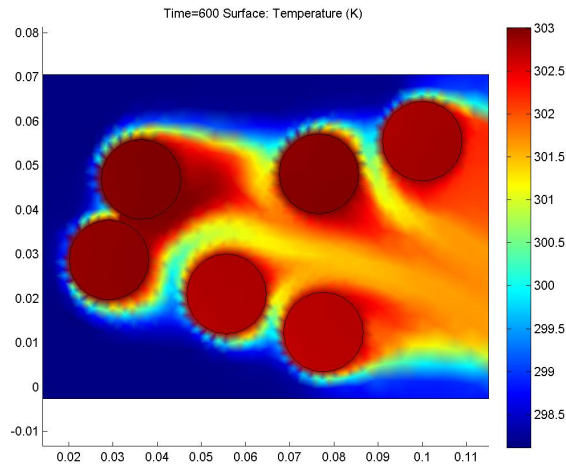


Figura 4.10: Perfil de temperatura del diseño N°21 en el Frente de Pareto. Esta es la solución que obtiene la menor diferencia de temperatura máxima de todas las configuraciones exploradas, con 0,627°C.

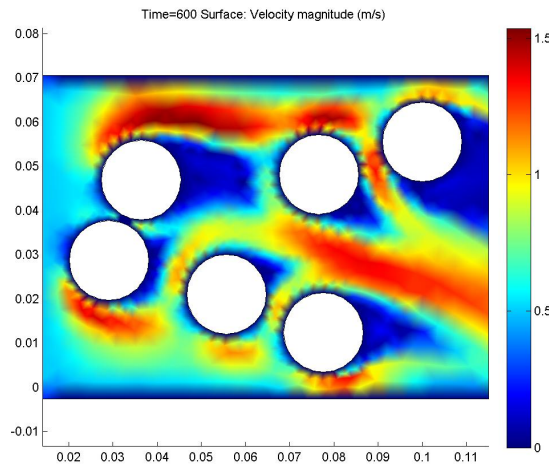


Figura 4.11: Perfil de velocidad del aire de la Solución N°21 del Frente de Pareto. Se observan los *jets* de alta velocidad de paso del aire en torno a las celdas, con formas similares, que permiten que la diferencia de temperatura sea baja.

4.1.4. Comparación de la metodología

Un problema de diseño como el presentado en este caso de estudio, resulta particularmente atractivo utilizar un algoritmo evolutivo para estudiar diferentes alternativas. Esto es por la necesidad de exploración del espacio de búsqueda, del cual no se tiene demasiada información a priori.

Una opción podría ser utilizar la optimización de un solo objetivo, que es más simple de implementar y requiere menor cantidad de cómputos para completar una generación. Sin embargo, en este caso, existiendo tres objetivos como la minimización de la temperatura máxima, la diferencia de temperatura máxima y el volumen, el uso de un algoritmo mono-objetivo reduce de forma importante la calidad de la información obtenida para el diseño de los módulos.

Como comparación se utilizó el algoritmo de Enjambre de Partículas (Particle Swarm Optimization, PSO) para la optimización de un objetivo. Se entregó a ambos algoritmos la misma población inicial. Para reducir el tiempo de simulación requerido para esta comparación, se utilizó como base de datos los resultados de las simulaciones con MOPSO y no se simularon aquellos individuos que ya habían sido simulados en el primer estudio.

Estas pruebas muestran la ventaja de utilizar una perspectiva multi-objetivo: Usando la temperatura máxima como objetivo de optimización, se obtuvo un resultado cercano a la menor temperatura encontrada en el caso multi-objetivo: 26,494°C de mínimo de temperatura máxima en el módulo con un volumen de 0,52 litros y una diferencia de temperatura de 0,96 °C, solución que puede observarse en la figura 4.15.

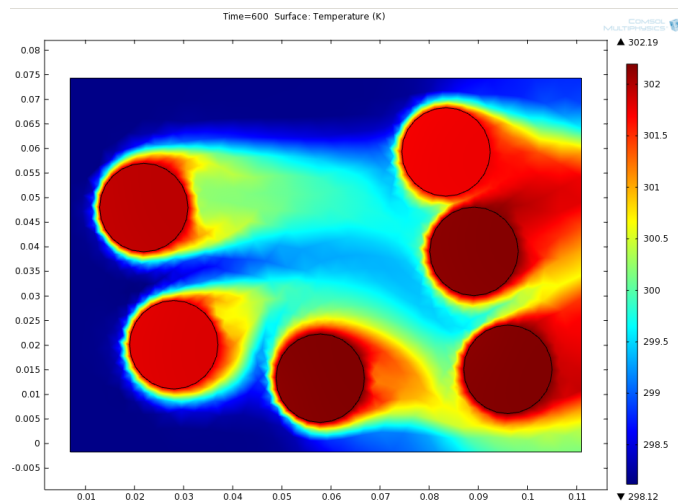


Figura 4.12: Diseño resultante tras operar PSO con un objetivo, minimización de la temperatura máxima.

Sin embargo, por naturaleza del algoritmo PSO, la anterior es la única solución óptima de la población, ya que solo se considera la temperatura máxima como objetivo y lo mismo lleva a que una mayor parte de la población se encuentre en torno a ésta, sin explorar alternativas de diferente volumen, por ejemplo. Un 82 % de los individuos que se encuentran en la última generación tienen volúmenes entre 0,5 y 0,56 Litros, con apenas un 6 % de ellos con volúmenes menores a 0,5 Litros. Luego, en la última generación, no existe información alguna sobre los módulos que tienen menores diferencias entre sus celdas y que en la optimización multi-objetivo llegan hasta los 0,258 Litros, ya que estas en general entregan mayores temperaturas máximas. En el caso de la exploración con MOPSO, la distribución de individuos en la población es mucho mayor. Un 35 % de los individuos tiene volumen menor a 0,4 Litros, pese a que un 42 % de las soluciones

tienen volúmenes entre 0,4 y 0,45 Litros. El histograma de distribución de las soluciones descrito anteriormente, puede observarse en la figura 4.13 para el caso de la optimización con un objetivo y en la figura 4.14 para el caso de la optimización multi-objetivo.

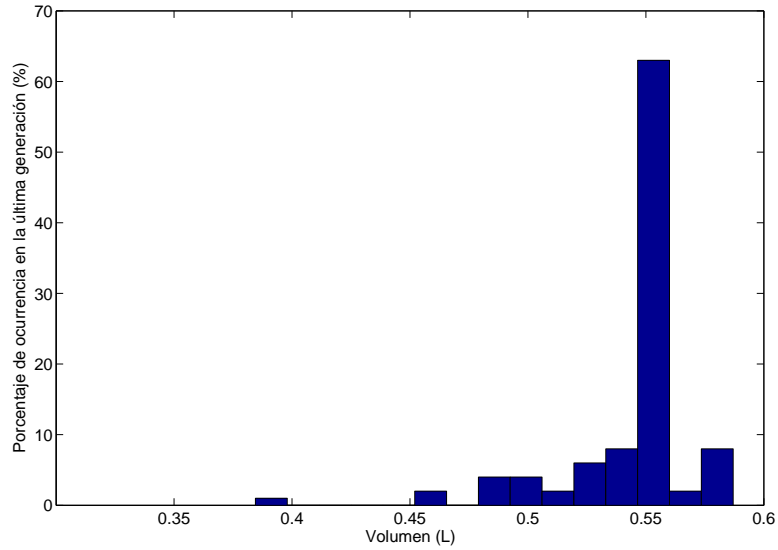


Figura 4.13: Distribución de las soluciones en la última generación del algoritmo PSO con un objetivo, minimización de temperatura máxima.

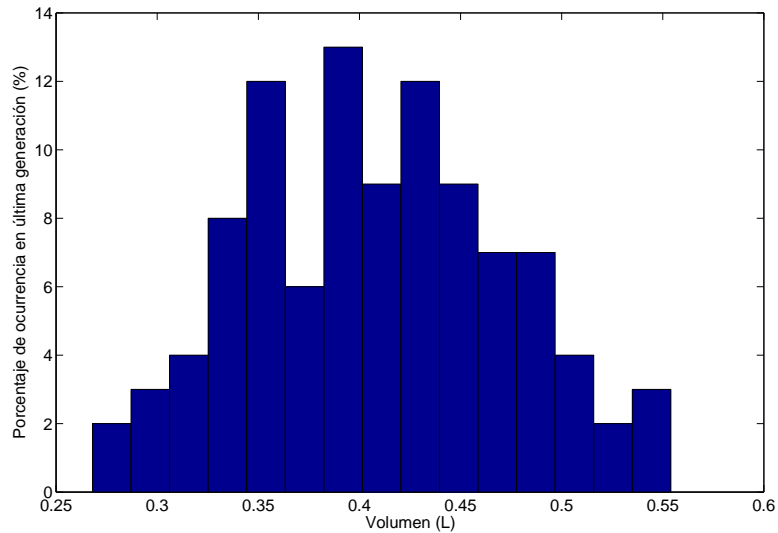


Figura 4.14: Distribución de las soluciones en la última generación del algoritmo MOPSO con tres objetivos: Minimización de temperatura máxima, volumen y diferencia de temperatura máxima.

Si se optimizara por volumen, la solución es trivial y es llevar la distancia entre celdas al mínimo, correspondiente a 0,1 milímetros en la simulación. Dicha solución no fue explorada o encontrada por el algoritmo evolutivo multi-objetivo. La operación llega a una solución no explorada por el algoritmo multi-objetivo, que tiene a todas las celdas en el límite del contacto, con un volumen de 0,218 Litros, pero con una temperatura máxima de 39,04°C y una diferencia de temperatura máxima de 8,4°C. Este elemento se hubiera encontrado en el Frente de Pareto del estudio con MOPSO, pero no se exploró una región con distancia entre celdas tan pequeña, por el forzamiento propio de los objetivos de temperatura máxima y diferencia de temperatura. La solución se observa en la figura 4.15.

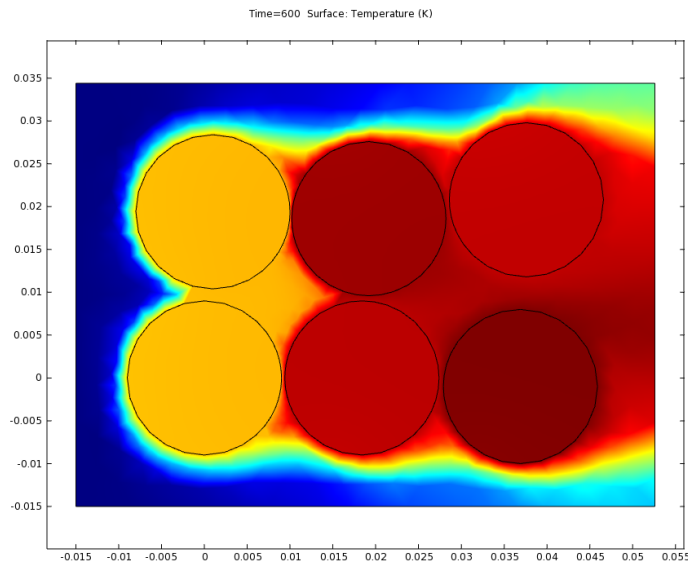


Figura 4.15: Diseño resultante tras operar PSO con un objetivo, minimización del volumen

Al optimizar por diferencia de temperatura con PSO se encontró un mínimo local, pero que es dominado por una solución encontrada en el caso multi-objetivo: En el caso simulado optimizando por diferencia de temperatura el algoritmo convergió a una solución con 0,8957 °C en separación máxima del valor de temperatura entre las celdas, 28,828°C en temperatura máxima y 0,598 Litros en volumen (ver figura 4.16. Este volumen es mayor que cualquiera encontrado entre las mejores soluciones utilizando MOPSO.

Observando los tres resultados, se concluye que el uso de una perspectiva multi-objetivo entrega información mucho más valiosa en el proceso de diseño, principalmente por la unión de funciones que no necesariamente apuntan en la misma dirección y que permite reorientar la exploración del espacio. Si se optimiza exclusivamente por temperatura máxima o diferencia de temperatura máxima, los volúmenes se disparan, ya que el diseño intenta separar las celdas y lograr una mayor ventilación. La exploración se hace luego en torno a volúmenes altos y se desprecian los volúmenes bajos, particularmente en las poblaciones finales. Dada la alta no-linealidad del problema térmico de los bancos de baterías, es relevante conocer el cambio progresivo y el costo en temperatura de incorporar soluciones de diferentes volúmenes.

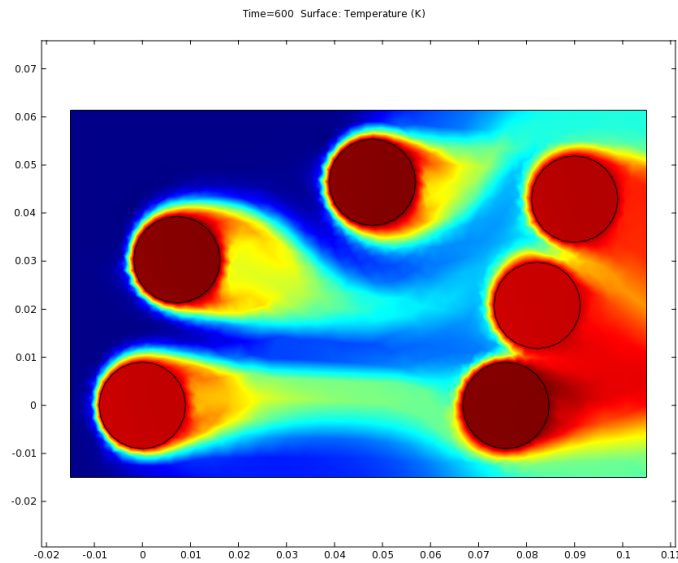


Figura 4.16: Diseño resultante tras operar PSO con un objetivo, minimización de la diferencia de temperatura.

Otra forma de demostrar el efecto que tiene el proceso de búsqueda y exploración con un algoritmo evolutivo multi-objetivo corresponde a comparar las soluciones encontradas con casos base generados por un diseñador humano solo a través de alguna decisión heurística y sin necesidad de ocupar herramientas sofisticadas de exploración del espacio. En general en el caso del diseño de intercambiadores de calor, que puede verse como una generalización del diseño de módulos de baterías, se utilizan dos geometrías típicas: grilla y grilla escalonada.

Usando grillas de dos filas y tres columnas, con una separación entre celdas de 27 milímetros en la vertical y en la horizontal (1,5 veces el diámetro de la celda), se obtiene un modulo con un volumen de 0,3122 litros con una temperatura máxima de 31,67°C y una diferencia de temperatura de 2,91°C. La geometría propuesta se muestra en la figura 4.17. Comparando con los casos en el Frente de Pareto, la solución más cercana en volumen corresponde a la N°8, que con un volumen de 0,306 litros obtiene una temperatura máxima de 29,889°C y una diferencia de temperatura de 2,84°C utilizando la misma simulación en COMSOL. La ganancia por la aplicación de la búsqueda con un algoritmo evolutivo para ese caso es relevante pero no significativa: Se reduce la temperatura máxima por 1,781°C y la diferencia de temperatura por 0,07°C ocupando un volumen 51 centímetros cúbicos más pequeño. Esta diferencia se explica ya que la grilla corresponde a una buena solución para volúmenes pequeños, según lo encontrado por el algoritmo.

Para volúmenes mayores, por ejemplo utilizando una separación de 36 milímetros en vertical y horizontal entre filas y columnas respectivamente para un volumen de 0,418 litros (como se muestra en la Figura 4.18), la temperatura máxima es 5,8°C más alta que la solución más cercana dentro del frente de Pareto en volumen (solución N°17): 33,35 °C en comparación a 27,987°C), lo que si representa una mejora significativa, mayor a 5°C. Además tiene una diferencia de temperatura 1°C más alta (2,89°C contra 1,89°C de la solución encontrada).

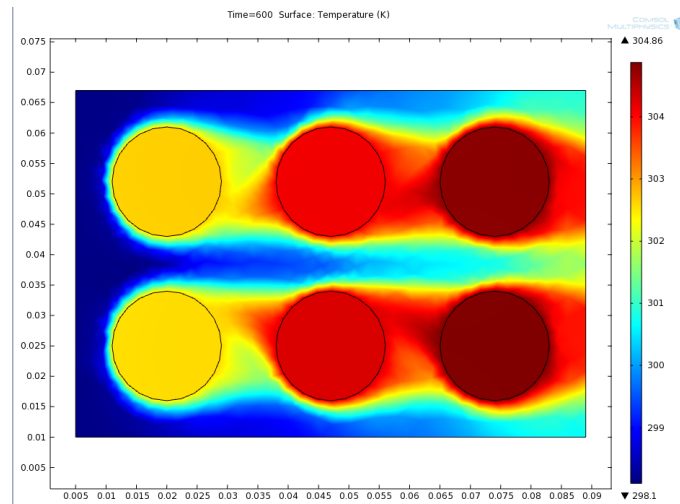


Figura 4.17: Caso base, configuración de grilla con separación de 27 milímetros.

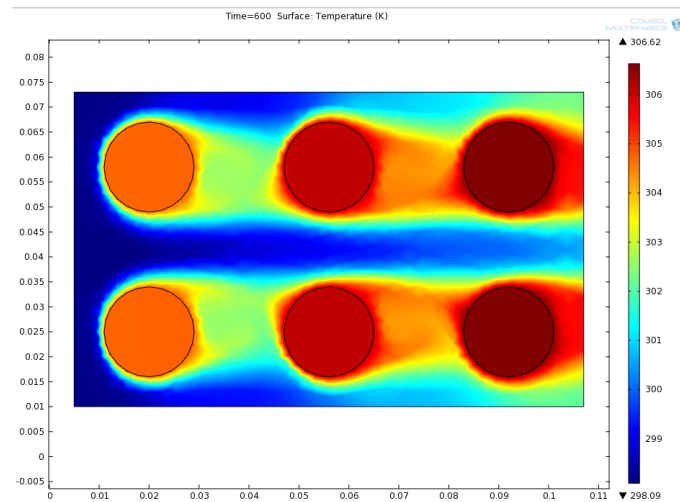


Figura 4.18: Caso base, configuración de grilla con separación de 36 milímetros en la horizontal y 27 milímetros en la vertical.

Comparando con una geometría en grilla escalonada (figura 4.19), las ventajas obtenidas son importantes: Con una separación entre celdas de 18 milímetros en la horizontal y 27 milímetros en la vertical, se obtiene un volumen de 0,5325 litros, una temperatura máxima de 29,25°C y una diferencia de temperatura máxima de 1,45°C, valores objetivo que son todos mayores a los de la solución N°23, que es la más cercana en términos de volumen.

Luego, como resultado de estas comparaciones puede señalarse que el uso de la herramienta diseñada no solo permite entregar información valiosa de los trade-off entre volumen y temperatura a través de las mejores soluciones, sino que también cumplir con su objetivo de optimización, reducir en varios grados Celsius el valor de la temperatura máxima y diferencia de temperatura en comparación con casos base. El costo de la mejora correspondería al hecho de que utilizar

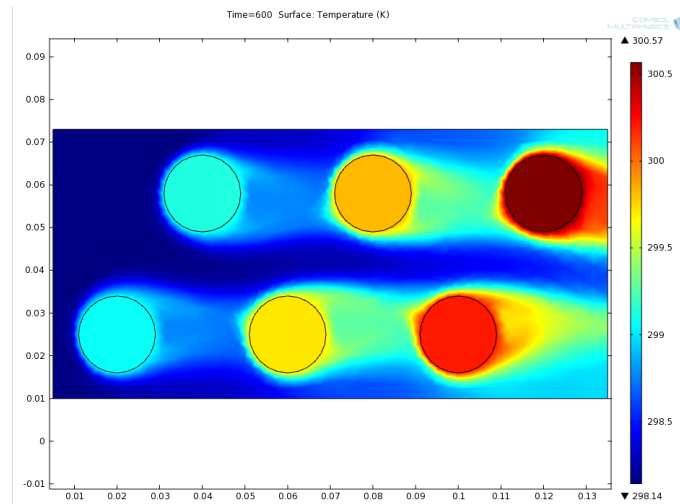


Figura 4.19: Caso de ejemplo, configuración de grilla escalonada con separación de 18 milímetros en la horizontal y 27 milímetros en la vertical.

la herramienta para encontrar soluciones óptimas requiere de una cantidad considerable de simulaciones térmicas, lo que toma un tiempo considerable (en este caso mayor a dos semanas) para obtener resultados, aún con computadores con gran capacidad de procesamiento y memoria.

4.2. Caso N°2: Seis módulos de celdas prismáticas

4.2.1. Descripción del caso de estudio

El objetivo de este caso de estudio es analizar el efecto del posicionamiento angular de módulos de baterías prismáticas. Para ello se estudia el ángulo de rotación de los módulos y se analizan sus implicancias térmicas a través de la simulación.

En el estudio se usan seis módulos de 11 celdas prismáticas SPB636395 conectadas en paralelo. Estos módulos están basados en los que utiliza el vehículo solar Eolian III de la Universidad de Chile, solo fueron modificados para generar simetría entre los módulos (igual ancho y largo). El estudio utiliza las restricciones similares de ventilación a las que tiene este prototipo, pensando en que la información podría ayudar a disponer los módulos con orientaciones adecuadas en próximas versiones del diseño del banco.

Los módulos tienen un ancho de 6,5 centímetros, un largo de 6,5 centímetros y un alto de 9,5 centímetros, y tienen 11 celdas. El confinamiento es una caja con 28 centímetros de largo, 30 centímetros de ancho, altura de 9,5 centímetros y fue diseñado intentando replicar las separaciones de los módulos con los bordes de la caja que se tienen en el Eolian III: Los centros de los módulos más extremos están a 6 centímetros de los bordes en la horizontal y a 8 centímetros de los bordes en la vertical. El banco a utilizar cuenta con una entrada de aire de 6 centímetros en la parte central de una de las caras frontales del banco y dos salidas de aire ubicadas en las caras laterales. Los

valores de flujo de aire respetan lo indicado en la sección 3.6.3 para presiones de entrada y salida. Las simulaciones se realizan en dos dimensiones, asumiendo simetría en altura. El aire ingresa a una temperatura constante de 20°C.

La geometría utilizada se muestra en la figura 4.20.

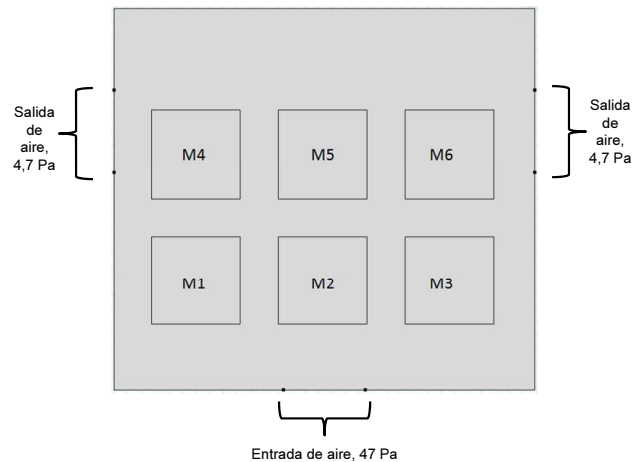


Figura 4.20: Geometría utilizada para el caso de estudio de seis módulos. La presión en Pascal aparece junto a la entrada y las salidas de aire.

Se asume que las posiciones de los módulos están definidas a priori y son fijas: Los centros de los módulos están separados a 9 centímetros uno de otro. Las variables de este caso corresponden al ángulo de rotación de cada módulo, ángulo que va desde -45° a 45° .

El módulo es representado como un solo elemento, generador de calor. La corriente queda determinada por la curva US06, con una corriente máxima de 99 Ampere para cada módulo de 11 celdas. El ángulo tiene un efecto importante en la refrigeración de las celdas o intercambiadores de calor en general, debido a que permite redirigir el flujo de aire que entra por convección forzada.

Los objetivos a optimizar para este caso son la temperatura máxima y la diferencia de temperatura máxima.

El algoritmo MOPSO fue programado para realizar 100 iteraciones con 100 individuos en cada generación para optimizar estas configuraciones. Para escoger estos valores se hicieron pruebas a diferentes tamaños de población, utilizando 10 generaciones para tamaños de 25 a 200 individuos. En los casos con menos de 50 partículas, se obtuvo convergencia acelerada a mínimos locales, sin varianza en el Frente de Pareto, lo cual llevó a tomar la decisión de escoger 100 individuos. En base a los resultados del mismo experimento, se determinó que el valor promedio de la velocidad de cambio para los ángulos lejanos a sus mejores posiciones correspondía a $4,2^\circ$, lo que indica que el espacio objetivo puede ser recorrido de un extremo a otro en aproximadamente 21 generaciones por cada módulo. Considerando el efecto que tiene reducir la velocidad cuando el elemento se acerca a un óptimo, se escoge 100 generaciones para permitir una exploración

completa.

El tiempo total de cálculo del algoritmo fue de 4 días y 21 horas, lo que significó un promedio de 33,48 segundos por simulación. Dichos valores fueron obtenidos utilizando un computador con un cuatro núcleos de 3,2 Ghz (procesador AMD Phenom II X4 840) y una memoria RAM de 4 GB.

4.2.2. Resultados

Completada la simulación, se obtienen 13 elementos en el Frente de Pareto. El Frente de Pareto se muestra en la Figura 4.21 y en la tabla 4.5. La temperatura máxima para los elementos que se encuentran en el Frente está entre 34,644°C y 36,247°C, en tanto que la diferencia de temperatura máxima está entre 6,486°C y 10,708 °C. Las soluciones se detallan en la tabla 4.7, donde se muestran los ángulos de cada solución en el Frente y en la tabla 4.6, donde se muestra el perfil de velocidad del aire para cada uno de los módulos.

Tabla 4.5: Frente de Pareto encontrado para el Caso de Estudio N°2 después de 100 generaciones del algoritmo MOPSO.

	Temperatura Máxima (°C)	Diferencia de Temperatura Máxima (°C)
1	34,644	10,708
2	34,881	9,470
3	34,914	9,388
4	35,060	9,355
5	35,144	9,201
6	35,160	9,138
7	35,169	9,115
8	35,176	9,037
9	35,179	7,824
10	35,407	7,591
11	35,836	7,497
12	35,963	6,855
13	36,247	6,486

Las soluciones presentan algunos patrones comunes entre ellas. Los módulos que se encuentran en los extremos más cercanos a las salidas de ventilación tienen siempre la misma orientación (positiva o negativa) y varianzas bajas, marcando que existe una forma óptima de colocar estos módulos de forma angular para aprovechar la recirculación del aire. En el caso del módulo M4, que se encuentra a la izquierda en la columna más alejada de la entrada de aire y más próximo a la salida de aire izquierda, el ángulo de orientación es siempre negativo (giro a favor de las manillas del reloj) y por otra parte, su valor se encuentra entre los -17° y los -32° para todos los casos. En el caso del módulo M6, la orientación es siempre positiva (en contra de las manillas del reloj) y los valores se encuentran entre 24° y 33° . Para el resto de los módulos existe una mayor varianza.

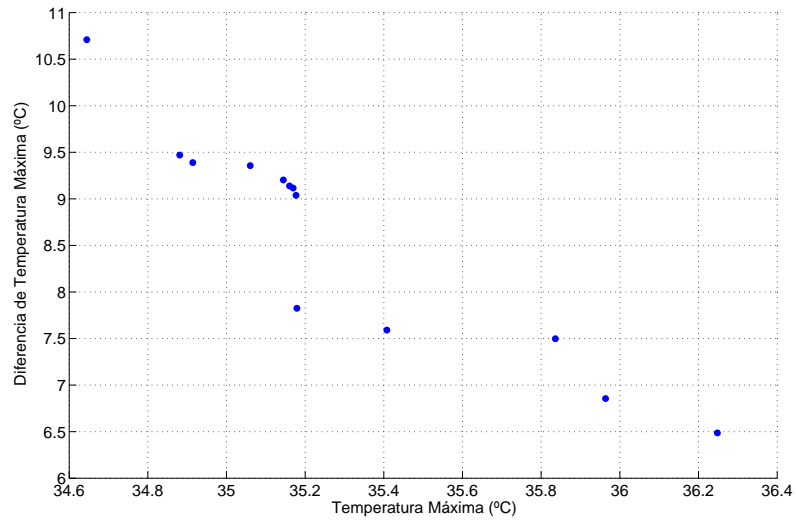
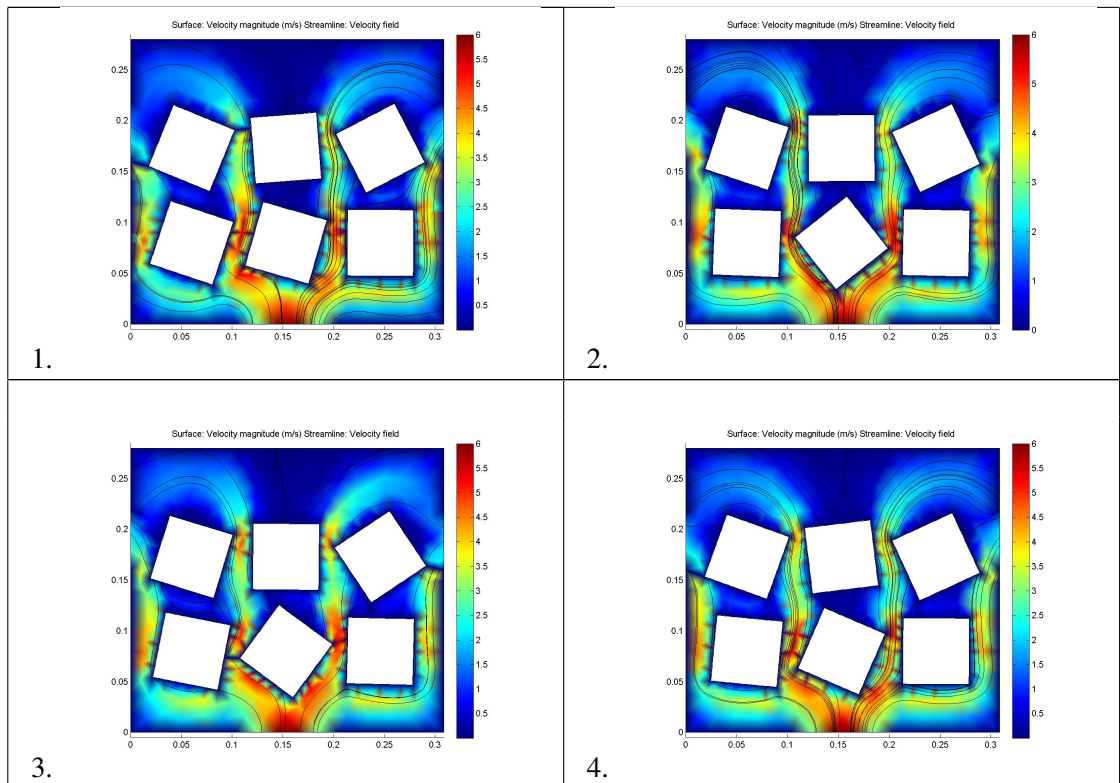
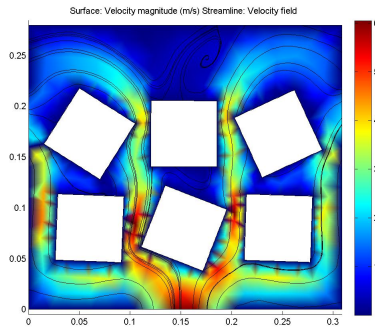


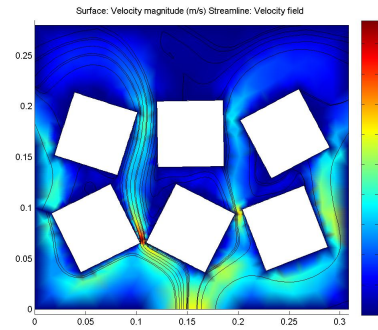
Figura 4.21: Frente de Pareto encontrado para el Caso de Estudio N°2 después de 100 generaciones del algoritmo MOPSO

Tabla 4.6: Soluciones óptimas obtenidas para el caso de orientación de 6 módulos de celdas prismáticas. Se muestra el perfil de velocidad del aire para cada solución.

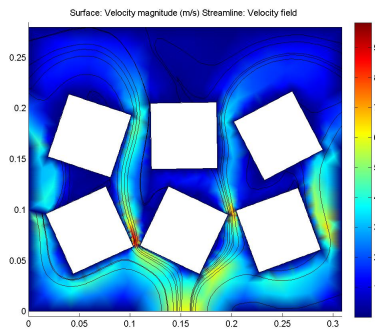




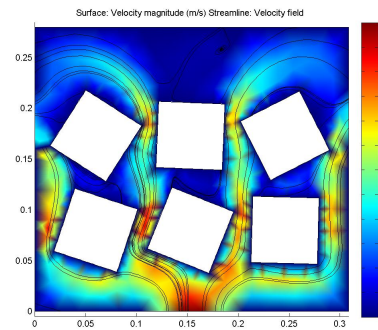
5.



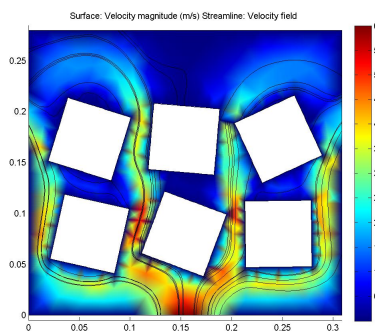
6.



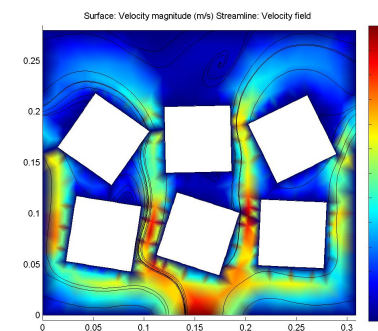
7.



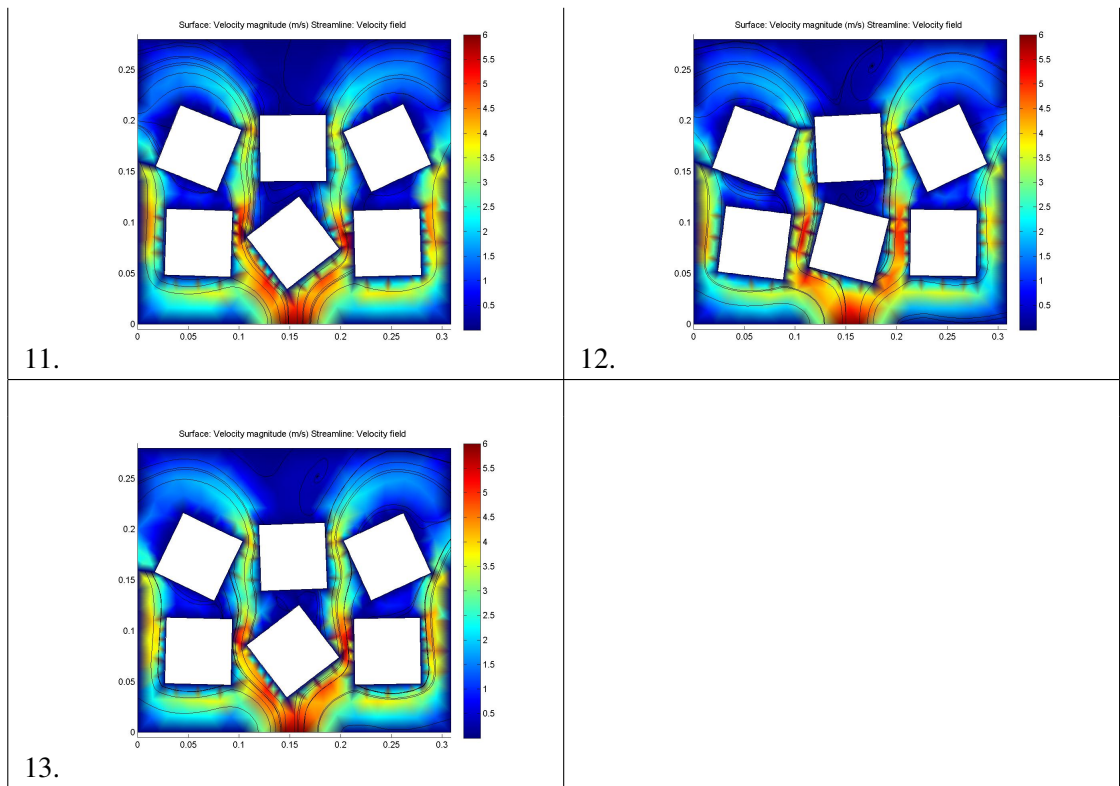
8.



9.



10.



Las menores diferencias de temperatura se encuentran para los diseños 12 y 13, que tienen la característica de utilizar las esquinas de los módulos para reducir el paso del aire a través de los espacios entre los módulos de la primera fila, la más cercana a la ventilación, de modo de lograr una mayor homogenización de la temperatura entre los seis elementos evaluados. Lo anterior se da en estos dos casos gracias a la combinación de ángulos positivos para los extremos de la primera fila y un ángulo negativo para el elemento central y que recibe la ventilación directamente. El diseño N°13 puede verse en la figura 4.22 que muestra el perfil de velocidades y en la figura 4.23 que muestra la temperatura en el instante final de simulación. El efecto anterior tiene el costo de una mayor temperatura máxima, con un valor más alto alcanzado durante la simulación que es 1,6°C más alto que el mínimo encontrado.

En las otras 11 soluciones, utilizando diferentes combinaciones de ángulos para los módulos se consigue redirigir la ventilación, manteniendo la velocidad de entrada (color rojo, en torno a los 6 m/s). Otro elemento que puede observarse en los elementos encontrados en el Frente es la influencia que tiene el ángulo en la forma de la circulación del aire detrás de los módulos y que sale en dirección a las tomas de aire. Las primeras siete soluciones tienen líneas de circulación del aire mucho más cercanas a las celdas, cosa que no se da tanto en las últimas celdas, y que tiene influencia directa en la reducción de la temperatura máxima para estas soluciones.

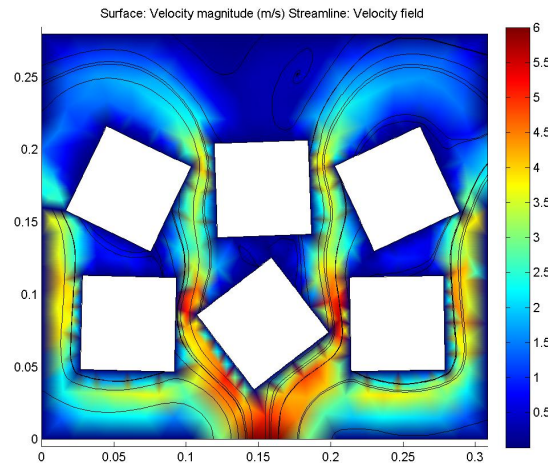


Figura 4.22: Perfil de velocidad del aire del diseño N°13 en el Frente de Pareto.

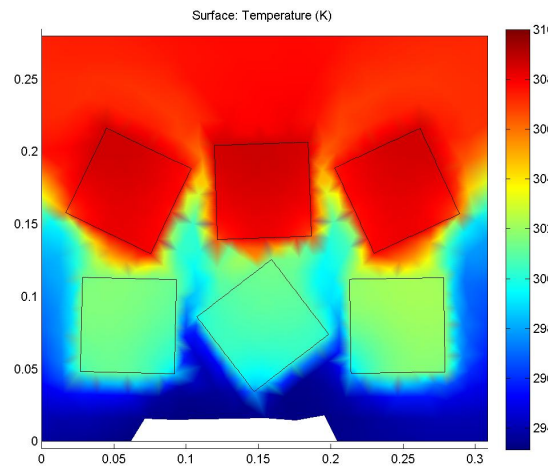


Figura 4.23: Perfil de temperatura de la Solución N°13 del Frente de Pareto.

4.2.3. Comparación de resultados

En base a los resultados, y la evolución de los mínimos de las funciones objetivo que se puede ver en las figuras 4.24 y 4.25, se puede observar que el algoritmo evolutivo produce mejora de los diseños para este caso de estudio, sin embargo, para evaluar el desempeño, se procede a comparar los resultados obtenidos con casos base que podrían ser diseñados sin necesidad de la herramienta.

El caso base directo es utilizar un ángulo de rotación de cero grados, como se utiliza actualmente en el vehículo Eolian que utiliza módulos de tamaño equivalente. En este caso, la temperatura máxima obtenida es igual a 38,29 °C y la diferencia de temperatura máxima corresponde a 12,43 °C, luego al incorporar cualquiera de las soluciones encontradas en el Frente

Tabla 4.7: Valores de los ángulos para cada solución en el Frente de Pareto.

	Ángulo M1	Ángulo M2	Ángulo M3	Ángulo M4	Ángulo M5	Ángulo M6
1	-18,4676	-15,9379	-0,3592	-22,2538	4,424	27,3627
2	-2,1799	-21,263	-1,825	-32,3247	-0,0518	24,8089
3	-7,0621	-14,2677	-0,7503	-20,1657	2,9354	25,173
4	-9,0764	-18,0149	-2,5724	-34,9396	1,2683	26,2712
5	-1,7668	37,6976	1,4454	-21,9655	0,3663	24,9912
6	-5,4607	-23,3193	-0,3691	-19,7261	6,9263	24,7077
7	-18,358	-21,7459	-1,4703	-32,6263	-2,3048	27,0692
8	-12,7652	-20,4423	0,6982	-17,5569	-4,869	24,8177
9	-1,3354	37,4722	0,9331	-25,6478	2,0636	24,8098
10	-2,2617	38,371	-1,1465	-18,3217	0,4849	25,1331
11	-11,1596	-36,4452	-1,316	-17,1491	-0,5409	33,7202
12	24,5372	-25,5714	20,1931	-18,544	0,8594	27,6078
13	26,4795	-27,0719	21,2211	-17,7196	0,8514	28,0574

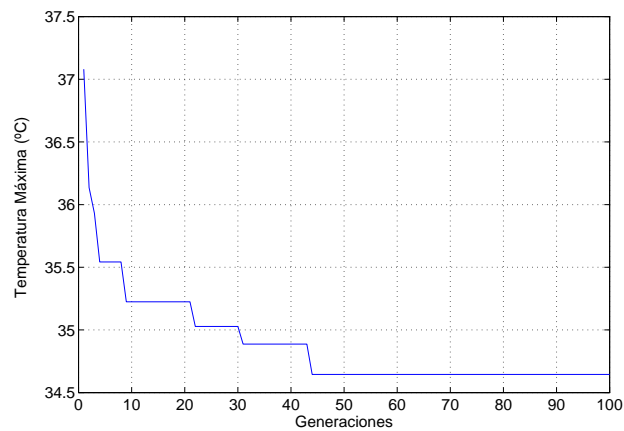


Figura 4.24: Evolución del mínimo de la temperatura máxima a lo largo de las generaciones para el caso de estudio N°2.

de Pareto se obtiene una mejora de al menos 2,05 °C en cuanto a temperatura máxima y 1,75 °C en diferencia de temperatura máxima, sin aumento en el espacio utilizado. La configuración con ángulo de rotación cero grados se muestra en la figura 4.26 con el perfil de velocidad del aire en el último instante de tiempo.

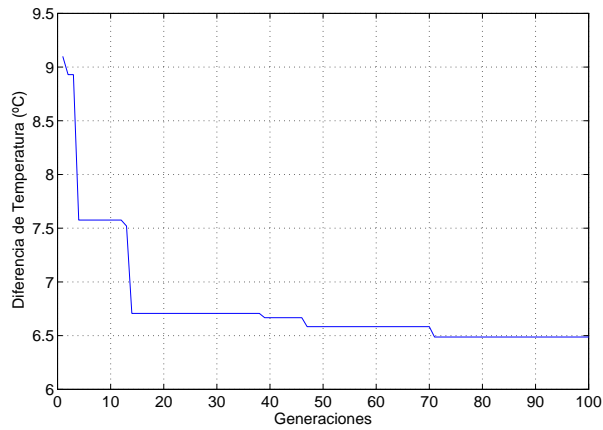


Figura 4.25: Evolución del mínimo de la diferencia de temperatura máxima a lo largo de las generaciones para el caso de estudio N°2.

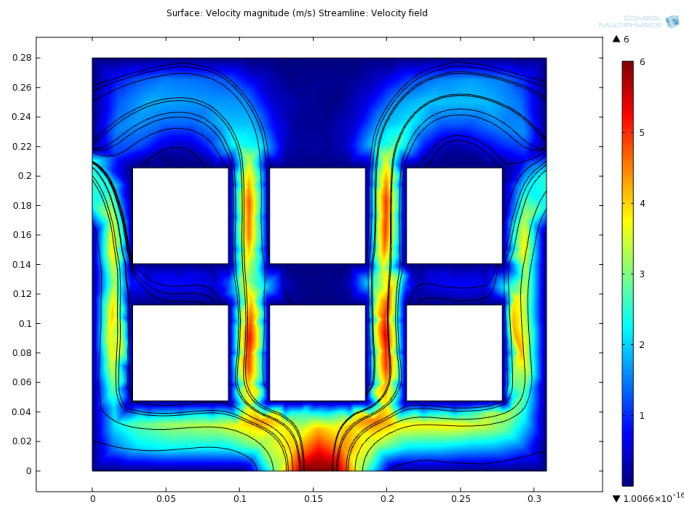


Figura 4.26: Perfil de velocidad del aire para la simulación del caso base con ángulo de rotación de cero grados para todos los módulos.

4.3. Caso N°3: Eolian III

4.3.1. Descripción del caso de estudio

Con la información recopilada de los dos casos de estudio teóricos presentados anteriormente, se puede concluir que el algoritmo evolutivo implementado en comunicación con simulaciones en COMSOL permite mejorar los diseños de bancos de baterías, sin embargo, no son casos directamente prácticos, lo que no permite evaluar el verdadero efecto de la metodología implementada en un sistema real. En ese sentido, se realizó una tercera aplicación del algoritmo

MOPSO para un banco de baterías, en este caso, el diseño del empaquetamiento para el vehículo solar Eolian III. El banco cuenta con 234 celdas de litio polímero SPB636395 separadas en 26 módulos de 9 celdas conectadas en paralelo. Los módulos están conectados en serie entre sí. El banco tiene una capacidad nominal de 40,5 Wh y un voltaje nominal de 96,2 Volt.

El diseño del banco de baterías de Eolian III presenta una gran cantidad de restricciones que lo hacen más realista: El problema consiste en la ubicación de 26 módulos de baterías en una caja con 61 centímetros de largo, 40 centímetros de ancho y 25 centímetros de alto. Cada módulo tiene las 9 celdas y dos placas de acrílico en los costados para mantener la estabilidad del banco. En la parte delantera del banco de baterías existe un espacio reservado de 9,5 centímetros de largo y 21 centímetros de ancho para elementos de electrónica como un contactor y cableado. Por restricciones de espacio y conexiones, los módulos deben ser colocados en dos pares de dos filas, con separaciones mínimas entre cada par. Además, las filas tienen asignados el número de módulos con los que pueden contar: seis para el par delantero, por la limitación que impone el contactor y siete en el segundo par de filas para totalizar los 26 módulos. Gráficamente puede verse la configuración del banco en la figura 4.27.

En este caso, por ser el vehículo destinado para competencia, el interés del equipo es determinar los valores de la temperatura para un uso constante a un valor de corriente cercano al máximo permitido por el banco. Se utilizará para las simulaciones una corriente máxima de 60 Ampere. Por ser una simulación estacionaria, la resistencia interna de las celdas debe mantenerse constante y no dependiente del tiempo, luego se utiliza un valor de 15 mohm por celda. Esto hace que cada módulo genere un calor constante de 6 W por Efecto Joule. Los módulos son simulados como un solo bloque generador de calor.

En la parte delantera del banco existe una entrada de aire, con un diámetro de 4 centímetros. En los bordes laterales existen salidas de aire, conectadas a un extractor de aire, también con un diámetro de 4 centímetros. La condición de borde utilizada para las entradas y salidas de aire en la simulación en COMSOL está dada por las presiones entregadas en la sección 3.6.3.

Debido a que el banco se encuentra con sus dimensiones definidas, los objetivos corresponden a lograr el mejor comportamiento térmico, por lo tanto se utilizan como función objetivo la temperatura máxima y la diferencia de temperatura máxima.

Las variables del problema corresponden a las diferentes separaciones entre los módulos dentro del banco:

- Separación entre la pared delantera de la caja y la primera fila de módulos (d1).
- Separación entre los pares de módulos que se encuentran en el mismo par de filas de módulos (d2).
- Separación en profundidad entre los dos pares de filas de módulos (d3).
- Separación entre módulos que se encuentran dentro de la misma fila (d4, para el par de filas delantero y d5 para el par de filas trasero, tomando como referencia la entrada de aire).

Las separaciones y su posición en el banco de baterías de Eolian III se muestran en la Figura 4.27 con los nombres d1, d2, d3, d4 y d5 indicando las separaciones correspondientes a las variables del problema. Las variables están restringidas dentro de un intervalo de modo que no se exceda el tamaño máximo del banco, correspondiente a la caja previamente construida. Los valores máximos y mínimos se indican en la Tabla 4.8.

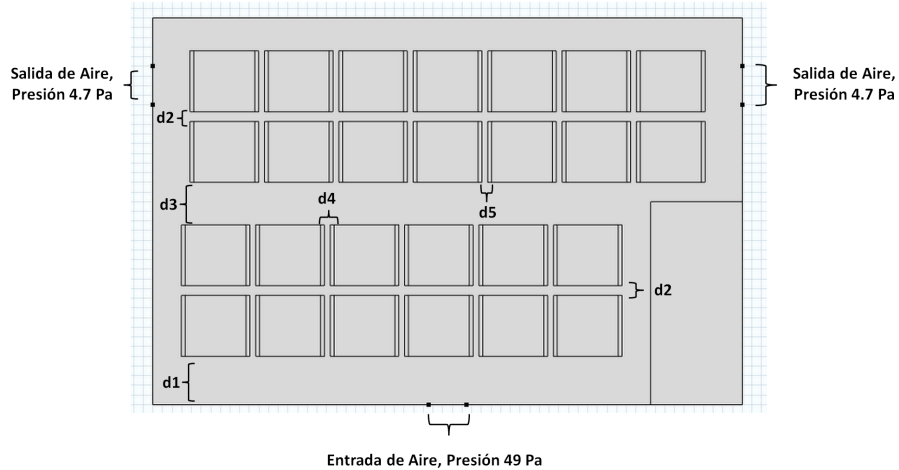


Figura 4.27: Geometría utilizada para la optimización del banco de baterías del vehículo solar Eolian III. Se muestran las variables que se utilizarán en la optimización con los nombres d1, d2, d3, d4 y d5 además de las entradas y salidas de aire.

Tabla 4.8: Valores máximos y mínimos para las variables del problema de diseño del banco de baterías del vehículo solar Eolian III.

Variable	Mínimo	Máximo
d1	40 [mm]	120 [mm]
d2	1 [mm]	10 [mm]
d3	1 [mm]	160 [mm]
d4	1 [mm]	20 [mm]
d5	1 [mm]	20 [mm]

4.3.2. Resultados

El equipo de Eolian III suministró un caso inicial para estudiar el diseño del módulo. Este elemento es entregado a la población inicial del algoritmo MOPSO para realizar la optimización. Los detalles de esta solución de diseño se muestran en la Tabla 4.9 y gráficamente en la figura 4.28.

La solución inicial con distancias entre los módulos sin optimizar alcanza una temperatura de 42,52 °C y una diferencia de temperatura máxima de 15,975 °C. Los perfiles de temperatura y velocidad del aire obtenidos por la simulación se muestran en las figuras 4.29 y 4.30, respectivamente.

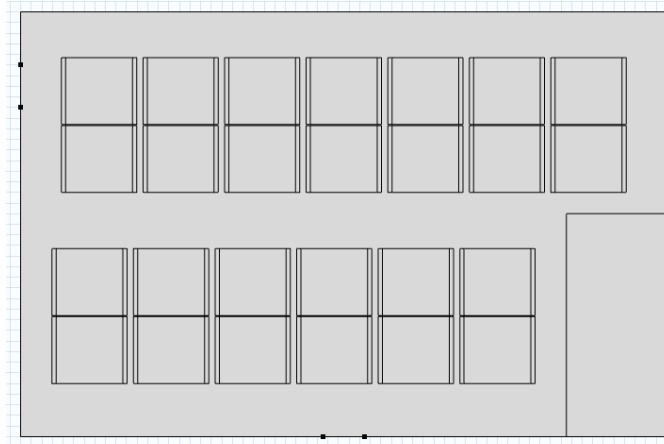


Figura 4.28: Caso inicial del banco de baterías de Eolian III.

Tabla 4.9: Valores de las variables del diseño del banco de baterías de Eolian III para el caso base entregado por el equipo para ser mejorado.

Variabes	Valor en Caso Base
d1	50 [mm]
d2	1 [mm]
d3	40 [mm]
d4	10 [mm]
d5	10 [mm]

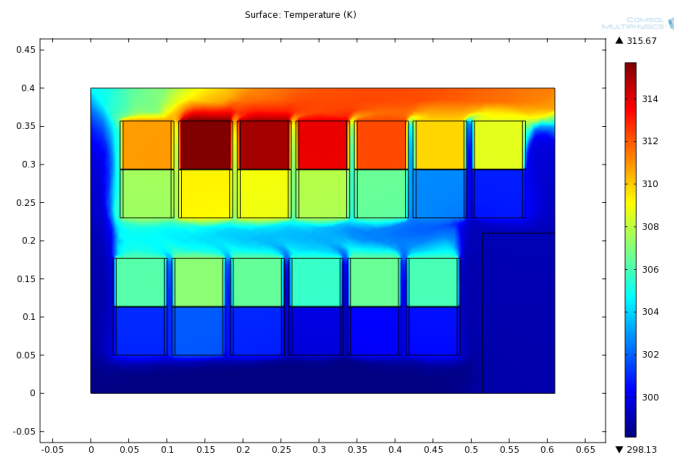


Figura 4.29: Perfil de temperatura del diseño inicial del banco de baterías de Eolian III.

Dado el espacio de exploración más acotado, el algoritmo MOPSO fue programado para realizar 50 iteraciones con una población de 100 individuos. Las simulaciones en COMSOL se realizan en dos dimensiones, asumiendo simetría en profundidad. Dado el espacio de exploración

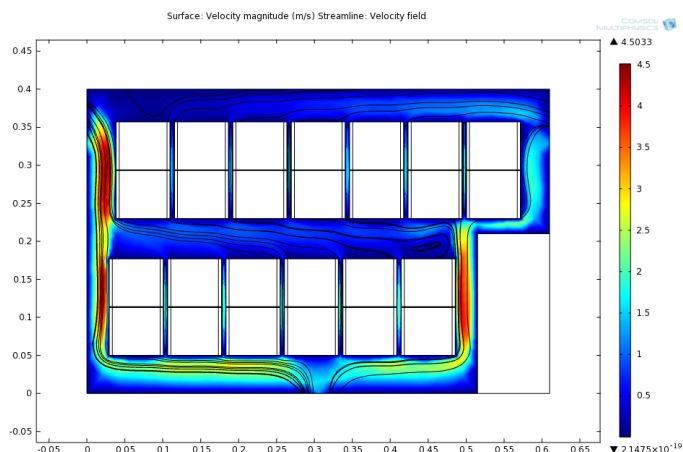


Figura 4.30: Perfil de velocidad del diseño inicial del banco de baterías de Eolian III.

más acotado, el algoritmo MOPSO fue programado para realizar 50 iteraciones con una población de 100 individuos. En este caso no se realizaron pruebas con menor o mayor cantidad de individuos, y los valores de iteraciones y número de población se basaron en el espacio de exploración, en comparación al utilizado en el caso de orientación angular.

Luego de completarse la evolución del algoritmo, converge hacia una solución óptima única con una temperatura máxima de 34,294 °C y una diferencia de temperatura máxima de 8,361 °C. La solución óptima se detalla en la tabla 4.10 y en la figura 4.31. En la figura 4.32 se muestra en su perfil de temperatura encontrado por la simulación en COMSOL y en la figura 4.33 en su perfil de velocidad del aire, también calculado por la simulación. Comparando con el caso inicial, la mejora lograda por la optimización multi-objetivo con respecto a este caso base es de 8,22 °C en temperatura máxima y 7,61°C en diferencia de temperatura máxima entre módulos, la mejora es significativa al ser mayor a 5°C.

Tabla 4.10: Valores de las variables para el caso optimizado del banco de baterías de Eolian III.

Variabes	Valor en Caso Optimizado
d1	50,3 [mm]
d2	2,6 [mm]
d3	47,8 [mm]
d4	11,06 [mm]
d5	12 [mm]

En el caso de este banco de baterías, la posición de la ventilación lleva a que se observe un efecto similar al del primer caso de estudio, en el cual existe una relación correlacionada entre la temperatura máxima y la diferencia de temperatura máxima. Este fenómeno lleva a que solamente se encuentre una solución óptima al terminar la optimización. El Frente de Pareto tuvo menos de 3 elementos en las 50 iteraciones del algoritmo y en las últimas 15, solamente se tuvo una solución óptima en el Frente. La evolución del mínimo para ambas funciones objetivo se muestra en las

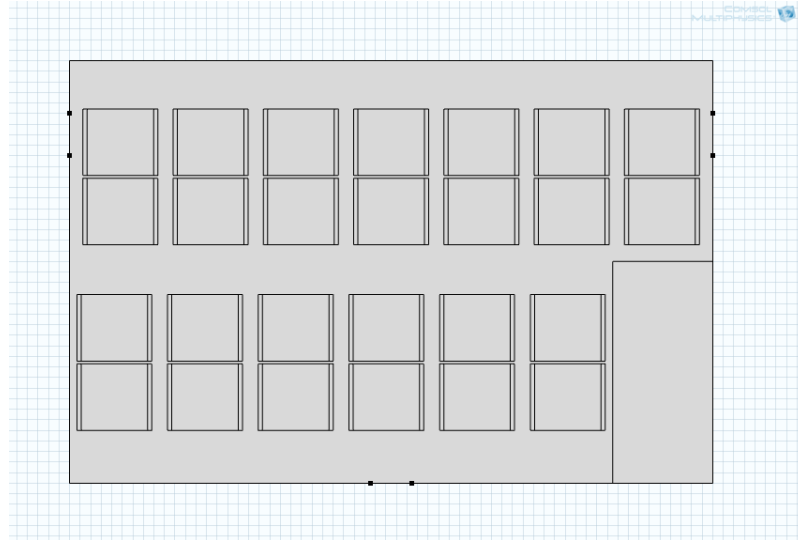


Figura 4.31: Geometría del diseño óptimo del banco de baterías de Eolian III encontrado por el algoritmo.

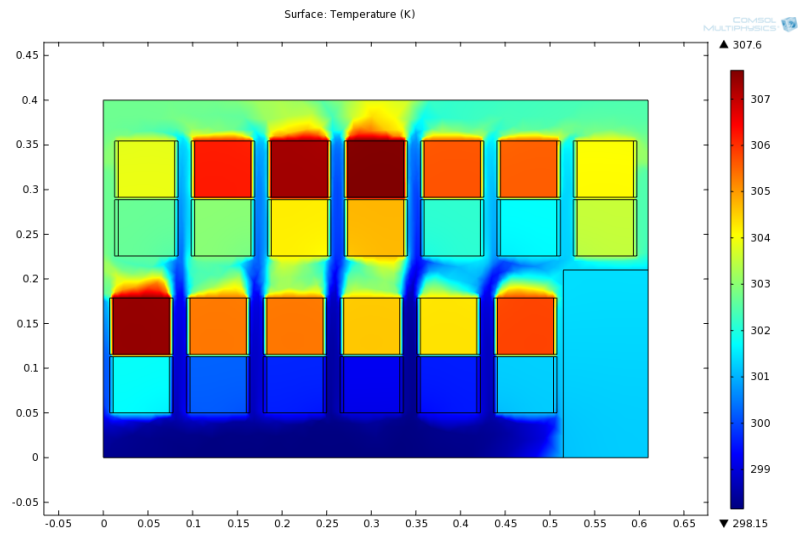


Figura 4.32: Perfil de temperatura del diseño óptimo del banco de baterías de Eolian III encontrado por el algoritmo.

figuras 4.34 y 4.35.

En el caso de algunas celdas de litio, como la SPB636395, puede ser deseable usarla a mayor temperatura para obtener más rendimiento según las curvas de descarga entregadas por el fabricante, pero existe un importante costo en la vida útil de las celdas. Por otra parte, no es adecuado que las celdas se encuentren por sobre los 40°C, una de las restricciones indicadas por el equipo a cargo de Eolian para la solución óptima encontrada. En ese sentido, la reducción de la temperatura máxima puede resultar como un elemento valioso a largo plazo para este banco,

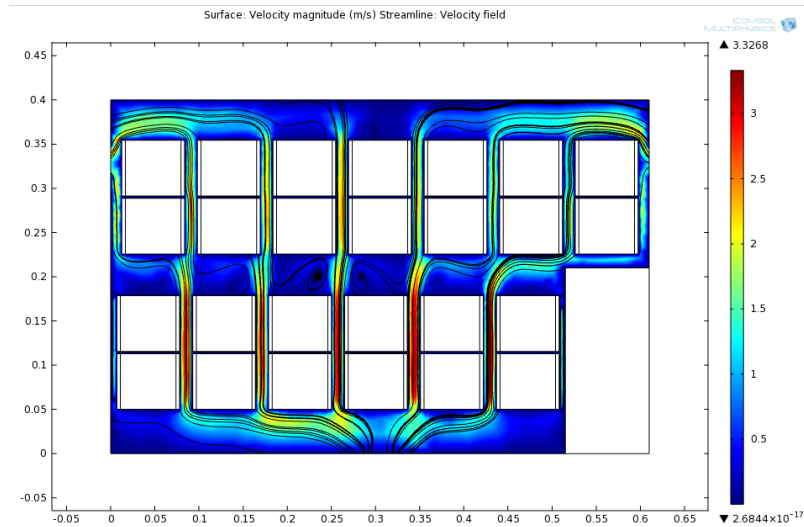


Figura 4.33: Perfil de velocidad del diseño óptimo del banco de baterías de Eolian III.

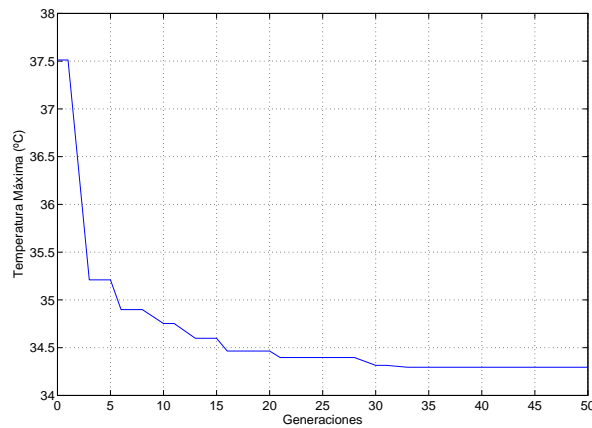


Figura 4.34: Evolución del mínimo de la temperatura máxima para el banco de baterías de Eolian III.

pero no necesariamente a corto plazo por el aumento de rendimiento con mayor temperatura. Sin embargo, la reducción en la diferencia de temperatura máxima es fundamental para el banco de baterías, ya que permite que las celdas funcionen adecuadamente y se descarguen de forma homogénea y eficiente con el uso del vehículo eléctrico. Es decir, una solución que permita reducir la diferencia máxima de temperatura en $7,61^{\circ}\text{C}$ es muy valiosa para un mejor funcionamiento del vehículo.

La solución indicada en este caso de estudio fue implementada, utilizando el archivo CAD entregado por COMSOL y funcionó adecuadamente, desde el punto de vista de temperatura, en las primeras pruebas realizadas al vehículo en operación previo a la competencia para la cual está destinado, Carrera Solar Atacama.

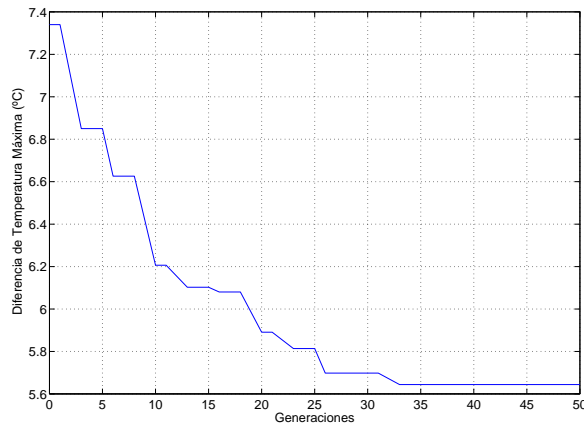


Figura 4.35: Evolución del mínimo de la diferencia de temperatura máxima para el banco de baterías de Eolian III.

4.3.3. Mejoras al sistema de ventilación

El caso de Eolian muestra que la metodología de optimización funciona adecuadamente para una situación mucho más restringida y de aplicación real.

A pesar de haber mejorado el desempeño térmico del banco en al menos 7°C según las simulaciones computacionales, el potencial de mejora de este banco de baterías para Eolian III es aún mayor. Debido a restricciones de construcción, previas al diseño del banco, la ubicación de la entrada de aire queda colocada al centro de una de las caras del confinamiento de los módulos. Lo anterior parecería ser una solución adecuada, sin embargo, el banco no es simétrico en su interior, debido a la presencia del contactor y los circuitos electrónicos que se encuentran en su parte delantera y que obligan a utilizar cuatro módulos menos en la primera fila del banco. Lo anterior tiene un efecto en la temperatura promedio de todos los módulos de la segunda fila reciben menos ventilación, algo que puede ser observado en las simulaciones y en mediciones y al mismo tiempo, hace que la ventilación se distribuya de forma desnivelada entre los módulos. Luego, una idea para mejorar el banco corresponde a cambiar la posición de la entrada de aire, permitiendo que este fluya de forma más equilibrada y así disminuyendo la temperatura máxima y la diferencia de temperatura máxima dentro del banco. Este cambio es lógico considerando que la posición del contactor es fija. Si bien este cambio sencillo no está disponible para esta iteración del banco de baterías de Eolian III, podría ser considerada para una próxima versión que considere cambio de la caja y la posición de las mangueras que permiten la ventilación.

Para decidir la posición de la entrada de aire de 40 [mm], se utilizó nuevamente el algoritmo evolutivo MOPSO, con la misma configuración que en el caso presentado anteriormente, agregando como variable la posición de la entrada de aire, en su límite izquierdo. Esta variable será denotada p_1 y tiene como valor mínimo 1 [mm] y como máximo 474 [mm]. La posición original corresponde a 285 [mm]. La disposición de variables y sus valores máximos y mínimos para este estudio se muestra en la figura 4.36 y en la tabla 4.11.

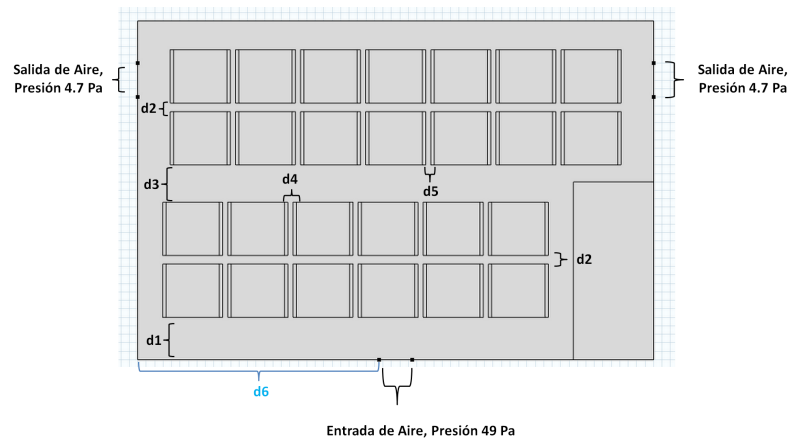


Figura 4.36: Geometría utilizada para la optimización del banco de baterías del vehículo solar Eolian III. Se muestran las variables que se utilizarán en la optimización con los nombres d1, d2, d3, d4, d5 y d6, además de las entradas y salidas de aire..

Tabla 4.11: Valores máximos y mínimos para las variables del problema de diseño del banco de baterías del vehículo solar Eolian III considerando la opción de modificar la posición de la entrada de aire.

Variable	Mínimo	Máximo
d1	40 [mm]	120 [mm]
d2	1 [mm]	10 [mm]
d3	1 [mm]	160 [mm]
d4	1 [mm]	20 [mm]
d5	1 [mm]	20 [mm]
d6	1 [mm]	474 [mm]

Uno de los elementos de la generación inicial será el diseño obtenido en la optimización, sin considerar cambios en la entrada de aire, es decir con la posición inicial de esta en 285 [mm]. Los valores de este caso se repasan en la tabla 4.12.

Tabla 4.12: Valores de las variables para el caso optimizado del banco de baterías de Eolian III sin cambiar la entrada de aire. Este diseño es parte de la población inicial de la optimización considerando la posibilidad de cambios en la entrada de aire.

Variabes	Valor en Caso Optimizado
d1	50,3 [mm]
d2	2,6 [mm]
d3	47,8 [mm]
d4	11,06 [mm]
d5	12 [mm]
d6	285 [mm]

Después de 50 iteraciones del algoritmo MOPSO, se obtiene un Frente de Pareto compuesto por cuatro soluciones. El tiempo total de simulación fue de 76 horas y 13 minutos (3 días, 4 horas y 13 minutos), lo que significa un promedio de 54,88 segundos por simulación. Los datos de tiempo de proceso se obtuvieron en el mismo computador que el caso para Eolian presentado anteriormente.

La mínima temperatura máxima encontrada corresponde a 32,47 °C y la mínima diferencia de temperatura equivale a 6,36°C. La separación de las funciones objetivo entre las soluciones extremas en el Frente de Pareto es de 0,17°C en temperatura máxima y 0,03°C en diferencia de temperatura, como se puede revisar en la tabla 4.13. Las soluciones se muestran a través del valor de sus variables en la tabla 4.14 y están separadas por solo milímetros una de otra, lo que indica que son casi equivalentes desde el punto de vista de diseño, lo que se puede corroborar por sus mínimas distancias en el espacio objetivo.

Tabla 4.13: Valores de funciones objetivo de las soluciones que se encuentran en el Frente de Pareto para el caso de la optimización del banco de baterías del Eolian III, considerando la opción de cambiar la posición de la entrada de aire.

Solución	Temperatura Máxima (°C)	Diferencia de Temperatura Máxima (°C)
1	32,48	6,4
2	32,62	6,39
3	32,63	6,37
4	32,64	6,36

Tabla 4.14: Valores de las variables de los diseños encontrados en el Frente de Pareto para el caso de la optimización del banco del Eolian III con cambio en la posición de la entrada de aire.

Solución	d1	d2	d3	d4	d5	d6
1	60,1 [mm]	2,6 [mm]	47,8 [mm]	11,06 [mm]	12,01 [mm]	249,9 [mm]
2	60,1 [mm]	2,6 [mm]	47,8 [mm]	11,05 [mm]	12,00 [mm]	249,7 [mm]
3	59,9 [mm]	2,6 [mm]	47,7 [mm]	11,06[mm]	12,01 [mm]	250,4 [mm]
4	60,1 [mm]	2,6 [mm]	47,9 [mm]	11,06 [mm]	12,00 [mm]	249,8 [mm]

La solución con menor temperatura máxima (N°1 del Frente de Pareto en la tabla 4.13) se muestra en la figura 4.37 y en los resultados de la simulación en la figura 4.38 (perfil de temperatura) y en la figura 4.39 (perfil de velocidad del aire).

Lo que se referencia en la tabla 4.13, indica que el cambiar la posición de la entrada de aire en 35 milímetros lleva a la disminución de la temperatura máxima en 1,82°C y la diferencia de temperatura máxima en 1,97°C comparando con el caso optimizado del banco de baterías de Eolian III. El resto de las variables se mantienen sin modificaciones con respecto al caso óptimo encontrado con la primera utilización de MOPSO. Lo anterior muestra que el cambio de posición de la entrada del aire representa un importante potencial de mejora para el banco de baterías de Eolian III, con la posibilidad de reducir otros 2°C la temperatura del banco en su uso continuo, según las simulaciones. Este cambio es propuesto para una futura implementación del banco de

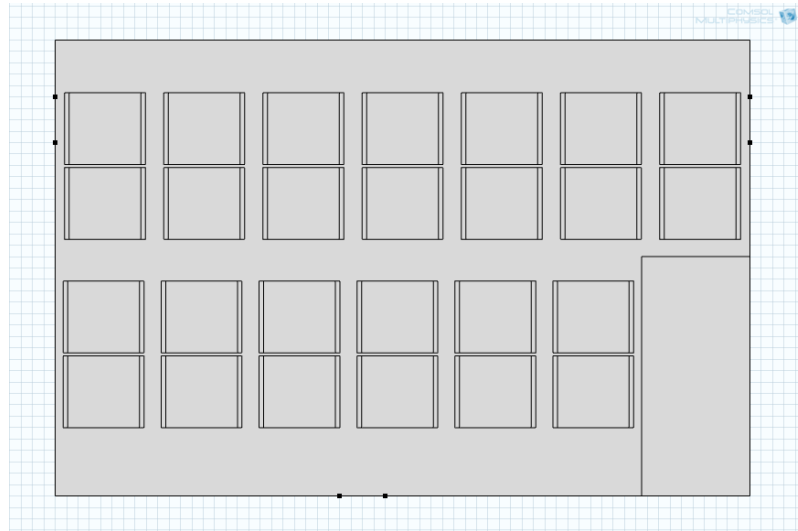


Figura 4.37: Geometría del diseño óptimo del banco de baterías de Eolian III, considerando el cambio de posición de entrada del aire, encontrado por el algoritmo evolutivo.

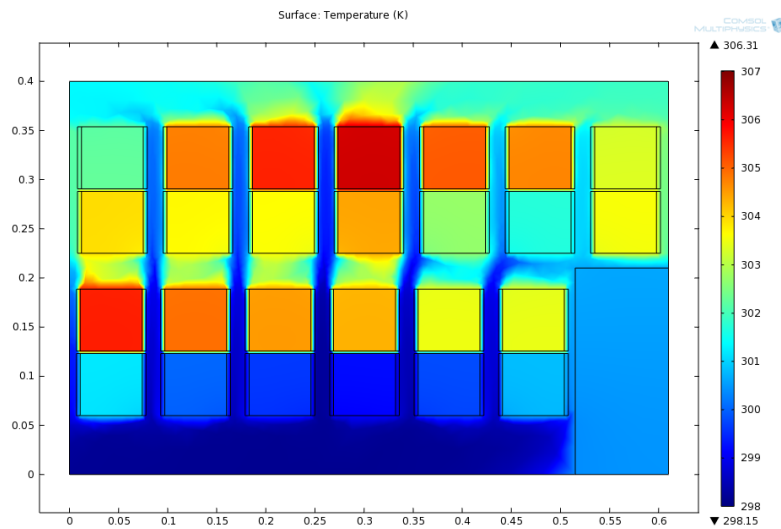


Figura 4.38: Perfil de temperatura del diseño óptimo del banco de baterías de Eolian III con cambio en la entrada de aire.

baterías, en el caso de que existan asimetrías motivadas por elementos electrónicos sobre los cuales no se requiere control de temperatura, como es el caso del diseño original de este banco.

Esta segunda mejora a un caso real recalca el potencial que tiene hacer uso de una herramienta de optimización de un banco de baterías antes de construirlo.

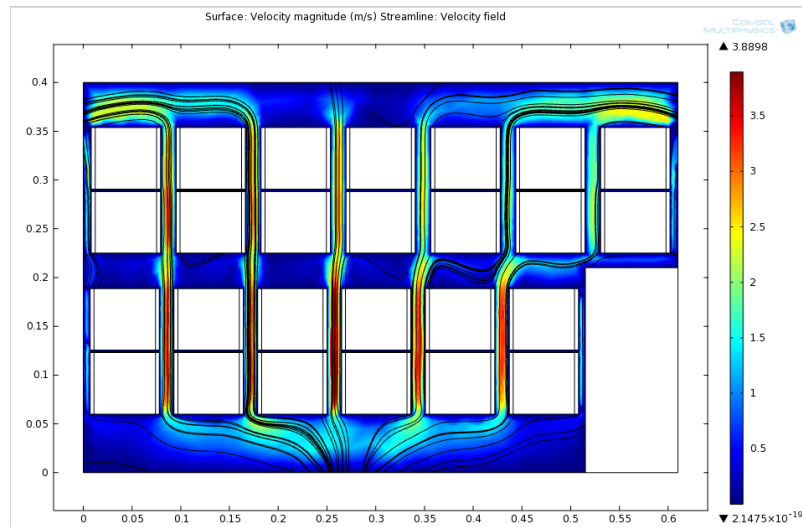


Figura 4.39: Perfil de velocidad del diseño óptimo del banco de baterías de Eolian III con cambio en la entrada del aire.

Capítulo 5

Conclusiones

5.1. Conclusiones generales

En esta memoria se propone una metodología para obtener diseños de módulos y bancos de baterías óptimos según su comportamiento térmico. La aplicación basada en dicha metodología fue implementada utilizando simulaciones computacionales y un algoritmo evolutivo multi-objetivo para realizar la optimización. La aplicación es utilizada y probada en tres casos de estudio, sobre los cuales se analizan los resultados obtenidos tanto del punto de vista de los diseños como en comparaciones con el diseño convencional humano.

En la primera parte se describe la situación actual del diseño de empaquetamientos de baterías a través de simulación computacional. Si bien el uso de programas CAE (Computer Aided Engineering) es ampliamente popular en la industria, los desarrollos que ponen mayor énfasis en optimización que en la simulación de diseños previamente concebidos son escasos. Tampoco se ha hecho uso intensivo de una metodología multi-objetivo, que permita obtener mayor información sobre los costos de priorizar ciertas características sobre otras, dentro de las muchas que un banco de baterías debe cumplir en seguridad, vida útil y precio. En este sentido, el desarrollo de herramientas computacionales que permitan diseño automatizado como la presentada en este trabajo son una necesidad actual de la creciente industria de baterías para la reducción de los tiempos de ingeniería requeridos para concebir un diseño adecuado para bancos para vehículos eléctricos o sistemas eléctricos de potencia.

La implementación computacional del sistema presentado en este trabajo incluye una modelación de la generación de calor y el comportamiento fluidodinámico de bancos de baterías para celdas cilíndricas y prismáticas. Esta modelación se realizó en COMSOL Multiphysics, utilizando el módulo de Transferencia de Calor Conjugado de este software con el equivalente en dos dimensiones. La decisión sobre el uso de las diferentes condiciones de borde, condiciones iniciales, mallas y equivalentes fue realizada a través de comparación dentro de COMSOL, con otros software de simulación y con trabajos previos que se encuentran en la literatura. El desarrollo de una simulación que pueda considerarse válida para realizar la comparación entre dos diseños

es parte fundamental para permitir resultados adecuados en la optimización de diferentes casos de estudio. El funcionamiento de la herramienta de optimización es posible gracias a la interfaz de comunicación entre MATLAB, programa en el cual se implementó el algoritmo evolutivo y COMSOL Multiphysics.

En este trabajo se estudia también el efecto de utilizar un perfil de conducción en el diseño del banco de baterías. En este estudio se utiliza generación de calor continua y fija y el perfil de manejo US06, diseñado para pruebas de vehículos de combustión interna y con motores eléctricos. Todo perfil induce un cierto sesgo en las soluciones y considerando que se utilizan algoritmos evolutivos que buscan maximizar el fitness, se podría caer en sobre-entrenamiento. En las pruebas realizadas, el perfil de uso US06 demostró ser consistente en su ordenamiento, aún cuando fue modificado, demostrando que es una prueba válida para comparar dos diseños de módulos de baterías para vehículos eléctricos.

Se modela el problema multi-objetivo del diseño óptimo de empaquetamientos de baterías utilizando tres funciones objetivo: temperatura máxima de las celdas del banco, diferencia de temperatura máxima entre las celdas de un mismo banco y volumen. Combinaciones de estas funciones objetivo son utilizadas en los casos de estudio. Las dos primeras funciones objetivo están basadas en la mejora de la vida útil de las celdas de ion-litio en tanto que el volumen tiene una interpretación del costo y complicación que significa para un fabricante de automóviles o BESS instalar un módulo o banco de mayor tamaño.

El algoritmo evolutivo escogido para la optimización fue MOPSO. La elección de este algoritmo se debe a su simplicidad y fácil adaptación a diferentes problemas en la programación, además del uso de una población de partículas con diferentes líderes que permite escapar de los mínimos locales y converger hacia soluciones óptimas con exploración del espacio distribuida. Para cada caso de estudio evaluado en este trabajo, se eligen valores de población y número de generaciones adecuados al problema, considerando el espacio a explorar y pruebas previas a comenzar la operación del algoritmo. En los tres casos de estudio, el costo de una simulación es bastante alto en términos de tiempo computacional, lo que hizo difícil hacer una evaluación completa que permita encontrar los valores óptimos de población y número de iteraciones para alcanzar los mejores resultados. Sin embargo, los valores utilizados en el trabajo permiten encontrar soluciones mejoradas y cuasi óptimas para todos los casos de estudio presentados.

Los casos de estudio de posicionamiento de seis celdas cilíndricas en posiciones aleatorias y de orientación para módulos de celdas prismáticas, corresponden a casos teóricos basados en información y aplicaciones reales. Ambos permiten mostrar el potencial de la herramienta de optimización multi-objetivo, obteniendo mejoras importantes en la reducción de la temperatura máxima y la diferencia de temperatura máxima entre las celdas para los bancos de baterías. La información multi-objetivo prueba ser un aporte, como se mostró en el caso de seis celdas aleatorias, debido a que permite una exploración más distribuida en el espacio y asimismo, entrega más opciones e información valiosa al diseñador, que le permitirían tomar una decisión adecuada dependiendo de las necesidades de su equipo.

El caso de estudio realizado en el vehículo solar-eléctrico Eolian III permite demostrar que mejoras importantes pueden lograrse con la optimización automatizada aún en casos donde las

variables están más restringidas. La metodología utilizada permite que el algoritmo evolutivo realice la exploración que normalmente le correspondería al equipo, tarea muy costosa, ya sea al hacerla a través de experimentos o simulaciones computacionales. En este caso además se incorporó una mejora que no estaba disponible para el equipo para esta implementación del banco de baterías, pero que en el caso de una futura versión podría mejorar el comportamiento térmico de este equipo. Este cambio corresponde a la modificación de la posición de la entrada de aire. El diseño para Eolian III fue implementado y permitirá comprobar luego de las competencias en las cuales participará, si las recomendaciones realizadas por la aplicación diseñada fueron adecuadas.

En definitiva, los primeros resultados de la implementación de un algoritmo evolutivo multi-objetivo funcionando en bucle cerrado con un simulador computacional para representar el comportamiento térmico de baterías de ion-litio son auspiciosos: La aplicación funciona adecuadamente y converge en tiempos aceptables en comparación con el tiempo que tarda un diseño en la actualidad, se mostraron mejoras en el sentido de la temperatura máxima y la diferencia de temperatura para los diferentes casos de estudio, los diseños encontrados permiten mostrar el balance entre las funciones objetivo utilizadas y además entregan al diseñador una población de soluciones sobre las cuales puede escoger. Además, la herramienta utilizando MOPSO y COMSOL es dúctil, lo que permite que eventualmente pueda incorporar más tipos de celdas de baterías, número de celdas en los módulos y bancos y otros perfiles de uso aplicados a vehículos eléctricos y sistemas eléctricos de potencia.

5.2. Trabajos Futuros

Existen diversas áreas en las que se puede trabajar para mejorar y completar la implementación como la presentada en esta memoria. El principal cuello de botella de la herramienta desarrollada corresponde al tiempo de simulación para cada diseño que se evalúa. Aún cuando estos tiempos fueron reducidos considerablemente al utilizar el equivalente de dos dimensiones y una malla suficientemente gruesa para obtener resultados aceptables en COMSOL Multiphysics, tiempos promedio cercanos a los 30 segundos todavía son muy altos para explorar adecuadamente en problemas que cuenten con mayor número de variables y rangos menos restringidos para estas. En total, entre los tres casos de estudio presentados en este trabajo, el tiempo ocupado exclusivamente en procesamiento computacional supera un mes. Mejorar el tiempo requerido para evaluar las funciones objetivo, ya sea a través de la programación en sistemas optimizados como en una GPU (Graphic Processor Unit) o la búsqueda de modelos que representen el comportamiento térmico de las baterías sin necesidad de simulaciones sería un importante avance para este trabajo y que permitiría evaluar los diseños con una mayor variedad de perfiles, más largos, incluso incorporando el efecto de degradación en el tiempo.

Del mismo modo, por las limitaciones en tiempo de simulación, hay espacio para mejorar en la búsqueda de una población y una cantidad de generaciones óptima para aprovechar al máximo tanto la exploración como el tiempo de cómputo. En este caso, por el tiempo disponible, se eligieron valores que si bien son adecuados y permiten encontrar soluciones óptimas en todos los problemas propuestos, pueden no ser exactamente los óptimos para cada una de las situaciones.

En la dirección contraria, también se puede hacer más complejo el problema incorporando como variable la solución de ventilación, esto es el tamaño de entradas de aire o velocidad, incluso la posibilidad de incorporar diferentes sistemas de refrigeración. También se puede hacer más compleja la simulación utilizando una aproximación más realista, de tres dimensiones en lugar de una de dos dimensiones. Estos elementos harían que el tiempo de simulación aumente, pero con las alternativas discutidas anteriormente podrían reducirse a valores que permitan realizar una exploración aceptable.

Otro punto que no se analiza en este trabajo corresponde al uso de otros algoritmos evolutivos como alternativa a MOPSO. Este aspecto también queda abierto para trabajo futuro, ya que diferentes técnicas de búsqueda podrían llegar a diferentes resultados en un número menor de generaciones o cálculos.

Por otra parte, para completar una herramienta que permita dar orientación completa a un diseñador de baterías de ion-litio, se requeriría una herramienta extra combinada a la de optimización térmica que permita elegir el número y tipo de celdas que se utilizarían para una aplicación particular dependiendo de los requerimientos de energía y potencia. Ambas herramientas combinadas entregarían un Frente de Pareto aún más amplio, con mayor variedad de opciones. Para hacer este avance sin embargo, nuevamente es fundamental reducir el tiempo de las simulaciones, ya que se requerirían más de estas para analizar casos con diferente número de celdas y exigencias eléctricas.

Finalmente, lo mostrado en este trabajo fue revisado exclusivamente dentro de un software de simulación computacional y debe ser validado fuera del computador para poder estar completamente seguros de la eficacia de los resultados. Esto debe hacerse con alguna plataforma real en un laboratorio que permita hacer las pruebas y comprobar las diferencias encontradas en la simulación, con el uso de sensores de temperatura o una cámara termográfica.

Bibliografía

- [1] *18650 battery test 2011.* 2011.
<http://lygte-info.dk/info/Batteries18650-2011%20UK.html>.
- [2] Battery, Hao Cheng: *Prismatic battery cells.* 2012.
- [3] Vetter, J., P. Novak, M.R. Wagner, C. Veit, K.C. Möller, J.O. Besenhard, M. Winter, M. Wohlfahrt-Mehrens, C. Vogler y A. Hammouch: *Ageing mechanisms in lithium-ion batteries.* Journal of Power Sources 147, 2005.
- [4] Pesaran, A., G.H. Kim y M. Keyser: *Integration Issues of Cells into Battery Packs for Plug-In and Hybrid Electric Vehicles.* EVS-24 International Battery, Hybrid and Fuel Cell Electric Vehicle Symposium, 2009.
- [5] Bianchi, E.: *Elementos de Electroquímica: Electrólisis y acumuladores reversibles.* 2008.
- [6] Communications, Woodbank: *MpowerUK Electropaedia.* 2012. <http://www.mpoweruk.com/>.
- [7] Buchmann, I.: *Batteries in a Portable World.* 1997.
<http://www.batteryuniversity.com>.
- [8] He, H., R. Xiong y J. Fan: *Evaluation of Lithium-Ion Battery Equivalent Circuit Models for State of Charge Estimation by an Experimental Approach.* Energies 2011 4, páginas 582–598, 2011.
- [9] D., Guasch y S. Silvestre: *Dynamic Battery Model for Photovoltaic Applications.* Progress In Photovoltaics: Research and Applications, 11, 2003.
- [10] Copetti, J.B., E. Lorenzo y F. Chenlo: *A General Battery Model for PV System.* Progress In Photovoltaics: Research and Applications, 1, 1993.
- [11] Pesaran, A. y T. Duong: *Electrothermal Analysis of Lithium Ion Batteries.* The 23rd International Battery Seminar Exhibit, 2006.
- [12] Wakihara, M. y O. Yamamoto: *Lithium ion Batteries: Fundamentals and performance.* Wiley-VCH, 1998.
- [13] Wendt, J.F.: *Computational Fluid Dynamics: An Introduction.* 2009.

- [14] IV, J.H.Lienhard y J.H.Lienhard V: *A Heat Transfer Textbook*. Massachusetts Institute of Technology, 2008.
- [15] Pesaran, A.: *Battery thermal models for hybrid vehicle simulations*. Journal of Power Sources 110, 2002.
- [16] Pesaran, A., G.Kim y K.Smith: *Accelerating Battery Design Using Computer-Aided Engineering Tools*. 2010.
- [17] Jarrett, A.: *Multi-Objective Design Optimization of Electric Vehicle Battery Cooling Plates Considering Thermal And Pressure Objective Functions*. 2011.
- [18] Hu, X.: *Battery Thermal Management in Electric Vehicles*. ANSYS, 2011.
- [19] Hu, X.: *A Foster Network Thermal Model for HEV/EV Battery Modeling*. IEEE Transactions on Industry Applications, 47(4), 2011.
- [20] Muratori, M.: *Thermal Characterization of Lithium-Ion Battery*. Politecnico di Milano, Facoltà de Ingegneria Industriale, 2009.
- [21] Mi, C.: *Advanced Electro-Thermal Modeling of Lithium-Ion Battery System for Hybrid Electric Vehicle Applications*. Vehicle Power and Propulsion Conference, 2007. VPPC 2007. IEEE, 47(4), 2007.
- [22] Pesaran, A., G.H. Kim, K. Smith, S. Santhanagopalan y K J. Lee: *Computer-Aided Engineering for Electric Drive Vehicle Batteries*. U.S. DOE Hydrogen Program And Vehicle Technologies Program Annual Merit Review Peer Evaluation Meeting, 2011.
- [23] Pesaran, A., A. Vlahinos y S.D.Burch: *Thermal Performance of EV and HEV Battery Modules And Packs*. 1997.
- [24] Pesaran, A., A. Vlahinos, G-H.Kim y D.Bharathan: *Electro-thermal modeling to improve battery design*. 2005 IEEE Vehicle Power and Propulsion Conference, 2005.
- [25] Pesaran, A. y G-H.Kim: *Battery Thermal Management System Design Modeling*. 22nd International Battery, Hybrid and Fuel Cell Electric Vehicle Conference and Exhibition (EVS-22), 2006.
- [26] Pesaran, A., D.Swan, J.Olson, J.T.Guerin, S.Burch, R.Rehn y G.D. Skellenger: *Thermal Analysis And Performance of a Battery Pack for a Hybrid Electric Vehicle*. NREL, 2002.
- [27] Pesaran, A., M. Zolot y M. Mihalic: *Thermal Analysis of Toyota Prius Battery Pack*. 2002.
- [28] Pesaran, A., M. Zolot, M. Mihalic, K.Kelly, M.Keyser y A.Hieronymus: *Thermal Evaluation of the Honda Isight Battery Pack*. 2001.
- [29] Coello, C.A.: *20 Years of Evolutionary Multi-Objective Optimization: What Has Been Done and What Remains To Be Done*. 2006.
- [30] Konak, A., D. Coit y A. Smith: *Multi Objective Optimization Using Genetic Algorithms: A tutorial*. Reliability Engineering and System Safety, 91, 2006.

- [31] Coello, C.: *Evolutionary Algorithms for Solving Multi-Objective Problems*. 2004.
- [32] Coello, C.: *A Tutorial on Multi-Objective Optimization Using Metaheuristics*. Computational intelligence in multi-criteria decision-making, 2009. MCDM '09., 2009.
- [33] Wright, J., H. Loosemore y R. Farmani: *Optimization of building thermal design and control by multi-criterion genetic algorithm*. Energy and Buildings, 34(9):959–972, 2002.
- [34] Hamdy, M., A. Hasan y K. Siren: *Impact of adaptive thermal comfort criteria on building energy use and cooling equipment size using a multi-objective optimization scheme*. Energy and Buildings, 43(9):2055–2067, 2011.
- [35] Wang, G. Zao, G. y H. Lin: *Multi-objective optimization design of the heating cooling channels of the steam-heating rapid thermal response mold using particle swarm optimization*. International Journal of Thermal Sciences, 50(5):790–802, 2011.
- [36] Wang, G. Zao, G. y H. Lin: *Research on optimization design of the heating cooling channels for rapid heat cycle molding based on response surface methodology and constrained particle swarm optimization*. Expert Systems with Applications, 38(6):6705–6719, 2010.
- [37] Sidy Ndao, J.: *Multi-objective thermal design optimization and comparative analysis of electronic cooling techniques*. International Journal of Heat and Mass Transfer, 52:4317–4326, 2009.
- [38] Toffolo, A. y A. Lazzareto: *Evolutionary Algorithms for Multi Objective Energetic And Economic Optimization In Thermal System Design*. Energy, 27:549–567, 2002.
- [39] Sambou, V., B. Lartigue y F. Monchoux: *Thermal Optimization Of Multilayered Walls Using Genetic Algorithms*. Energy and Buildings, 41(10):1031–1036, 2009.
- [40] Nariman-Zadeh, N., K. Atashkari, A. Jamali, A. Pilechi y X. Yao: *Inverse modelling of multi-objective thermodynamically optimized turbojet engines using GMDH-type neural networks and evolutionary algorithms*. Engineering Optimization, 37(5):437–462, 2004.
- [41] Raju, U., G. Chrysler y A. Kasaire: *Multi-Objective optimization entailing computer architecture and thermal design for microprocessors*. 11th Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems, 2008. IOTHERM 2008., 2008.
- [42] Mazumder, P. y K. Shahookar: *VLSI Cell Placement Techniques*. ACM Computing Surveys, 1991.
- [43] Nicosia, G., S. Rinaudo y E. Sciacca: *An Evolutionary Algorithm based approach to robust Analog Circuit design using constrained Multi Objective Optimization*. Knowledge-Based Systems, 21(3):175–183, 2008.
- [44] Stanley, T. y T. Mudge: *A Parallel Genetic Algorithm for Multiobjective Microprocessor Design*. Proceedings of the Sixth International Conference on Genetic Algorithms, 1995.

- [45] Bright, M. y T. Arslan: *Multi-Objective Design Strategy for High-Level Low Power Design of DSP Systems*. Proceedings of the 1999 IEEE International Symposium on Circuits and Systems, 1999. ISCAS '99., 1999.
- [46] Thomson, R. y T. Arslan: *On the Impact of Modelling, Robustness, and Diversity to the Performance of a Multi-Objective Evolutionary Algorithm for Digital VLSI System Design*. The 2003 Congress on Evolutionary Computation, 2003. CEC '03., 2003.
- [47] Kashfi, F., S. Hatami y M. Pedram: *Multi-Objective Optimization Techniques for VLSI Circuits*. 12th International Symposium on Quality Electronic Design (ISQED), 2011, 2011.
- [48] Sait, S., H. Youssef y J.Khan: *Fuzzy Evolutionary Techniques for VLSI Placement*. 2003.
- [49] Deb, K. y T. Goel: *A Hybrid Multi Objective Evolutionary Approach to Engineering Shape Design*. Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization (EMO-2001), 2007.
- [50] Coello, C., A. Arias-Montaña y E. Mezura-Montes: *Evolutionary Algorithms Applied to Multi-Objective Aerodynamic Shape Optimization*. Computational Optimization, Methods and Algorithms Studies in Computational Intelligence Volume 356, 2011, 2009.
- [51] Dufo-López, R. y J. L. Bernal-Agustín: *Multi Objective Design of a PV Wind Diesel Hydrogen Battery Systems*. Renewable Energy, 33(12):2559–2572, 2008.
- [52] Mohamed, F. y H. Koivo: *Micro Grid Online Mangment and Balancing Using Multi Objective Optimization*. Power Tech, 2007 IEEE Lausanne, 2006.
- [53] Moghaddam, A. Anvari, A. Seifi y T. Niknam: *Multi Objective Operation Management of Renewable Micro Grid with Back Up Microturbine /Fuel Cell / Battery Hybrid Power Source*. Journal of Intelligent Learning Systems and Applications, 5(1), 2011.
- [54] Dufo-López, R., J. L. Bernal-Agustín y J. M. Yusta-Loyo: *Multi Objective Optimization Minimizing Cost and Life Cycle Emissions of Stand Alone PV Wind Diesel Systemas with Batteries Storage*. Applied Energy, 88(11):4033–4041, 2011.
- [55] Dufo-López, R. y J. L. Bernal-Agustín: *Design of Isolated Hybrid Systems Minimizing Costs and Pollutant Emissions*. Renewable Energy, 31(14):2227–2244, 2005.
- [56] Hu, X., Z. Wang y L. Liao: *Multi Objective Optimization of HEV Fuel Economy and Emissions Using Evolutionary Computation*. SAE 2004 World Congress Exhibition, 2004.
- [57] Sarikhani, A. y O. A. Mohammed: *Hybrid GA-PSO Multi-Objective Design Optimization of Coupled PM Synchronous Motor-Drive Using Physics-Based Modeling Approach*. IEEE Transactions on Magnetics, 2010.
- [58] Vlahinos, A., K.Kelly, A. Pesaran y T. Penney: *Empowering Engineers to Generate Six Sigma Quality Designs*. First Annual Quality Paper Symposium, American Society for Quality, Automotive Division, 2003.

- [59] Li, Zhe Zhe, Tai Hong Cheng, Dong Ji Xuan, Ming Ren, Gui Ying Shen y Yun De Shen: *Optimal Design for Cooling System of Batteries using Design of Experiments and Response Surface Method*. International Journal of Precision Engineering and Manufacturing, 13(9):1641–1645, 2012.
- [60] Mousavi, M., S. Hoque, S. Rahnamayan, I. Dincer y G.F. Naterer: *Optimal Design of an Air-Cooling System for a Li-Ion Battery Pack in Electric Vehicles with a Genetic Algorithm*. 2011 IEEE Congress on Evolutionary Computation (CEC), 2011.
- [61] P.Kumar y P.Bauer: *Parameter Extraction of Battery Models Using Multiobjective Optimization Genetic Algorithms*. 14^o International Power Electronics and Motion Control Conference, 2010.
- [62] Coello, C. y M. Salazar Lechuga: *MOPSO: A Proposal for Multiple Objective Particle Swarm Optimization*. Proceedings of the 2002 Congress on Evolutionary Computation, 2002. CEC '02, 2002.
- [63] Sabbah, R., R. Kizilel, J.R. Selman y S. Al-Hallaj: *Active (air-cooled) vs. passive (phase change material) thermal management of high power lithium-ion packs: Limitation of temperature rise and uniformity of temperature distribution*. Journal of Power Sources, 182:630–638, 2008.
- [64] Molicel: *Product Datasheet IMR-18650E*. 2012.
- [65] Berdichevsky, G., K. Kelty, JB Straubel y Erik Toomre: *The Tesla Roadster Battery System*. 2007.
- [66] Leo-Industries: *Superior Lithium Polymer Battery SPB636395*. 2012.
- [67] Burgos, C. y S. Fehlandt: *Proyecto: Construcción Caja Banco de Baterías de EOLIAN 2 para WSC*. 2011.
- [68] Agency, Environmental Protection: *Federal Test Procedure Revisions*. 1996. <http://www.epa.gov/otaq/sftp.htm>.
- [69] Yakushi, Bob: *EV / HEV Safety, Nissan Motor Company*. 2012.

Apéndice A

Proceso de Fabricación de celdas de baterías de ion-litio

A.1. Fabricación de Ánodos y Cátodos

La fabricación de celdas de baterías de ion-litio es similar a la que se ha establecido para las celdas de níquel-cadmio, con modificaciones menores debido a la alta reactividad del litio como material. El material activo, correspondiente a carbono para el ánodo y litio-metal para el cátodo llega a la planta de fabricación como polvo. Los materiales deben ser revestidos con los colectores de corriente, de aluminio en el caso del cátodo y de cobre en el caso del ánodo.

El proceso completo para construir las láminas o tiras de ánodos y cátodos tiene cinco etapas principales: Mezcla, Revestimiento, Compresión, Secado y Cortado. En la figura A.1 se muestra el proceso para el ánodo y en la figura A.2 para el cátodo. La principal diferencia está en el uso de los materiales, con los procesos siendo prácticamente idénticos para ambos electrodos.

A.2. Fabricación de la Celda

La fabricación de la celda corresponde a formar la estructura tipo *sandwich* que permite al separador quedar entre cátodo y ánodo en el electrolito. Este proceso se realiza posteriormente a la fabricación de las tiras de ánodo y cátodo, descritas en A.1.

Las celdas prismáticas sólidas y prismáticas de tipo *pouch* se fabrican en base a pilas de láminas de ánodos y cátodos, que luego son conectadas adecuadamente. Las celdas cilíndricas se fabrican en base a rollos, y solo tienen un ánodo y un cátodo enrollado en forma de espiral alrededor de la celda. Esto hace que las celdas cilíndricas sean más fáciles y baratas de fabricar que las prismáticas, ya que no hay que hacer conexiones entre terminales internos.

Luego de haber insertado ánodos y cátodos en el *case* de la celda, se procede a insertar el

electrolito a través de un orificio dejado en la cobertura de la celda. Este proceso se realiza en una sala seca, ya que el electrolito reacciona violentamente con la humedad, generando gases tóxicos. Una vez que se ha insertado el electrolito, la celda se sella y se comienza un proceso de pruebas, de carga y descarga que permiten la activación de materiales. Este proceso se denomina Formación y permite generar la interfaz sólida entre electrolito y ánodo. La primera carga se realiza a través de un aumento progresivo de voltaje, contrario al proceso habitual que inicia con corriente constante y concluye con voltaje constante. Después de la formación se realizan otras pruebas de carga y descarga para comprobar la calidad, resistencia interna y capacidad de la celda, además de descartar celdas falladas. Con este proceso, existe una mayor probabilidad de que las fallas denominadas "muertes infantiles" por problemas de fabricación de la celda se den dentro de la fábrica y no en manos del cliente.

Otras pruebas optativas que se realizan en algunos laboratorios corresponden a utilizar espectrómetros de masa, calorímetros, pruebas de vibraciones, máquinas de stress mecánico y escaneo microscópico.

El proceso de fabricación de la celda prismática se muestra en la figura A.3 y el de la celda cilíndrica se muestra en la figura A.4

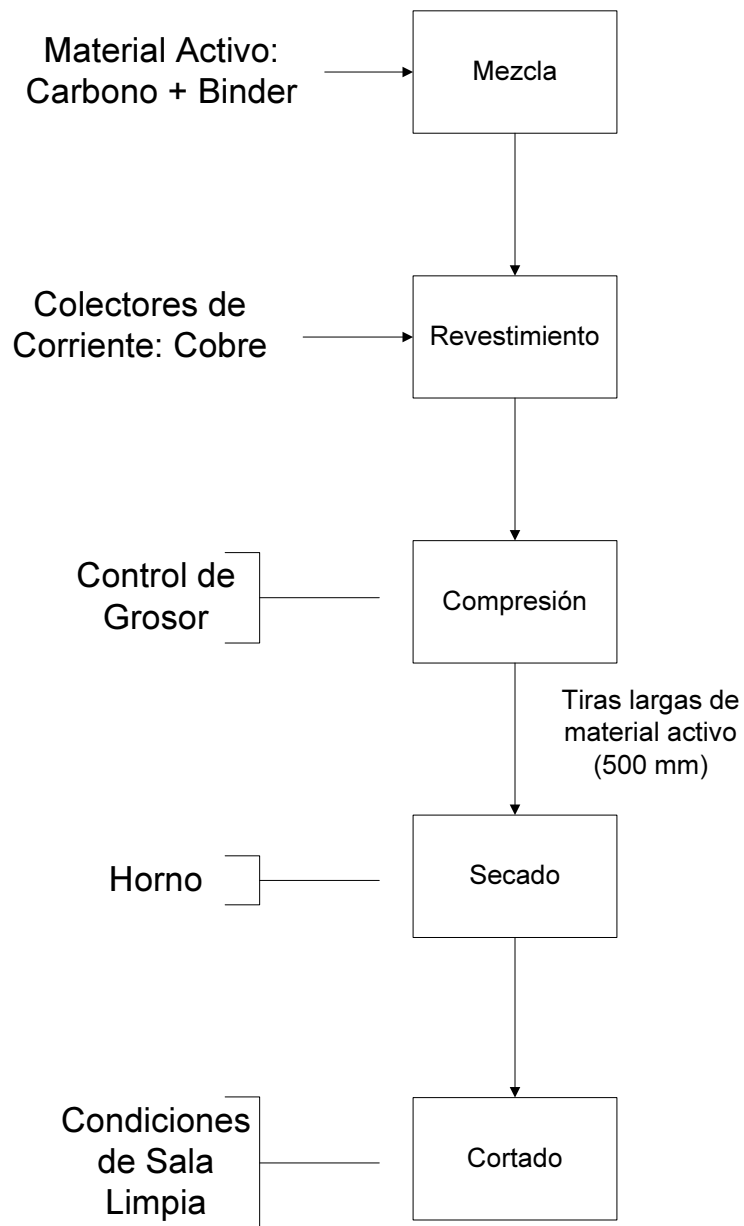


Figura A.1: Proceso de fabricación de las tiras de ánodo.

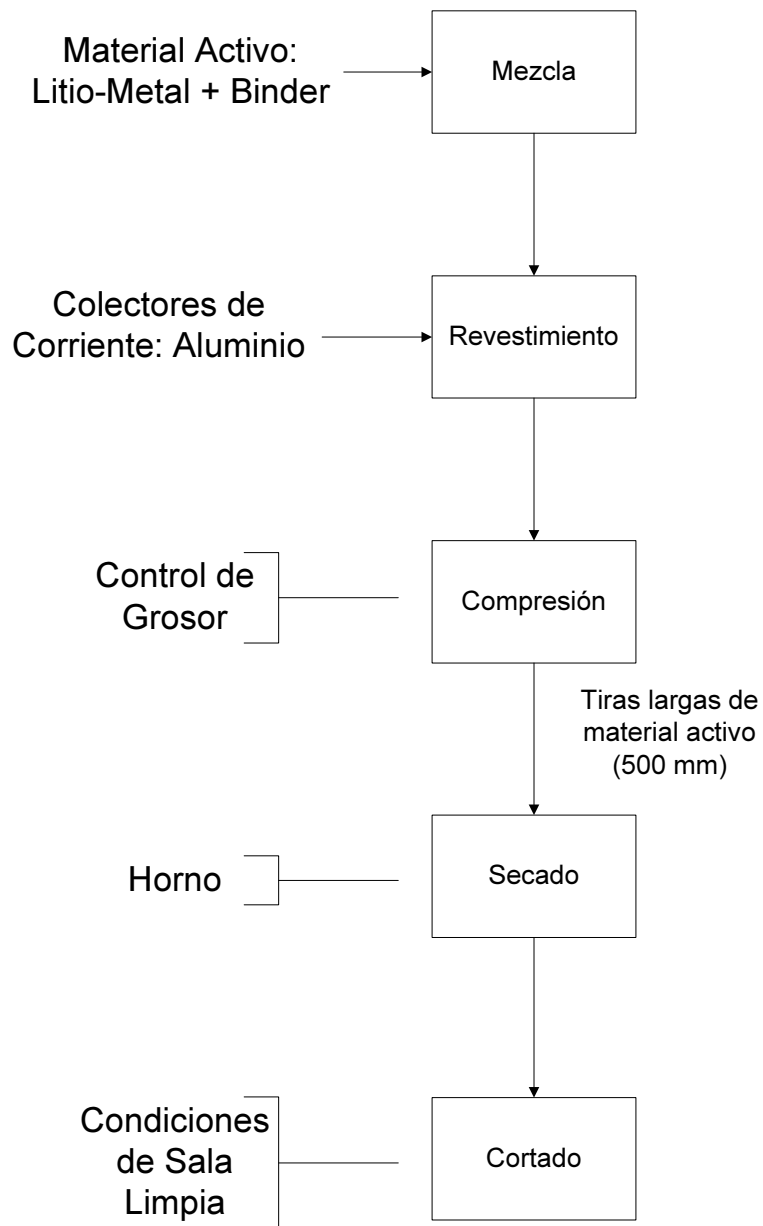


Figura A.2: Proceso de fabricación de las tiras de cátodo.

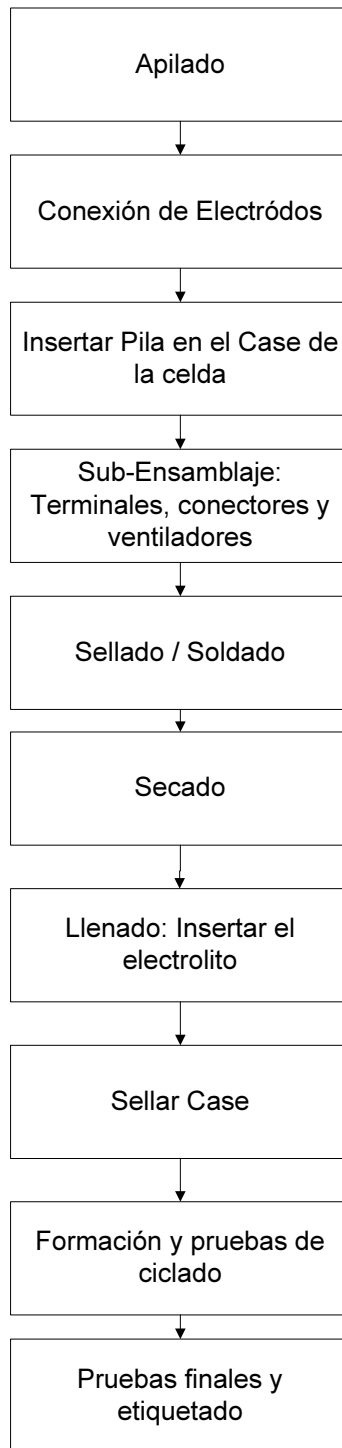


Figura A.3: Proceso de fabricación de las celdas prismáticas.

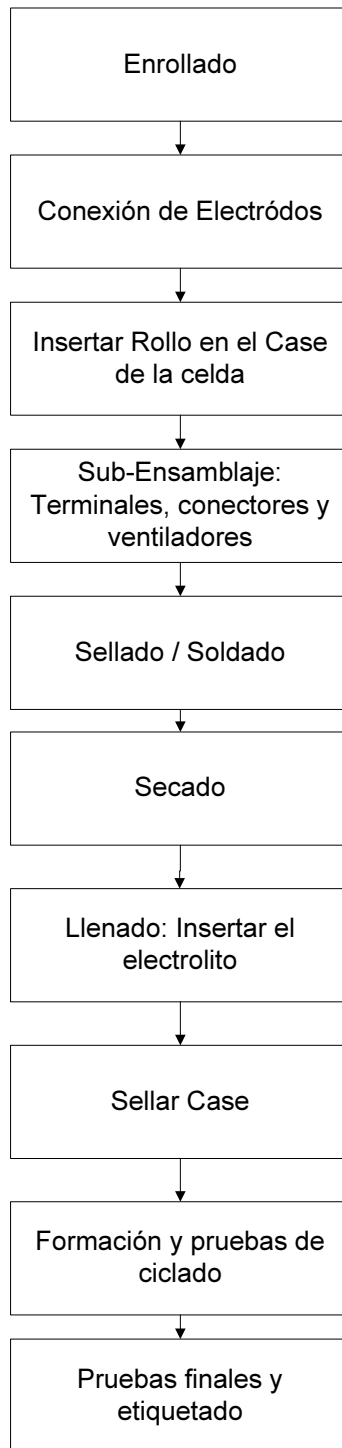


Figura A.4: Proceso de fabricación de las celdas cilíndricas.

Apéndice B

Criterios de Decisión en Problemas Multi-Objetivo

Existen tres formas de buscar las soluciones óptimas de un problema multi-objetivo y luego escoger la más adecuada bajo algún criterio. Estas estrategias son: tomar la decisión A Priori, de forma Progresiva o a Posteriori. Para cada una de estas tres opciones existen diferentes alternativas de implementación, las que se describirán a continuación. La mayor parte de estas estrategias de decisión no tiene directa relación con el algoritmo de búsqueda que se utilice para encontrar el Frente de Pareto, sin embargo, puede ser necesario hacer pequeños cambios a las arquitecturas para que funcione de forma ordenada. En general se ha indicado que los algoritmos evolutivos trabajan y funcionan mejor en el formato de búsqueda “A Posteriori”, que corresponde a buscar las soluciones en primer lugar y luego se entrega el Frente de Pareto al operador, quien decide con un algún criterio sólo una vez que tiene todas las soluciones a la vista.

B.1. Criterios A Priori

Los criterios a priori asumen que existen algunos objetivos deseables antes de que se realice la búsqueda de las soluciones óptimas, por lo tanto, se modifican las funciones objetivo y la modelación del problema para hacer calzar esta visión con la resolución del problema. Esta modificación puede cambiar la estructura y forma del Frente de Pareto, por lo tanto, se está induciendo soluciones óptimas en el problema que no necesariamente son las mismas que se encontrarán bajo otra modelación del mismo.

- **Método del Criterio Global (Global Criterion Method):** Consiste en la minimización de una función que mide la cercanía existente entre la solución y una solución ideal. Existen diferentes variantes de este método, por ejemplo, asumiendo que existen regiones ideales, utilizando diferentes medidas de distancia o bien ponderando los errores en las diferentes funciones objetivo para priorizar algunas más importantes sobre otras. En definitiva, el resultado siempre es la solución que minimiza la distancia vectorial.

- **Programación con Objetivos (Goal Programming):** En este método, similarmente al del Criterio Global, se definen valores deseados para cada objetivo y se incorpora lo anterior como una función objetivo o restricción más. Incluso podría usarse como única función objetivo. La función objetivo que se agrega es la sumatoria de los módulos de las diferencias entre los valores del individuo evaluado en las funciones objetivos y los valores deseados por el diseñador. Se puede generalizar también utilizando pesos y ponderadores para las diferentes funciones y diferencias positivas y negativas.
- **Método de Alcance de Objetivos (Goal Attainment Method):** Corresponde en agregar una nueva restricción al problema, que tiene una variable α por determinar. Esta variable buscará su mínimo valor para mantener el problema en la zona factible, de modo tal que se cumpla que:

$$(B.1) \quad b + \alpha w \geq f(x)$$

Donde b representa los valores deseados puestos por el diseñador, $f(x)$ representa los valores de la función objetivo y w corresponde a un vector de ponderaciones colocado como factor de seguridad. Tanto b como w y $f(x)$ son vectores en tanto que α es un valor numérico. Se busca el primer punto en el cual la recta $b + \alpha w$ intersecta la región factible. Luego, el problema de optimización de α se reduce a dos opciones:

- $\alpha \geq 0 \rightarrow$ El objetivo puesto por el diseñador es infactible
- $\alpha \leq 0 \rightarrow$ El objetivo puesto por el diseñador es factible y todavía puede ser aumentado.

En general, se hace una iteración de este algoritmo, sin embargo, podrían hacerse varias veces con el diseñador cambiando sus aspiraciones, lo que la convertiría en una técnica progresiva.

- **Método Lexicográfico:** Este método es uno de los más populares en la resolución de problemas multi-objetivo, ya que reduce el problema a uno de un objetivo, pero optimizado muchas veces. Lo anterior puede ser eficiente en algunos casos en los cuales existen objetivos claramente más importantes que otros. El método lexicográfico obliga al diseñador a ordenar los objetivos por importancia (o bajo algún criterio, incluso aleatoriamente), y los va minimizando uno a uno. Cada vez que se encuentra la solución óptima, no se almacena la solución encontrada sino que se impone que la función objetivo mantenga ese valor como restricción y se optimiza la siguiente función en el orden. Es decir, se optimiza secuencialmente, lo cual obviamente hace que se explore el espacio de manera no equitativa. Usualmente, no se usa este método en algoritmos evolutivos.
- **Optimización Min-Max:** En este método, se intenta determinar los incrementos relativos más pequeños que se pueden realizar en cada función objetivo después de extraer los valores extremos de las funciones objetivo.
- **Teoría de Utilidad (Multi-Attribute Utility Theory):** Corresponde a la construcción, por parte del diseñador o investigador, de una región de utilidad (como las de las curvas de oferta

y demanda) para cada una de las funciones objetivo. Lo importante está en el detalle de las “esquinas” es decir, cuánto vale el aumentar una función objetivo en desmedro de la otra. Se agrega entonces como objetivo el maximizar la utilidad. El problema de esta estrategia es que funciona para un número acotado de objetivos ya que al contar con más, rápidamente se complejiza el problema de diseñar las funciones de utilidad y se hace muy costoso en tiempo hacerlo.

- **Valor del Trade-Off:** Consiste en calcular las derivadas parciales de cada objetivo con respecto a los otros. En general, se definen valores de umbrales y de aspiración para cada derivada parcial, que permiten tomar la decisión correcta, que es llevar todas dichas derivadas a cero. Es un mecanismo muy complejo computacionalmente y al igual que la Teoría de Utilidad requiere de un intensivo trabajo del operador.
- **ELECTRE:** ELECTRE es una sigla que significa “Elimination and Et Choice Translating Algorithm”. Corresponde a una relación de orden entre diferentes soluciones, por lo tanto es una forma de resolver el problema en sí. Se definen valores de umbral para dos medidas: concordancia y discordancia entre cada solución. Cuando se tienen todas las relaciones, se decide gráficamente cuales soluciones dominan a otras.
- **PROMETHEE:** PROMETHEE es una sigla que significa “Preference Ranking Organization Method for Environment Evolution”. También es una forma de resolver el problema en sí, es un método de ranking en base a seis funciones pre-definidas.

B.2. Criterios Progresivos

Los criterios progresivos son aquellos en los cuales existe una suerte de interacción entre el operador y el programa de optimización. La búsqueda va cambiando sus prioridades y objetivos según lo que se va logrando y utilizando el feedback del operador. Normalmente los criterios progresivos funcionan de la siguiente manera:

- Buscan una solución no dominada en el espacio de objetivos.
- Entregan la solución como output, reciben la reacción del operador, quien modifica los objetivos de forma acorde a sus necesidades y deseos para cada objetivo.
- Repetir los dos primeros pasos hasta llegar a una solución satisfactoria.

Los métodos progresivos más utilizados son los siguientes:

- **ProTrade:** El nombre completo de ProTrade es “Probabilistic Trade-Off Development Method”. Está diseñado para problemas que tienen funciones objetivo probabilísticas y restricciones probabilísticas. La solución encontrada pasa por una función de utilidad multi-atributo antes de ser entregada al operador. Una vez que este la ve, puede modificar las funciones de utilidad y las probabilidades, reordenando así la importancia de los objetivos para encontrar soluciones mejores en uno u otro objetivo.

- **STEP / STEM:** Consiste en una técnica iterativa, en la cual se articulan preferencias y se va convergiendo paso a paso a una solución mejor. Para esto, se define una función objetivo especial, ponderada al estilo Optimización Min-Max y en cada paso, se obtiene una solución que es evaluada por el operador. Si todavía no es satisfactoria, el operador elige algunos objetivos que podrían ser “disminuidos” en una cierta cantidad, con el objetivo de mejorar otros (más importantes) y se coloca lo anterior en las restricciones. El problema de este método es que no lograr capturar perfectamente los trade-off ya que se están interviniendo las funciones objetivo.
- **SEMOPS:** La sigla SEMOPS corresponde a Sequential Multi-Objective Problem Solving Method. Este método diseña una función objetivo nueva basada en aspiraciones y objetivos máximos para cada función. Los objetivos máximos son los ideales, y probablemente no sean alcanzables en tanto que las aspiraciones son realistas. Cada vez que un nivel de objetivo sea superado, el valor de una variable se hará menor a 1. En base a lo anterior, se modifican las restricciones, las aspiraciones y se resuelve un nuevo problema intentando nuevamente llevar todas las variables a valores menores a 1.

B.3. Criterios A Posteriori

En general los criterios a posteriori son los más utilizados en algoritmos evolutivos ya que corresponden a aquellos en que se realiza la búsqueda en primer lugar y luego se toma la decisión por parte del operador en base a algún criterio. Es importante considerar que una estrategia “A Posteriori” es simplemente que el operador elija sin un criterio formado y simplemente basado en su conocimiento y sus aspiraciones con el problema, sin utilizar ninguna técnica especial en apariencia.

- **Combinación Lineal de Pesos:** Corresponde a tomar la decisión final en base a una función mono-objetivo, pero después de haber realizado la búsqueda en el espacio multi-objetivo. Es decir, se evalúa con pesos al final (dados por el operador) de modo tal de que no se intervenga la exploración del espacio, sino que solamente es un criterio de desempate entre soluciones óptimas según Pareto.
- **Método del Trade-Off (épsilon constraint):** Consiste en resolver un nuevo sub-problema en el cual se busca minimizar las diferentes funciones objetivo ordenadas secuencialmente, permitiendo que cada una de las otras funciones se mueva a lo más un valor pequeño (épsilon, ϵ). Sirve para encontrar soluciones mejores priorizando una función, pero una vez que ya se tiene el Frente de Pareto.
- **Muestreo Independiente:** Consiste en elegir una única función objetivo y elegir la solución óptima en base al valor máximo o mínimo que se tenga en dicha función, olvidándose del resto. Este muestreo puede ser aleatorio o bien siempre elegir la misma función, dado que es más importante que el resto. Puede interpretarse como utilizar la estrategia de suma ponderada cuando se coloca un peso de valor uno a una función objetivo y cero a todas las otras.

- **Selección Modificada:** Este método fue propuesto en el algoritmo VEGA (Vector Evaluated Genetic Algorithm) y corresponde también a una selección en base a una separación de poblaciones, optimizando una función objetivo en cada población (luego reduciendo el problema a mono-objetivo) y luego realizando recombinación para obtener nuevas soluciones. No es clara que la exploración del espacio sea adecuada, por eso puede usarse en fase final luego de tener un set de elementos Pareto óptimos solo para hacer más distribuido el rango de soluciones.
- **Uso de Ranking, Nichos y Elitismo:** Los mecanismos que utilizan los algoritmos evolutivos para ajustar el Fitness son considerados métodos a posteriori, ya que se calculan una vez que ya se tiene una generación de elementos y el operador no tiene influencia en ellos.
- **Métodos Híbridos:** Estas alternativas “clásicas” permiten terminar el progreso de un algoritmo cuando la búsqueda ya ha terminado. La Distribución Paralela consiste en generar “poblaciones vecinas” en un algoritmo evolutivo que corresponden a una población a través de perturbaciones, y luego escoger los mejores individuos padres en base a todas las poblaciones, en tanto que la Búsqueda Local de soluciones es una exploración numérica en torno a cada solución del Frente de Pareto encontrado.

Apéndice C

Imágenes banco de baterías Eolian III

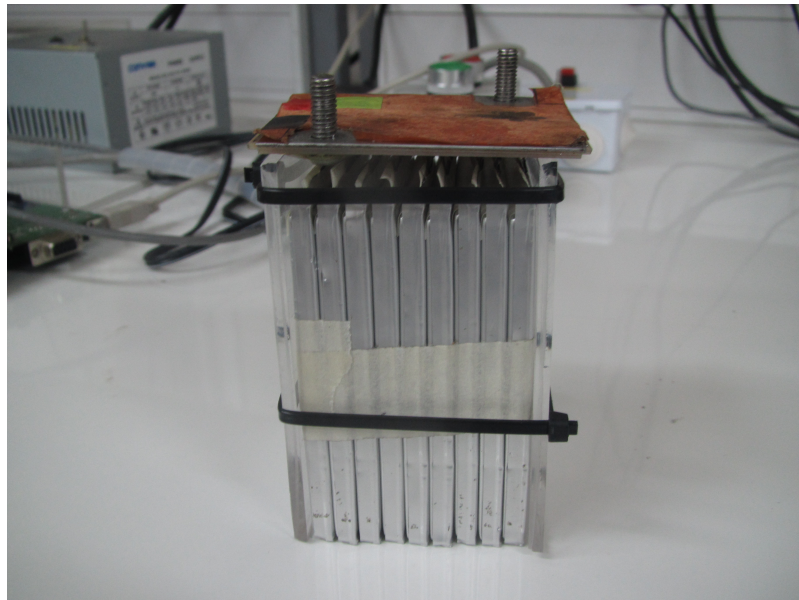


Figura C.1: Módulo de baterías Eolian III.

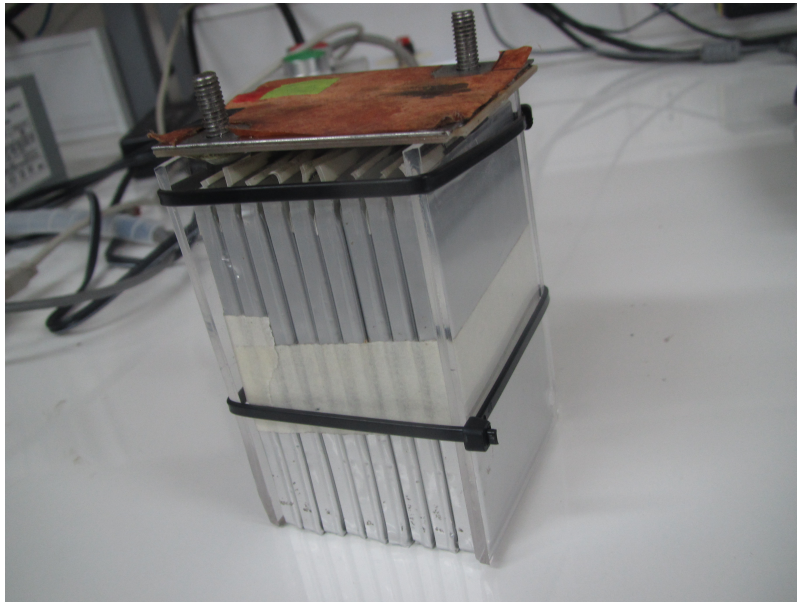


Figura C.2: Módulo de baterías Eolian III.



Figura C.3: Banco de baterías Eolian III.



Figura C.4: Banco de baterías Eolian III.

Apéndice D

Resultados completos caso celdas aleatorias

Tabla D.1: Valores de la posición de las celdas 1, 2, 3, 4, 5 y 6 en la horizontal.

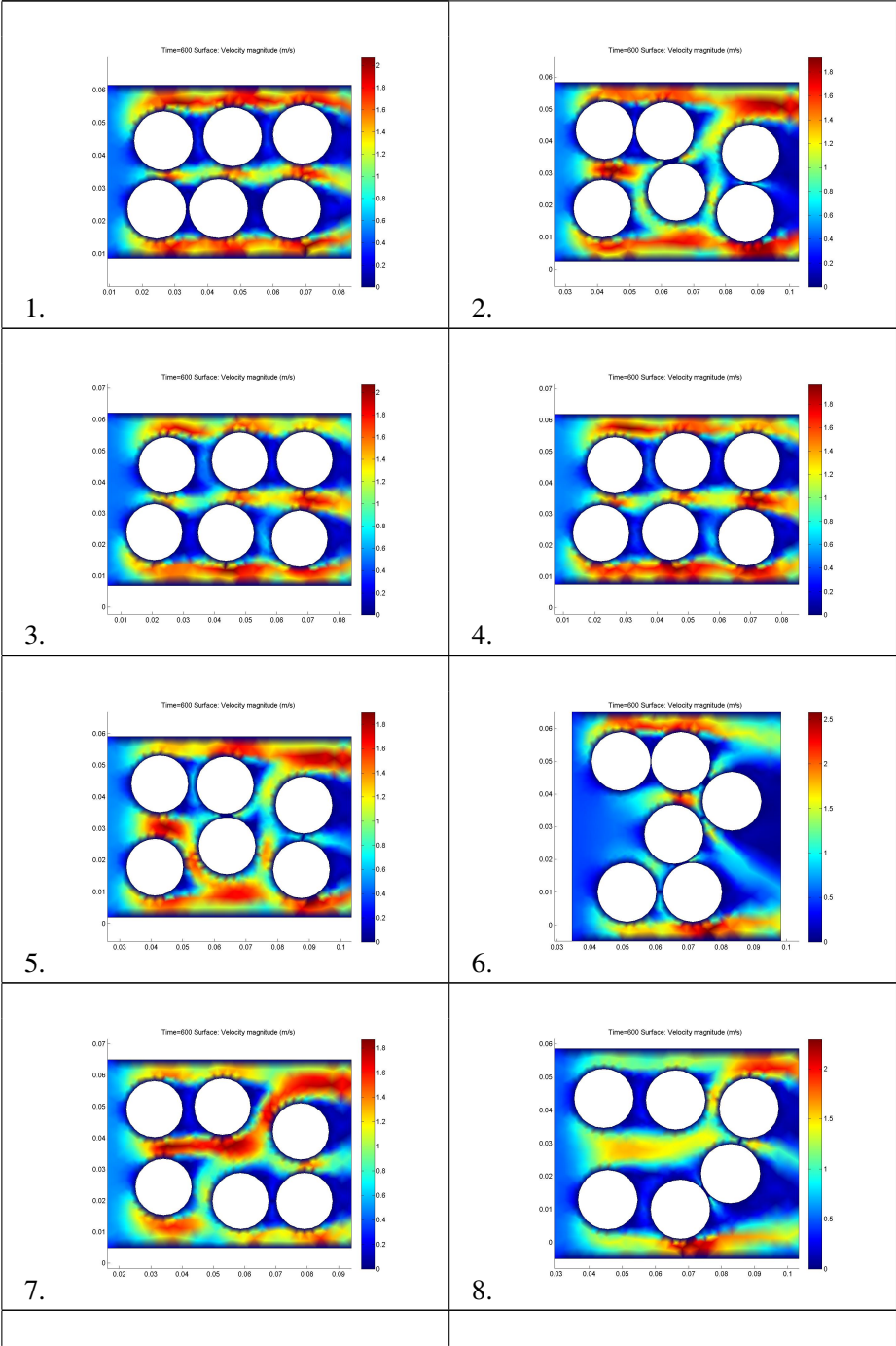
Solución	x1	x2	x3	x4	x5	x6
1	0,0244	0,0433	0,0656	0,0265	0,0476	0,0689
2	0,0415	0,0878	0,0610	0,0422	0,0862	0,0647
3	0,0207	0,0437	0,0673	0,0247	0,0481	0,0689
4	0,0221	0,0442	0,0687	0,0265	0,0482	0,0704
5	0,0412	0,0412	0,0412	0,0412	0,0412	0,0412
6	0,0833	0,0656	0,0495	0,0677	0,0513	0,0713
7	0,0587	0,0529	0,0791	0,0779	0,0341	0,0312
8	0,0827	0,0444	0,0883	0,0661	0,0675	0,0456
9	0,0171	0,0645	0,0210	0,0747	0,0361	0,0895
10	0,0420	0,0589	0,0919	0,0438	0,0654	0,0676
11	0,0562	0,0375	0,0961	0,0807	0,0953	0,0763
12	0,0250	0,0464	0,0704	0,0245	0,0565	0,0807
13	0,0474	0,0388	0,0957	0,0630	0,0668	0,0911
14	0,0420	0,0630	0,0932	0,0393	0,0620	0,0684
15	0,0280	0,0992	0,0780	0,0825	0,0575	0,0393
16	0,0860	0,0388	0,0918	0,0202	0,0652	0,0585
17	0,0690	0,0545	0,0309	0,0885	0,0525	0,0927
18	0,0595	0,0681	0,0210	0,0822	0,0217	0,0876
19	0,0441	0,0580	0,0547	0,0244	0,0737	0,0844
20	0,0775	0,0782	0,0230	0,0428	0,0189	0,0560
21	0,0364	0,0291	0,0557	0,1000	0,0776	0,0766
22	0,0776	0,0391	0,0697	0,0160	0,0931	0,0616
23	0,0280	0,0219	0,0891	0,0960	0,0578	0,0834

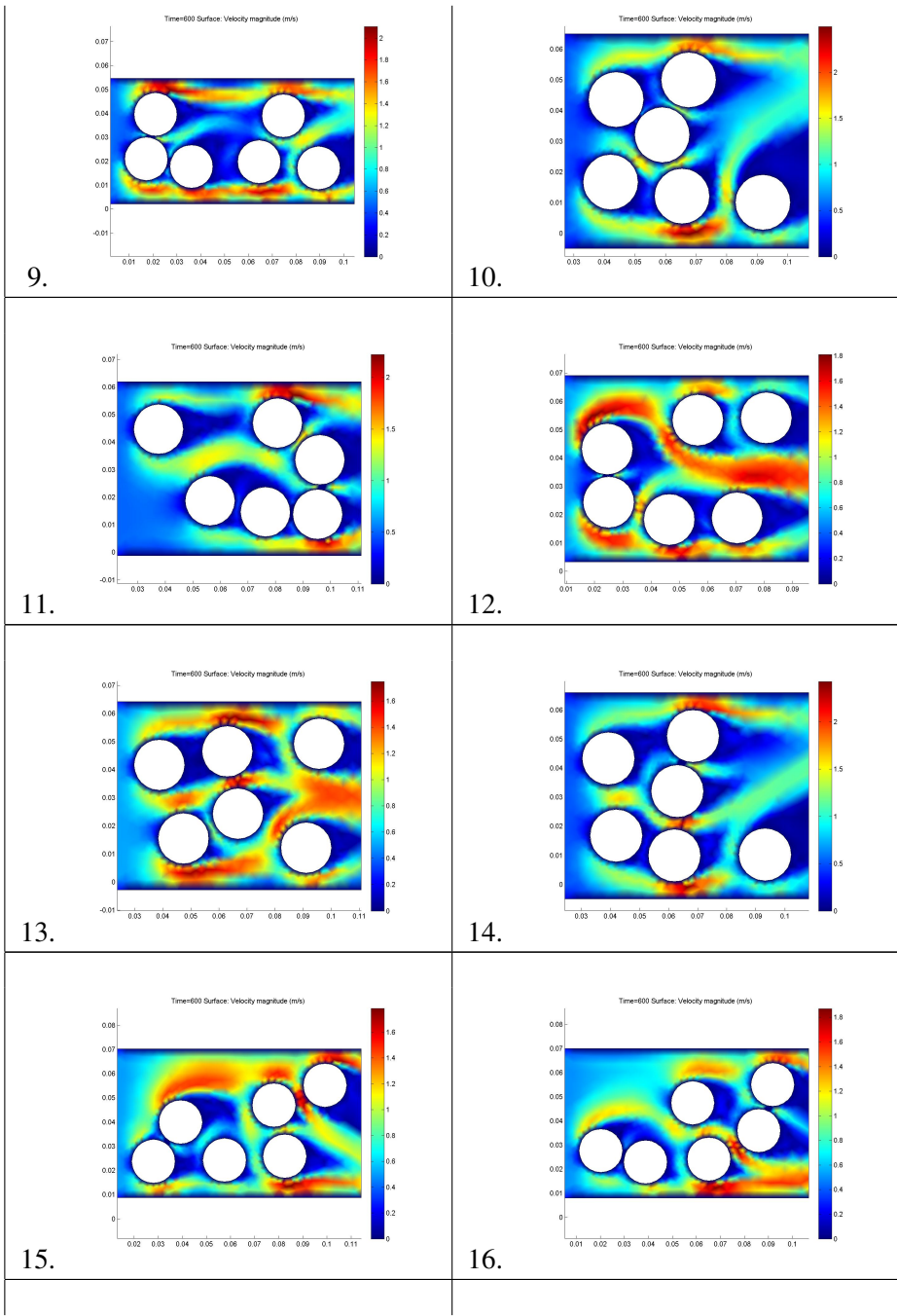
Tabla D.2: Valores de la posición de las celdas 1, 2, 3, 4, 5 y 6 en la vertical.

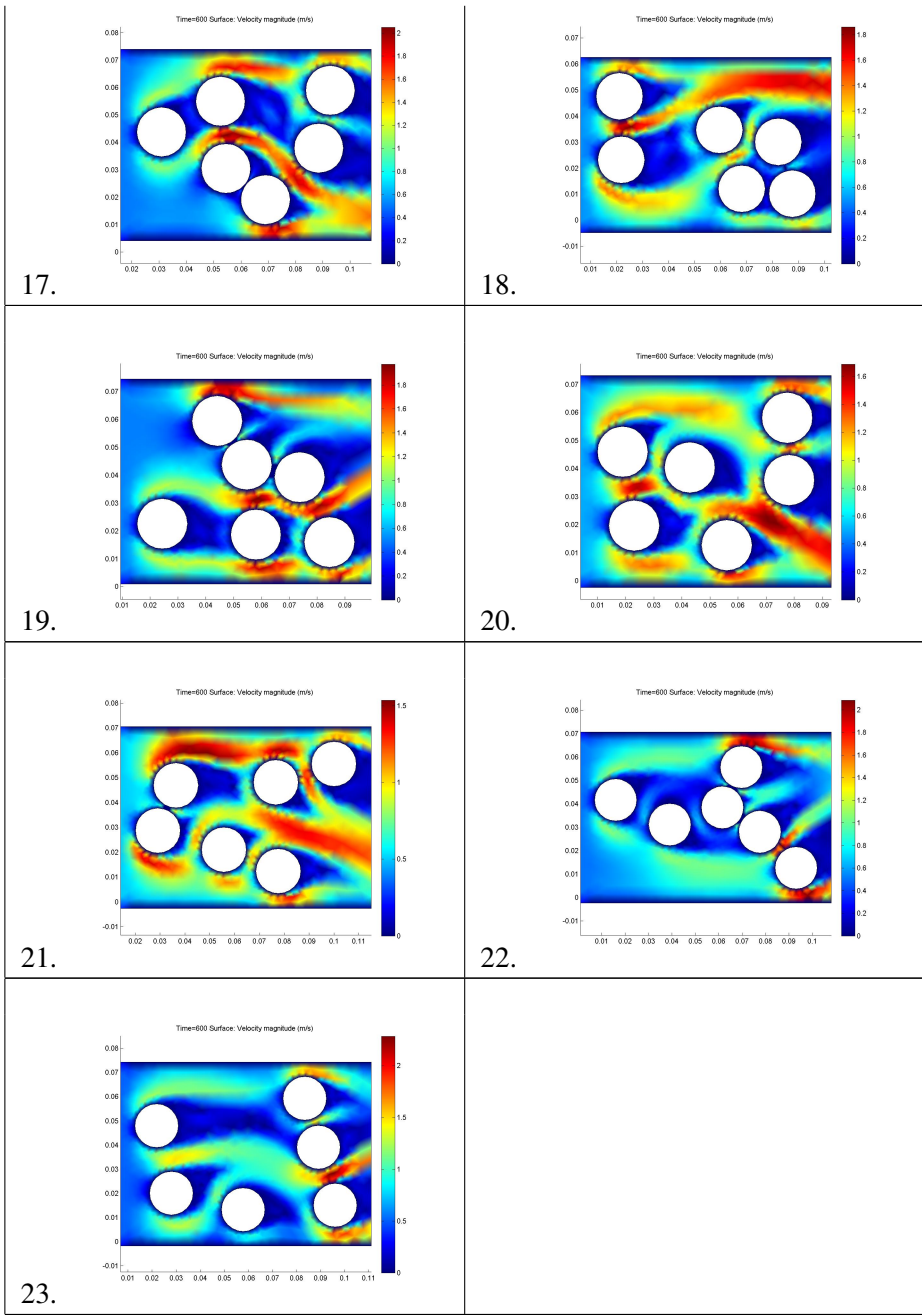
Solución	y1	y2	y3	y4	y5	y6
1	0,0236	0,0238	0,0236	0,0445	0,0457	0,0464
2	0,0189	0,0362	0,0432	0,0434	0,0174	0,0241
3	0,0240	0,0238	0,0219	0,0454	0,0469	0,0471
4	0,0239	0,0243	0,0225	0,0456	0,0469	0,0469
5	0,0178	0,0374	0,0437	0,0441	0,0170	0,0245
6	0,0378	0,0277	0,0500	0,0500	0,0100	0,0100
7	0,0199	0,0500	0,0199	0,0421	0,0244	0,0492
8	0,0208	0,0436	0,0406	0,0431	0,0100	0,0129
9	0,0210	0,0198	0,0396	0,0390	0,0178	0,0172
10	0,0167	0,0321	0,0101	0,0436	0,0121	0,0500
11	0,0187	0,0447	0,0336	0,0469	0,0138	0,0147
12	0,0244	0,0183	0,0189	0,0432	0,0534	0,0542
13	0,0155	0,0417	0,0492	0,0465	0,0243	0,0122
14	0,0168	0,0321	0,0103	0,0433	0,0100	0,0510
15	0,0238	0,0552	0,0470	0,0259	0,0243	0,0400
16	0,0360	0,0230	0,0551	0,0277	0,0242	0,0475
17	0,0191	0,0305	0,0438	0,0379	0,0550	0,0590
18	0,0347	0,0121	0,0476	0,0301	0,0232	0,0102
19	0,0595	0,0187	0,0438	0,0226	0,0393	0,0160
20	0,0581	0,0359	0,0197	0,0404	0,0460	0,0127
21	0,0469	0,0287	0,0210	0,0555	0,0124	0,0481
22	0,0281	0,0312	0,0557	0,0417	0,0127	0,0385
23	0,0201	0,0480	0,0391	0,0151	0,0133	0,0593

D.1. Perfil de velocidad

Tabla D.3: Soluciones óptimas obtenidas para el caso de 6 celdas 18650 en ubicaciones aleatorias.



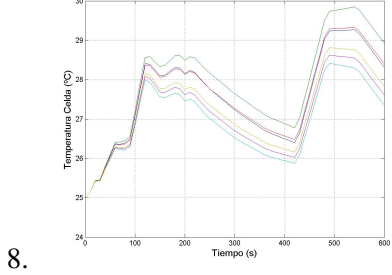
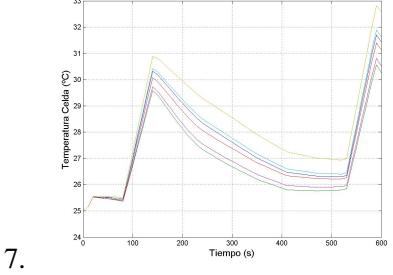
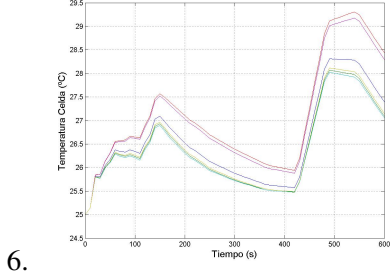
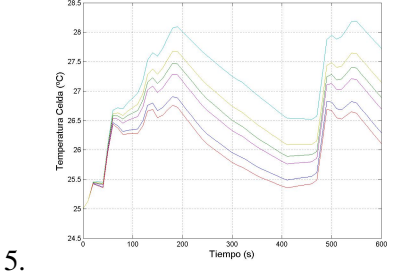
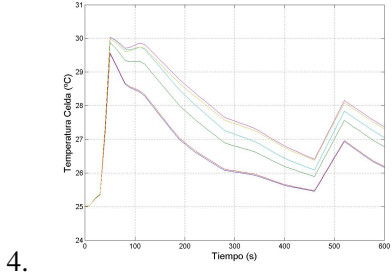
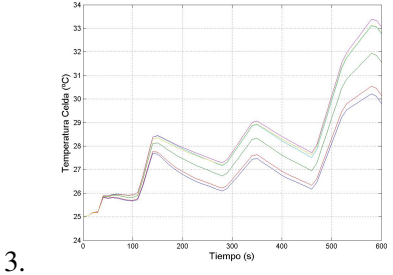
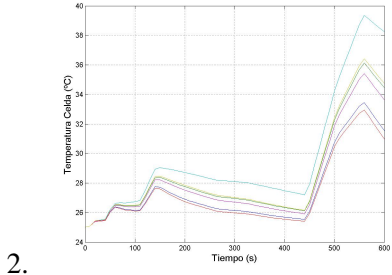
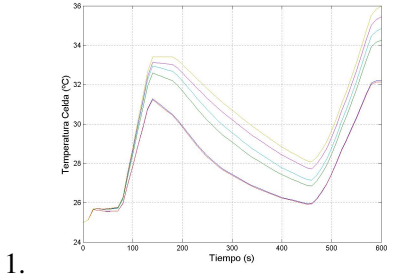


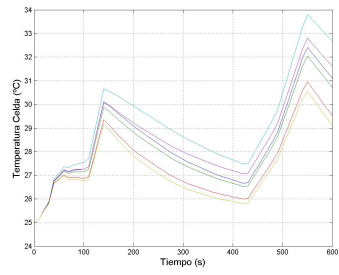


D.2. Temperatura celdas en función del tiempo

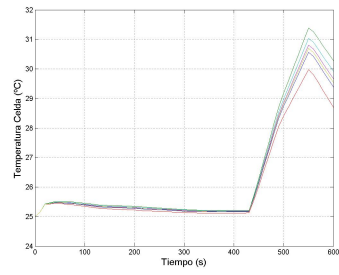
Tabla D.4: Soluciones óptimas obtenidas para el caso de 6 celdas 18650 en ubicaciones aleatorias.

--	--

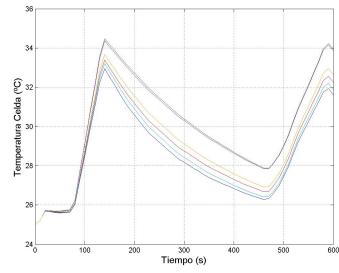




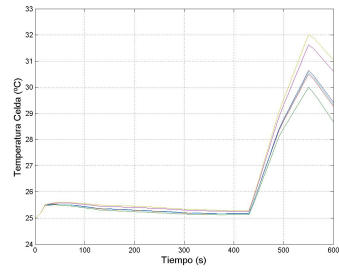
9.



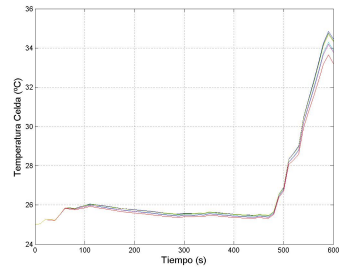
10.



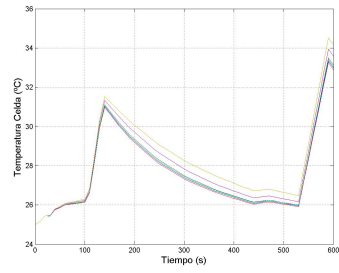
11.



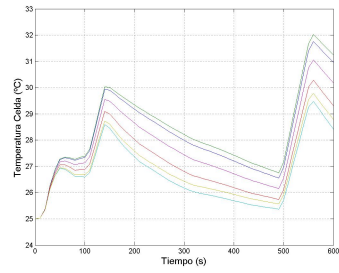
12.



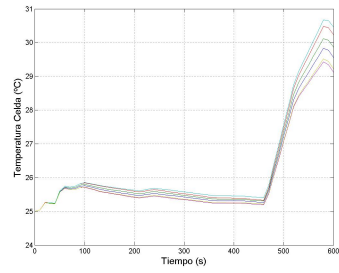
13.



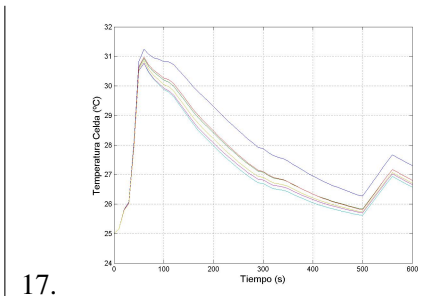
14.



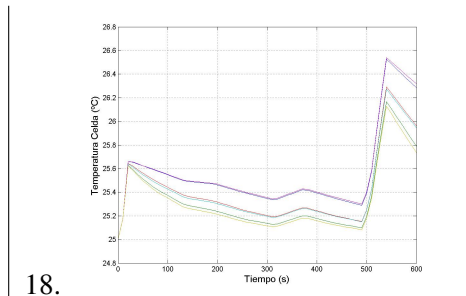
15.



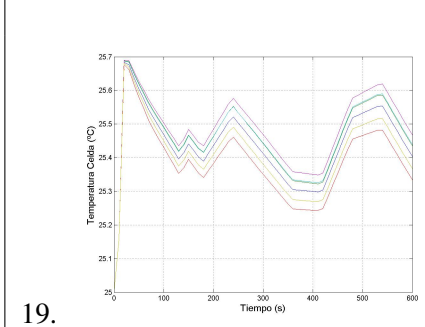
16.



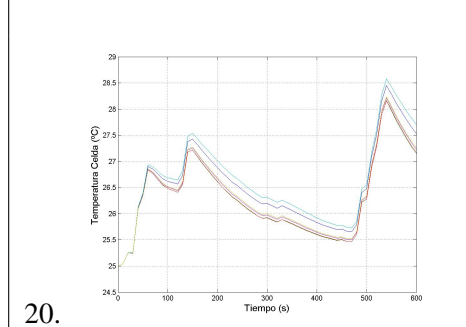
17.



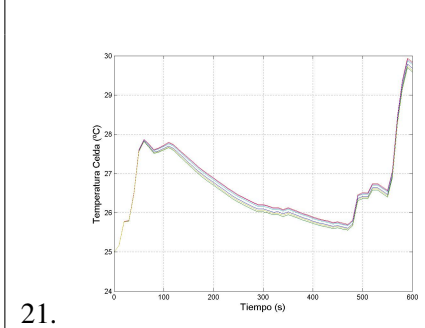
18.



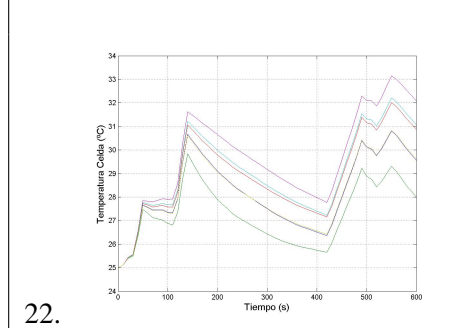
19.



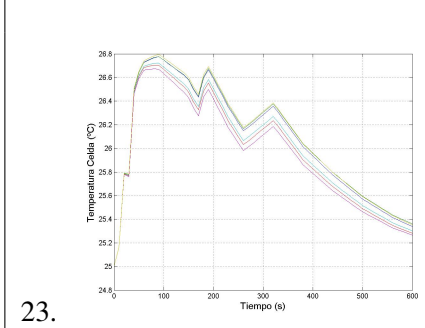
20.



21.



22.



23.