

**DESARROLLO DE LA VERSIÓN V2.0 DE LA HERRAMIENTA SOFTWARE
UIS-PROPELLER PARA EL DISEÑO DE HÉLICES**

LUZ ANGELA RINCÓN GALLEGO



**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO MECÁNICAS
ESCUELA DE INGENIERÍA MECÁNICA
BUCARAMANGA**

2013

**DESARROLLO DE LA VERSIÓN V2.0 DE LA HERRAMIENTA SOFTWARE
UIS-PROPELLER PARA EL DISEÑO DE HÉLICES**

LUZ ANGELA RINCÓN GALLEGO

**Trabajo de grado presentado como requisito para optar al título de
Ingeniero Mecánico**

Director

JORGE ENRIQUE MENESES FLÓREZ

MSc. Ingeniero Mecánico



**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO MECÁNICAS
ESCUELA DE INGENIERÍA MECÁNICA
BUCARAMANGA**

2013

DEDICATORIA

*A Dios quien me concedió el milagro de la vida y ha iluminado mi camino.
A la memoria de mi abuela Ana Isabel, una mujer adelantada a su época, quien me enseñó
el verdadero significado de la fortaleza.
A mis padres Raúl y Luz María, quienes me han amado, apoyado, guiado y fortalecido
durante el camino.
A mi hermana Andrea Paola, quien siempre ha creído en mí y en cada una de mis
metas.
A mi tía Elizabeth, la soñadora, quien me enseñó a creer en lo increíble.
A mis grandes amigos, Yuli, Madeleidy, Ricardo, Jair y Diego, quienes han compartido no
solo mis alegrías sino también mis tristezas.
A mis familiares y todas esas personas que de una u otra forma han marcado la diferencia
en mi vida.*

Luz Ángela

AGRADECIMIENTOS

A Dios.

A mis padres por su apoyo incondicional.

Al profesor Jorge Enrique Meneses, director de éste proyecto, por todo el apoyo, sugerencias y acompañamiento ofrecido durante el desarrollo del mismo.

A Cristian Aldana y Sergio Granados autores de la primera versión de “UIS Propeller” por las sugerencias e información dada durante la etapa inicial de éste proyecto.

Y a todas las personas que de manera directa o indirecta contribuyeron a que “UISPropellerV2.0” fuera una realidad.

TABLA DE CONTENIDO

	Pág.
INTRODUCCIÓN	18
1. FORMULACIÓN DEL PROBLEMA	21
1.1 IDENTIFICACIÓN DEL PROBLEMA	21
1.2 JUSTIFICACION PARA SOLUCIONAR EL PROBLEMA	24
1.3 OBJETIVOS DEL TRABAJO DE GRADO	25
1.3.1 Objetivo General	25
1.3.2 Objetivos Específicos	25
2. MARCO TEÓRICO	27
2.1 TEORÍA AERODINÁMICA HÉLICES PARA ULTRALIVIANOS (TEORÍA DE ELEMENTOS DE PALA)	28
2.2 TEORÍA AERODINÁMICA HÉLICES PARA AEROGENERADORES	29
2.2.1 Teoría de momento axial	30
2.2.2 Teoría de elementos de pala	30
2.2.3 Teoría de momento de elementos de pala (BEM)	31
2.3 GENERALIDADES SOBRE API's DE SOLIDWORKS	32
3. ACTUALIZACIÓN DE LA FUNCIÓN DE DISEÑO HÉLICES PARA ULTRALIVIANO (HÉLICES PROPULSORAS)	38
3.1 PROCESO DE DISEÑO DE HÉLICES PARA ULTRALIVIANO	38
3.2 PROCESO DE ACTUALIZACIÓN DE LA FUNCIÓN PARA EL DISEÑO DE HÉLICES DE ULTRALIVIANO	39
3.2.1 Verificación de ecuaciones del proceso de diseño de hélices para ultralivianos	40

3.2.2	Modificación del código VB.NET (Reducción de líneas de código)	49
3.2.3	Desarrollo de la nueva estrategia para el modelado de la hélice para ultraliviano en SolidWorks	57
4.	DESARROLLO DE LA NUEVA FUNCIÓN DE DISEÑO DE HéLICES PARA AEROGENERADORES DE EJE HORIZONTAL (HAWTs)	68
4.1	PROCESO DE DISEÑO DE HéLICES PARA AEROGENERADORES	69
4.1.1	Identificación y determinación de los criterios básicos de diseño	69
4.1.2	Definición de la forma de la pala	77
4.1.2.1	Cálculo del número de Reynolds	77
4.1.2.2	Selección del perfil aerodinámico (airfoil)	79
4.1.2.3	Aplicación de la teoría de Momento de Elementos de Pala (BEM)	86
4.2	PROGRAMACIÓN DE LA FUNCIÓN DE DISEÑO DE HéLICES PARA AEROGENERADORES	96
4.2.1	Interfaz Visual con el usuario	97
4.2.2	Programación en lenguaje VB.NET del proceso de diseño de hélices para HAWTs	98
4.2.2.1	Declaración de variables	98
4.2.2.2	Obtención de valores suministrados	99
4.2.2.3	Creación del código de programación	99
4.2.3	Programación de los API's de Integración con SolidWorks	102
4.3	PRUEBAS FUNCIONALES DE LA FUNCIÓN DE DISEÑO DE HéLICES PARA HAWTs	104
4.3.1	Pruebas de diseño	105
4.3.2	Pruebas de modelado	110
5.	DESCRIPCIÓN DEL SOFTWARE	114
5.1	PANTALLA DE INICIO	114

5.2	FUNCIÓN DE DISEÑO DE HÉLICES PROPULSORAS (ULTRALIVIANOS)	115
5.2.1	Pantalla de inicio	115
5.2.2	Pantalla de cubo y perfiles	120
5.3	FUNCIÓN DE DISEÑO DE HÉLICES PROPULSADAS (AEROGENRADORES)	122
5.3.1	Pantalla de inicio	122
5.3.2	Módulo forma de la hélice	125
5.3.3	Módulo vista de perfiles girados	126
5.3.4	Módulo de desempeño C_p vs λ	126
5.4	ABOUTBOX	127
6.	CONCLUSIONES	129
7.	RECOMENEDACIONES	131
	BIBLIOGRAFÍA	132
	ANEXOS	135

LISTA DE FIGURAS

	Pág.
Figura 1. Proceso de diseño y modelado de hélices	28
Figura 2. Modelo de flujo axial	30
Figura 3. Modelo de la teoría de elementos de pala	31
Figura 4. Fuerzas, velocidades y ángulos de flujo sobre un elemento de pala a un radio r	32
Figura 5. Barra de Herramientas Macro	33
Figura 6. Ventana de SolidWorks: Extrusión rectangular	34
Figura 7. Ventana VBA; Código obtenido mediante grabación del macro	35
Figura 8. Ventana Interfaz Visual <i>UIS-Propeller V2.0</i> -Diseño Hélice Ultraliviano	38
Figura 9. Proceso de actualización de la función de diseño de hélices para ultralivianos de <i>UIS-Propeller</i>	40
Figura 10. Introducción de variables de entrada en Mathcad	42
Figura 11. Obtención de resultados mediante la utilización de ecuaciones en Mathcad	43
Figura 12. Estimación de valores mediante la utilización de gráficas (Weick, F.E. 1926. Practical Application of the blade element theory III)	44
Figura 13. Corrección Ecuación línea 4159 del código de programación	45
Figura 14. Líneas de código VB adicionadas al programa para calcular y verificar el cumplimiento del mínimo porcentaje de error permitido para diámetro variable.	47
Figura 15. Resultados del proceso de diseño para diámetro variable una vez modificado el código	48
Figura 16. Creación de la interfaz visual de la función de diseño de hélices para ultraliviano <i>UIS-Propeller</i>	50
Figura 17. Ciclo <i>For</i> aplicado al cálculo de hélices para cada uno de los propósitos de vuelo	52

Figura 18. Modelado de hélice para ultraliviano, bajo diferentes condiciones de diseño (<i>UIS-Propeller V1.0</i>)	58
Figura 19. Modelado de hélice para ultraliviano, bajo diferentes condiciones de diseño (<i>UIS-Propeller V2.0</i>)	60
Figura 20. Primer paso construcción del cubo de la hélice para ultraliviano	61
Figura 21. Segundo paso construcción del cubo de la hélice para ultraliviano	62
Figura 22. Tercer paso construcción del cubo de la hélice para ultraliviano	62
Figura 23. Cuarto paso construcción del cubo de la hélice para ultraliviano	63
Figura 24. Quinto paso construcción del cubo de la hélice para ultraliviano	63
Figura 25. Sexto paso construcción del cubo de la hélice para ultraliviano	64
Figura 26. Séptimo paso construcción del cubo de la hélice para ultraliviano	65
Figura 27. Octavo paso construcción del cubo de la hélice para ultraliviano	66
Figura 28. Hélices modeladas en SolidWorks por <i>UIS-Propeller V2.0</i>	67
Figura 29. Diagrama del proceso de desarrollo de la función de diseño de hélices para HAWTs	68
Figura 30. Turbinas eólicas de dos y tres palas	70
Figura 31. Curvas de C_p vs TSR para rotores de 1, 2 y 3 palas	71
Figura 32. Establecimiento de los rangos de TSR en función del número de palas	73
Figura 33. Cálculo iterativo de los criterios básicos de diseño en Mathcad	76
Figura 34. Cálculo de propiedades aerodinámicas del perfil NACA 64(2)-415 en <i>Javafoil</i>	82
Figura 35. Proceso de puntuación de perfiles realizado en Mathcad	85
Figura 36. Proceso de Selección del perfil en Mathcad	86
Figura 37. Método BEM para evaluar a y a'	88
Figura 38. Angulo de ataque y vectores de elevación y arrastre de un perfil aerodinámico	88
Figura 39. Regresión polinómica de C_l vs α y C_d vs α ($Re=400000$) para el perfil NACA63(2)421	90

Figura 40. Curvas de a vs r/R , correspondientes a la aplicación de la corrección de Glauert, el modelo de pérdidas de W. Z Shen <i>et al</i> y sin corrección alguna para pérdidas en el extremo de la pala.	92
Figura 41. Ventana de inicio de la función de diseño de hélices para HAWTs	97
Figura 42. Código VB del proceso iterativo para la estimación de la forma de la pala de una HAWT	100
Figura 43. Código VB.NET para la obtención de las coordenadas geométricas de la pala de una HAWT	102
Figura 44. Adición de librerías de SolidWorks a <i>UIS-Propeller</i>	103
Figura 45. Pala de una HAWT modelada por <i>UIS-Propeller</i> a través de los API's de Integración con SolidWorks 2013	104
Figura 46. Resultados de la ejecución de <i>UIS-Propeller</i> en la etapa de pruebas	106
Figura 47. Modelado de la hélice diseñada durante el proceso de pruebas	109
Figura 48. Error de modelado de la hélice cuando la ventana de SolidWorks abre minimizada	110
Figura 49. Proceso de recubrimiento de perfiles con errores de modelado, en caso que SolidWorks abra minimizado	111
Figura 50. Operación de recubrimiento en ventana maximizada	111
Figura 51. Proceso de recubrimiento de perfiles modificado	112
Figura 52. Resultado de la corrección pertinente al modelado de la hélice	113
Figura 53. <i>UIS-Propeller</i> Splash Screen	114
Figura 54. Pantalla de inicio de <i>UIS-Propeller V2.0</i>	115
Figura 55. Pantalla de inicio, función de diseño de hélices propulsoras	116
Figura 56. Ventanas de ayuda del software (AR y CR)	117
Figura 57. Descripción de la pantalla de inicio de la función de diseño de hélices para ultralivianos	118
Figura 58. Pala de hélice, generada en SolidWorks 2013	119

Figura 59. Descripción de la pantalla cubo y perfiles de la función de diseño de hélices para ultralivianos	120
Figura 60. Resultado final de la ejecución de la función de diseño de hélices para ultralivianos	121
Figura 61. Pantalla de inicio función de diseño de hélices propulsadas	122
Figura 62. Descripción de la pantalla de inicio de la función de diseño de hélices para aerogeneradores	124
Figura 63. Modelado de la hélice resultante de la ejecución de la función de diseño de hélices para aerogeneradores	125
Figura 64. Modulo forma de la hélice de la función de diseño de hélices para aerogeneradores	125
Figura 65. Módulo vista de perfiles girados de la función de diseño de hélices para aerogeneradores	126
Figura 66. Módulo de desempeño C_p vs λ de la función de diseño de hélices para aerogeneradores	127
Figura 67. AboutBox de <i>UIS-Propeller V2.0</i>	127
Figura 68. Ícono de notificación de <i>UIS-Propeller</i> en la barra de tareas, acceso al AboutBox	128

LISTA DE TABLAS

	Pág.
Tabla 1. Descripción de variables de entrada UIS-Propeller Ultraliviano	41
Tabla 2. Rango de valores de TSR en función del número de palas	72
Tabla 3. Criterios básicos para el diseño HAWTs	76
Tabla 4. Porcentajes de puntuación para la selección del perfil	81
Tabla 5. Valores de valores de $\frac{C_{l_{max}}}{C_{d_{min}}}$ y $\left(\frac{L}{D}\right)_{C_{l_{max}}}$ para NACA 64(2)-415	83
Tabla 6. Valores de α_0 para perfiles NACA	84
Tabla 7. Variables de entrada proceso de prueba	105
Tabla 8. Resultados iniciales de diseño durante la etapa de pruebas	107
Tabla 9. Resultados del diseño geométrico etapa de pruebas	108
Tabla 10. Resultados de C_p para cada estación de la hélice en etapa de pruebas	109

LISTA DE ANEXOS

	Pág.
ANEXO A. Explicación esquemática de la metodología para el modelado de la pala en SolidWorks	136
ANEXO B. Proceso de diseño de hélices para ultralivianos	137
ANEXO C. Nomenclatura utilizada por Weick en el diseño de hélices para ultralivianos	158
ANEXO D. Programación de API's de integración SolidWorks en lenguaje VB para el modelado de hélices para ultraliviano	160
ANEXO E. Estrategia de modelado del cubo de la hélice implementada en <i>UIS-Propeller V1.0</i>	165
ANEXO F. Programación de API's de integración SolidWorks en lenguaje VB para la construcción del cubo de hélices para ultraliviano	168
ANEXO G. Nomenclatura de variables empleadas por <i>UIS-Propeller</i> en el diseño de hélices para aerogeneradores de eje horizontal	173
ANEXO H. Valores de las propiedades empleadas en proceso de selección del perfil.	177
ANEXO I. Ecuaciones de Cl_{max}/Cl_{min} y $(L/D)_{Cl_{max}}$ a $Re \leq 400000$ y $400000 < Re \leq 5000000$ para cada tipo de perfil	178
ANEXO J. Referencia bibliográfica de las ecuaciones utilizadas en <i>UIS-Propeller</i>	181

RESUMEN

TÍTULO: DESARROLLO DE LA VERSIÓN V2.0 DE LA HERRAMIENTA SOFTWARE *UIS-PROPELLER* PARA EL DISEÑO DE HÉLICES*.

AUTOR: RINCON-GALLEGO, Luz Ángela**.

PALABRAS CLAVE: Ultraliviano, Aerogenerador de eje horizontal (HAWT), Teoría de Elementos de Pala, Teoría de Momento Axial, Teoría de Momentos de Elementos de Pala (BEM), NACA, Perfil aerodinámico, API's, Software.

DESCRIPCIÓN:

UIS-Propeller V2.0, surgió como respuesta al constante interés a nivel nacional por el aprovechamiento del recurso eólico, el cual es evidente en el cada vez mayor número de Colombianos que practican la aviación experimental como actividad deportiva (paramotores y vehículos ultralivianos) y la fuerte tendencia actual a nivel mundial por la utilización del viento como fuente de energía.

Con lo anterior en mente, se actualizó y mejoró la función de diseño de hélices para vehículos ultralivianos desarrollada en la primera versión del software. Esta, se fundamenta en la teoría de elementos de pala, desarrollada a principios del siglo pasado, por la NACA para el diseño de hélices para aeronaves, obteniendo de esta forma las coordenadas geométricas de la hélice. Posteriormente a través de API's, este diseño es modelado en una herramienta CAD (SolidWorks 2013).

Por otro lado, una función de diseño de hélices para aerogeneradores de eje horizontal (HAWTs), fue adicionada a *UIS-Propeller*, aplicando para ello la Teoría de Momentos de Elementos de Pala (BEM) y los diferentes perfiles NACA empleados para aplicaciones eólicas (NACA2509, NACA4412, NACA4415, al igual que las series NACA63, NACA64 y NACA65). Esta, al igual que la función anterior, genera las coordenadas geométricas de la hélice para su posterior modelado en SolidWorks 2013.

* Trabajo de grado

** Facultad de Ingenierías Físico-Mecánicas, Escuela de Ingeniería Mecánica, Universidad Industrial de Santander, Director: MSc. Ing. Jorge Enrique Meses Flórez.

ABSTRACT

TITLE: DEVELOPMENT OF THE SECOND VERSION OF *UIS-PROPELLER* SOFTWARE TOOL FOR PROPELLER DESIGN*.

AUTHOR: RINCON-GALLEGO, Luz Ángela**.

KEYWORDS: Ultralight, Wind Turbine, Blade Element Theory, Axial Momentum Theory, Blade Element Momentum Theory (BEM), NACA, Airfoil, API's, Software.

DESCRIPCIÓN:

UIS-Propeller V2.0 came out as an answer to the growing national interest for wind resource utilization, fact that is evident in the every time higher number of Colombians practicing experimental aviation as a sport (paramotors and ultralight aircrafts) and the current strong worldwide tendency of wind energy extraction.

Keeping in mind the above, in this project the function for aircraft propeller design developed in the first version of *UIS-Propeller* was updated and improved. This is based on the Blade Element Theory developed early last century by NACA for aircraft propeller design, thereby obtaining, the blade coordinates, which through the API's are subsequently exported to SolidWorks 2013 generating the 3D propeller model.

On the other side, a function for the design of Horizontal Axis Wind Turbines (HAWT's) propellers was added to *UIS-Propeller*. To achieve this, the Blade Element Momentum Theory (BEM), and NACA airfoils commonly used for wind turbines applications (NACA2509, NACA4412, NACA4415, and the NACA63, NACA64 and NACA65 series), were applied. In the same way than the function for aircraft propellers, this function generates the blade coordinates for its subsequent 3D modeling in SolidWorks 2013.

* Work Degree

** Faculty of Physico-Mechanical Engineerings, School of Mechanical Engineering, Universidad Industrial de Santander Project director: MSc. Ing. Jorge Enrique Meneses Flórez.

INTRODUCCIÓN

En los últimos años, el vuelo como actividad deportiva ha adquirido un gran apogeo a nivel nacional (vuelo de paramotores y ultralivianos), sin embargo, es poco el desarrollo de la industria aeronáutica en nuestro país, razón por la cual los amantes de este deporte, deben importar las hélices utilizadas o fabricarlas de forma artesanal, exponiéndose por tanto a riesgos durante la operación de las hélices; en consecuencia de la problemática anterior, en el año 2009 fue desarrollada la versión V1.0 de la herramienta software *UIS-Propeller*, de diseño de hélices para equipos de vuelo ultraliviano. Esta primera versión del software, presentó sin embargo ciertas incongruencias en el diseño de hélices a diámetro variable, al igual que algunas interferencias geométricas en el modelado de la misma en el Software CAD SolidWorks; haciéndose necesario, el desarrollo de una nueva versión del software.

Este proyecto de grado presenta la versión V2.0 de *UIS-Propeller*, la cual continúa ofreciendo soluciones a las problemáticas de los amantes de la aeronáutica, a la vez que ofrece un software actualizado, más confiable y eficiente. En adición, esta versión V2.0, posee nuevas funciones referentes al aprovechamiento del viento como fuente de energía renovable i.e. un nuevo módulo de cálculo y diseño de hélices para aerogeneradores.

La herramienta software trabaja bajo la plataforma Visual Studio 2010, donde la ejecución del proceso de cálculo y diseño genera como variable de salida, las coordenadas geométricas de la hélice diseñada (para ultraliviano o aerogeneradores según sea el caso), para su posterior modelado en el Software CAD, SolidWorks 2013. A este respecto, cabe aclarar que, con el propósito de presentar una herramienta de diseño actualizada a los más recientes avances CAD

disponibles en el mercado. La interfaz de comunicación Visual Studio-SolidWorks establecida a través de API's¹ fue desarrollada con SolidWorks 2013, en la lugar de SolidWorks 2010 tal como había sido propuesto en el plan de este proyecto.

En cuanto a la estructuración del presente libro, este está compuesto por siete capítulos, los cuales se distribuyen de la siguiente manera:

En el primer capítulo, se describirá la formulación del problema que dio como resultado el desarrollo de este proyecto, y sus respectivos objetivos.

En el capítulo dos, serán explicados los fundamentos teóricos bajo los cuales fue desarrollado *UIS-Propeller*.

En el capítulo tres, se describen cada uno de los pasos seguidos durante el proceso de actualización de la función de diseño de hélices para ultralivianos, entre los cuales se cuentan el estudio del proceso de diseño de la hélice, la verificación de las ecuaciones implementadas, la modificación del código de programación en VB.NET bajo la plataforma Visual Studio 2010 y finalmente el desarrollo de la nueva estrategia de modelado de la hélice en SolidWorks 2013.

El capítulo cuatro, trata del diseño de hélices para aerogeneradores, desde el proceso de diseño implementado (BEM²), pasando por la programación del software en VB.NET (adicionando los API's de integración con SolidWorks 2013) y finalizando con las pruebas funcionales realizadas a este nuevo módulo del Software.

¹ Interfaz de Programación de Aplicaciones (API, del Inglés Application Programming Interface)

² Teoría de momentos de elementos de pala (BEM, del Inglés Blade Element Momentum)

Adicionalmente, la interfaz visual del software y cada uno de sus módulos serán descritos en el capítulo cinco y como es natural, en el capítulo 6 y 7 se darán algunas conclusiones y recomendaciones respectivamente.

NOTA: Al final del libro se adjunta a manera de anexo parte de la información empleada durante el desarrollo de este software.

1. FORMULACIÓN DEL PROBLEMA

1.1 IDENTIFICACIÓN DEL PROBLEMA

En vista del constante interés del hombre por el aprovechamiento del viento, y del poco desarrollo que la industria aeronáutica tiene en nuestro país, en la escuela de Ingeniería Mecánica de la Universidad Industrial de Santander nació la iniciativa de desarrollar una serie de proyectos de grado enfocados a impulsar este tipo de industria y generar soluciones y alternativas factibles para profesionales y amantes del vuelo como actividad deportiva, desarrollando la línea de investigación en aviación experimental conocida como el proyecto Lilienthal, de aquí que en el año 2009, se desarrollara como proyecto de grado la herramienta software *UIS Propeller V1.0*. Dicha herramienta software calcula y diseña hélices para aviación experimental basándose en la teoría de elementos aerodinámicos, utilizando la plataforma Visual Studio 2005, y posteriormente mediante la implementación de APIS es enlazada con el software CAD SolidWorks, generando de esta forma el modelado de la respectiva hélice.

A pesar de que la herramienta software *UIS-Propeller* diseña de manera satisfactoria la pala de la hélice, dicho software presenta algunas interferencias geométricas a la hora de adicionar el cubo de la hélice en SolidWorks, limitando de esta forma la variedad de hélices a ser modeladas, al igual que la variedad de acoplamientos a ser empleados por el usuario, razón por la cual los usuarios del software han de realizar por si mismos el modelado del cubo en SolidWorks, o realizar el proceso de tallado del cubo de manera artesanal.

Además de lo anteriormente expuesto, la interfaz del software se limita a Visual Studio 2005 y SolidWorks 2006, motivo por el cual se presentan incongruencias de

código a la hora de realizar la conexión Visual Studio-SolidWorks de versiones más recientes, para el respectivo modelado de la hélice, por tanto, la herramienta software no puede ser empleada de forma satisfactoria por los potenciales usuarios actuales.

Por otro lado, la creciente demanda energética a nivel global, acompañada de los grandes niveles de contaminación y la preocupación mundial por el cambio climático, han sido algunas de las grandes cuestiones del siglo XXI, tanto así que la inversión en investigación, desarrollo y explotación de fuentes de energía renovable ha experimentado un gran incremento en los últimos años, fuentes de las cuales la energía eólica es una de las más importantes a ser aprovechadas.

Globalmente, el aprovechamiento del recurso eólico ha ido adquiriendo un gran apogeo, siendo de acuerdo con *Global Wind Energy Council (GWEC)*, el continente Europeo el líder del mercado eólico con 86,647 MW instalados para finales del 2010, mostrando un gran desarrollo en países como Alemania (27,191 MW), España (20,623 MW), Francia (5,797 MW), Dinamarca (3,749 MW), entre otros, de igual forma en el continente Asiático se ha evidenciado un gran crecimiento de la capacidad instalada estando China (44,733 MW) e India (13,065 MW) a la cabeza. Mientras que en el continente Americano es Estados Unidos quien lleva la delantera (40,298 MW), los países latinoamericanos han mostrado creciente interés por el aprovechamiento del recurso eólico siendo Brasil (927 KW) y México (519 MW) los países con mayor capacidad instalada³. Dicho progreso ha sido observado en los últimos años también a nivel nacional, muestra de ello es la instalación por

³ Global Wind Report: Annual market updates 2011. *Global Wind Energy Council (GWEC)*. March, 2012. Available from internet:

http://www.gwec.net/fileadmin/documents/NewsDocuments/Annual_report_2011_lowres.pdf>

parte de EPM en el departamento de la Guajira del parque eólico Jepírachi el cual cuenta con una capacidad instalada de 19,5 MW⁴ y el parque eólico Wayúu por parte de las empresas Isagen S.A. E.S.P. y Wayuu S.A. E.S.P, el cual cuenta con una capacidad instalada de 20 MW⁵. Hecho a partir del cual se ha despertado a nivel nacional un creciente interés en el aprovechamiento de la energía del viento, no sólo por parte de las grandes empresas generadoras de energía sino también como recurso aprovechable para la generación de electricidad en viviendas aisladas mediante la implementación de aerogeneradores de baja potencia.

Mundialmente hablando, se experimenta un gran desarrollo tecnológico no solo en materiales de construcción, sino también en diseños aerodinámicos basados en la fuerza de elevación que permiten sacar el máximo provecho posible a las turbinas eólicas, en las cuales el diseño apropiado de las palas (hélices) juega un papel protagónico, motivo por el cual se ha prestado especial atención al diseño de las mismas, no solo por parte de las empresas líderes del mercado sino también por parte de grupos de investigación. Sin embargo, a pesar de lo anteriormente expuesto a nivel nacional no se cuenta con empresas que diseñen y fabriquen hélices para aerogeneradores (hélices propulsadas), sólo se cuenta con un número reducido de empresas distribuidoras.

Dado que el proceso de diseño de hélices es un proceso de gran complejidad, que puede llegar a ser tedioso realizado de forma manual, la sistematización del proceso de diseño mediante la utilización de herramientas software de diseño y herramientas CAD/CAM representan una alternativa viable para facilitar y

⁴Disponible en internet:

<http://www.epm.com.co/site/Home/Institucional/Nuestrasplantas/Energía/ParqueEólico.aspx>>

⁵ Disponible en internet: http://www.isagen.com.co/comunicados/boletin_8_nov15.pdf>

acelerar proceso de diseño y manufactura de hélices, al mismo tiempo que se garantiza la confiabilidad de las mismas mediante la reducción de posibles errores humanos.

1.2 JUSTIFICACION PARA SOLUCIONAR EL PROBLEMA

Dada la creciente demanda energética, que ha llevado a la búsqueda de fuentes alternativas, siendo la energía eólica una de las fuentes con mayor acogida, cuyo desarrollo promete seguir creciendo, se han desarrollado nuevas tecnologías y técnicas de diseño y fabricación basados en la teoría aerodinámica. Sin embargo pese al gran apogeo de la misma, a nivel nacional no se cuenta con empresas dedicadas al diseño y manufactura de hélices para aerogeneradores; de manera tal que las hélices distribuidas a nivel nacional son netamente importadas.

En vista de lo anterior, el desarrollo de la versión V2.0 de la herramienta software de diseño *UIS Propeller* la cual contenga un módulo de cálculo y diseño de hélices para aerogeneradores, representa un paso hacia el desarrollo de partes para turbinas eólicas a nivel nacional, facilitando el diseño sistemático de hélices para aerogeneradores que responda a las preferencias y especificaciones del usuario y/o fabricante y por ende fomentando la utilización del viento como fuente aprovechable de energía.

De igual forma, siendo consecuentes con el avance tecnológico y el desarrollo de nuevas herramientas software que brindan un mayor número de funcionalidades, al igual que mayor confiabilidad, el permanecer a la vanguardia teniendo en cuenta que los potenciales usuarios actuales y la Universidad Industrial de Santander cuentan con versiones más actualizadas de los software de soporte (Visual Studio, SolidWorks) se hace un factor más que relevante para el desarrollo

de una nueva versión de la herramienta software UIS Propeller que presente no solo nuevas funcionalidades sino también mejor desempeño y confiabilidad a los usuarios.

1.3 OBJETIVOS DEL TRABAJO DE GRADO

1.3.1 Objetivo General.

Contribuir con la misión de la Universidad Industrial de Santander de ser participe activo del desarrollo tecnológico y resolución de problemáticas de la industria local y nacional, mediante el lanzamiento de una versión mejorada del software *UIS-Propeller V1.0* que contenga un módulo de diseño de hélices para aerogeneradores.

1.3.2 Objetivos Específicos.

- ▶ Optimizar la herramienta software *UIS-Propeller V1.0* mediante la sintetización del código de programación del diseño de hélices para ultralivianos (hélices propulsoras), logrando de esta manera un software más eficiente.
- ▶ Mejorar el funcionamiento de la herramienta software *UIS-Propeller V1.0* mediante la corrección y ajuste de las múltiples interferencias geométricas presentadas en el modelado del cubo de la hélice, las cuales limitan la variedad de hélices diseñadas por la herramienta software *UIS-Propeller V1.0*.
- ▶ Actualizar la herramienta software *UIS-Propeller V1.0* a Microsoft Visual Studio 2010 y Solid Works 2010.

▶ Desarrollar y adicionar un nuevo módulo de cálculo y diseño de hélices para aerogeneradores en la plataforma Visual Studio 2010, a la herramienta software *UIS-Propeller V1.0* que cuente con las siguientes variables de entrada a ser establecidas por el usuario:

- Sistema de unidades
- Numero de palas
- Potencia nominal
- Velocidad del viento
- Densidad del aire

Una vez ejecutado el programa el usuario obtiene como variables de salida las dimensiones y coordenadas correspondientes de cada una de las estaciones del perfil aerodinámico (también llamado perfil alar) que conforman la pala del aerogenerador.

▶ Vincular los resultados del proceso de diseño de hélices para aerogeneradores obtenidos en la interfaz Microsoft Visual Studio 2010 con el software CAD SolidWorks 2010, mediante la implementación de API's que permitan el modelado de la hélice en dicho software.

NOTA: Con el propósito de presentar una herramienta de diseño actualizada a los más recientes avances CAD disponibles en el mercado. La interfaz de comunicación Visual Studio-SolidWorks establecida a través de API's fue desarrollada con SolidWorks 2013, en la lugar de SolidWorks 2010.

2. MARCO TEÓRICO

La versión V2.0 de *UIS-Propeller*, desarrollada en este proyecto de grado, cuenta con la actualización de la función de diseño de hélices para vehículos ultralivianos⁶ y una nueva función de diseño de hélices para aerogeneradores.

Los cálculos para el diseño de hélices de ultralivianos se basan en la teoría de Elementos de Pala (conocida también como teoría de Elementos Aerodinámicos o Blade Element Theory) implementada por Fred E. Weick en su reporte para la NACA⁷ dividido en cuatro partes⁸, tal como fue expuesto en la versión anterior de *UIS-Propeller*. Por otro lado, los cálculos correspondientes al diseño de hélices para aerogeneradores se basaron en la aplicación de la teoría de Momento de Elementos de Pala o BEM (Blade Element Momentum; por sus siglas en Inglés).

Al igual que la versión anterior de *UIS-Propeller*, la versión V2.0 genera una matriz con las coordenadas geométricas (x , y , z) de los puntos de los perfiles aerodinámicos en cada estación de la hélice para su posterior exportación al software CAD (SolidWorks), mediante la implementación de API's, ejecutando de esta forma el modelado de la hélice (ver ANEXO A).

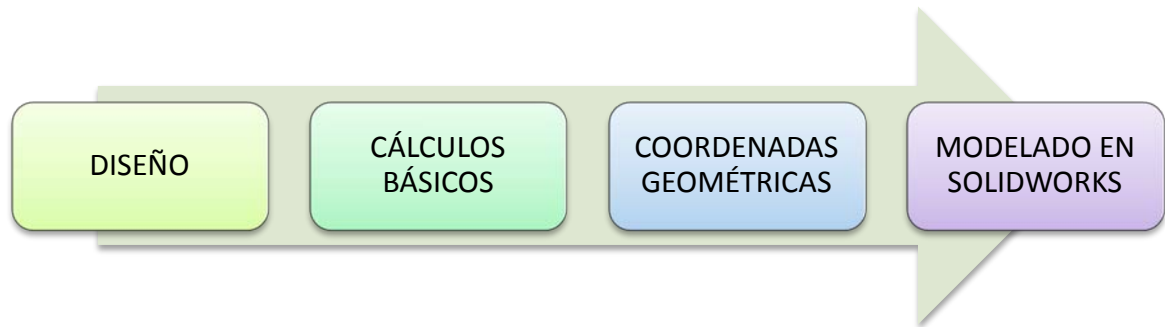
Con el fin de lograr un mayor entendimiento del proceso de diseño y modelado de hélices para ultralivianos y aerogeneradores llevado a cabo durante el desarrollo de éste proyecto, a continuación se presentan los diferentes conceptos necesarios para tal fin.

⁶ La función de diseño de hélices para vehículos ultralivianos, fue desarrollado en la versión V1.0 de *UIS-Propeller*.

⁷ NACA: National Advisory Committee for Aeronautics

⁸ Weick, F.E (1926). Practical application of the blade element theory -I-II-III and IV.

Figura 1. Proceso de diseño y modelado de hélices



2.1 TEORÍA AERODINÁMICA HÉLICES PARA ULTRALIVIANOS (TEORÍA DE ELEMENTOS DE PALA)

Tal como se mencionó anteriormente, el diseño de hélices para ultralivianos es llevado a cabo mediante la aplicación de la teoría de elementos de pala modificada por Fred E. Weick, la cual consiste en dividir la pala de la hélice en un número determinado de estaciones (o partes), para posteriormente realizar los cálculos de las fuerzas que actúan sobre dicha estación. En el método presentado por Weick en su serie de cuatro papers, se propone emplear solo las fuerzas calculadas para uno de los elementos de pala y a partir de estas obtener las características de la hélice completa.

Cabe aclarar, que aun siendo la teoría de elementos de pala bastante conocida en la industria aeroespacial, los resultados obtenidos mediante dicho método han de ser combinados con datos obtenidos de pruebas de hélices modelo en túneles de viento y pruebas de hélices a escala real en vuelo, esto, con el fin de realizar las correcciones pertinentes logrando de esta forma resultados confiables y satisfactorios en la práctica.

2.2 TEORÍA AERODINÁMICA HÉLICES PARA AEROGENERADORES

La pala es el elemento más importante de la hélice o rotor, ya que es éste el encargado de extraer y convertir la energía cinética del viento en energía mecánica. Por lo anteriormente expuesto el correcto diseño aerodinámico de las palas juega un papel indispensable en la eficiencia y desempeño del rotor.

Teniendo en cuenta que, la pala de la hélice es un elemento aerodinámico cuya sección transversal es un perfil aerodinámico (airfoil o superficie de sustentación), lo primero que se ha de tener en cuenta a la hora de diseñar dichas palas, es el desarrollo de la teoría aerodinámica más conveniente, la cual permita la determinación de los parámetros geométricos de la hélice (distribución de la cuerda, espesor, ángulo de giro y reducción gradual de cuerda y espesor a lo largo de la hélice *-tapered blades-*).

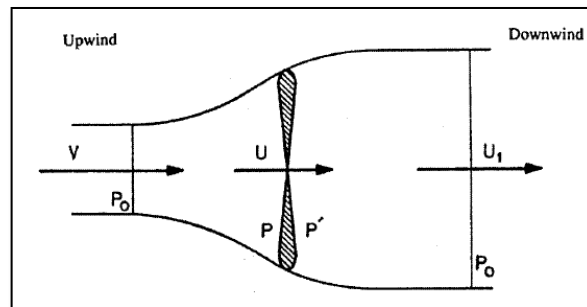
Considerando que, el vuelo de un rotor es un fenómeno bastante complejo que no puede ser modelado con exactitud, diseñadores actuales han optado por realizar ciertas asunciones con el fin de modelar situaciones aerodinámicas aproximadas. La aproximación más utilizada, es el modelo de flujo unidireccional, incompresible, no viscoso mediante la aplicación de la teoría de Momento Axial, incluyendo los efectos de la estela rotacional (wake effect), sumando a estos la teoría de Elementos de Pala. La combinación de estas dos teorías, es lo que conocemos como teoría de Momento de Elementos de Pala (BEM, del inglés *Blade Element Momentum*).⁹

⁹ HABALI, S.M y SALEH, I.A. Local design, testing and manufacturing of small mixed airfoil wind turbine blades of glass fiber reinforced plastics. Part I: Design of the blade and root. Energy Conversion and Management [online]. 2000, no. 41 [cited 13 de March 2012], pp. 249-280.

2.2.1 Teoría de momento axial. Para la aplicación de esta teoría se realizan las siguientes suposiciones:

- ▶ El flujo es completamente axial
- ▶ El flujo es rotacionalmente simétrico
- ▶ No ocurre fricción cuando el aire pasa a través del rotor

Figura 2. Modelo de flujo axial



Fuente: HABALI, S.M. *et al*, 2000, pp. 254.

De la Figura 2 se tiene que:

U : Velocidad local del viento

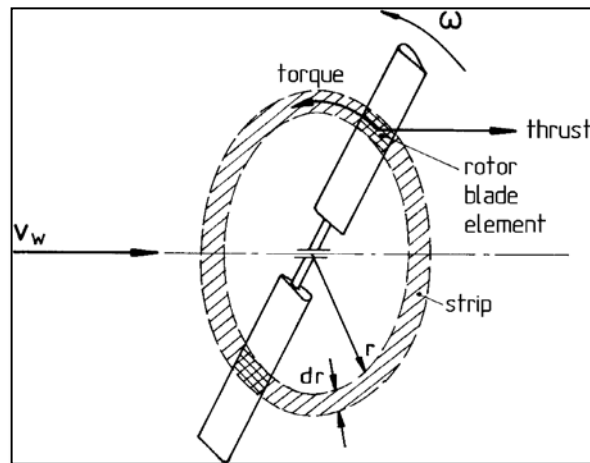
V : Velocidad del viento

P : Potencia

2.2.2 Teoría de elementos de pala. Tal como se mencionó anteriormente en el diseño de hélices para ultralivianos, la teoría de Elementos de Pala consiste en dividir la pala de una hélice en un número determinado de secciones o elementos (de longitud dr), para el posterior cálculo de las fuerzas que actúan en cada uno de dichos elementos (figura 3).

Tal como HABALI, S.M *et al* mencionan en su paper “*Local design, testing and manufacturing of small mixed airfoil wind turbine blades of glass fiber reinforced plastics. Part I: Design of the blade and root*”, esta teoría se caracteriza por suponer que las fuerzas y los momentos que actúan sobre la pala se deben únicamente a la elevación y el arrastre, así mismo, debido a que las fuerzas en cada elemento de pala son calculadas a la velocidad de viento local (en función del dr), se supone que no hay interferencia entre los elementos de hélice adyacentes. De igual forma el flujo alrededor del elemento de hélice a un radio r ha de ser considerado como flujo en dos dimensiones.

Figura 3. Modelo de la teoría de elementos de pala

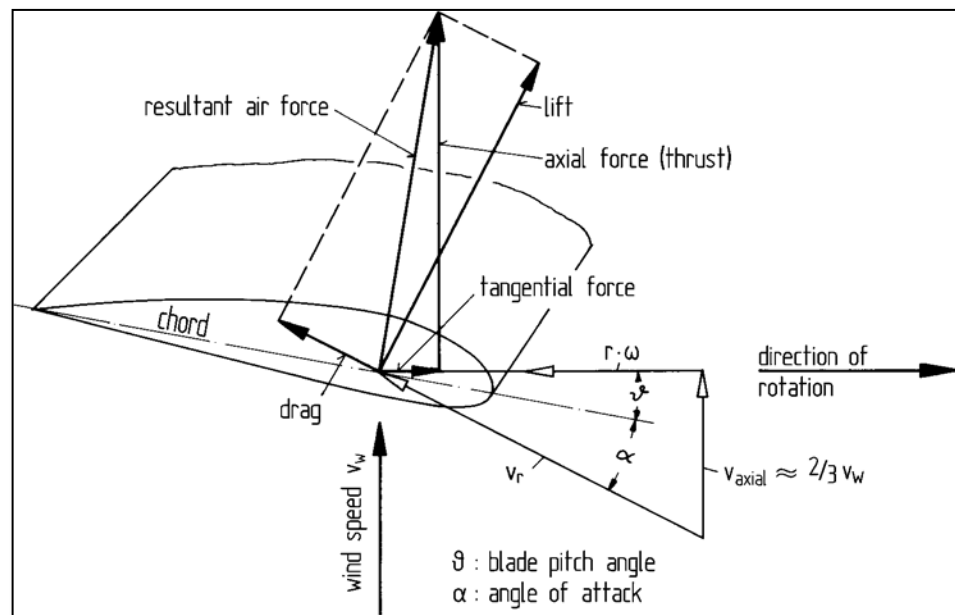


Fuente: HAU, E., 2006, pp. 94

2.2.3 Teoría de momento de elementos de pala (BEM). La teoría BEM es la combinación de la teoría de elementos de pala, que como se estudió anteriormente divide la hélice en n elementos (de longitud dr) independientes y la teoría de momento axial, la cual toma en consideración la pérdida del momento en el plano del rotor, debido al efecto aerodinámico del flujo de aire pasando a través de la hélice.

La aplicación de ésta teoría permite el cálculo de los factores de inducción a y a' (axial y tangencial respectivamente); los cuales son necesarios para el cálculo de la velocidad angular W de incidencia en el plano del rotor.

Figura 4. Fuerzas, velocidades y ángulos de flujo sobre un elemento de pala a un radio r



Fuente: HAU, E., 2006, pp. 95

2.3 GENERALIDADES SOBRE API's DE SOLIDWORKS¹⁰

API (del Inglés, *Application Programming Interface*) o IPA (Interfaz de programación de aplicaciones) es un conjunto de funciones y procedimientos ofrecidos en bibliotecas (en este caso librerías de SolidWorks) para ser utilizadas por otro software.

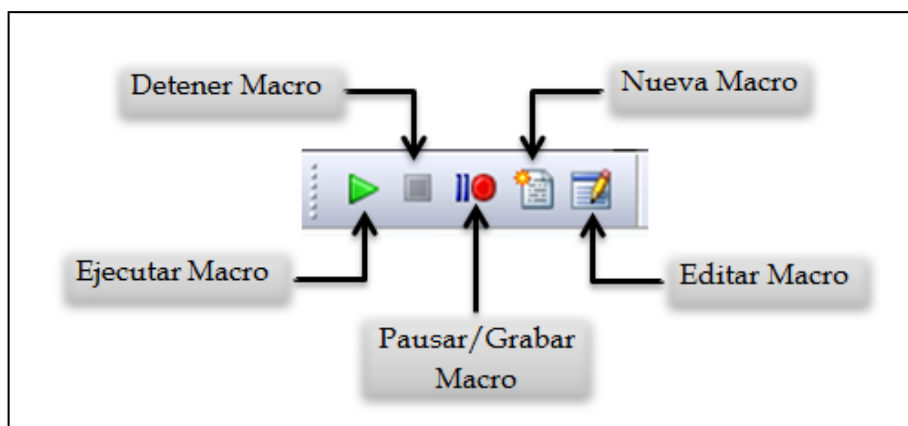
¹⁰ La información presentada fue obtenida de la página web de ayuda para API's de SolidWorks : <http://help.solidworks.com/2013/English/api/sldworksapiproguide/Welcome.htm>

Los API pueden ser utilizados para automatizar y personalizar SolidWorks, ya que contienen cientos de funciones que pueden ser llamadas desde Visual Basic para aplicaciones (VBA), VB.NET, Visual C#, Visual C++ 6.0 y Visual C++/CLI. Dichas funciones proveen acceso directo a las diferentes funcionalidades de SolidWorks. Reduciendo tiempo y esfuerzo a los usuarios del software, a la vez que se reducen errores en el modelado a gran escala de productos, evitando la realización de procedimientos repetitivos, sean estos complejos o no.

El lenguaje preestablecido por SolidWorks en la creación de dichos API's es VBA (del Inglés, *Microsoft Visual Basic for Applications*), el cual se encuentra embebido a SolidWorks. Microsoft VBA, permite ejecutar y editar, VBA macros en SolidWorks.

La forma más rápida y sencilla de programar con los API's de SolidWorks, es grabando una macro (archivo de extensión *.swp), la cual contiene todos los llamados de los API de SolidWorks correspondientes a las acciones desarrolladas durante el modelado de una pieza específica. Para grabar una macro en SolidWorks se ha de llevar a cabo el siguiente procedimiento:

Figura 5. Barra de Herramientas Macro




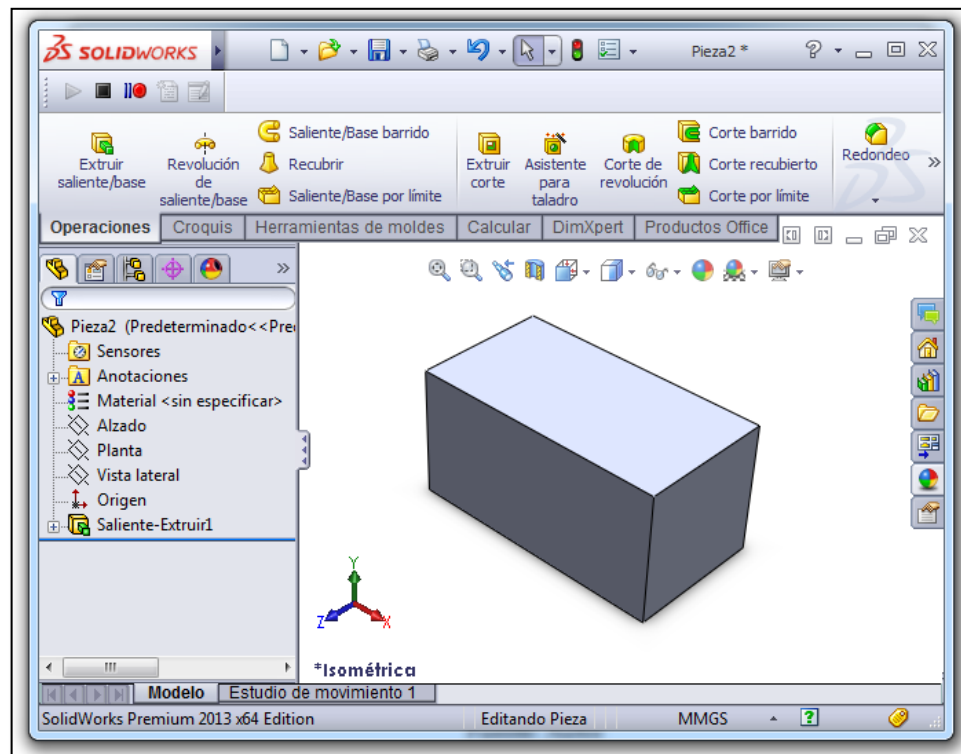



- I. Habilitar la barra de herramientas macro (figura 5) , siguiendo los siguientes pasos:
 - a. Clic en el menú *Opciones* de la barra de menús
 - b. Clic en *Personalizar*
 - c. En la pestaña *Barras de Herramientas* se selecciona el ítem *Macro*
 - d. Clic en aceptar
- II. Clic en *Pausar/Grabar Macro*  en la barra de herramientas macro.
- III. Se realiza la operación que se desea grabar. A continuación se muestra a manera de ejemplo una extrusión rectangular (figura 6):

Figura 6. Ventana de SolidWorks: Extrusión rectangular



IV. Una vez se ha realizado el modelado de la pieza, se da clic en Detener Macro  en la barra de herramientas macro.

V. En el cuadro de dialogo, escribir un nombre para el archivo, seleccionar el tipo de macro y luego dar clic en Guardar.

VI. Si se desea detener el macro mientras está grabando dar clic en Pausar/Grabar  en la barra de herramientas macro. Para reanudar, clic nuevamente en Pausar/Grabar .


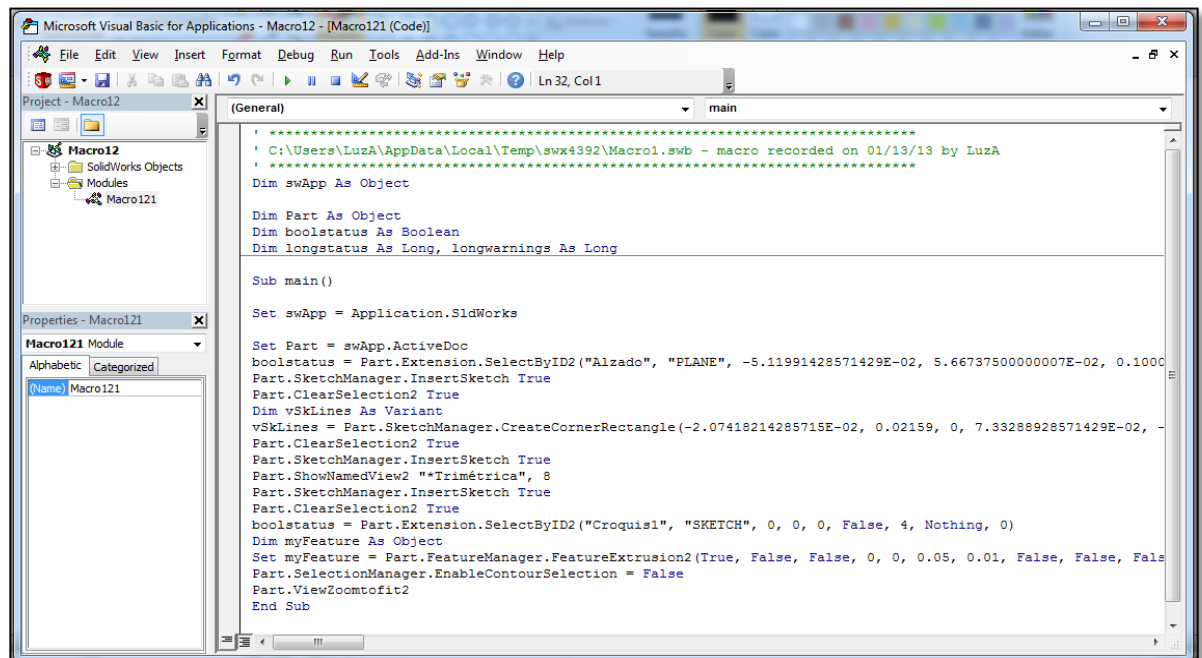
Para ver el código VBA se da clic en Editar Macro  en la barra de herramientas macro, y en el cuadro de dialogo seleccionar el archivo deseado. A continuación puede observarse el código obtenido al grabar la macro de SolidWorks (figura 7).

Figura 7. Ventana VBA; Código obtenido mediante grabación del macro



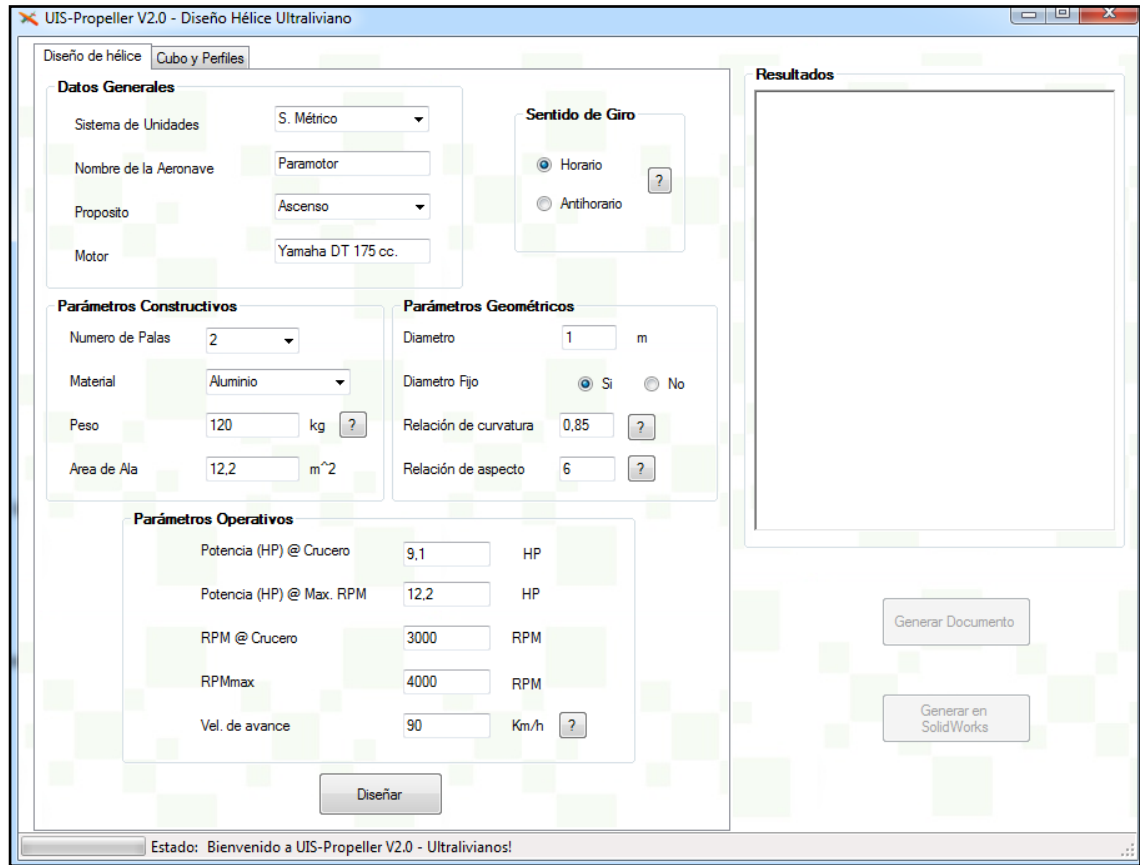
- a. Dim swApp As Object
- b. Dim Part As Object
- c. Dim boolstatus As Boolean
- d. Dim longstatus As Long, longwarnings As Long
- e. Sub main()
- f. Set swApp = Application.SldWorks
- g. Set Part = swApp.ActiveDoc
- h. boolstatus = Part.Extension.SelectByID2("Alzado", "PLANE", -5.11991428571429E-02, 5.66737500000007E-02, 0.100000000000023, False, 0, Nothing, 0)
- i. Part.SketchManager.InsertSketch True
- j. Part.ClearSelection2 True
- k. Dim vSkLines As Variant
- l. vSkLines = Part.SketchManager.CreateCornerRectangle(-2.07418214285715E-02, 0.02159, 0, 7.33288928571429E-02, -2.58308928571429E-02, 0)
- m. Part.ClearSelection2 True
- n. Part.SketchManager.InsertSketch True
- o. Part.ShowNamedView2 "*Trimétrica", 8
- p. Part.SketchManager.InsertSketch True
- q. Part.ClearSelection2 True
- r. boolstatus = Part.Extension.SelectByID2("Croquis1", "SKETCH", 0, 0, 0, False, 4, Nothing, 0)
- s. Dim myFeature As Object
- t. Set myFeature = Part.FeatureManager.FeatureExtrusion2(True, False, False, 0, 0, 0.05, 0.01, False, False, False, False, 1.74532925199433E-02, 1.74532925199433E-02, False, False, False, False, True, True, True, 0, 0, False)
- u. Part.SelectionManager.EnableContourSelection = False
- v. End Sub

Seguidamente son explicadas algunas de las sentencias de VBA utilizadas en el procedimiento anterior:

- ▶ *Sub Main()*: Indica la creación de un nuevo procedimiento en Microsoft Visual Basic (del tipo Sub), y habilita las operaciones en Solidworks, es decir define donde inicia la macro.
- ▶ *Set swApp = Application.SldWorks*: Esta línea es utilizada para conectar la aplicación de SolidWorks que se está ejecutando con la variable swApp definida en la primera línea. Una vez realizada esta conexión se utiliza “swApp” para referirse a la aplicación de SolidWorks en lugar de Application.SldWorks.
- ▶ *Set Part = swApp.ActiveDoc*: Se utiliza para crear una pieza en el documento activo de Solidworks.
- ▶ *Part.Extension.SelectByID2*: Se utiliza para seleccionar entidades.
- ▶ *Part.SketchManager.InsertSketch True*: Habilita la creación de croquis.
- ▶ *Part.SketchManager.CreateCornerRectangle*: Se especifican las dimensiones del rectángulo a ser croquizado.
- ▶ *Part.ClearSelection2 True*: Es utilizado para deseleccionar entidades.
- ▶ *Part.FeatureManager.FeatureExtrusion2*: Es utilizado para crear la Extrusión del croquis seleccionado previamente.
- ▶ *Part.SelectionManager.EnableContourSelection = False*: Es utilizado para deseleccionar el contorno de la pieza.
- ▶ *End Sub()*: Define dónde finaliza la macro.

3. ACTUALIZACIÓN DE LA FUNCIÓN DE DISEÑO DE HÉLICES PARA ULTRALIVIANO (HÉLICES PROPULSORAS)

Figura 8. Ventana Interfaz Visual *UIS-Propeller V2.0*-Diseño Hélice Ultraliviano



3.1 PROCESO DE DISEÑO DE HÉLICES PARA ULTRALIVIANO

Tal como fue mencionado en el numeral 2.2 del marco teórico de éste proyecto, el proceso de diseño de hélices para ultralivianos implementado en *UIS-Popeller* se basa en la teoría de elementos de pala modificada, la cual fue expuesta por Fred E. Weick en los reportes “*Practical application of the blade element theory - I-II-III and IV*”, realizados para la NACA. Dicha teoría, permite el diseño paso a paso de la hélice para ultraliviano bajo las condiciones iniciales especificadas por el usuario

(ver figura 8). El proceso de diseño de hélices para ultraliviano paso a paso, se encuentra detallado en el ANEXO B, el cual corresponde al procedimiento implementado en la versión 1.0 de *UIS Propeller*.

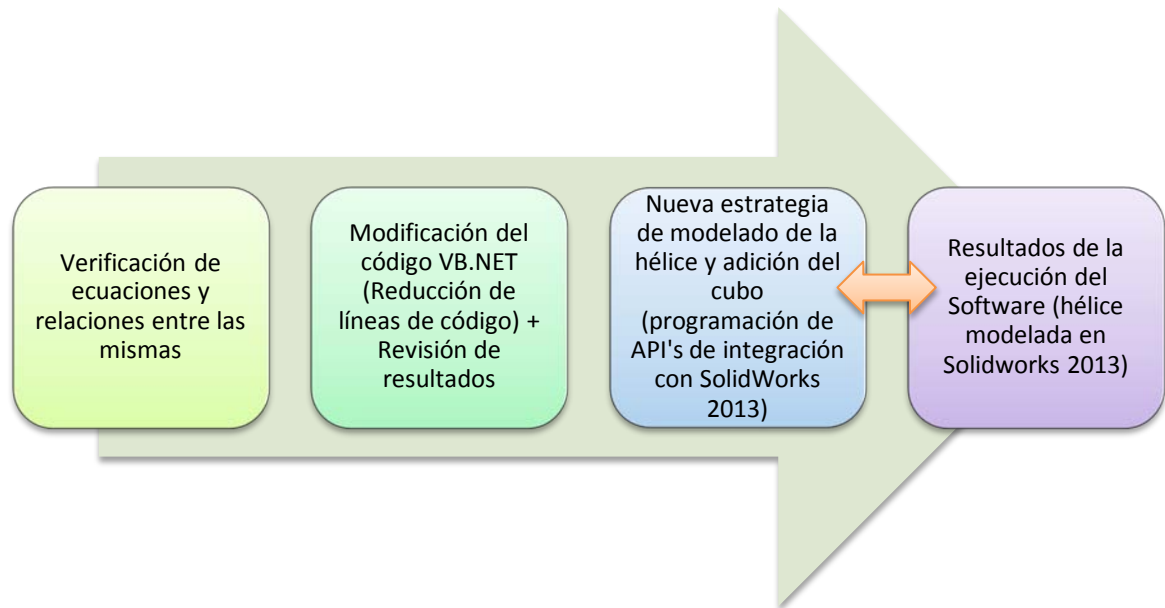
3.2 PROCESO DE ACTUALIZACIÓN DE LA FUNCIÓN PARA EL DISEÑO DE HÉLICES DE ULTRALIVIANO

Para desarrollar el proceso de actualización de *UIS-Propeller*, fue indispensable tomar en consideración las desventajas y/o falencias presentes en la primera versión del mismo (ver numeral 1.1), dichas desventajas son enumeradas a continuación:

- ▶ El software no implementa ciclos o bucles de programación, por lo cual se compone de 14593 líneas de código, siendo por ende vulnerable a errores durante la ejecución del mismo.
- ▶ Es incompatible con versiones recientes de SolidWorks.
- ▶ Tiende a presentar errores durante el modelado de la hélice para propósitos de vuelo diferentes a *Estándar*.
- ▶ Presenta múltiples interferencias geométricas (modelado en SolidWorks) en el proceso de construcción del cubo de la hélice.

La versión actualizada de *UIS-Propeller* se desarrolló bajo la plataforma de Visual Studio 2010, en lenguaje VisualBasic.NET. En adición, el modelado de la hélice se llevó a cabo mediante la implementación de API's de SolidWorks 2013. A continuación, se explicará de manera detallada cada uno de los pasos llevados a cabo durante el proceso de actualización de la función de diseño de hélices para ultraliviano de *UIS-Propeller* (ver figura 9).

Figura 9. Proceso de actualización de la función de diseño de hélices para ultralivianos de *UIS-Propeller*



3.2.1 Verificación de ecuaciones del proceso de diseño de hélices para ultralivianos¹¹¹². Con el propósito de garantizar a los usuarios del software la confiabilidad de *UIS-Propeller*, se verificaron cada una de las ecuaciones utilizadas durante el proceso de diseño de hélices, al igual que las relaciones existentes entre las mismas para cada uno de los propósitos de vuelo (Ascenso, Estándar, Crucero y Velocidad), todo esto tanto para diámetro fijo como variable.

Para el proceso de diseño de la hélice son necesarias una serie de variables de entrada que han de ser proporcionadas por el usuario, tal como se puede observar en la tabla 1:

¹¹ También conocidas como hélices propulsoras.

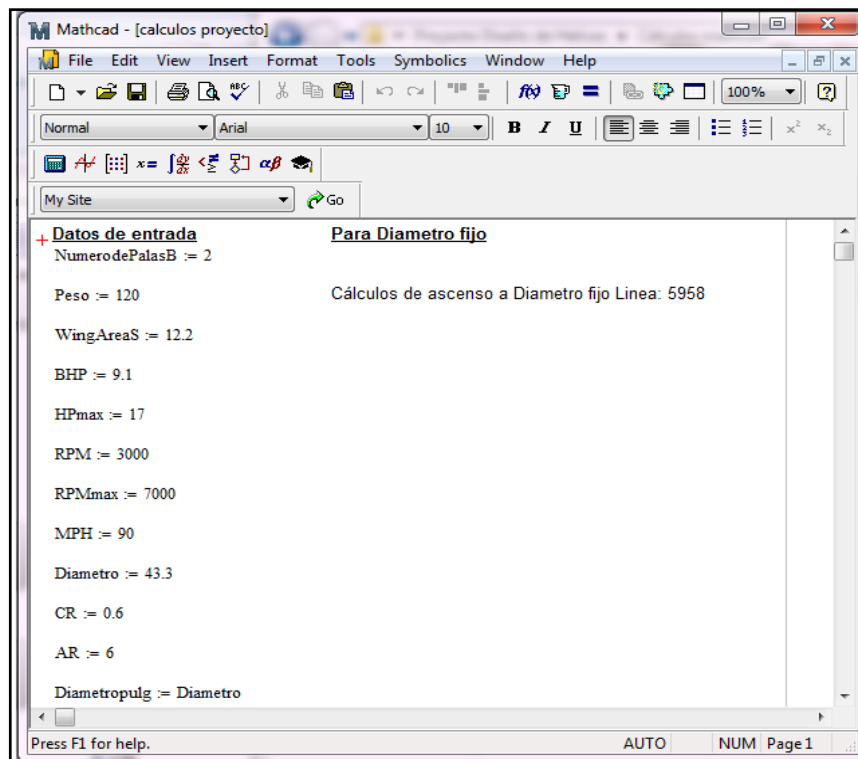
¹² Para lograr un mayor entendimiento de las ecuaciones que serán presentadas se puede recurrir al ANEXO B. (proceso de diseño de hélices) y al ANEXO C. (Nomenclatura utilizada por Weick en el diseño de hélices).

Tabla 1. Descripción de variables de entrada UIS-Propeller Ultraliviano

VARIABLES DE ENTRADA UIS-Propeller	
Datos Generales	
Unidades	Seleccionar el sistema de unidades deseado: <ul style="list-style-type: none"> ✓ Sistema Métrico ✓ Sistema Americano
Aeronave	Escribir el nombre de la aeronave
Propósito	Seleccionar el propósito de vuelo: <ul style="list-style-type: none"> ✓ Ascenso ✓ Estándar ✓ Crucero ✓ Velocidad
Motor	Escribir la referencia del motor
Sentido de Giro	
Sentido de Giro	Seleccionar el sentido de giro deseado: <ul style="list-style-type: none"> ✓ Horario ✓ Antihorario
Datos constructivos	
Número de palas	Seleccionar el número de palas deseado, 2, 3 ó 4.
Material	Seleccionar el material deseado: <ul style="list-style-type: none"> ✓ Madera ✓ Aluminio ✓ Epoxy/Fibra de vidrio
Peso	Peso del aeronave con pasajero incluido
Área del ala	Proporcionar dato
Datos operativos de la hélice	
Potencia HP a crucero	Proporcionar dato
Potencia HP a máx. RPM	Proporcionar dato
RPM a crucero	Proporcionar dato
RPM máx.	Proporcionar dato
Velocidad de avance	Velocidad de avance de la aeronave bajo condiciones de crucero o de velocidad máxima.
Datos geométricos	
Diámetro	Proporcionar dato
Diámetro Fijo o variable	Seleccionar dato
Relación de curvatura	Proporcionar dato
Relación de Aspecto	Proporcionar dato

Una vez conocidas las variables de entrada, se procedió a comparar las ecuaciones utilizadas en el software, con las ecuaciones expuestas por Weick en sus papers¹³. Teniendo en cuenta que varias de estas fueron obtenidas mediante regresiones matemáticas de los monogramas presentados por Weick, una comprobación de dichas regresiones fue llevada a cabo con la ayuda del software matemático Mathcad (Figura 10).

Figura 10. Introducción de variables de entrada en Mathcad

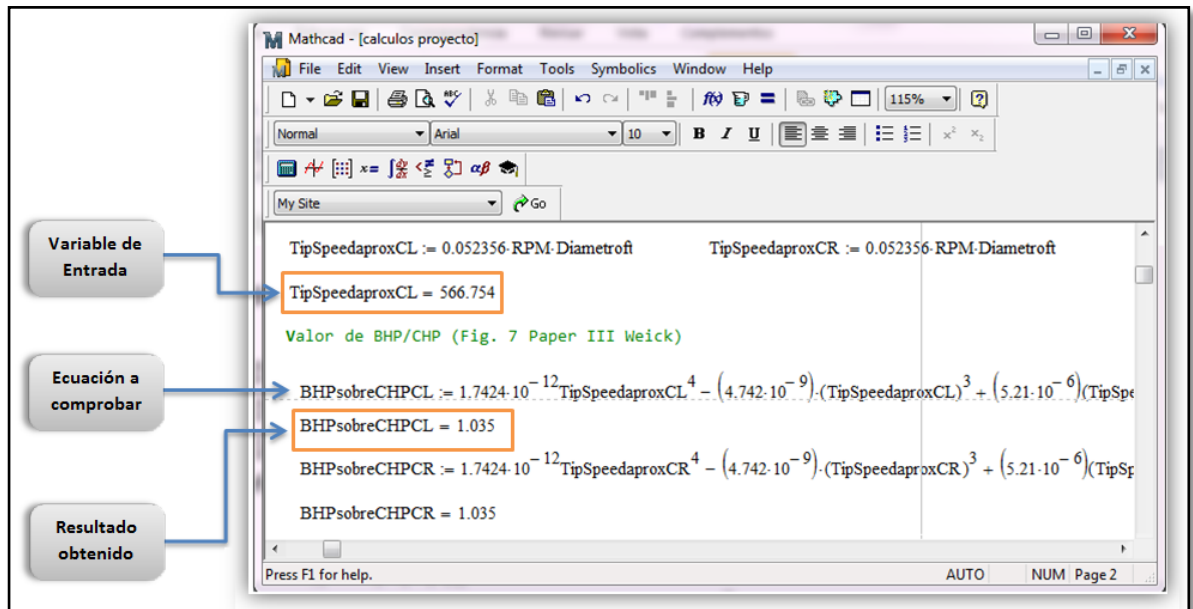


Para efectuar la fase de comprobación mencionada anteriormente, fue necesario introducir las ecuaciones correspondientes al procedimiento de diseño básico de la hélice en Mathcad. Este consistió en dar diferentes valores a las variables de entrada de las ecuaciones, para posteriormente contrastar sus resultados con los

¹³ Weick, F.E (1926). Practical application of the blade element theory -I-II-III-IV.

valores obtenidos al realizar dicho procedimiento de forma manual con la ayuda de gráficas y monogramas¹⁴. En las figuras 11 y 12, se ejemplifica el procedimiento descrito anteriormente.

Figura 11. Obtención de resultados mediante la utilización de ecuaciones en Mathcad

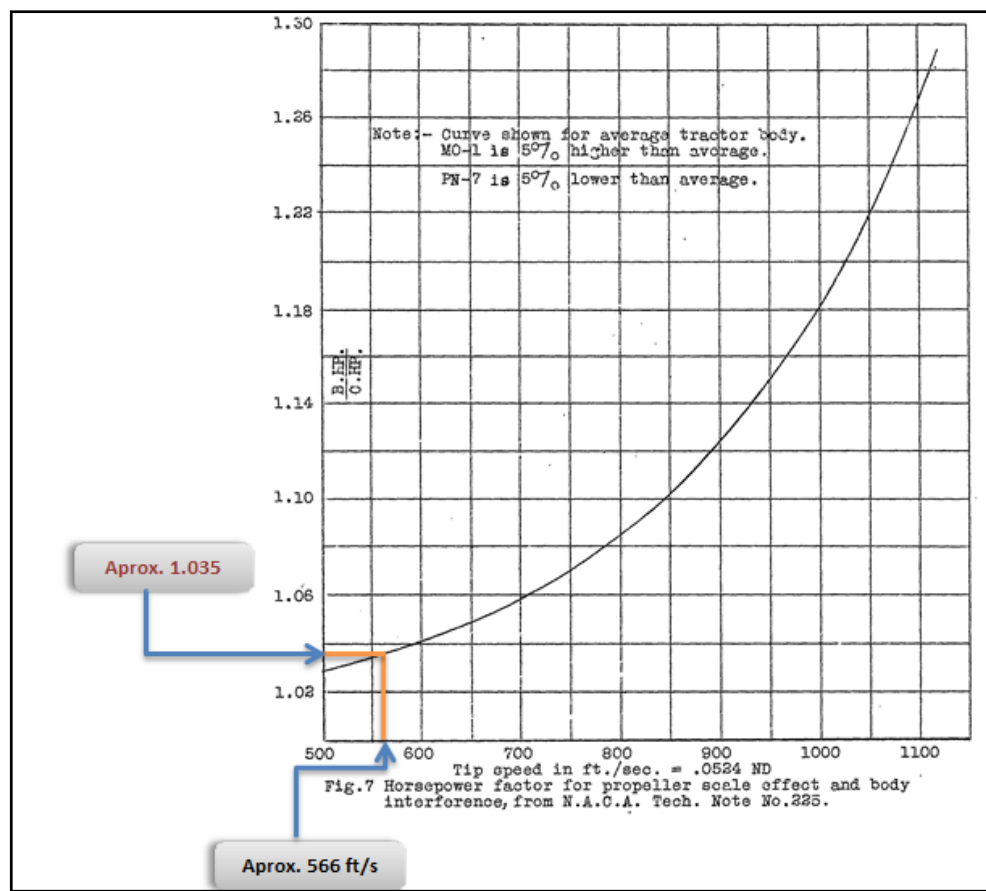


En la figura 11 se puede apreciar el valor de la velocidad de punta o *Tip Speed* (el cual es 566.754), hallado con las variables de entrada proporcionadas por el usuario del software (figura 10) para el propósito de vuelo ascenso (*CL*, del Inglés Climb), el cual, es a su vez la variable de entrada a la ecuación (Regresión) que desea verificarse (en este caso particular, la relación entre la potencia de freno *BHP* y la

¹⁴ Los datos presentados en monogramas y gráficas por Fred E. Weick en sus papers, son en su mayoría el resultado de pruebas experimentales, tanto a hélices modelo en túneles de viento, como a hélices a escala real en vuelo, realizadas por la NACA a principios del siglo pasado, las cuales aún hoy en día conservan su validez.

potencia considerando el factor de forma¹⁵ para mayor información sobre estas variables ver ANEXO B). Una vez reemplazado el valor de *tip speed* en la ecuación, se obtuvo que $BHP/CHP = 1.035$. Posteriormente, con el valor de este y con la ayuda del monograma correspondiente (figura 7 del paper “*Practical Application of the blade element theory III*”), fue comprobada la fiabilidad de dicho resultado.

Figura 12. Estimación de valores mediante la utilización de gráficas (Weick, F.E. 1926. Practical Application of the blade element theory III)

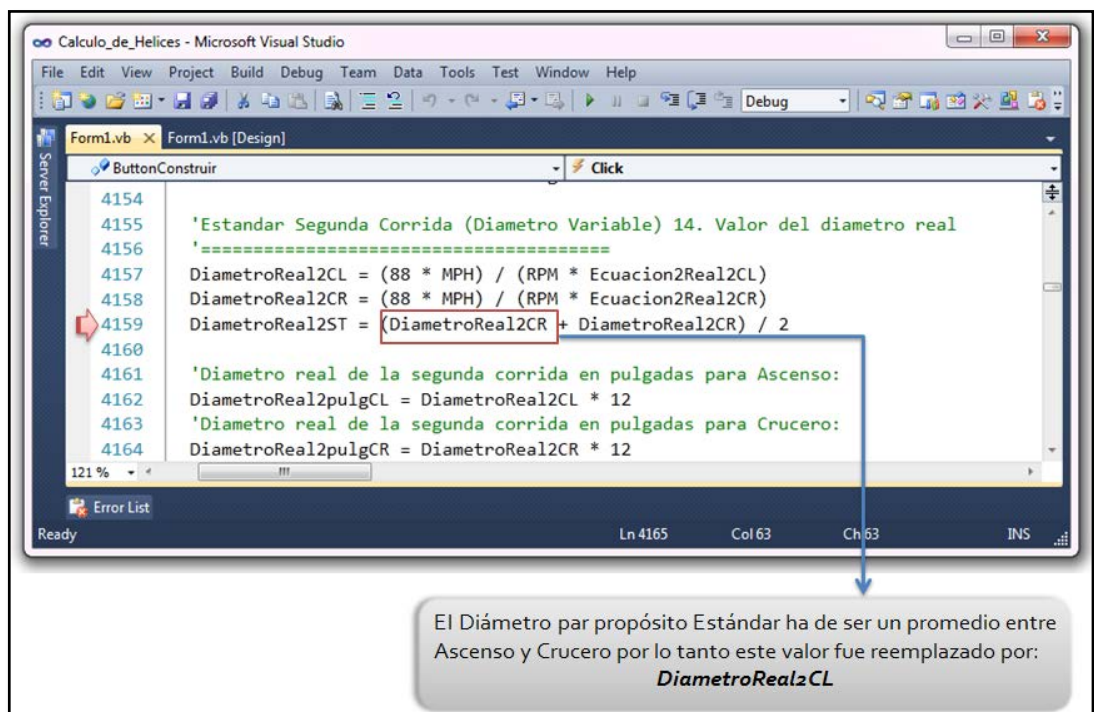


¹⁵ Para hélices diferentes a las Navales Estándar (en cuyo caso el valor es igual a 1), el factor de forma depende de la velocidad máxima, la distorsión de la hélice y la interferencia del fuselaje.

Cabe aclarar que con el propósito de garantizar la efectividad de las regresiones, el procedimiento descrito anteriormente, se realizó para diferentes valores de las variables de entrada de las ecuaciones revisadas.

Luego de revisar las ecuaciones implementadas en el software, se comprobó que estas coinciden con las utilizadas por la NACA para el diseño de hélices y que aquellas obtenidas mediante regresiones, arrojan resultados coincidentes con los que serían obtenidos de las gráficas y monogramas. Por otro lado, se encontró un pequeño, pero significativo error en la ecuación (ubicada en la línea de código 4159) para el cálculo de diámetro real (diámetro de hélice variable) correspondiente al propósito de vuelo estándar (para el que la mayoría de propiedades es un promedio entre vuelo en ascenso y vuelo en crucero), el cual fue corregido satisfactoriamente (figura 13).

Figura 13. Corrección Ecuación línea 4159 del código de programación



Una vez comprobadas las ecuaciones, se verificaron las relaciones existentes entre cada una de estas a los diferentes propósitos de vuelo (Ascenso, Estándar, Crucero y Velocidad), para diámetro fijo y variable. Por consiguiente, cada línea del código VB desarrollado en la primera versión del software fue revisada.

Tal como puede observarse en la figura 8 y en la tabla 1, *UIS-Propeller* ofrece al usuario la posibilidad de establecer si desea un diámetro fijo (en caso de que cuente con restricciones dimensionales) o un diámetro variable; en cuyo caso el software desarrolla un proceso iterativo de cálculo para establecer el diámetro bajo el cual la hélice tendrá un mejor desempeño de acuerdo a las condiciones establecidas previamente por el usuario (variables de entrada), tal iteración ha de finalizar una vez satisfecha la condición presentada en (2). Donde para la primera iteración el valor de Diámetro aproximado (D_{approx}) es el del diámetro proporcionado por el usuario y el Diámetro calculado (D_{calc}) corresponde al hallado mediante la ecuación (1). Si el porcentaje de error obtenido no cumple la condición establecida, el D_{calc} será el D_{approx} de la siguiente iteración y así sucesivamente hasta que el porcentaje de error sea aceptable.

$$D_{calculado} = \frac{88 * MPH}{N * V/nD} \quad (1)^{16\ 17}$$

$$Error \rightarrow D_{approx} - D_{calc} \leq 0.05 * D_{approx} \quad (2)$$

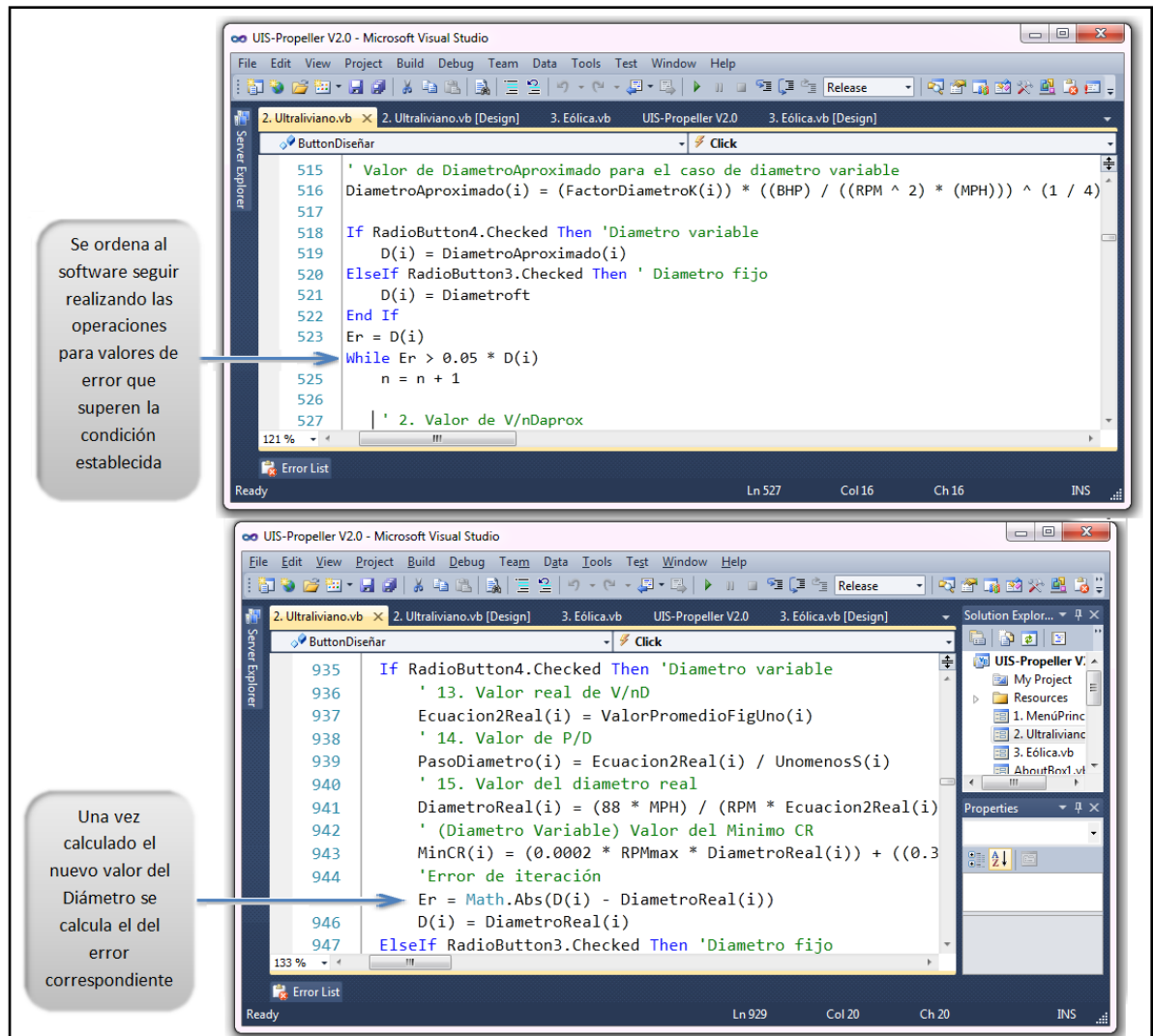
Luego de repasar las líneas del código VB de *UIS-Propeller V1.0*, se encontró que el software no verifica el cumplimiento del mínimo porcentaje de error permitido (2),

¹⁶ Para mayor entendimiento de las ecuaciones se ver ANEXO B y/o ANEXO C.

¹⁷ En el ANEXO J se muestra la referencia bibliográfica de la cual fue tomada cada una de las ecuaciones utilizadas en este proyecto.

por lo cual fue necesario agregar nuevas líneas de código que se encarguen de comprobar dicha condición, según lo establecido en el párrafo anterior (figura 14).

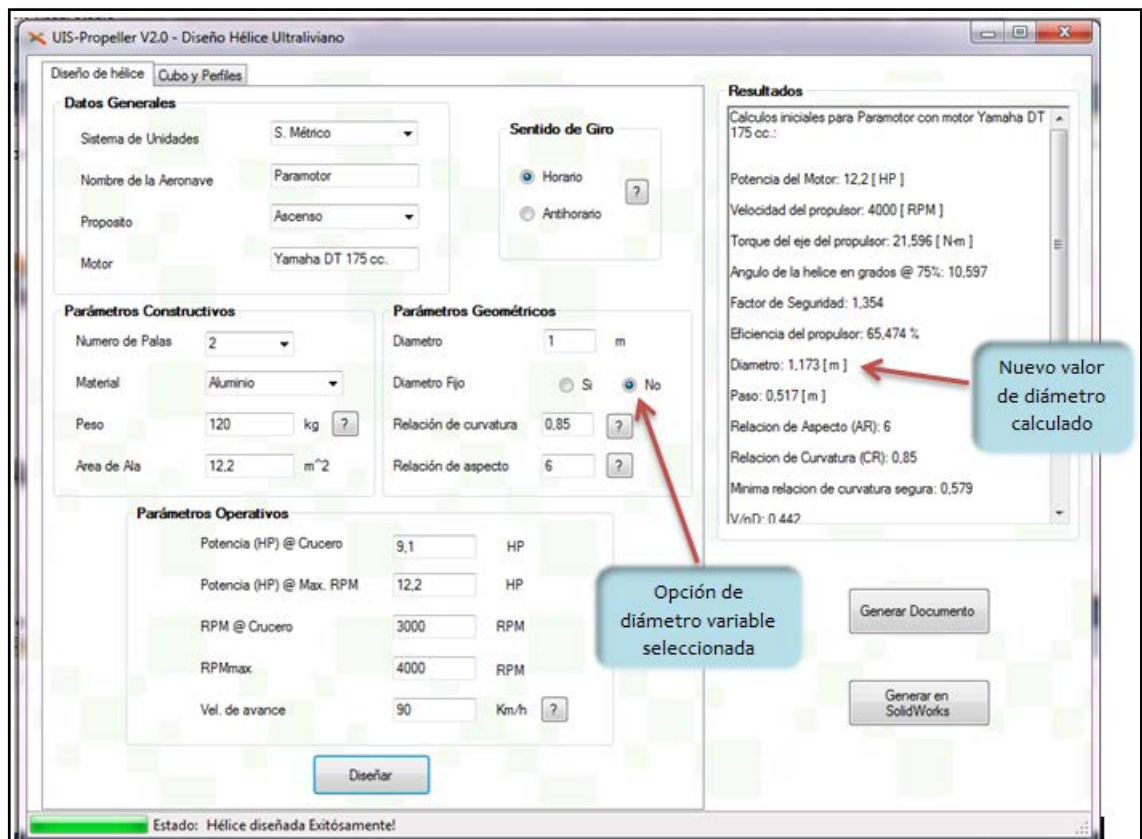
Figura 14. Líneas de código VB adicionadas al programa para calcular y verificar el cumplimiento del mínimo porcentaje de error permitido para diámetro variable.



Durante el proceso de revisión, también se advirtió que a pesar de que el software hallaba el valor del diámetro para el caso de diámetro variable y las propiedades correspondientes al mismo, no muestra al usuario el valor calculado; sumado a

esto, también se advirtió que el software no realizaba el modelado de la hélice con las dimensiones halladas, sino con las establecidas por el usuario en primera instancia (diámetro fijo). Por lo anteriormente expuesto, se hizo necesaria la modificación del código correspondiente al cálculo de coordenadas geométricas para el caso de diámetro variable y la adición de una línea de código en la cual se ordenara al software mostrar el nuevo valor del diámetro al usuario. La figura 15 muestra los resultados del proceso de diseño para diámetro variable, donde en el recuadro de resultados puede ser apreciado el nuevo valor del diámetro.

Figura 15. Resultados del proceso de diseño para diámetro variable una vez modificado el código



3.2.2 Modificación del código VB.NET (Reducción de líneas de código). Como se mencionó al inicio del numeral 3.2, una de las desventajas de la primera versión de *UIS-Propeller* es la gran extensión de líneas de código (14593 líneas de código), las cuales podrían ser reducidas de manera significativa mediante la implementación de ciclos (bucles) de programación tales como son los ciclos *for* y *while*, adicionando a esto el condicional *if*, y demás estrategias de programación derivadas de la aplicación de los ciclos mencionados (tales como *Break*, *Continue*, *Exit*, y sus respectivos contadores, vectores y matrices)¹⁸.

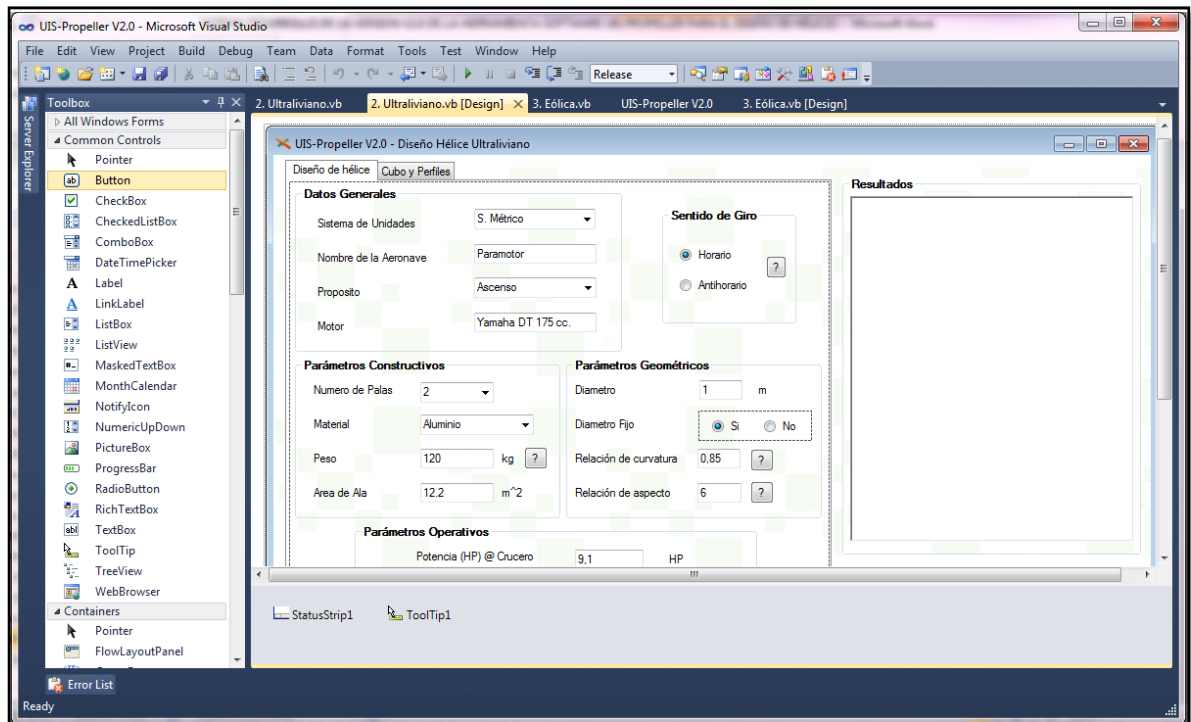
Cabe aclarar que previo a la modificación del código programado en la primera versión de *UIS-Propeller*, se ha de crear la nueva interfaz visual bajo la plataforma Visual Studio 2010 mediante la implementación de las Windows Applications Forms (Aplicaciones de Formularios de Windows)¹⁹, y sus respectivas *Windows Forms*, que son los botones, cuadros de texto, barras de progreso y demás controles y herramientas necesarias para el desarrollo de aplicaciones y software. La interfaz visual de la función de diseño de hélices para ultralivianos de *UIS-Propeller V2.0* se muestra en la figura 16, al igual que la ventana del *Toolbox* (caja de herramientas) que Microsoft Visual Studio ofrece a sus usuarios.

¹⁸ Para información detallada acerca del lenguaje de programación en Visual Basic 2010 (al igual que de versiones anteriores y posteriores, éste lenguaje y los demás soportado por Microsoft Visual Studio), se recomienda visitar la Guía de Programación en Visual Basic de la página web de Microsoft Visual Studio, la cual se encuentra disponible en el siguiente enlace:

<[http://msdn.microsoft.com/es-co/library/y4wf33f0\(v=vs.100\).aspx](http://msdn.microsoft.com/es-co/library/y4wf33f0(v=vs.100).aspx)>

¹⁹ Una descripción más detallada de la interfaz visual de *UIS-Propeller V2.0* será dada a conocer en el capítulo 5.

Figura 16. Creación de la interfaz visual de la función de diseño de hélices para ultraliviano *UIS-Propeller*



Una vez creada la interfaz visual, se procedió a modificar la programación del proceso diseño de hélices.

El primer paso a seguir, es declarar como públicas las variables a ser empleadas en el diseño de hélices, de ésta forma podrán ser utilizadas en todo el programa sin inconveniente alguno. Dichas variables se han de definir “As String” o “As Double” según sea el caso (caracteres alfabéticos o numéricos respectivamente), tal como se muestra a continuación:

- ▶ Caracteres alfabéticos: `Public Unidades As String = Nothing`
- ▶ Caracteres numéricos: `Public RPM As Double = Nothing`

Durante la declaración de variables, es igualmente importante especificar cuáles son escalares, cuales son vectores y cuales son matrices, del siguiente modo:

- ▶ Escalares: `Public Diametroft As Double = Nothing`
- ▶ Vectores: `Public DiametroAproximado As Double()`
- ▶ Matrices: `Public X As Double(,)`

Luego se procedió a ordenar la obtención de los datos de entrada mediante, la utilización de los comandos “*CStr*” y “*Cdbl*”, según sea el caso (caracteres alfabéticos o numéricos respectivamente), tal como se muestra a continuación:

- ▶ En caso de ser un carácter alfabético: `Motor = CStr(TextBoxMotor.Text)`
- ▶ En caso de ser un carácter numérico: `CR = Cdbl(TextBoxCR.Text)`

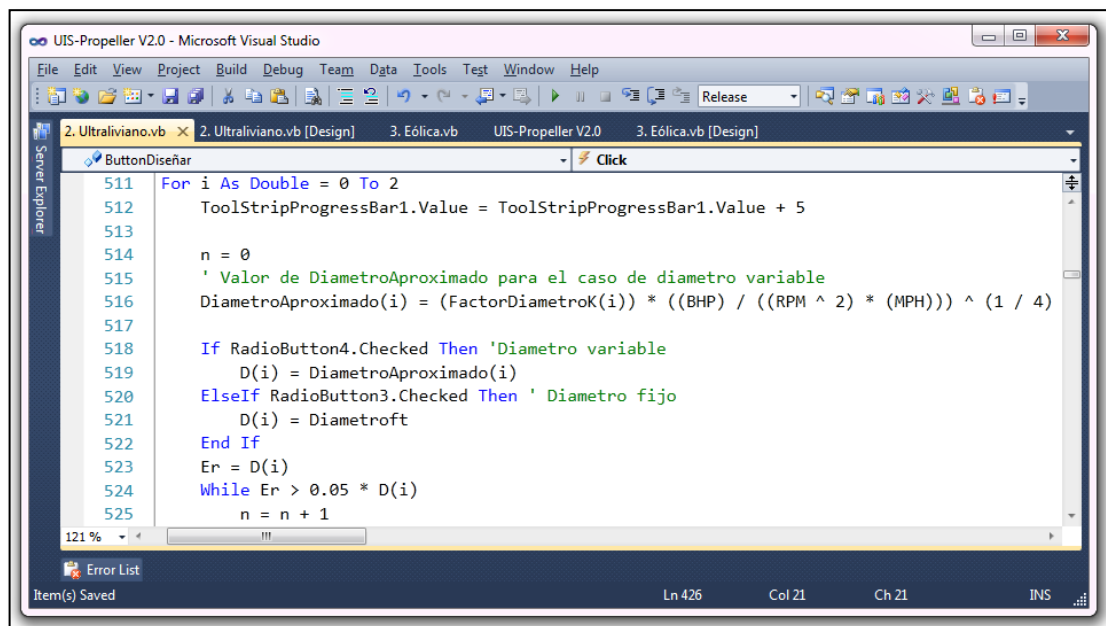
Es así como, declaradas las variables y adquiridos los datos suministrados por el usuario (variables de entrada), se procede a realizar el proceso de diseño inicial de la hélice, para diámetro variable o diámetro fijo según haya sido establecido. Tal como fue mencionado en el numeral 3.2.1, si el diámetro es variable, se debe realizar todo el procedimiento dos veces o hasta que la condición de error sea aceptable, de tal forma se garantiza la confiabilidad del diámetro calculado; acercándolo más al ideal para el propósito de vuelo seleccionado (Estándar, Ascenso, Crucero o Velocidad).

Antes de iniciar el proceso de programación, se ha de aclarar que el procedimiento para vuelo estándar es realizado separadamente de los otros propósitos, ya que éste es un promedio entre los resultados obtenidos para el propósito ascenso y el propósito crucero.

De acuerdo a lo expuesto en los dos párrafos anteriores, se vislumbra la necesidad de emplear un ciclo *for* para *i* (contador de los vectores) que vaya de 0 a 2, lo que equivale a 3 corridas del ciclo *for*; una para cada propósito, siendo el propósito ascenso calculado en la primera corrida del ciclo *for*, al cual le corresponde el indicativo (0); el propósito crucero calculado en la segunda corrida del ciclo *for*, al cual le corresponde el indicativo (1); de igual forma el propósito velocidad es calculado en la tercera corrida del ciclo *for*, por consiguiente le corresponde el indicativo (2). Las propiedades para el propósito estándar son calculadas por fuera del ciclo *for*, dado que como se mencionó anteriormente, estas son valores promedio entre ascenso y crucero.

En la figura 17, se puede observar la aplicación del ciclo *for*, para el diseño de hélices y las variables expresadas en términos vectoriales mediante el contador *i*.

Figura 17. Ciclo *For* aplicado al cálculo de hélices para cada uno de los propósitos de vuelo



```
511 For i As Double = 0 To 2
512     ToolStripProgressBar1.Value = ToolStripProgressBar1.Value + 5
513
514     n = 0
515     ' Valor de DiametroAproximado para el caso de diametro variable
516     DiametroAproximado(i) = (FactorDiametroK(i)) * ((BHP) / ((RPM ^ 2) * (MPH))) ^ (1 / 4)
517
518     If RadioButton4.Checked Then 'Diametro variable
519         D(i) = DiametroAproximado(i)
520     ElseIf RadioButton3.Checked Then ' Diametro fijo
521         D(i) = Diametroft
522     End If
523     Er = D(i)
524     While Er > 0.05 * D(i)
525         n = n + 1
```

Dada la necesidad del cumplimiento de la condición de error mínima ($\text{error} \leq 0.05 \cdot D$), se hace necesaria la implementación del ciclo *while* justo antes de introducir las ecuaciones de diseño, por tanto; para el caso de diámetro variable sólo se abandonará el ciclo *while* cuando el error sea menor o igual al 5% del diámetro aproximado. Para diámetro fijo, las operaciones han de realizarse una sólo vez por lo cual se introduce el siguiente código o condicional:

```
If RadioButton3.Checked And n = 1 Then
Exit While
End If
```

Donde en este caso particular el RadioButton3 equivale a la opción de selección de diámetro fijo y n equivale a una sumatoria llevada a cabo al cada vez que el ciclo *while* sea recorrido (adicionando una unidad cada vez; figura 17).

Una vez finalizado el ciclo *for*, para la obtención de los valores definitivos (resultados del proceso de diseño) de las propiedades correspondientes a cada uno de los propósitos de vuelo, se hace uso de la instrucción *Select Case*, de la siguiente manera:

```
Select Case Proposito
Case 0 ' Ascenso
    'Valor del Mínimo CR
    If RadioButton4.Checked Then ' Diámetro variable
        MinCRr = MinCR(0)
        :
Case 1 'Estándar
    'Valor del Mínimo CR
```

```

If RadioButton4.Checked Then ' Diámetro variable

MinCRr=0.0002*RPMmax*((D(0)+D(1))/2)+((0.3((AR*NumerodePalasB)/2))
-4.3)
    :
Case 2 ' Crucero
    'Valor del Mínimo CR
    If RadioButton4.Checked Then ' Diámetro variable
        MinCRr = MinCR(1)
        :
Case 3 ' Velocidad
    'Valor del Mínimo CR
    If RadioButton4.Checked Then ' Diámetro variable
        MinCRr = MinCR(2)
        :
End Select

```

En el extracto de código anterior, \vdots simboliza la continuación del código para otras variables, así mismo, se puede apreciar el cálculo de las propiedades para vuelo estándar como un promedio de entre ascenso y crucero.

Una vez finalizado el proceso de diseño inicial de la hélice, se procede a realizar el diseño geométrico de la misma. Para dicho propósito se definieron las coordenadas de los perfiles para cada estación en forma de vectorial de la siguiente manera:

```

cX = {1, 0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1, 0.05,...}
cyu = {0.07, 0.35, 0.56, 0.74, 0.87, 0.95, 0.99, 0.998, 0.95,...}
cyl = {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.25, 0.41, ...}

```

Para el diseño geométrico de la hélice, se definen 7 estaciones (al 15%, 30%, 45%, 60%, 75%, 90% y 100% del radio de la hélice), donde cada una equivale a un perfil aerodinámico conformado por 27 puntos (equivalente al número de elementos que conforman los vectores mostrados anteriormente) con excepción de los perfiles al 30% y 45%, los cuales están conformados por 28 puntos.

Nuevamente se hace necesario el uso de ciclos *for*, para el cálculo de las coordenadas geométricas, siguiendo el siguiente patrón:

```
For i As Double = 0 To nel
  Anchobr(i)
  'Valor de hu y hl para cada posición
  hu(i)
  hl(i)
  'Centro de gravedad para cada posición
  CGx(i)
  CGy(i)
  'Ubicación del centro de gravedad en la vista frontal
  PFCG(i)
  'Ubicación del centro de gravedad en la vista lateral
  PLCG(i)
  'Datos del perfil normal para cada estación
  If i = 1 Then ' 15%
  Vector cX
  ElseIf i = 2 Then ' 30%
  Vector cX
  :
End If
If i = 1 Then ' 15%
```

```

Vector cyu
Vector cyl
ElseIf i = 2 Then ' 30%
Vector cyu
Vector cyl
    ⋮
End If
    ⋮
For j As Double = 0 To nc
    X(i, j) = Anchobr(i) * cX(j) - CGx(i)
    yul(i, j) = hl(i) * cyl(j) + hu(i) * cyu(j) - CGy(i)
Next
Next

```

En el código anterior, *nel* hace referencia al número de elementos o estaciones, mientras que *nc* al número de puntos (o coordenadas) que conforman cada perfil.

De la programación anterior, se obtuvo de forma matricial los valores de las coordenadas *x* y *yul* (coordenada *y* superior (*u*; del Inglés upper) e inferior (*l*; del Inglés lower), correspondientes a matrices 7x27 o 7x28 según sea el caso. Esto para el perfil normal.

Una vez calculados estos valores, se efectúan nuevamente ciclos *for*, para hallar el valor de las coordenadas del perfil girado al ángulo de giro (*φb* o *phib*) correspondiente a cada estación de la hélice. Finalizando con esto el proceso de diseño geométrico de la hélice.



NOTA: El ciclo *for* fue a su vez empleado en la programación de los API's de integración con SolidWorks 2013, como será expuesto en el numeral 3.2.4.

Cabe destacar, que luego de emplear ciclos o bucles para la programación del software, se redujo el número de líneas de programación a 2473; lo que equivale a una reducción de líneas programadas del 83%, haciendo por ende más eficiente la nueva versión de *UIS-Propeller*, hecho que se evidencia en el tiempo de respuesta del mismo.

Por último, resulta importante mencionar que para garantizar la confiabilidad de *UIS-Propeller*, conforme se avanzó en el proceso de reducción del código de programación, se fue ejecutando el software, para determinar los valores de las propiedades arrojados por cada una de las ecuaciones de diseño, lo anterior con el objeto de comparar dichos valores con los hallados durante el proceso de verificación de las ecuaciones para el diseño de hélices realizado con la ayuda de Mathcad (numeral 3.2.1).

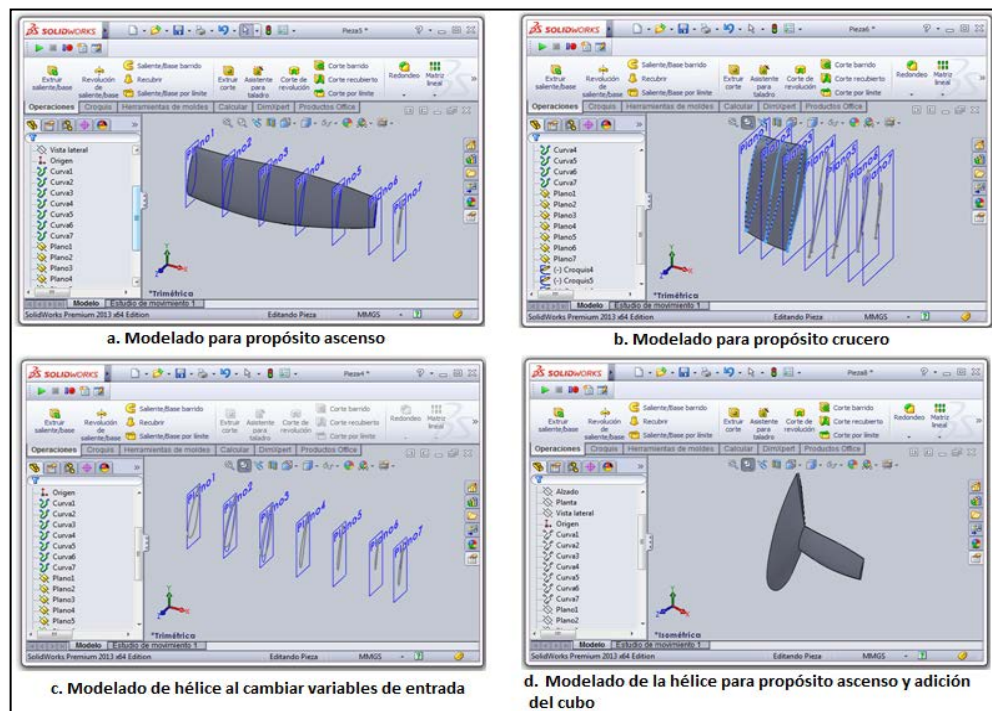
3.2.3 Desarrollo de la nueva estrategia para el modelado de la hélice para ultraliviano en SolidWorks. Como se indicó anteriormente, en la versión 1.0 de *UIS-Propeller* se generan múltiples interferencias geométricas durante la etapa de modelado de la hélice en SolidWorks, hecho por el cual es imperiosa la modificación de la estrategia llevada a cabo.

La primera tarea desarrollada, fue el repaso de las órdenes programadas mediante los API's de SolidWorks en *UIS-Propeller V1.0*, con la intención de conocer y entender el procedimiento empleado para el modelado de la hélice. De dicho estudio, se reconoció que en primera instancia mediante la instrucción *PART.InsertCurveFilePoint()*, se grafican los perfiles de la hélice

correspondientes a cada estación de la misma, sin embargo con el fin de utilizar estos perfiles para la operación *Recubrir* , es necesario convertir tales curvas (perfiles) en entidades de croquis a través de la operación *Convertir entidades* , y posteriormente realizar el recubrimiento de los croquis para obtener la hélice (sin el cubo).

Posterior a lo expuesto en el párrafo anterior, se siguió con la identificación de la causa de las interferencias geométricas, para lo cual se ejecutó el software bajo diferentes condiciones (propósito, sentido de giro, dimensiones, entre otros) y por consiguiente para diferentes dimensiones del cubo de la hélice. La figura 8, muestra cuatro intentos de modelado de la hélice bajo diferentes condiciones de diseño, ninguno de los cuales fue modelado de manera exitosa.

Figura 18. Modelado de hélice para ultraliviano, bajo diferentes condiciones de diseño (UIS-Propeller V1.0)



De estos intentos de modelar la hélice, se pudo advertir, que los procesos de modelado poseen un error común entre ellos: SolidWorks no siempre selecciona los perfiles indicados para ejecutar la operación *Convertir entidades*, hecho que resulta evidente en todos los casos presentes en la figura 8. En consecuencia de lo anterior, se realizó la revisión del proceso de selección de las curvas (perfiles).

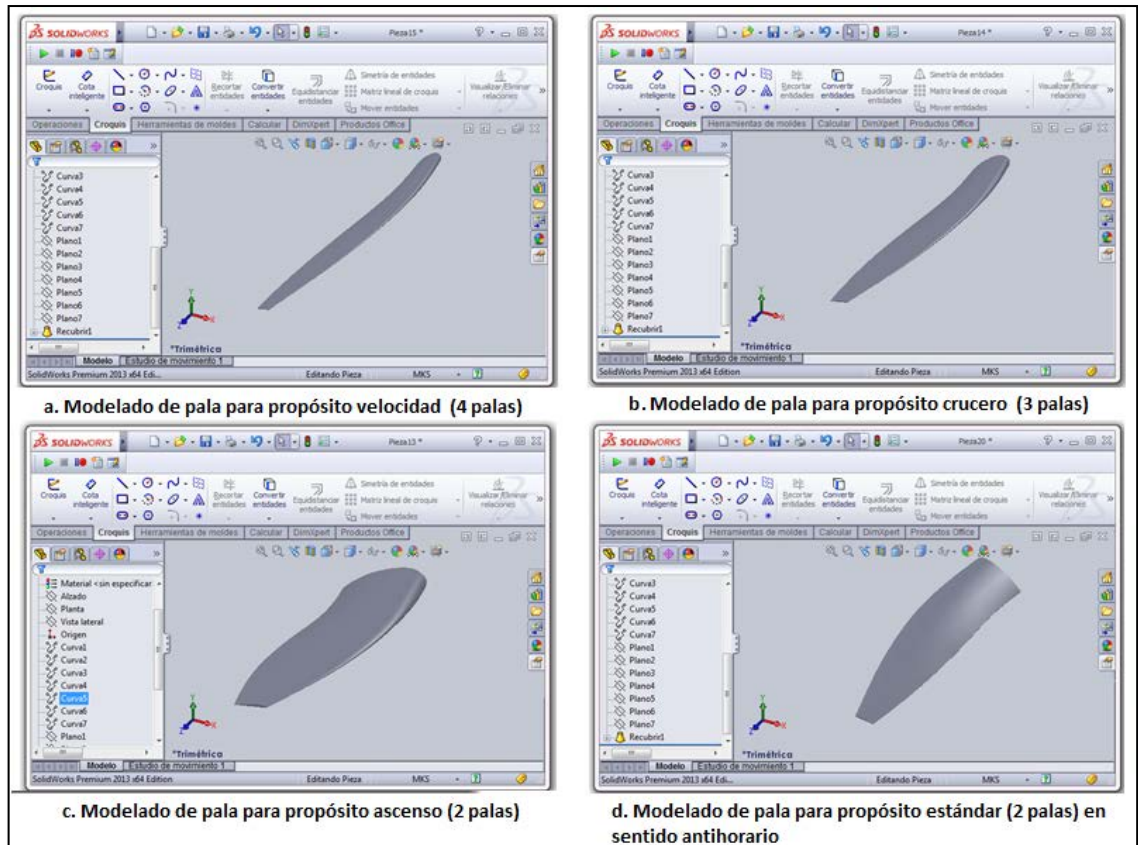
Es importante tener en cuenta, que previo a la conversión de las curvas en entidades de croquis, son creados *planos* en cada estación de la hélice (7 en total) paralelos a la vista lateral y una vez creada la totalidad de tales planos, las curvas de los perfiles son seleccionadas, para seguidamente ser convertidas en *entidades de croquis*. De ahí que el programa sea susceptible a no seleccionar los perfiles indicados para cada estación de la hélice. En consecuencia de lo anterior, se ideó un proceso de modelado que creará un *plano* en cada estación, e inmediatamente la respectiva curva fuera convertida en una *entidad de croquis*. Cabe aclarar que los perfiles generados por el proceso de diseño geométrico no son cerrados, razón por la cual, cada perfil ha de cerrarse una vez transformado en *entidad de croquis*. Para mayor entendimiento de lo explicado, ver el ANEXO D, donde se muestra el código (API's) implementado en *UIS-Propeller V2.0*, en el cual se hizo uso del ciclo *for*, el condicional *if* y de vectores.

Una vez cambiada la metodología de modelado de la pala en SolidWorks, se procede a ejecutar el modelado de la pala en *UIS-Propeller V2.0* bajo distintas condiciones de diseño. Las palas resultantes son mostradas en la figura 19, de las cuales se puede concluir que el modelado fue exitoso.


Luego de corregir los errores presentados en el modelado de la pala, se han de corregir las interferencias geométricas evidenciadas en el proceso de construcción del cubo de la hélice; siguiendo la misma secuencia de revisión del modelado

utilizada anteriormente. Dentro de este marco, el primer paso a seguir es el análisis de la estrategia empleada (ver ANEXO E).

Figura 19. Modelado de hélice para ultraliviano, bajo diferentes condiciones de diseño (*UIS-Propeller V2.0*)



Después de revisar la estrategia de modelado del cubo empleada en la versión 1.0 de *UIS-Propeller*, se dedujo que las múltiples interferencias geométricas y errores de modelado producidos, se deben al grado de complejidad de las relaciones geométricas empleadas con el fin de generar una hélice de estética armoniosa. En este sentido, se opta por desarrollar un procedimiento de modelado más simple, con las mínimas relaciones geométricas posibles. El procedimiento empleado es presentado a continuación:

NOTA: Antes de iniciar el proceso de construcción del cubo de la hélice, se sugiere activar la opción *Pausar/Grabar Macro* , tal como fue explicado en el numeral 2.3 del marco teórico, de esta forma el código VB empleado durante el modelado del cubo podrá ser guardado.


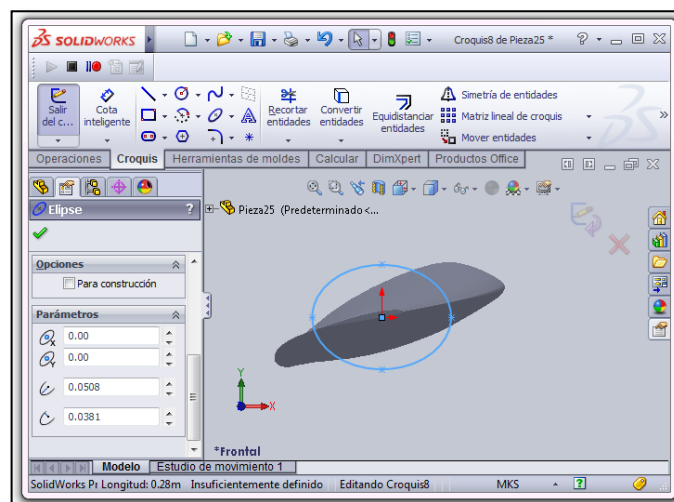
- I. Sobre el plano de *Planta*, dibujar una *Elipse*  (*croquis8*) cuyo radio en dirección *y* sea igual a $\frac{1}{2}$ del espesor del cubo y cuyo radio en dirección *x* sea igual radio del cubo (figura 20).

Figura 20. Primer paso construcción del cubo de la hélice para ultraliviano




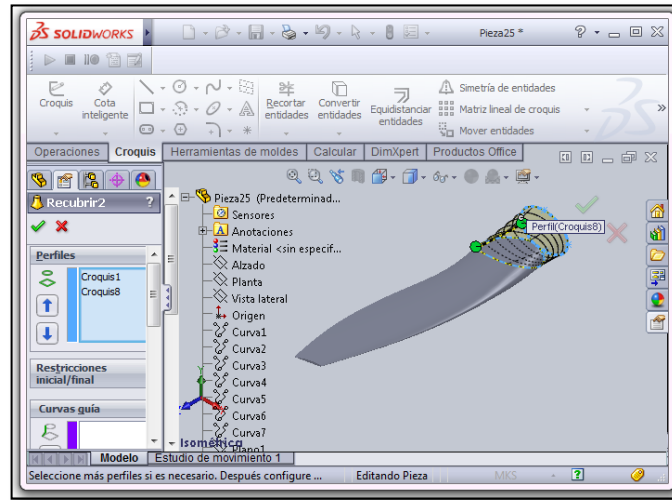
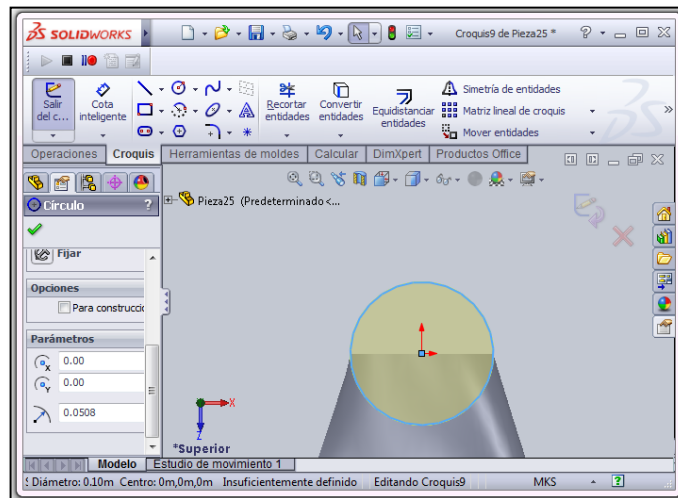
- II. Realizar una operación de *recubrimiento*  desde el *croquis1* hacia el *croquis8* (figura 21).

Figura 21. Segundo paso construcción del cubo de la hélice para ultraliviano



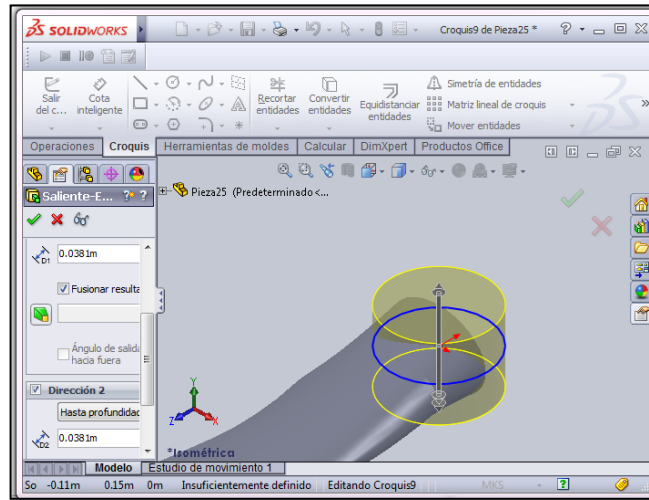
- III. Sobre el plano *Alzado*, dibujar un *Círculo* (croquis9) centrado en el *origen*, cuyo radio sea igual radio del cubo (figura 22).

Figura 22. Tercer paso construcción del cubo de la hélice para ultraliviano



- IV. Extruir *saliente/base* del *croquis9*, seleccionar ambas direcciones, con una profundidad igual a $\frac{1}{2}$ del espesor del cubo (figura 23).

Figura 23. Cuarto paso construcción del cubo de la hélice para ultraliviano





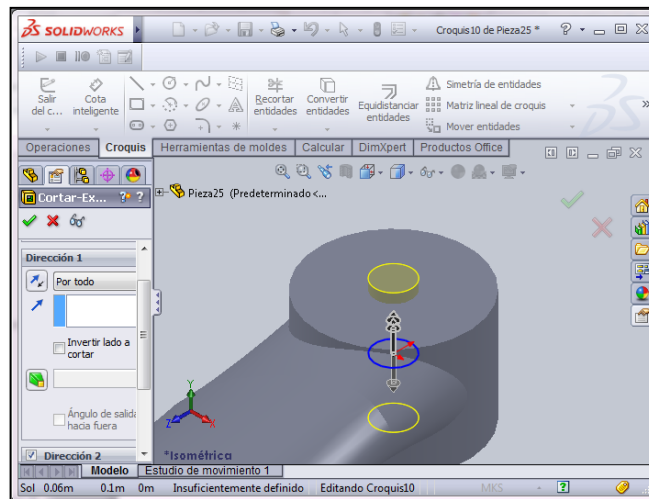
- V. Utilizar la operación *Extruir corte*  sobre la cara superior del cubo, para lo cual se dibuja un *Círculo*  de radio igual a $\frac{1}{2}$ del diámetro del orificio central del cubo. Seguidamente realizar el corte en extrusión y seleccionar la opción de profundidad *por todo* (figura 24).

Figura 24. Quinto paso construcción del cubo de la hélice para ultraliviano




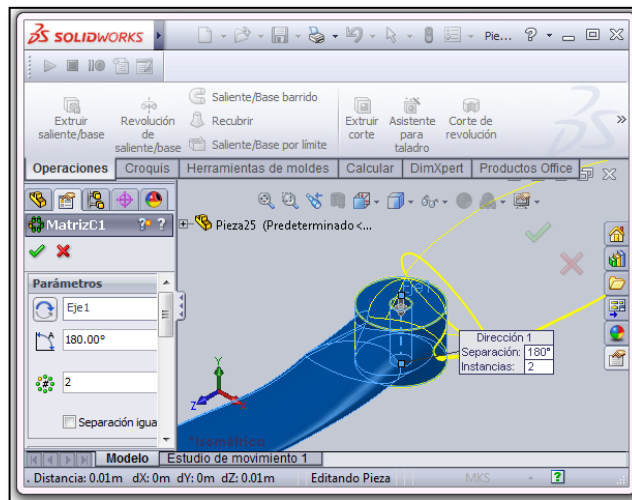
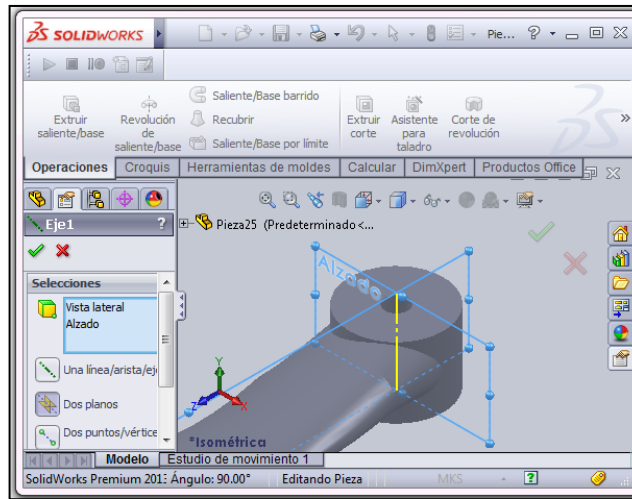

- VI. Se crea un Eje (Eje 1) entre los planos de Alzado y Vista Lateral, para luego crear una Matriz circular  de ángulo 180° (el ángulo varía en función del número de palas) y con número de instancias igual a 2 (el número de instancias equivale al número de palas) (figura 25).

Figura 25. Sexto paso construcción del cubo de la hélice para ultraliviano



- VII. Dibujar sobre la cara superior del cubo un *Círculo*  de radio igual a $\frac{1}{2}$ del diámetro de los pernos y cuyo centro esté ubicado en x igual a $\frac{1}{2}$ del diámetro


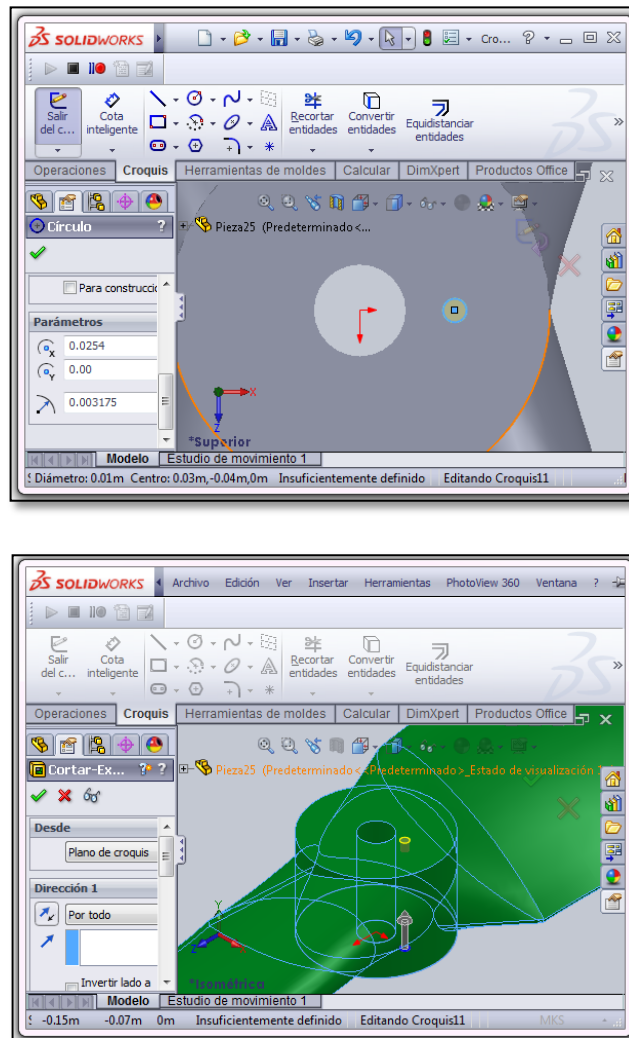
de ubicación de los pernos y en y igual a 0. Una vez creado el croquis, se procede a *Extruir un corte*  de dicho croquis (figura 26).

Figura 26. Séptimo paso construcción del cubo de la hélice para ultraliviano




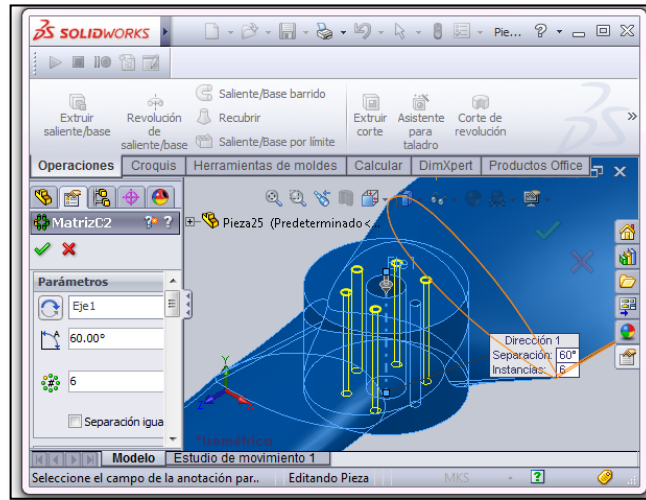
VIII. El último paso consiste es crear una *Matriz circular*  de la extrusión hecha en el paso anterior cuyo número de instancias es igual al número de pernos (figura 27).

Figura 27. Octavo paso construcción del cubo de la hélice para ultraliviano

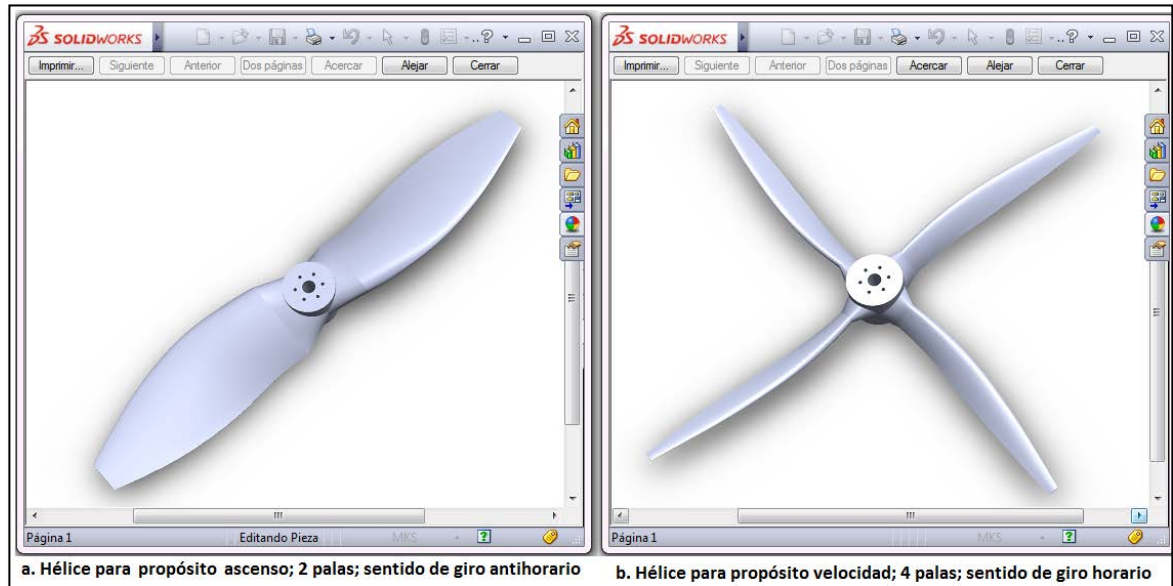


Al finalizar la construcción del cubo, la macro generada por SolidWorks ha de ser guardada (ver numeral 2.3) para su posterior utilización en la programación de los API's.

En lo referente a los API's de integración con SolidWorks, es importante subrayar que gracias a la nueva estrategia de modelado (construcción del cubo) desarrollada, la programación llevada a cabo, resultó ser sencilla y de fácil entendimiento. De ahí, que el programa sea poco (o nada) susceptible a generar interferencias geométricas²⁰. La figura 28, muestra el modelado de dos hélices (bajo diferentes condiciones de diseño) ejecutados por *UIS-Propeller V2.0*.

²⁰ Si se desea lograr una mayor comprensión a cerca de la programación de los API's desarrollada para éste propósito, se sugiere ver el ANEXO F.

Figura 28. Hélices modeladas en SolidWorks por *UIS-Propeller V2.0*

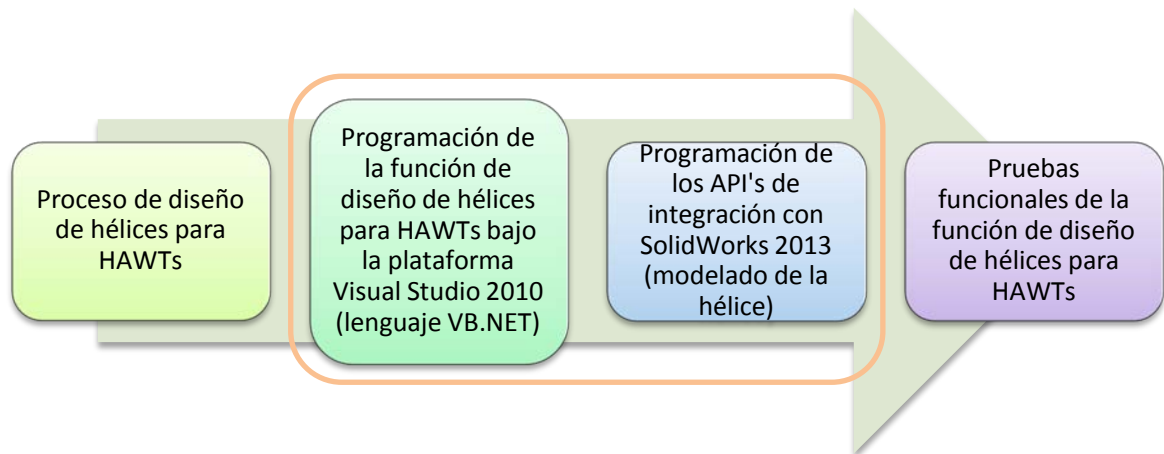


Tal como se evidencia en la figura 28, la nueva estrategia de modelado de hélices y construcción del cubo, mediante la aplicación de API's de integración con SolidWorks 2013, arroja resultados satisfactorios a los usuarios de *UIS-Propeller V2.0*; restando sólo la adición de unos pocos ajustes estéticos en el modelado de las uniones entre la hélice y el cubo de la misma, si así lo prefiere el usuario.

4. DESARROLLO DE LA NUEVA FUNCIÓN DE DISEÑO DE HéLICES PARA AEROGENERADORES DE EJE HORIZONTAL (HAWTs)

En este capítulo, se describirá el proceso llevado a cabo para desarrollar la nueva función de diseño de hélices para aerogeneradores de eje horizontal (HAWTs, del Inglés *Horizontal Axis Wind Turbines*). Dicho proceso fue realizado como se muestra en la figura 29.

Figura 29. Diagrama del proceso de desarrollo de la función de diseño de hélices para HAWTs



Como puede verse en la figura 29, el primer paso a desarrollarse, es el diseño detallado de las hélices para aerogeneradores, el cual, como fue mencionado en el numeral 2.2, está basado en la teoría aerodinámica de *Momento de Elementos de Pala* o BEM (del Inglés *Blade Elementum Theory*), seguido de la programación de tal proceso de diseño, para lo cual las ecuaciones serán transformadas en códigos de programación en lenguaje Visual Basic; por otro lado, las coordenadas geométricas obtenidas durante el diseño, serán convertidas en instrucciones de API's y de esta forma realizar el modelado de la hélice. Para finalizar, se llevarán a cabo pruebas

de funcionamiento a *UIS-Propeller V2.0*, con el propósito de verificar la efectividad del Software.

Cada uno de los pasos mencionados en el párrafo anterior, será descrito a continuación (numerales 4.2.1, 4.2.2, 4.2.3 respectivamente).

4.1 PROCESO DE DISEÑO DE HÉLICES PARA AEROGENERADORES

NOTA: Para lograr un mayor entendimiento de las variables utilizadas en el proceso de diseño a ser descrito, se sugiere revisar el ANEXO G²¹. Por otro lado, las referencias bibliográficas de donde cada una de las ecuaciones utilizadas para este propósito fue tomada, se encuentran tabuladas en el ANEXO J.

4.1.1 Identificación y determinación de los criterios básicos de diseño.

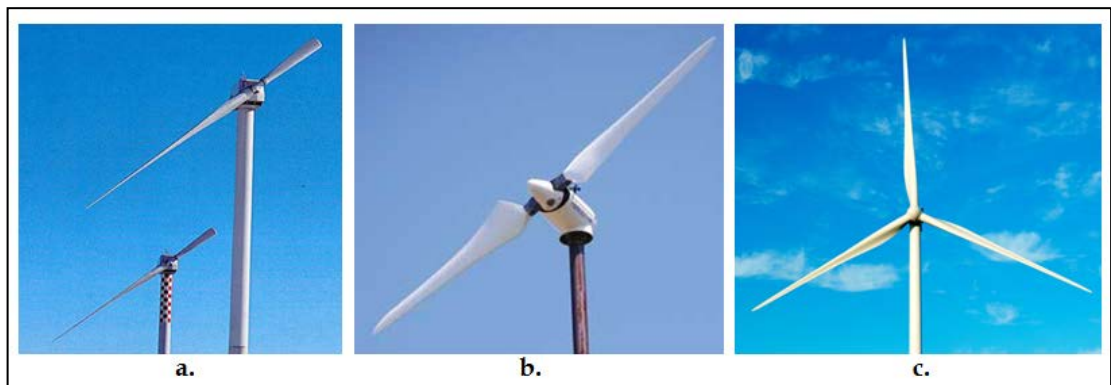
Tomando en consideración que el propósito de *UIS-Propeller* es diseñar la hélice para un aerogenerador, basándose en los requerimientos y/o preferencias del usuario, son criterios básicos de diseño: la potencia nominal deseada [Watts] (potencia generada por la turbina) y las condiciones ambientales de la zona donde va a ser instalada la turbina, tales como velocidad del viento [m/s; ft/s], densidad del aire [kg/m³; lbm/ft³] y temperatura [°C; °F].

Otro criterio igualmente importante, es el número de palas del aerogenerador, donde para turbinas eólicas de eje horizontal las configuraciones varían dependiendo del propósito de la turbina eólica. Para la generación de energía eléctrica, las configuraciones más comunes son aquellas de dos (Figura 30b) o tres

²¹ ANEXO G: Nomenclatura de variables empleadas por *UIS-Propeller* en el diseño de hélices para aerogeneradores de eje horizontal.

palas (Figura 30c), sin embargo en turbinas pequeñas (de uso doméstico; <50KW), es común encontrar configuraciones de 1 pala (Figura 30a). Las configuraciones de 1 y 2 palas resultan más económicas que las de 3 palas, debido al uso de menor material de fabricación, sin embargo, estas rotan a mayor velocidad (poseen mayor relación de velocidad de punta o *TSR*, del Inglés *Tip Speed Ratio*), por lo cual su visión resulta molesta para un observador, adicionando a esto una mayor generación de ruido. Por otro lado, las turbinas de 3 palas poseen una rotación de apariencia apacible (figura 31), perturbando el medio ambiente en menor medida, a esto se le suma el hecho de que turbinas eólicas con mayor número de palas capturan más energía del viento (son más eficiencia). Las configuraciones de más de 3 palas, son descartadas para la generación de energía eléctrica, ya que al aumentar el número de palas se debe disminuir el grosor de las palas (con el fin de mantener la solidez optima²² del rotor), surgiendo la necesidad de emplear materiales más costosos y métodos de fabricación más complejos.

Figura 30. Turbinas eólicas de dos y tres palas



Fuentes: (a) <http://www.powerhousewind.co.nz>
(b) http://spinoff.nasa.gov/Spinoff2009/er_3.html;
(c) <http://www.vestas.com/en/media/images.aspx>

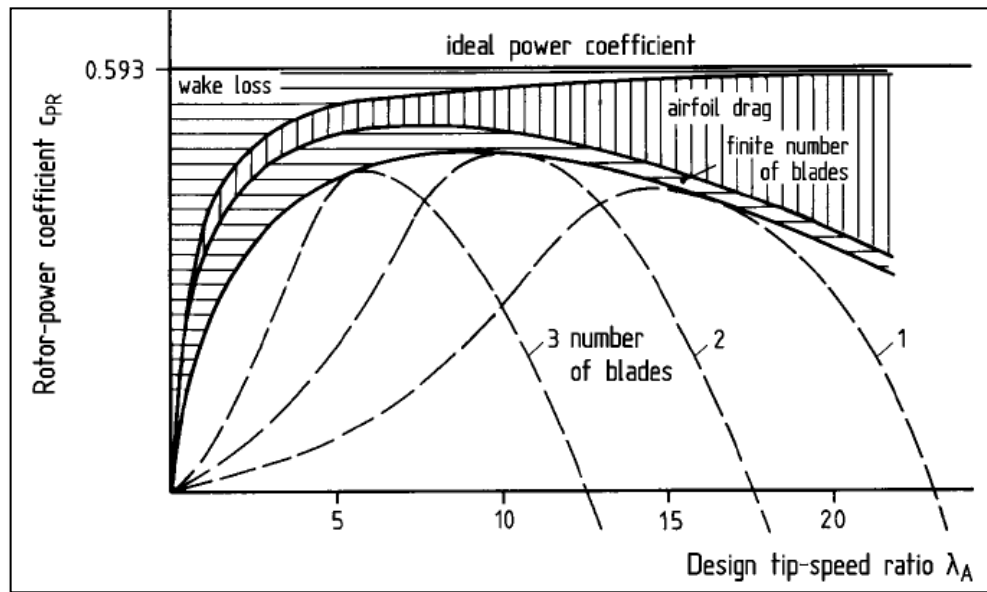
²² La solidez es definida como la fracción del área anular del volumen de control que es cubierta por las palas del rotor.

De acuerdo a lo mencionado en el párrafo anterior, para el diseño de las palas de la hélice, los usuarios de *UIS-Propeller* podrán elegir entre configuraciones de HAWTs de 1, 2 o 3 palas.

En este orden de ideas, la consulta de múltiples materiales bibliográficos referentes a turbinas eólicas, indicó que la *TSR* es dependiente estrictamente del número de palas del rotor. La *TSR*, es definida como la relación entre la velocidad en el extremo de la pala y la velocidad nominal del viento incidente; tal como se observa en la ecuación (3), siendo Ω la velocidad angular del rotor (en radianes por segundo), R el radio [m] y V_w la velocidad nominal del viento incidente [m/s].

$$\lambda = \Omega * \frac{R}{V_w} \quad (3)$$

Figura 31. Curvas de C_p vs *TSR* para rotores de 1, 2 y 3 palas



Fuente: HAU, E., 2006, pp. 98.

En la figura 31, se puede apreciar la curva de C_p (coeficiente de potencia) vs TSR ; (curva de potencia del rotor) para diferentes números de palas, entendiéndose por C_p a la cantidad de energía cinética contenida en el viento que es capturada por el aerogenerador, en otras palabras, el C_p es una medida de la eficiencia del rotor. Como puede verse en la figura 31, el coeficiente ideal de potencia tiene un valor de 0.593, este valor es normalmente conocido como *Límite de Betz*, cuyo significado físico indica que una turbina eólica (ideal) puede como máximo capturar el 59.3% de la energía cinética del viento incidente. Así mismo, de la ecuación (3) y la figura 31, puede deducirse que las dimensiones de la turbina eólica (Radio) son una función de la potencia nominal generada y viceversa.

Tabla 2. Rango de valores de TSR en función del número de palas

B	λ	
	λ_d	λ_{up}
1	8	17
2	4	15
3	4	8

Fuente: MANWELL, J.F. *et al*, 2002, pp. 125.

La velocidad angular del rotor es un criterio de vital importancia en el diseño de las palas del mismo, por lo tanto, el generador a ser acoplado debe poseer igual valor de velocidad angular; en caso contrario, se hace necesario el uso de una caja multiplicadora.

Existen dos formas de calcular la velocidad angular, la primera forma consiste calcularla en función del ángulo de giro de la hélice (β)²³ y la otra consiste en un proceso iterativo, en el cual la velocidad angular del rotor es calculada como una función de la velocidad angular del generador, de la relación de velocidad de punta λ (*TSR*) y del radio del rotor²⁴. Llegado este punto, es importante destacar que en la gran mayoría de las situaciones, la velocidad angular del rotor no coincide con la velocidad angular de los generadores disponibles en el mercado, por tanto el uso de una caja multiplicadora se hace necesario, surgiendo por ende otro inconveniente: la relación de transmisión a utilizar no siempre podrá ser cumplida. Tomando en consideración lo anterior, se decidió calcular la velocidad angular del rotor mediante el segundo método (proceso iterativo), para lo cual el usuario deberá suministrar el valor de la velocidad angular del generador que desee utilizar.

A continuación se explicará el proceso iterativo realizado, el cual fue desarrollado con la ayuda de la herramienta matemática Mathcad:

- I. El primer paso consistió en asignar un límite de iteración (inferior y superior) al *TSR* en función del número de palas (tabla 2; figura 32):

Figura 32. Establecimiento de los rangos de *TSR* en función del número de palas

$\lambda_d :=$	if $B = 1$	$\lambda_{up} :=$	if $B = 1$
	8		17
	if $B = 2$		if $B = 2$
	4		15
	if $B = 3$		if $B = 3$
	4		8

²³ HABALI, S. M. *et al*, 2000, pp.267.

²⁴ BELTRÁN L. g. *et al*, 2011

II. Dado que ya se cuenta con los valores de TSR aceptables para un rotor de B número de palas, se procede a calcular el valor de C_p que corresponde a dichas propiedades, tomando como primera suposición $\lambda = \lambda d$. Para realizar este cálculo, se hizo necesario convertir en ecuaciones las curvas de C_p vs λ (figura 31) para turbinas de una, dos y tres palas. Las regresiones matemáticas pertinentes fueron realizadas con la ayuda de Excel. A continuación se muestran las ecuaciones obtenidas:

$$C_{p_{B=1}} = 0.0002 * \lambda^4 - 0.0098 * \lambda^3 + 0.1855 * \lambda^2 - 1.4537 * \lambda + 4.1834 \quad (4)$$

$$C_{p_{B=2}} = -2E - 05 * \lambda^5 + 0.001 * \lambda^4 - 0.0202 * \lambda^3 + 0.1902 * \lambda^2 - 0.7529 * \lambda + 1.1724 \quad (5)$$

$$C_{p_{B=3}} = -0.0017 * \lambda^4 + 0.044 * \lambda^3 - 0.4368 * \lambda^2 + 1.952 * \lambda - 2.8464 \quad (6)$$

III. Con los datos obtenidos hasta el momento se cuenta con información suficiente para calcular el radio del rotor:

$$R = \sqrt{\frac{2 * Pe}{C_p * \eta * \pi * \rho * Vw^3}} \quad (7)$$

Donde,

Pe : Potencia nominal generada por el rotor [W]

C_p : Coeficiente de potencia

η : Eficiencia del rotor (se asume de 0.94)

ρ : Densidad del aire [Kg/m³]

Vw : Velocidad del viento [m/s]

IV. Ahora se aplica la ecuación (3) para calcular el nuevo valor de λ (como una función del R calculado), adicionando el valor de la relación de transmisión

(n) entre el eje del rotor y el generador. Para los cálculos iniciales, el valor de n se asume igual a 1.

$$\lambda_c = \frac{\Omega_{rad} * R}{V_w * n} \quad (8)$$

Donde,

Ω_{rad} : Velocidad angular del generador en rad/s

- V. Se calcula el valor de velocidad angular del rotor Ω_r , según el valor de la relación (n) de transmisión, donde para la primera iteración este es igual a la velocidad angular del generador Ω :

$$\Omega_r = \frac{\Omega}{n} \quad (9)$$

- VI. Finalmente, se procede a comprobar si el valor del λ cumple con el mínimo error de iteración admisible ($error \leq 0.005$), el cual fue definido como el valor absoluto de la diferencia entre el λ asumido y el λ calculado:

$$error = |\lambda - \lambda_c| \quad (10)$$

Si el valor del error cumple la condición establecida, se finaliza el proceso iterativo, de lo contrario éste continúa para diferentes valores de λ que se encuentren dentro del rango de valores aceptables (figura 32). Si una vez abarcados todo el rango de valores no se cumple la condición de error, se aumenta el valor de la relación de transmisión (n) y se realizan los cálculos nuevamente. Este proceso debe ser realizado tantas veces como sea necesario.

En la figura 33, se muestra el proceso iterativo realizado en Mathcad.

Figura 33. Cálculo iterativo de los criterios básicos de diseño en Mathcad

```

Calc:=
n ← 0
error ← 1
while error > 0.005
  n ← n + 1
  for λ ∈ λd, λd + 0.01 .. λup
    if B = 1
      Cp ← 0.0002 · λ4 - 0.0098 · λ3 + 0.1855 · λ2 - 1.4537 · λ + 4.1834
    if B = 2
      Cp ← -2 · 10-5 · λ5 + 0.001 · λ4 - 0.0202 · λ3 + 0.1902 · λ2 - 0.7529 · λ + 1.1724
    if B = 3
      Cp ← -0.0017 · λ4 + 0.044 · λ3 - 0.4368 · λ2 + 1.952 · λ - 2.8464
    Rr ← √(2 · Pe / (Cp · η · π · ρ · Vw3))
    λc ← (Ωrad · Rr) / (Vw · n)
    Ωr ← Ω / n
    error ← |λ - λc|
    if error ≤ 0.005
      λ ← λc
      break
    if λ > λup
      continue
  out ← [ Rr
          λ
          Cp
          n
          Ωr
          error ]

```

La tabla 3 resume los criterios básicos de diseño que fueron identificados y sus unidades (si corresponde).

Tabla 3. Criterios básicos para el diseño HAWTs

Criterio	Nomenclatura	Unidades
Potencia nominal del rotor	Pe	[W]
Velocidad del viento	Vw	[m/s]
Densidad del aire	ρ	[kg/m ³]
Temperatura ambiente	Ta	[°C]
Número de palas	B	[-]
Radio del rotor	Rr	[m]
Velocidad angular del rotor	Ωr	[rpm]
Relación de Velocidad de punta	λ	[-]

4.1.2 Definición de la forma de la pala. Durante este proceso, se calculará el número de Reynolds bajo el cual opera la turbina eólica deseada (basándose en los criterios básicos previamente establecidos), la selección del perfil aerodinámico (airfoil), y finalmente la geometría del rotor.

4.1.2.1 Cálculo del número de Reynolds. Este es un factor de gran importancia en el diseño de turbinas eólicas, dado que el rendimiento aerodinámico de estas depende en cierta medida del número de Reynolds (11) bajo el cual se esté operando. Sabiendo que,

$$Re = \frac{Vr * c}{\nu} \quad (11)$$

Donde,

Vr : Velocidad relativa del viento sobre la pala [m/s]

c : Cuerda de la hélice [m]

ν : Viscosidad cinemática del aire [m²/s]

Con propósitos prácticos, el Reynolds es calculado con aquellas propiedades que den como resultado su máximo valor, en este caso la raíz²⁵ de la pala.

La raíz, es la sección con mayor cuerda de la hélice, debido a que mayores esfuerzos serán soportados por esta sección. Esta, es normalmente ubicada al 25%²⁶ del Radio de la hélice (12) y su cuerda *root* (13) es calculada como sigue:

²⁵ La raíz de la pala es también conocida como root (por su traducción al inglés), y es la sección de la pala donde empieza la transición hacia la brida del cubo de la hélice (hub). Basándose en la teoría BEM, esta corresponde a la primera estación de la pala.

²⁶ 10% del radio del rotor corresponde al cubo, 15% corresponde a la raíz, por tanto midiendo desde el centro del cubo la raíz llega hasta el 25% del radio del rotor.

$$r_{root} = 0.25 * Rr \quad (12)$$

$$c_{root} = \frac{8 * \pi * r_{root} * (1 - \cos(\varphi_{root}))}{B * Cl_{dis}} \quad (13)$$

Donde,

r_{root} : Radio de la raíz [m]

Cl_{dis} : Coeficiente de elevación de diseño (se asume igual a 1)

φ_{root} : Ángulo aparente del viento sobre la raíz de la pala.

B : Número de palas del rotor

El ángulo aparente del viento en la raíz de la pala (14) es calculado a través de la siguiente expresión:

$$\varphi_{root} = \sin^{-1} \left(\frac{2 * Vw}{3 * Vr_{root}} \right) \quad (14)$$

Donde,

Vr_{root} : Velocidad relativa del viento en la raíz de la pala [m/s]

$$Vr_{root} = Vw + \Omega r * r_{root} \quad (15)$$

Con estos valores ya puede ser calculada la cuerda en la raíz de la pala.

Ahora, la viscosidad cinemática está definida de la siguiente forma:

$$v = \frac{\mu}{\rho} \quad (16)$$

Donde,

ν : Viscosidad cinemática [m²/s]

ρ : Densidad del aire [kg/m³]

μ : Viscosidad dinámica [Pa*s]

Continuando, se calcula la viscosidad dinámica que, según Sutherland es función de la temperatura ambiente y el tipo de fluido (en este caso el viento) (17):

$$\mu = \mu_0 \left[\frac{T_0 + C_s}{T_a + C_s} \right]^{\frac{3}{2}} \quad (17)$$

Con:

T_0 : Temperatura de referencia; $T_0 = 291.15$ [K]

μ_0 : Viscosidad de referencia a T_0 ; $\mu_0 = 18.27E-6$ [Pa*s]

C_s : Constante de Sutherland. Para aire estándar; $C_s = 120$ K.

T_a : Temperatura media del ambiente [K]. $T_a[K] = T_a[^\circ C] + 273.5$ (18)

Llegado este punto, es posible calcular el Re bajo el cual opera la turbina eólica que se está diseñando.

4.1.2.2 Selección del perfil aerodinámico (airfoil). Para este propósito, se tomaron como referencia los perfiles NACA comúnmente utilizados en aplicaciones eólicas: NACA2509, NACA4412, NACA4415, NACA63-215, NACA63(4)-221, NACA63-415, NACA63(4)-421, NACA64(2)-415, NACA64(4)-421, NACA65(2)-415 y NACA65(4)-421; perfiles entre los cuales el usuario podrá elegir.

Por otro lado, no todos los usuarios de UIS-Propeller tendrán certeza acerca del tipo de perfil a utilizar, por tanto, se decidió incluir en el software, una opción de selección automática, mediante la cual se ordena al programa seleccionar el perfil

de acuerdo a los criterios básicos de diseño (numeral 4.1.1). Es decir, el usuario podrá seleccionar el perfil de una lista o podrá ordenar al software seleccionar en su lugar.

Cabe aclarar que para el proceso de selección, sólo se tomaron en consideración los perfiles NACA mencionados en el primer párrafo de este numeral.

A este respecto es importante enfatizar, que dado que la elevación (Lift) es la causante del movimiento del rotor, esta es empleada en todo momento por el aerogenerador. Todo lo contrario que el arrastre (Drag), que representa la resistencia al movimiento del rotor (Figura 3). Por tanto, se desea un perfil que bajo las condiciones dadas, posea un gran valor de elevación y un mínimo de arrastre, i.e es decir los valores de L/D y Cl/Cd han de ser los máximos posibles. Así mismo el valor de α_0 (ángulo de cero elevación) resulta importante a la hora de decidirse por un perfil aerodinámico.

La selección se llevó a cabo mediante un proceso de puntuación²⁷. Para el cual se consideraron la relación entre el coeficiente de elevación máximo y el coeficiente de arrastre mínimo $\left(\frac{Cl_{max}}{Cd_{min}}\right)$, la relación entre elevación y arrastre $\left(\frac{L}{D}\right)$ en el punto de máximo Coeficiente de elevación (Cl_{max}) y el ángulo de cero levantamiento (o elevación) α_0 ; calculados al Re calculado previamente. Respecto a lo anterior es importante aclarar que el Re varía para cada estación de la pala ($Re_i = f(R_i)$), sin embargo, tal variación no es significativa de una estación a otra, razón por la cual todos los cálculos de propiedades se harán bajo un mismo número de Reynolds (Re en la raíz de la pala).

²⁷ El proceso de selección empleado en este proyecto está basado en el propuesto por Gómez-Rivera, W. *et al*, 2010, pp. 307.

En la Tabla 4 se muestra el porcentaje de puntuación asignado a cada uno de los parámetros:

Tabla 4. Porcentajes de puntuación para la selección del perfil

Parámetro	% Puntuación
$\frac{Cl_{max}}{Cd_{min}}$	55
$\left(\frac{L}{D}\right)_{Cl_{max}}$	35
α_0	10

Los valores de los parámetros mencionados previamente (exceptuando α_0), a diferentes números de Re, se obtuvieron a través del software de libre distribución *JavaFoil*²⁸. Este posee una base de datos de perfiles aerodinámicos, y en caso de no contar con el perfil deseado, permite al usuario agregar sus coordenadas geométricas. Así mismo una vez calculadas las propiedades aerodinámicas, es posible guardar los resultados como archivos de texto.

Con el propósito de obtener mayor exactitud en los resultados, se agregaron a *JavaFoil* las coordenadas geométricas de los perfiles a analizar (obtenidas de la base de datos del grupo de aerodinámica aplicada de la Universidad de Illinois²⁹), ya que estas cuentan con un mayor número de coordenadas geométricas (puntos).

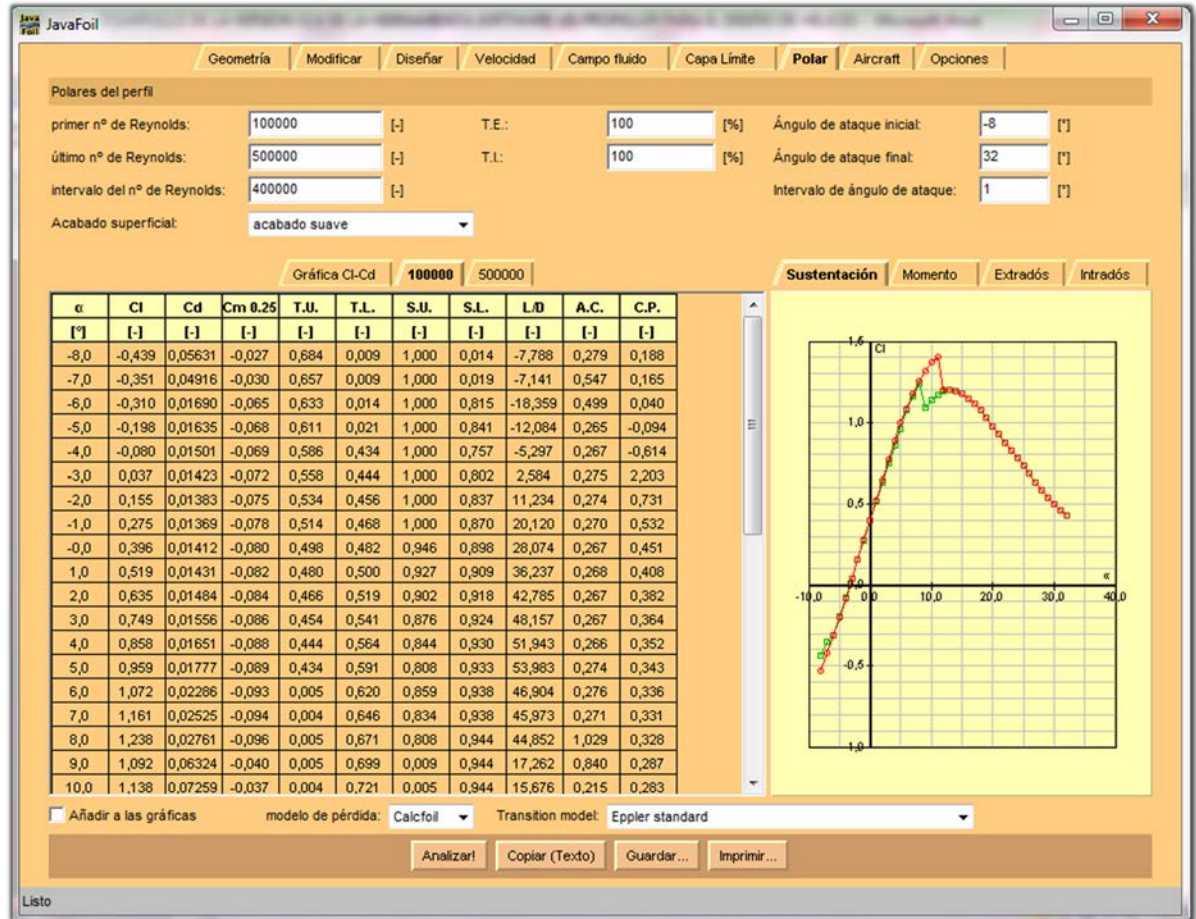
Los cálculos de propiedades fueron ejecutados para Re de 100000, 400000, 1000000 y 5000000, para un rango de ángulos de ataque $\alpha = [(-8) - 32]$.

²⁸ JavaFoil calcula las propiedades aerodinámicas de perfiles bajo diferentes condiciones previamente establecidas por el usuario.

²⁹ http://www.ae.illinois.edu/m-selig/ads/coord_database.html

La Figura, muestra los resultados obtenidos en *JavaFoil* para el perfil NACA64(2)-415 a números para Re de 100000.

Figura 34. Cálculo de propiedades aerodinámicas del perfil NACA 64(2)-415 en *Javafoil*



Fuente: <http://www.mh-aerotoools.de/airfoils/javafoil.htm>

Una vez calculadas las propiedades para los diferentes perfiles, se tabularon los valores de $\frac{Cl_{m\acute{a}x}}{Cd_{m\acute{i}n}}$ y $\left(\frac{L}{D}\right)_{Cl_{m\acute{a}x}}$ a tabla 5 muestra los valores de dichas propiedades para perfil NACA64(2)-415. (La tabla con la totalidad de perfiles puede ser consultada en el ANEXO H).

Tabla 5. Valores de $\frac{Cl_{max}}{Cd_{min}}$ y $\left(\frac{L}{D}\right)_{Cl_{max}}$ para NACA 64(2)-415

N° Reynolds	Perfil	$\frac{Cl_{max}}{Cd_{min}}$	$\left(\frac{L}{D}\right)_{Cl_{max}}$	$\left(\frac{L}{D}\right)_{max}$	Cl_{max}
100000	64(2)-415	61.23	12.74	39.99	1.72
400000	64(2)-415	110.43	36.36	53.70	1.40
1000000	64(2)-415	125.56	42.91	55.74	1.41
5000000	64(2)-415	126.66	59.88	72.90	1.43

Fuente: <http://www.mh-aerotoools.de/airfoils/javafoil.htm>

Respecto al ángulo de cero levantamientos α_0 , para los perfiles de la serie NACA de cuatro dígitos, dicho valor fue obtenido de la tabla III del reporte 460 de la NACA³⁰. Mientras que los valores α_0 para la serie NACA de 5 dígitos fueron calculados según la ecuación (19) obtenida del reporte 824 de la NACA³¹.

$$\alpha_0 = \alpha_i - \frac{57.3}{2 * \pi} * Cl_i \quad (19)$$

Donde,

α_0 : Ángulo de cero elevación

α_i : Ángulo de ataque de diseño

Cl_i : Coeficiente de Elevación de diseño

Los valores de α_i y Cl_i (Tabla 6) son también obtenidos del reporte 824 de la NACA³².

³⁰ JACOBS, E. *et al*, NACA Report 460, pp. 57.

³¹ ABBOTT, I. *et al*, NACA Report 824, 1945, pp. 14.

³² ABBOTT, I. *et al*, NACA Report 824, 1945, pp. 90-91.

Tabla 6. Valores de α_0 para perfiles NACA

Perfil NACA	Cl_i	α_i	α_0
2509	-	-	-3.8
4412	-	-	-3.9
4415	-	-	-2.0
63 Family	0.75	0	-6.84
64 Family	0.76	0.74	-6.19
65 Family	0.8	1.6	-5.59

Fuente: NACA, reportes 460 y 824

Luego, con los valores de, $\frac{Cl_{m\acute{a}x}}{Cd_{min}}$ y $\left(\frac{L}{D}\right)_{Cl_{max}}$ calculados, se procedió a obtener una expresión general para un rango de Re determinado, i.e. una ecuación para cada uno de los siguientes rangos: $Re \leq 400000$, $400000 < Re \leq 5000000$ y $Re > 5000000$. Para este fin, se realizaron interpolaciones de la forma (20):

$$f_1(x) = f(x_0) + \frac{f(x_1) - f(x_0)}{x_1 - x_0} * (x - x_0) \quad (20)$$

Aplicando (20) a las propiedades requeridas:

$$\left(\frac{Cl_{max}}{Cd_{min}}\right)_{Re_i} = \left(\frac{Cl_{max}}{Cd_{min}}\right)_{Re_{i-1}} + \frac{\left(\frac{Cl_{max}}{Cd_{min}}\right)_{Re_{i+1}} - \left(\frac{Cl_{max}}{Cd_{min}}\right)_{Re_{i-1}}}{Re_{i+1} - Re_{i-1}} * (Re_i - Re_{i-1}) \quad (20.a)$$

$$\left(\left(\frac{L}{D}\right)_{Cl_{max}}\right)_{Re_i} = \left(\left(\frac{L}{D}\right)_{Cl_{max}}\right)_{Re_{i-1}} + \frac{\left(\left(\frac{L}{D}\right)_{Cl_{max}}\right)_{Re_{i+1}} - \left(\left(\frac{L}{D}\right)_{Cl_{max}}\right)_{Re_{i-1}}}{Re_{i+1} - Re_{i-1}} * (Re_i - Re_{i-1}) \quad (20.b)$$

El ANEXO I, contiene las ecuaciones de $\frac{Cl_{m\acute{a}x}}{Cd_{min}}$ y $\left(\frac{L}{D}\right)_{Cl_{max}}$ correspondientes a $Re \leq 400000$ y $400000 < Re \leq 5000000$ para cada tipo de perfil.

Con las propiedades ya calculadas, se procede a realizar el proceso de puntuación para cada perfil de la forma que indica la ecuación (21):

$$puntuaci\acute{o}n_{perfil} = 0.55 * \left(\frac{Cl_{max}}{Cd_{min}}\right)_{perfil} + 0.35 * \left(\left(\frac{L}{D}\right)_{Cl_{max}}\right)_{perfil} + 0.1 * (\alpha_0)_{perfil} \quad (21)$$

La figura 35 muestra un extracto de la programación realizada en Mathcad para el proceso de puntuación del perfil aerodinámico.

Figura 35. Proceso de puntuación de perfiles realizado en Mathcad

Donde,

- **a2509**: Representa $\left(\frac{Cl_{max}}{Cd_{min}}\right)_{Rei}$ para el perfil NACA 2509
- **b2509**: representa $\left(\left(\frac{L}{D}\right)_{Cl_{max}}\right)_{Rei}$
- **αo2509**: el ángulo de cero Elevación
- pun2509**: la puntuación obtenida para dicho perfil.

Y así sucesivamente para los otros perfiles.

Calcsp:= if $Re \leq 400000$

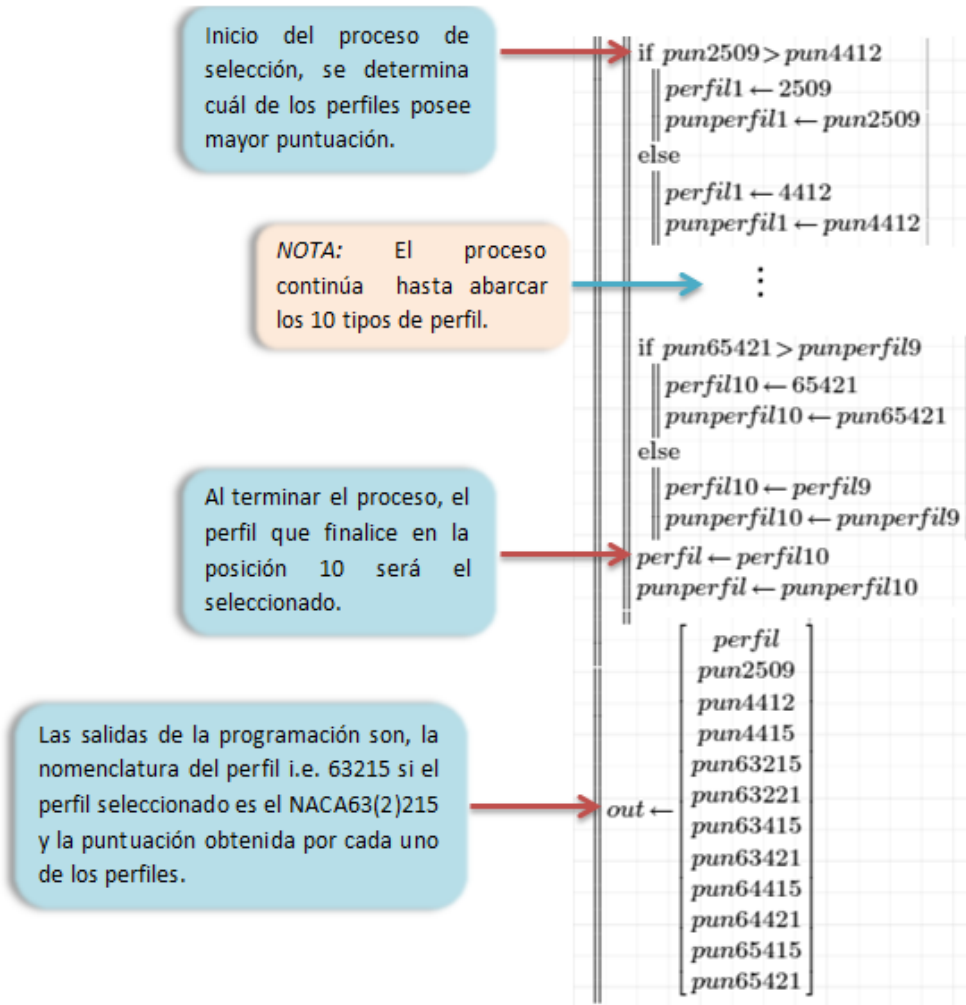
```

a2509 ← 83.11 + 5.74 · 10-5 · (Re - 100000)
b2509 ← 57.07 + 6.94 · 10-6 · (Re - 100000)
αo2509 ← -2
pun2509 ← 0.55 a2509 + 0.35 b2509 + 0.1 |αo2509|
a4412 ← 76.11 + 2.42 · 10-4 · (Re - 100000)
b4412 ← 41.87 + 3.12 · 10-5 · (Re - 100000)
αo4412 ← -3.9
pun4412 ← 0.55 a4412 + 0.35 b4412 + 0.1 |αo4412|

```

Finalmente, será seleccionado el perfil que presente una mayor puntuación bajo los criterios de diseño establecidos en la figura 36, se evidencia el proceso realizado en Mathcad:

Figura 36. Proceso de Selección del perfil en Mathcad



4.1.2.3 Aplicación de la teoría de Momento de Elementos de Pala (BEM). Tal como se ha mencionado en reiteradas ocasiones, el método utilizado en este proyecto para el diseño de hélices para HAWTs es el BEM. Este modelo es el resultado de la combinación de la teoría de momento axial y de la teoría de elementos de pala (ver numeral 2.2).

El modelo BEM fue desarrollado por Glauert en 1935, y en su momento, la teoría original asumió que el rotor estaba conformado por un número infinito de palas,

despreciando por tanto las pérdidas que ocurren en el extremo de la pala. Es debido a lo anterior, que a través de los años se le han modificado y adicionado ciertos parámetros, que permiten al modelo acercarse al funcionamiento real de la hélice. El modelo de corrección para un número finito de palas más utilizado es conocido como el factor de corrección de Prantl. Por otro lado, está comprobado que cuando el factor de inducción axial " a ", es aproximadamente superior a 0.3, el método BEM pierde su validez, por tanto otro factor de corrección es necesario (factor de corrección de Glauert).

Éste numeral trata el diseño de hélices para aerogeneradores basándose en el modelo BEM y tomando en consideración las pérdidas en el extremo de la pala.

El modelo de corrección de pérdidas utilizado para este proyecto es el desarrollado por el Departamento de Energía Eólica del laboratorio Risø de Dinamarca y el Departamento de Ingeniería Mecánica de la Universidad Técnica de Dinamarca en el año 2004³³. Este, integra los factores de corrección desarrollados por Prantl y Glauert, adicionando ciertas modificaciones que serán presentadas más adelante.

El modelo BEM, consiste en calcular para cada una de las estaciones de la pala, los coeficientes de inducción axial a y tangencial a' , el ángulo de ataque y el coeficiente de empuje. Para lo cual debe desarrollarse un proceso iterativo figura 37.

³³ SHEN, W. Z. *et al*, 2005, pp. 464-474.

Figura 37. Método BEM para evaluar a y a'

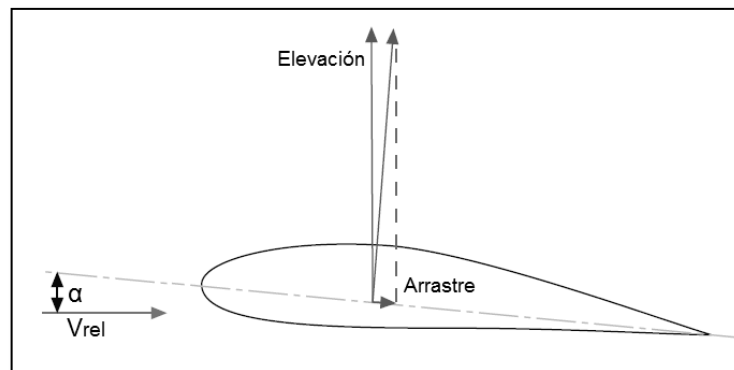
Step	Action required
1	Initialise a and a' with zero values
2	Evaluate the flow angle using eqn.
3	Evaluate the local angle of incidence, $a = \phi - \beta$
4	Determine C_L and C_D from tables (if available) or from formula
5	Calculate a and a'
6	Check on convergence of a and a' , if not sufficient go to step 2, else go to step 7
7	Calculate local forces on the element

Fuente: DIXON, S.L. 1998, pp. 346.

A continuación, se explicara el procedimiento llevado a cabo:

- I. **Determinación del ángulo de ataque α :** El ángulo de ataque es el ángulo comprendido entre la línea de cuerda y la velocidad relativa del viento sobre el perfil (figura 38).

Figura 38. Angulo de ataque y vectores de elevación y arrastre de un perfil aerodinámico



Una anotación importante respecto al α , es que a pesar de que la hélice incrementa su coeficiente de elevación (C_l) al incrementar su α , si el valor de

éste es demasiado grande, la hélice experimenta sacudidas (fenómeno conocido como stalls), decayendo con ello su Cl . Tomando en consideración lo anterior y el coeficiente de arrastre (Cd), se hace necesario calcular un ángulo de ataque al cual el valor Cl/Cd sea el máximo posible. Para este propósito, se analizaron los datos de Cl y Cd en función de Re y α obtenidos a través de JavaFoil, para cada uno de los perfiles (numeral 4.1.2.2), de lo cual se evidenció un comportamiento similar para en todos los perfiles: presentan sus mayores valores de Cl/Cd y L/D en valores de α comprendidos entre aproximadamente 4.5° y 6° , por tanto, el ángulo de ataque α fue establecido igual a 5° .

$$\alpha = \varphi - (\beta + \theta_p) \quad (22)$$

Donde,

φ : Ángulo de flujo

β : Ángulo de giro

φ_p : ángulo de paso (se asume =0)

Dado que, se busca mantener el valor de Cl/Cd a lo largo de la pala y que α está definido como muestra la ecuación (22), cada estación de la misma debe estar giradas a un ángulo β_i (23).

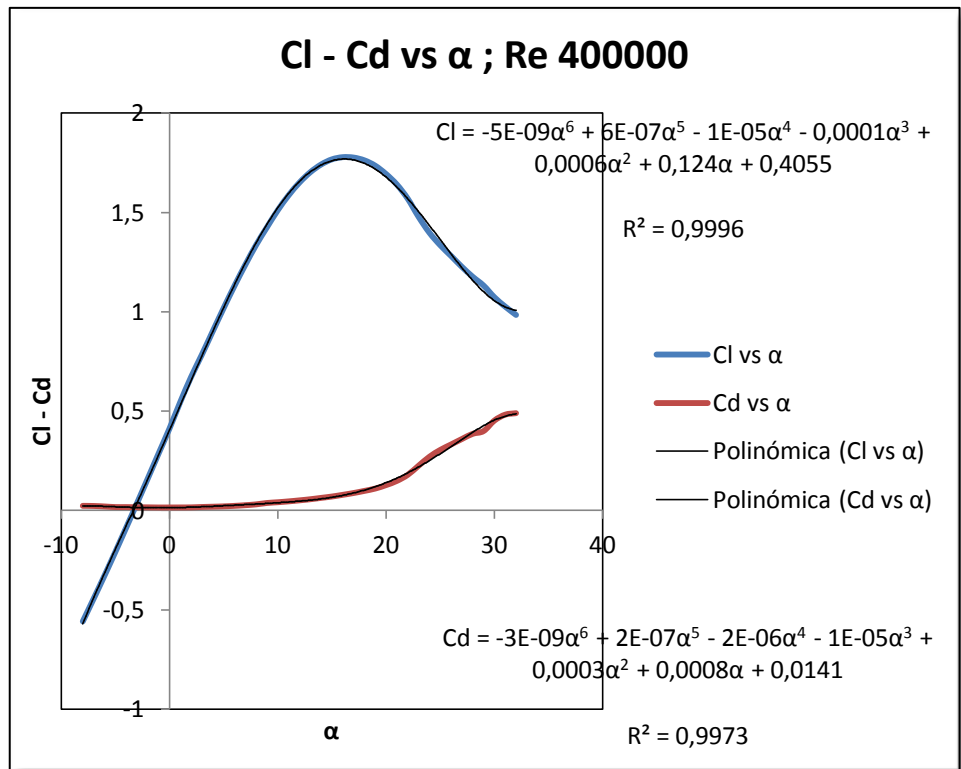
$$\beta = \varphi - \alpha \quad (23)$$

En este caso particular,

$$\beta = \varphi - 5 \quad (24)$$

II. **Cálculo de C_l y C_d .** A fin de que UIS-Propeller calculara los coeficientes de elevación y arrastre, correspondientes a cada tipo de perfil en función Re bajo el cual opere la turbina y del α , con los datos obtenidos del JavaFoil y la ayuda de Excel figura 39, se realizaron regresiones matemáticas. Como resultado se obtuvieron para cada perfil una serie de ecuaciones ($C_l=f(\alpha)$ y $C_d=f(\alpha)$), a los diferentes rangos de Re que fueron previamente establecidos.

Figura 39. Regresión polinómica de C_l vs α y C_d vs α ($Re=400000$) para el perfil NACA63(2)421



UIS-Propeller realizará interpolaciones entre las ecuaciones mencionadas, para el valor de Re de operación de la hélice tal como se hizo en el numeral 4.1.2.3 para el cálculo del L/D y el C_l/C_d .

III. Establecimiento del número de elementos de pala. El número de elementos de pala (nel) utilizados para este propósito fue de 20 (normalmente de 10 a 20). Por tanto el incremento (inc) de una estación a otra está dado por la ecuación (25):

$$inc = \frac{Rr - r_{root}}{nel} \quad (25)$$

Siendo,

Rr : El radio del rotor

r_{root} : Radio hasta el cual llega de la raíz de la pala (12) (Ver numeral 4.1.2.1)

IV. Inicio del proceso iterativo. A continuación, se mostrará paso por paso el proceso iterativo llevado a cabo para el cálculo de la geometría de la pala. Los valores del factor de pérdida en la punta de la pala y la corrección de Glauert serán considerados.

NOTA: El subíndice i hace referencia a la estación de la pala $0 \leq i \leq 19$ ó $1 \leq i \leq 20$.

a. El primer paso consiste en calcular el valor de TSR local λr_i (26)

$$\lambda r_i = \lambda \left(\frac{r_i}{Rr} \right) \quad (26)$$

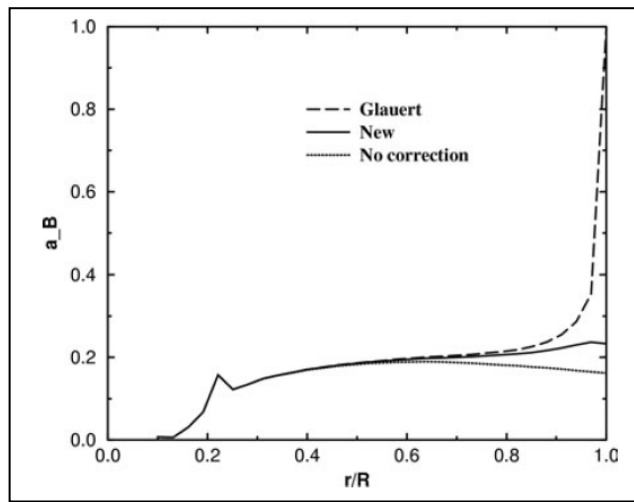
b. Luego, se asignan los valores de los coeficientes de inducción axial y tangencial para la primera iteración. En este caso: $a_i=0$ y $a'_i=0$

c. Se calcula el ángulo de flujo local φr_i , como una función de a y a' (27)

$$\varphi_i = \tan^{-1} \left(\frac{1 - a_i}{\lambda r_i * (1 + a'_i)} \right) \quad (27)$$

- d. Llegado a este punto, se procede a hallar los factores de corrección aplicados a BEM. Como se mencionó anteriormente, el modelo de pérdidas aplicado es el desarrollado por W. Z Shen *et al.* Este modelo, arroja resultados de a y a' que se acercan más al comportamiento real de los aerogeneradores (ver figura 40). A continuación se presentan las ecuaciones correspondientes:

Figura 40. Curvas de a vs r/R , correspondientes a la aplicación de la corrección de Glauert, el modelo de pérdidas de W. Z Shen *et al* y sin corrección alguna para pérdidas en el extremo de la pala.



W. Z Shen *et al*, 2004

- Coeficiente de corrección de Prantl:

$$F_i = \frac{2}{\pi} * \cos^{-1} \left(e^{-\frac{B}{2} * \left(\frac{Rr - r_i}{r_i * \sin(\varphi_i)} \right)} \right) \quad (28)$$

- ▶ Coeficiente de corrección por W. Z Shen *et al*

$$F1_i = \frac{2}{\pi} * \cos^{-1} \left(e^{-g_i * \frac{B}{2} * \left(\frac{Rr-r_i}{r_i * \sin(\varphi_i)} \right)} \right) \quad (29)$$

Con:

$$g = e^{-0.125(B * \lambda r_i - 21)} + 0.1 \quad (30)$$

- e. A continuación, se calculan los coeficientes de fuerza tangencial (Ctn) y normal (Cn), los cuales son función del Cl y Cd .

$$Ctn_i = Cl * \sin(\varphi_i) - Cd * \cos(\varphi_i) \quad (31)$$

$$Cn_i = Cl * \cos(\varphi_i) + Cd * \sin(\varphi_i) \quad (32)$$

- f. Se procede a calcular la cuerda de la hélice para la estación i :

$$cr_i = \frac{8 * \pi * r_i * (1 - \cos(\varphi_i))}{B * Cl} \quad (33)$$

- g. Se calcula la solidez la cual está dada por la siguiente ecuación:

$$\sigma_i = \frac{B * cr_i}{2 * \pi * r_i} \quad (34)$$

- h. Llegado este punto, se calculan los nuevos valores de a y a' (ac y $a'c$). Cuyas ecuaciones formulan de acuerdo al nuevo modelo de perdidas empleado (d) de la siguiente manera:

$$ac_i = \frac{2 + Y1_i - \sqrt{4 * Y1_i * (1 - F_i) + Y1_i^2}}{2 * (1 + F_i * Y1_i)} \quad (35)$$

$$a'c_i = \frac{1}{\frac{(1 - a_i * F_i) * Y2}{1 - a_i} - 1} \quad (36)$$

Donde,

$$Y1 = 4 * F * \frac{\sin(\varphi_i)^2}{\sigma r_i * Cn_i * F1_i} \quad (37)$$

$$Y2 = \frac{4 * F_i * \sin(\varphi_i) * \cos(\varphi_i)}{\sigma r_i * Ctn_i * F1_i} \quad (38)$$

- i. Cuando el factor de interferencia axial a , es mayor a aproximadamente 1/3, el modelo BEM, deja de ser aplicable, por tanto, el coeficiente de empuje es calculado como de la siguiente manera³⁴:

$$si \ a \leq \frac{1}{3} \quad C_{Ti} = 4 * ac_i * F_i * (1 - ac_i * F_i) \quad (39. a)$$

$$si \ a > \frac{1}{3} \quad C_{Ti} = 4 * [acr^2 * F_i^2 + (1 - 2 * acr * F_i) * ac_i * F_i] \quad (39. b)$$

Donde,

acr = factor de inducción axial crítico, $acr= 1/3$

³⁴ W. Z Shen *et al*, 2004

- j. Se calcula la diferencia entre el factor de inducción axial y tangencial calculado, y los valores supuestos:

$$difa_i = |a_i - ac_i| \quad (40)$$

$$difa'_i = |a'_i - a'_c_i| \quad (41)$$

Si el valor de (40) o (41) es superior a 1×10^{-7} , se deben realizar los cálculos nuevamente. Este proceso iterativo debe realizarse para cada estación de la hélice.

Una vez finalizado el proceso iterativo, se calcula el valor de coeficiente de potencia local Cp_i (42).

$$Cp_i = \frac{8 * \Delta\lambda}{\lambda^2} \left(F_i * \sin(\varphi_i)^2 (\cos(\varphi_i) - \lambda r_i * \sin(\varphi_i)) * (\sin(\varphi_i) + \lambda r_i * \cos(\varphi_i)) * \left(1 - \left(\frac{Cd}{Cl} \right) * \cot(\varphi_i) \right) * \lambda r_i^2 \right) \quad (42)$$

Siendo,

$$\Delta\lambda = \frac{\lambda - \lambda_{root}}{nel} \quad (43)$$

El ángulo de giro es calculado como fue indicado previamente en (23):

$$\beta r_i = \varphi_i - \alpha$$

Ahora bien, con el tipo de perfil, las distribuciones de cuerda cr y de ángulo de giro β calculados, se cuenta con la información necesaria para realizar el

modelado de la hélice en un software CAD. En el numeral 4.2.2, se explicará el procedimiento llevado a cabo para tal fin.

- V. **Cálculo del Cp.** Para finalizar, una vez realizado el proceso iterativo para cada una de las estaciones de la pala, el Cp total es calculado como una sumatoria de los Cp de cada estación:

$$Cp_{total} = \sum_{i=1}^{20} Cp_i \quad (44)$$

4.2 PROGRAMACIÓN DE LA FUNCIÓN DE DISEÑO DE HÉLICES PARA AEROGENERADORES

Este numeral tratará de manera general el proceso de programación realizado bajo la plataforma Visual Studio 2010, en lenguaje de programación VB.NET, de la nueva función de diseño de hélices para aerogeneradores.

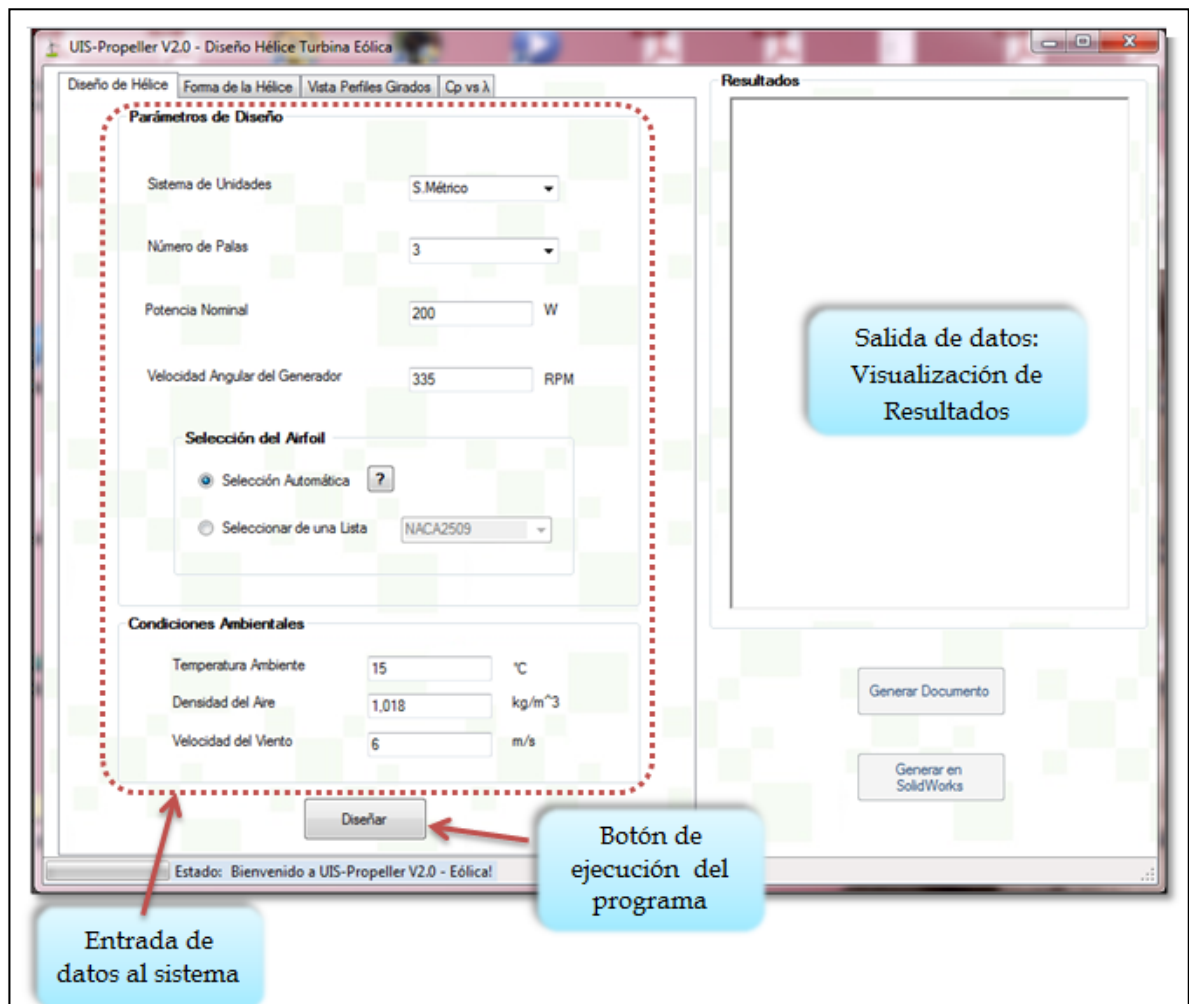
En primera instancia, se dará a conocer a groso modo la interfaz visual dispuesta para esta función, y los elementos que la componen.

Luego, basándose en el proceso de diseño de hélices para HAWTs explicado en el numeral anterior y en el programa desarrollado en Mathcad para dicho propósito, se explicarán algunas de las líneas de código generadas. De igual forma, se explicará el proceso de obtención de coordenadas geométricas (puntos) de cada perfil.

Para finalizar, se dará una breve explicación de la programación de los API's de integración con SolidWorks realizada para el modelado de la hélice del aerogenerador en el software CAD.

4.2.1 Interfaz Visual con el usuario. Previo a la programación del proceso de diseño en lenguaje VB, es necesaria la creación de una interfaz visual en la que se contemplen las variables de entrada al sistema, así mismo como las salidas de datos que se esperan de él.

Figura 41. Ventana de inicio de la función de diseño de hélices para HAWTs



La interfaz visual con el usuario, ha de ser intuitiva (de fácil comprensión y uso) a fin de evitar confusiones a la hora de proporcionar los requerimientos del diseño y/o interpretar los resultados producto del mismo. Con tal propósito, esta fue creada a través de las *Applications Windows Forms* de Visual Studio que, como se mencionó en el numeral 3.2.2, con la utilización de sus *Windows Forms*, facilita la creación de una interfaz organizada y agradable.

En la figura 41, se puede ver la disposición final de la ventana de inicio para la nueva función de diseño de *UIS-Propeller*:

4.2.2 Programación en lenguaje VB.NET del proceso de diseño de hélices para HAWTs. El proceso de programación de la nueva función de *UIS-Propeller*, es desarrollado de manera similar a aquel para hélices propulsadas (numeral 3.2.2). En vista de esto, éste numeral describirá dicho proceso de manera general. A continuación se enumerarán los pasos seguidos:

- I. Declaración de variables.
- II. Obtención de valores suministrados.
- III. Creación del código de programación.

4.2.2.1 Declaración de variables. El primer paso consiste en declarar todas las variables empleadas en el diseño como públicas; discriminando entre escalares, vectores y matrices, de este modo se podrá hacer uso de ellas en todo el programa.

4.2.2.2 Obtención de valores suministrados. Una vez declaradas las variables, se ha de ordenar al software, obtener los valores y/o especificaciones suministradas por el usuario, discriminándolas con los comandos “*CStr*” y “*Cdbl*” según sea el caso (caracteres alfabéticos o numéricos respectivamente).

4.2.2.3 Creación del código de programación. Ahora se procede a programar el diseño de la hélice, utilizando para ello los ciclos *for* y *while*, y el condicional *if*.

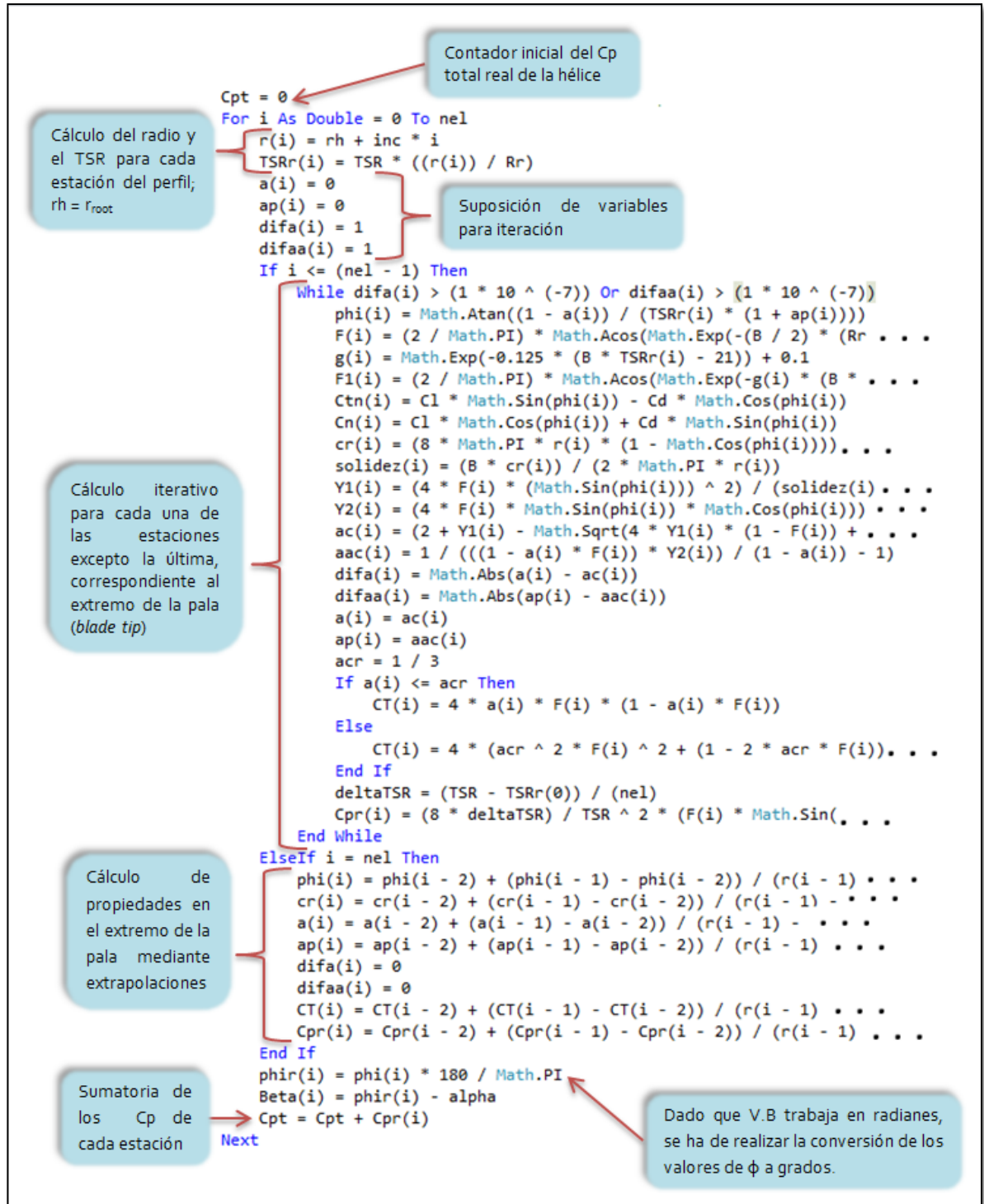
A continuación, a manera de ejemplo se explicará el código de programación utilizado en el proceso iterativo para la estimación de la forma de la pala (ver numeral 4.1.2.3 - IV):

Antes de iniciar con el proceso iterativo, se debe especificar la dimensión de los vectores (variables) que serán utilizados, donde la dimensión de cada uno de estos equivale al número de elementos en el que será dividida la pala para su diseño, así:

```
ReDim r(nel)
```

Una vez redimensionados los vectores, el proceso quedaría como muestra la figura 42:

Figura 42. Código VB del proceso iterativo para la estimación de la forma de la pala de una HAWT



Adviértase que en la figura 42, las propiedades en el extremo de la pala son calculadas mediante extrapolaciones en función del radio de la pala. Esto, debido a que el factor de corrección $F1$ introducido por W.Z. Shen *et al.*, al igual que el factor F de Prantl, es igual a cero en la punta de la pala; por tanto el proceso iterativo realizado para las otras estaciones de la pala no es válido en el extremo de la misma. Cabe aclarar que al dividirse la pala en 20 estaciones (distancia entre estaciones puede considerarse como mínima), los resultados obtenidos mediante la extrapolación son confiables.

Al llegar aquí, las coordenadas geométricas obtenidas de la base de datos de la Universidad de Illinois, fueron convertidas en vectores de tal manera que pudieran utilizarse en el ciclo *for*, necesario para la obtención de los puntos que conforman cada una de las estaciones del perfil en forma matricial i.e. $x(nel, nc)$, siendo *nel*, el número de estaciones de la pala y *nc* igual al número de puntos que conforman el perfil. El código empleado puede observarse en la figura 43:

Figura 43. Código VB.NET para la obtención de las coordenadas geométricas de la pala de una HAWT

```

If perfil = 2509 Or perfil = 4412 Or perfil = 4415 Then
    ccx = 0.3
ElseIf perfil = 654421 Then
    ccx = 0.4
Else
    ccx = 0.35
End If

For i As Double = 0 To nel
    For j As Double = 0 To nc
        x(i, j) = (cx(j) - ccx) * cr(i)
        y(i, j) = cy(j) * cr(i)
    Next
Next

' A continuación se muestra el proceso para girar los perfiles en función del Ángulo de Calado β (Beta)
' Conversión de Beta a radianes
Betarad(i) = Beta(i) * Math.PI / 180

For j As Double = 0 To nc
    If perfil = 2509 Or perfil = 4412 Or perfil = 4415 Then
        If x(i, j) = 0 Then
            xg(i, j) = -Math.Sin(Math.Atan(x(i, j) / y(i, j)) - Betarad(i)) * y(i, j)
        Else
            xg(i, j) = Math.Cos(Math.Atan(y(i, j) / x(i, j)) - Betarad(i)) / (Math.Cos(Math.Atan(y(i, j) / x(i, j)))) * x(i, j)
        End If
    Else
        xg(i, j) = Math.Cos(Math.Atan(y(i, j) / x(i, j)) - Betarad(i)) / (Math.Cos(Math.Atan(y(i, j) / x(i, j)))) * x(i, j)
    End If


    yg(i, j) = Math.Tan(Math.Atan(y(i, j) / x(i, j)) - Betarad(i)) * xg(i, j)
Next

```

Ubicación en x del centro de giro del perfil en función de su tipo

Cálculo de las coordenadas geométricas en forma matricial

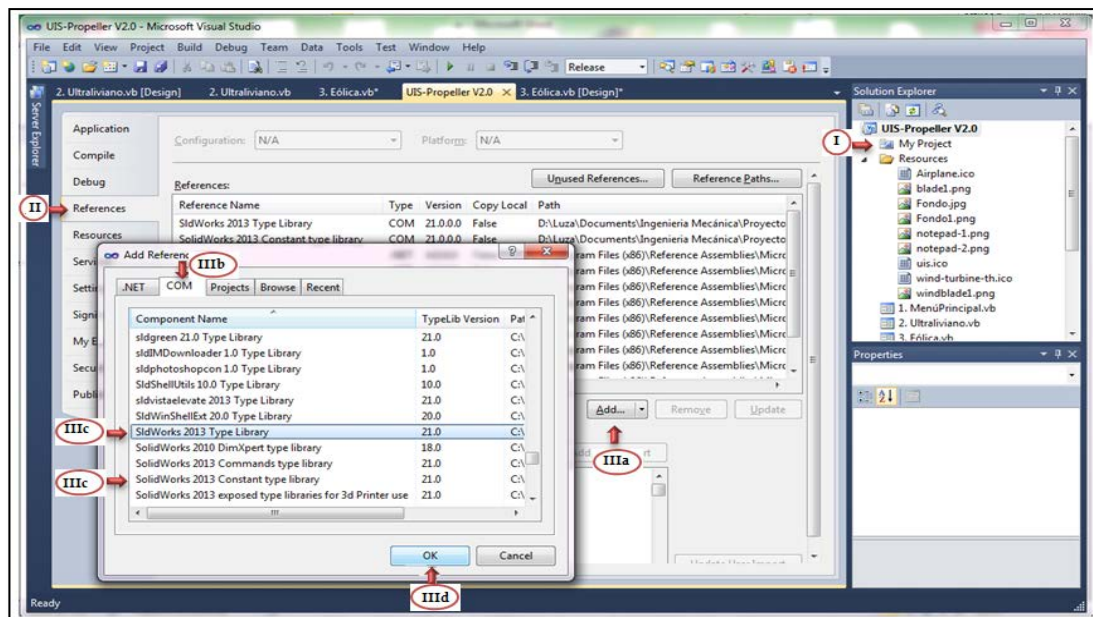
Cabe concluir que obtenidas las coordenadas geométricas de la pala, es posible realizar su modelado en SolidWorks (4.2.3).

4.2.3 Programación de los API's de Integración con SolidWorks. La programación de los API's para este propósito, fue desarrollada de igual modo que aquél para hélices de ultralivianos. Por tanto, el primer paso seguido, fue la obtención del código de programación al grabar el modelado en SolidWorks con la herramienta Pausar/Grabar Macro .

Posteriormente, para la utilización del código obtenido, han de incluirse las librerías de SolidWorks al programa de diseño, tal como sigue:

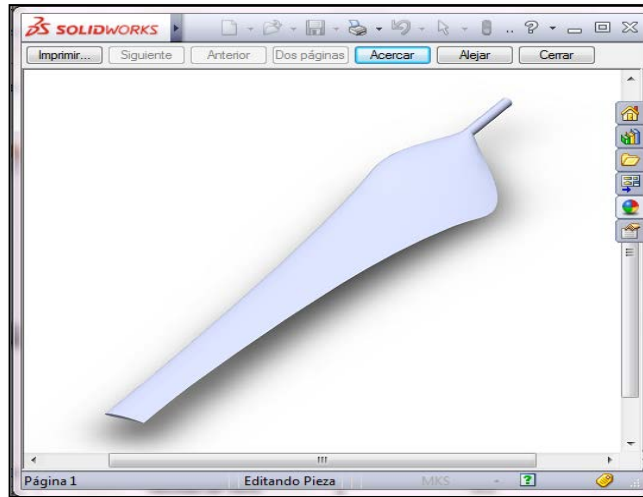
- I. En el *Solution Explorer*, dar doble clic en la opción *My Project*, de tal forma que toda la información concerniente al programa en desarrollo sea mostrada,
- II. Una vez allí, doble clic sobre la pestaña *References*,
- III. Luego, clic sobre *Add*, allí son mostrados todos los tipos de referencias que pueden ser incluidas al proyecto, en este caso el tipo de interés es COM, dando clic sobre la pestaña correspondiente, vemos una serie de componentes; seleccionamos “*SolidWorks2013 Type Library*”, y cliqueamos OK. Luego repetimos este paso para “*SolidWorks2013 Constant type library*”. Con esto han sido agregadas las librerías necesarias.

Figura 44. Adición de librerías de SolidWorks a *UIS-Propeller*



Finalmente se procede a adicionar el código a *UIS-Propeller*. La figura 45 muestra la hélice obtenida por el software.

Figura 45. Pala de una HAWT modelada por *UIS-Propeller* a través de los API's de Integración con SolidWorks 2013



4.3 PRUEBAS FUNCIONALES DE LA FUNCIÓN DE DISEÑO DE HÉLICES PARA HAWTs

Una vez terminado el proceso de programación, la nueva función de *UIS-Propeller* fue sometida a diferentes pruebas, con el propósito de garantizar el correcto funcionamiento de la misma, bajo diferentes condiciones.

Las pruebas se dividieron en dos tipos:

- ▶ Pruebas de diseño: Cuyo propósito fue el de garantizar que los resultados obtenidos durante el proceso de diseño de las mismas fuera correcto.
- ▶ Pruebas de modelado: Donde se verificó que el software realizara el modelado de la hélice previamente diseñada sin ningún inconveniente bajo cualquier circunstancia.

Estos, serán descritos a continuación:

4.3.1 Pruebas de diseño. Para garantizar la confiabilidad de los resultados obtenidos por el software, se llevó a cabo en el software el diseño de la hélice propuesta por Beltrán, L. G *et al*³⁵, 2011 en su proyecto de grado.

Las variables de entrada al software utilizadas según lo propuesto fueron:

Tabla 7. Variables de entrada proceso de prueba

Sistema de Unidades	Métrico
Potencia nominal	5000 [W]
Numero de palas	3
Velocidad angular del generador	217 [rpm]
Velocidad del viento	6 [m/s]
Temperatura ambiente	15 [°C]
Densidad del aire	1.018 [kg/m ³]
Perfil aerodinámico	E-387 ³⁶

Fuente: Beltrán, L. G *et al*, 2011

Como se puede observar el tipo de perfil utilizado por Beltrán, L. G *et al*, no coincide con aquellos empleados en *UIS-Propeller*. Se sabe que $(Cl/Cd)_{max}$ del E-387 es igual a 105.87 para un Re de 500000.

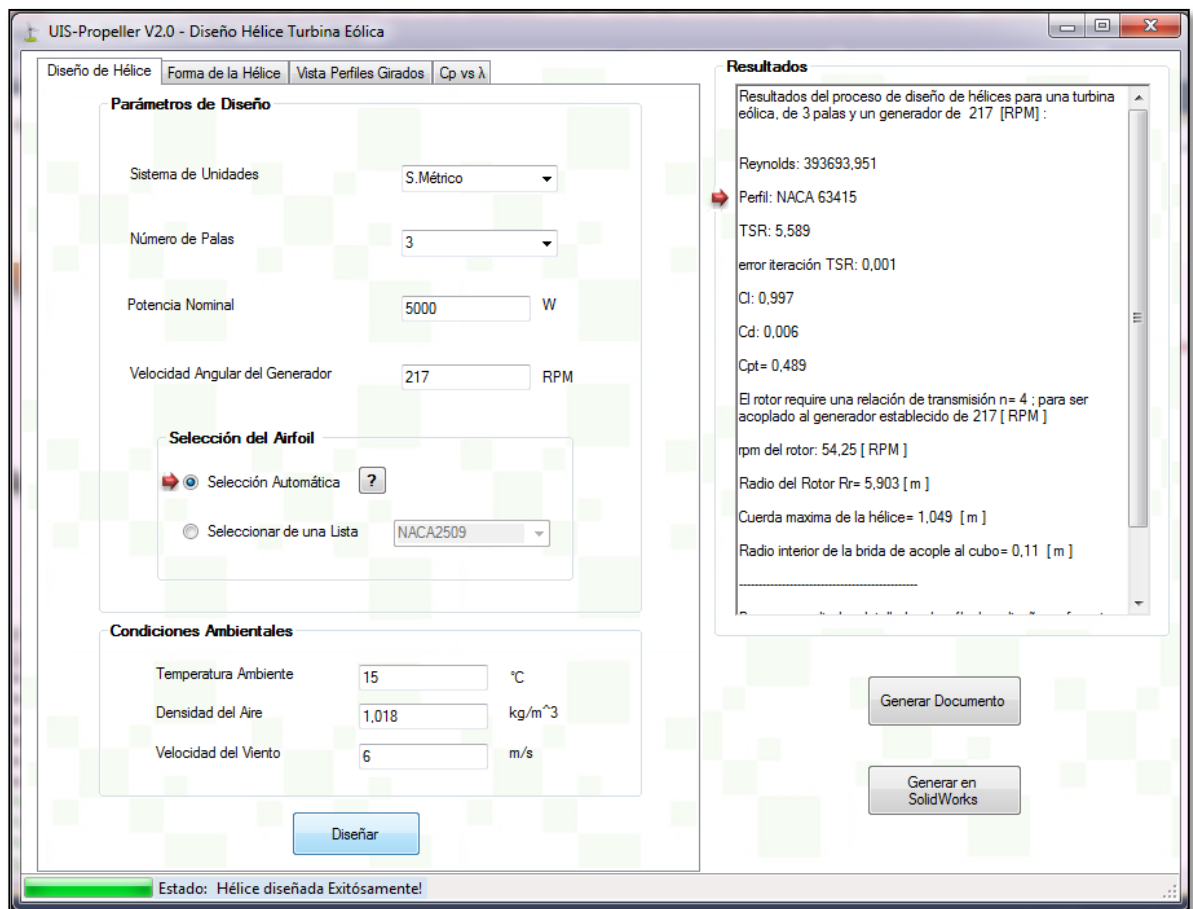
³⁵ BELTRÁN DUARTE, Luis Gabriel y FIGUEROA MADERO, Juan Carlos. Diseño de un sistema eólico de baja potencia para la zona aledaña al municipio de Gameza – Boyacá. Bucaramanga, Colombia: Universidad Industrial de Santander, 2011, pp. 77-122.

³⁶ El E-387, fue desarrollado a principios de los años 60 por el Alemán Richard Eppler, para aplicaciones en Sailplanes. Es aún, un perfil aerodinámico comúnmente utilizado en aplicaciones con bajo Re (a bajas velocidades).

Por lo anterior, con la ayuda de JavaFoil se seleccionó un perfil aerodinámico entre los presentados por nuestro software, cuyas propiedades fueran aproximadamente iguales bajo esta condición de Re . Obteniendo que para las condiciones establecidas, el perfil NACA 63(2)415, era el que mostraba mayor similitud $(C_l/C_d)_{\max} = 108.3$.

Luego, fue ejecutado UIS-Propeller marcando la opción de selección automática de perfil y se encontró que la selección del software era coincidente con la realizada de forma manual i.e. Perfil = NACA 63(2)415 (figura 46).

Figura 46. Resultados de la ejecución de *UIS-Propeller* en la etapa de pruebas



En la tabla 8, se puede ver una comparación entre los resultados obtenidos por Beltrán, L. G *et al* en su proyecto y los obtenidos mediante la ejecución de *UIS-Propeller*, para el proceso inicial de diseño

Tabla 8. Resultados iniciales de diseño durante la etapa de pruebas

Parámetro	Beltrán, L. G <i>et al</i>	UIS-Propeller
Reynolds	500000.000	393693.951
Perfil	E-387	NACA 63(2)415
TSR (λ)	5.280	5.589
Velocidad angular rotor [RPM]	54.160	54.250
Relación de transmisión (n)	4	4
Radio del rotor [m]	5.582	5.903
Error de iteración1	0.062	0.001

Fuentes: Beltrán L. G *et al*, 2011 y *UIS-Propeller* V2.0, 2013

En el proceso inicial (el cual es independiente del tipo de perfil empleado) de diseño se pueden observar algunas discrepancias entre los valores debidas mayormente a que *UIS-Propeller* realiza el proceso de iteración un mayor número de veces, al haberse establecido su error mínimo aceptable igual a 0.005.

Luego se analizaron los resultados producto del diseño geométrico de la hélice (tabla 9). Para éste propósito se tomaron los resultados obtenidos en los que Beltrán L. G *et al* denominan primera iteración, ya que los resultados dados por estos como definitivos corresponden a los obtenidos luego de haber recortado la cuerda de la pala con fines de facilidad de fabricación. Así mismo, para poder realizar la comparación de valores, se redujo el número de elementos de pala a 8 (normalmente *UIS-Propeller* trabaja con 20).

Tabla 9. Resultados del diseño geométrico etapa de pruebas

nel	Radio Medio (m)		a		a'		Velocidad Relativa W (m/s)		ϕ (Grados)		β o calado (Grados)		Cuerda (m)	
	Beltrán. L.G et al	UIS-Propeller V2.0	Beltrán. L.G et al	UIS-Propeller V2.0	Beltrán. L.G et al	UIS-Propeller V2.0	Beltrán. L.G et al	UIS-Propeller V2.0	Beltrán. L.G et al	UIS-Propeller V2.0	Beltrán. L.G et al	UIS-Propeller V2.0	Beltrán. L.G et al	UIS-Propeller V2.0
1	1.395	1.476	0.331	0.324	0.111	0.100	9.67	8.38	24.53	23.74	18.29	18.74	0.97	1.049
2	1.954	2.108	0.331	0.328	0.059	0.052	12.41	11.98	18.86	17.74	12.62	12.74	0.795	0.84
3	2.512	2.741	0.33	0.33	0.036	0.031	15.31	15.57	15.22	14.05	8.98	9.05	0.658	0.689
4	3.07	3.373	0.332	0.332	0.024	0.021	18.29	19.16	12.66	11.59	6.43	6.59	0.558	0.578
5	3.628	4.006	0.332	0.333	0.018	0.015	21.33	22.76	10.83	9.83	4.6	4.83	0.479	0.494
6	4.186	4.638	0.331	0.337	0.013	0.011	24.4	26.35	9.47	8.5	3.24	3.45	0.41	0.428
7	4.745	5.271	0.332	0.347	0.01	0.009	27.49	29.94	8.38	7.4	2.14	2.4	0.338	0.368
8	5.303	5.903	0.331	0.357	0.008	0.006	30.6	33.54	7.54	6.29	1.29	1.29	0.214	0.309

Fuentes: Beltrán L. G *et al*, 2011 y UIS-Propeller V2.0, 2013

De los resultados presentes en la tabla 9, puede decirse que las diferencias entre los valores obtenidos por Beltrán L. G *et al* y los obtenidos por UIS-Propeller, son mayormente consecuencia de las diferencias entre las propiedades de los perfiles utilizados en cada uno de los casos (E-387 y NACA 63(2)415), sumando a esto los números de Re para los cuales fueron calculados Cl y Cd y las dimensiones del radio. Sin embargo, a pesar de lo anteriormente mencionado, las propiedades conservan cierta similitud entre sí, manteniendo la hélice dentro del mismo rango de dimensiones. Adicional a esto, el valor total de captación eólica (Cp) en ambos casos es el mismo

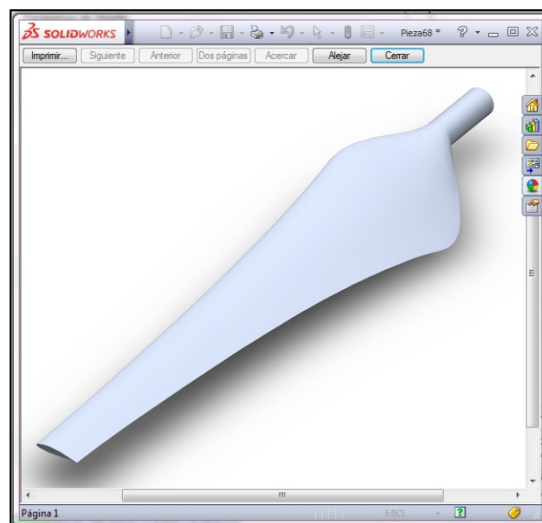
Tabla 10. Resultados de C_p para cada estación de la hélice en etapa de pruebas

nel	Radio Medio (m)		C_p	
	Beltrán. L.G et al	UIS-Propeller V2.0	Beltrán. L.G et al	UIS-Propeller V2.0
1	1.5	1.476	0.0025	0.025
2	2.1	2.108	0.0137	0.039
3	2.7	2.741	0.0260	0.053
4	3.3	3.373	0.037	0.066
5	3.9	4.006	0.0495	0.078
6	4.5	4.638	0.0606	0.088
7	5.1	5.271	0.0711	0.087
8	5.6	5.903	0.0799	0.086
Total			0.4893	0.522

Fuentes: Beltrán L. G *et al*, 2011 y *UIS-Propeller V2.0*, 2013

Luego de las comparaciones realizadas, puede concluirse que la nueva función de diseño de hélices para aerogeneradores de *UIS-Propeller V2.0*, realiza el diseño de la misma de manera satisfactoria. La figura 47 muestra el modelado de la hélice diseñada en SolidWorks 2013.

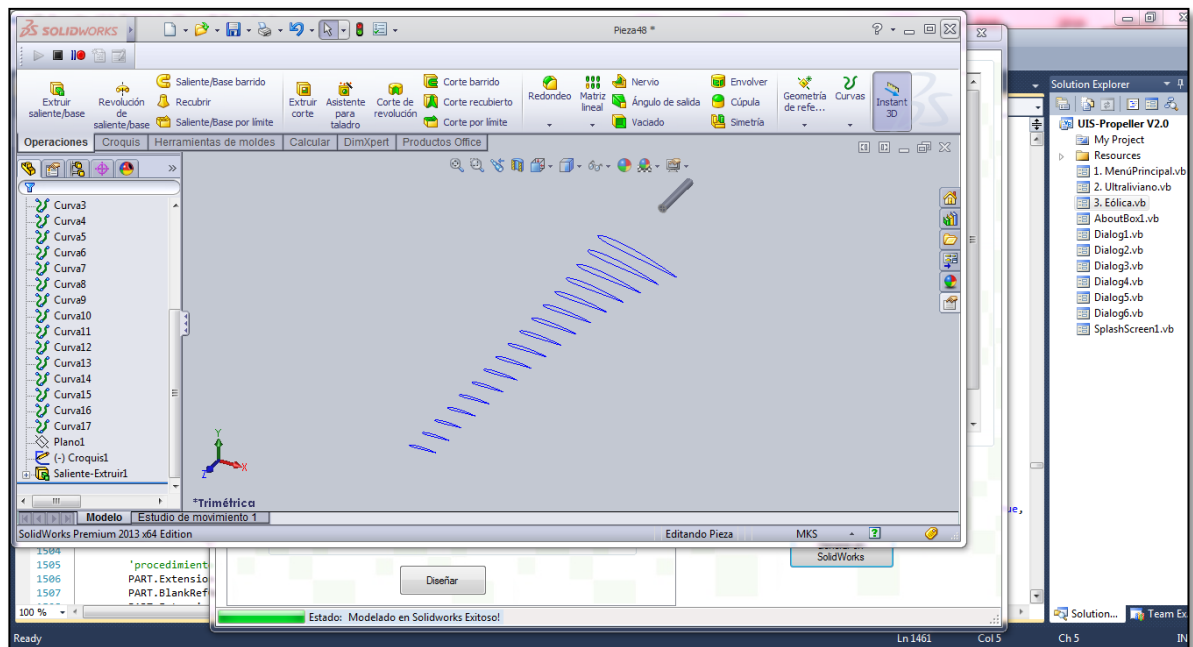
Figura 47. Modelado de la hélice diseñada durante el proceso de pruebas



4.3.2 Pruebas de modelado. Tal como se mencionó al inicio de esta sección de pruebas, el propósito de la misma es el de garantizar al usuario que podrá realizar el modelado de cualquier hélice que haya diseñado con *UIS-Propeller*. Para dicho propósito se ejecutó el programa bajo diferentes especificaciones (variables de entrada).

Luego de múltiples pruebas, se encontró que el programa modelaba exitosamente la hélices siempre y cuando SolidWorks 2013 abriera maximizado, pero, si por el contrario el programa abría minimizado, al ordenar a *UIS-Propeller* realizar el modelado de la hélice se presentaban errores de construcción (para mayor entendimiento ver figura 48).

Figura 48. Error de modelado de la hélice cuando la ventana de SolidWorks abre minimizada



En la figura 48, puede apreciarse que los errores surgen a la hora de realizar el recubrimiento de los perfiles, por tanto el procedimiento llevado a cabo para dicho fin fue analizado. A continuación se muestra la forma original del procedimiento (figura 49):

Figura 49. Proceso de recubrimiento de perfiles con errores de modelado, en caso que SolidWorks abra minimizado

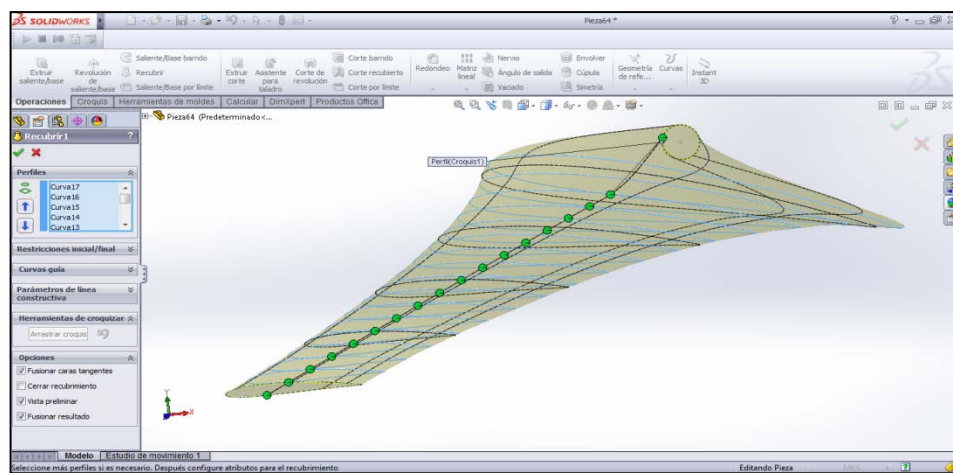
```

PART.Extension.SelectByID2("Curva20", "REFERENCECURVES", 0, 0, 0, True, 0, Nothing, 0)
PART.Extension.SelectByID2("Curva1", "REFERENCECURVES", 0, 0, 0, True, 0, Nothing, 0)
PART.Extension.SelectByID2("Croquis1", "SKETCH", 0, 0, 0, True, 0, Nothing, 0)
PART.ClearSelection2(True)
PART.Extension.SelectByID2("Curva20", "REFERENCECURVES", 0, 0, 0, False, 1, Nothing, 0)
PART.Extension.SelectByID2("Curva19", "REFERENCECURVES", 0, 0, 0, True, 1, Nothing, 0)
PART.Extension.SelectByID2("Curva1", "REFERENCECURVES", 0, 0, 0, True, 1, Nothing, 0)
PART.Extension.SelectByID2("Croquis1", "SKETCH", 0, 0, 0, True, 1, Nothing, 0)
PART.FeatureManager.InsertProtrusionBlend(False, True, False, 1, 6, 6, 1, 1, True, True, ...

```

Dado que el código aparentaba ser correcto, se ejecutó nuevamente el software teniendo SolidWorks maximizado, luego dando clic en editar recubrimiento, se pudieron observar los puntos del perfil que eran seleccionados como guía de la operación (figura 50).

Figura 50. Operación de recubrimiento en ventana maximizada



Puede observarse que los puntos seleccionados por SolidWorks como guía de recubrimiento no son los idóneos. En la figura 49, también puede ser apreciado un cambio en la dirección de la línea guía entre el punto que se encuentra sobre la curva 1 y el punto correspondiente al círculo donde inicia el eje de la brida de acople.

Por lo anterior, surgió la necesidad de editar dicho procedimiento, especificando de manera más detallada cuáles eran los puntos que se deseaban establecer como guía. El nuevo procedimiento es mostrado en la figura 51:

Figura 51. Proceso de recubrimiento de perfiles modificado

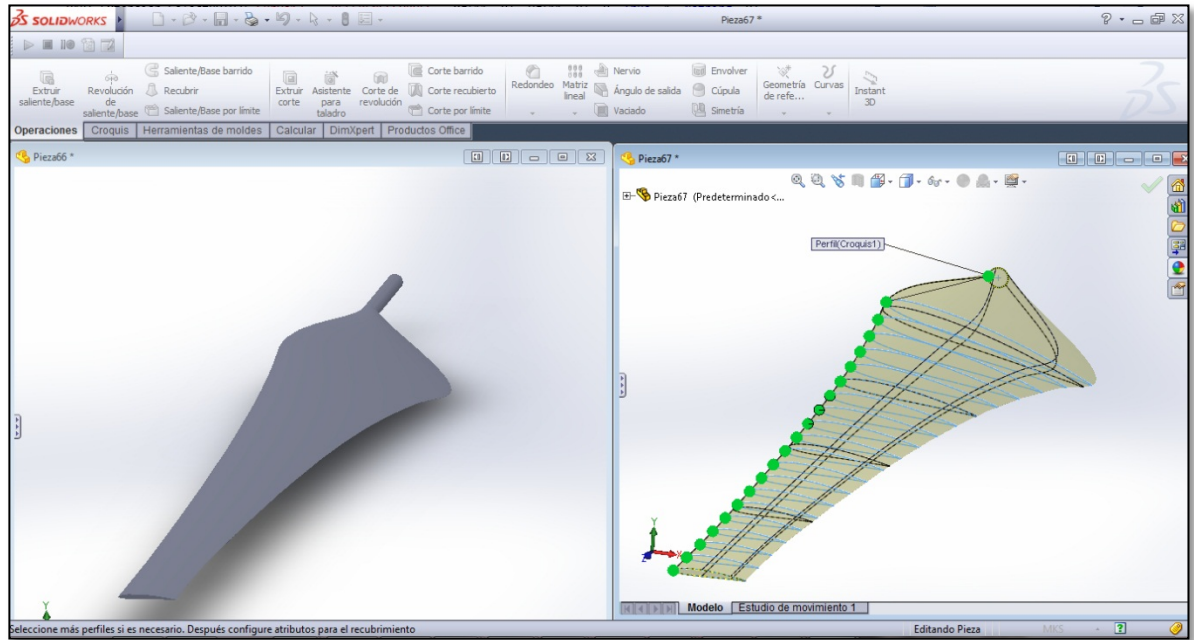
```

PART.Extension.SelectByID2("Curva20", "REFERENCECURVES", xg(19, 0), yg(19, 0), 0, True, 0, Nothing, 0)
PART.Extension.SelectByID2("Curva19", "REFERENCECURVES", xg(18, 0), yg(18, 0), 0, True, 0, Nothing, 0)
    ⋮
PART.Extension.SelectByID2("Curva1", "REFERENCECURVES", xg(0, 0), yg(0, 0), 0, True, 0, Nothing, 0)
PART.Extension.SelectByID2("Croquis1", "SKETCH", -rcroot, 0, (0.1 * Rr), True, 0, Nothing, 0)
PART.ClearSelection2(True)
PART.Extension.SelectByID2("Curva20", "REFERENCECURVES", xg(19, 0), yg(19, 0), 0, False, 1, Nothing, 0)
PART.Extension.SelectByID2("Curva19", "REFERENCECURVES", xg(18, 0), yg(18, 0), 0, True, 1, Nothing, 0)
    ⋮
PART.Extension.SelectByID2("Curva1", "REFERENCECURVES", xg(0, 0), yg(0, 0), 0, True, 1, Nothing, 0)
PART.Extension.SelectByID2("Croquis1", "SKETCH", -rcroot, 0, (0.1 * Rr), True, 1, Nothing, 0)
PART.FeatureManager.InsertProtrusionBlend(False, True, False, 1, 6, 6, 1, 1, True, True, False, 0, 0, . . .

```

La modificación del proceso de recubrimiento, especifica los puntos guías deseados para el desarrollo de tal operación. En la figura 52 puede verse una hélice modelada con el nuevo procedimiento (en una ventana minimizada) y sus respectivos puntos guía.

Figura 52. Resultado de la corrección pertinente al modelado de la hélice



Una vez modificado el código correspondiente al recubrimiento de la hélice, fue necesario verificar la nueva estrategia para diferentes configuraciones de hélice, sin embargo no fueron hallados más errores de modelado, dando por terminada la etapa de pruebas de *UIS-Propeller V2.0*.

5. DESCRIPCIÓN DEL SOFTWARE

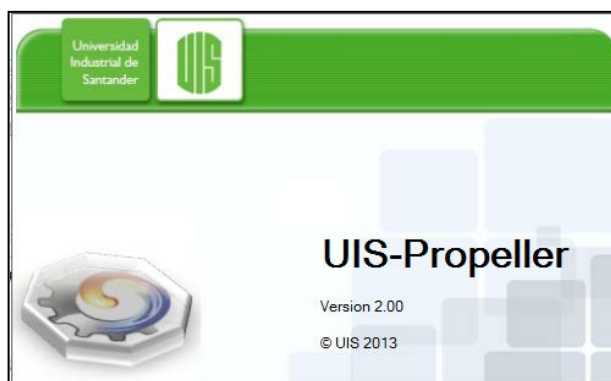
UIS-Propeller, es un software para el diseño de hélices, que en primera instancia surgió como respuesta a la falta de un método eficiente y confiable para el diseño de hélices propulsoras, las cuales son utilizadas por deportistas locales practicantes del vuelo experimental. Ahora en su segunda versión, presenta además de la función actualizada de diseño de hélices propulsoras más eficiente y eficaz, una nueva función para el diseño de hélices propulsadas (hélices para aerogeneradores).

UIS-Propeller posee una interfaz intuitiva y agradable, que facilita al usuario su utilización. A continuación serán descritos cada uno de los módulos que conforman *UIS-Propeller V2.0*.

5.1 PANTALLA DE INICIO

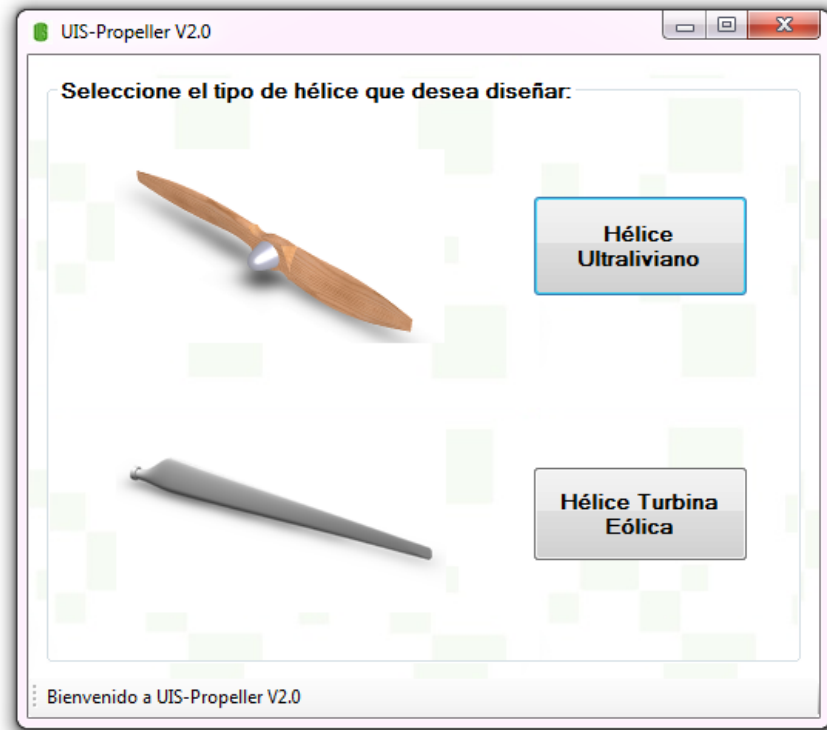
Al iniciar el software la siguiente “*Splash Screen*” (Figura 53) podrá ser observada por unos instantes. En esta se presenta información general del software, tal como el nombre, versión y último año de actualización.

Figura 53. *UIS-Propeller V2.0* Splash Screen



Posterior a esto aparecerá en pantalla un menú de selección, en el cual el usuario podrá elegir el tipo de hélice que desee diseñar.

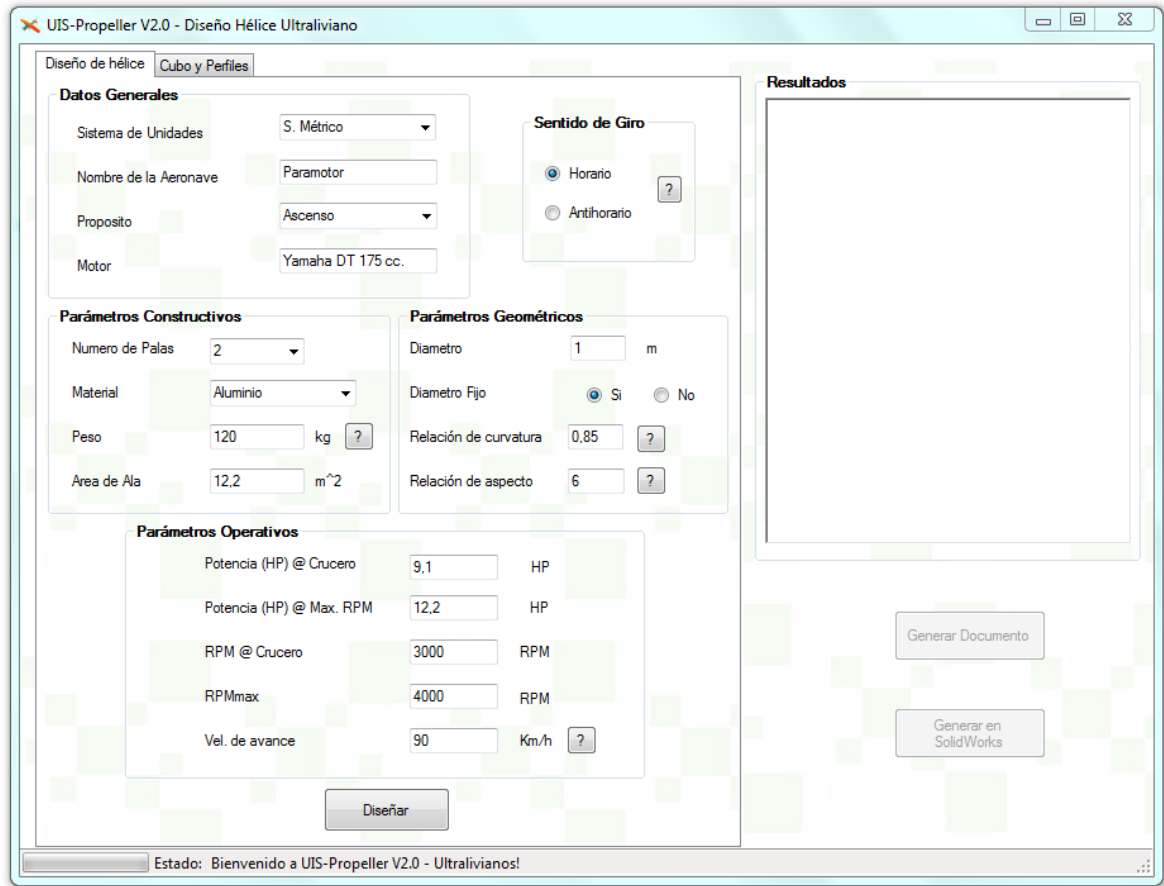
Figura 54. Pantalla de inicio de *UIS-Propeller V2.0*



5.2 FUNCIÓN DE DISEÑO DE HÉLICES PROPULSORAS (ULTRALIVIANO)

5.2.1 Pantalla de inicio. La figura 55 presenta la pantalla de inicio de ésta función del software (corresponde a la pestaña diseño de hélice). La cual conserva la estructura presentada en su primera versión. Esta, es conformada por una sección de entrada de datos (izquierda), donde el usuario podrá ingresar los requisitos y/o preferencias para el diseño de la hélice.

Figura 55. Pantalla de inicio, función de diseño de hélices propulsoras



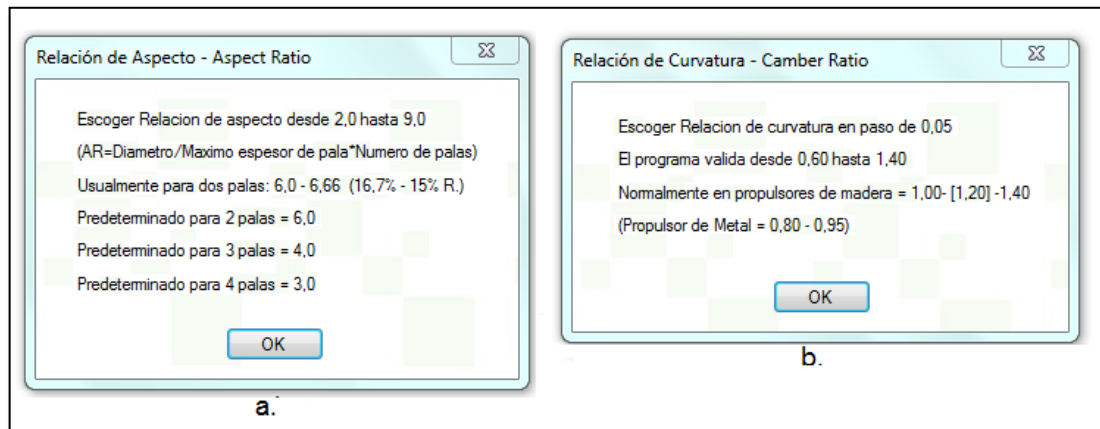
A continuación se enumerarán y describirán cada uno de sus paneles:

- a. **Datos Generales:** En este panel, el usuario podrá seleccionar el tipo de sistema de unidades que desee emplear: *Americano* o *Métrico*, así mismo que el propósito de vuelo para el cual desee diseñar la hélice (*Ascenso*, *Crucero*, *Estándar* o *Velocidad*). La casilla *motor*, corresponde a la referencia del motor que será empleado para la propulsión de la hélice en cuestión.
- b. **Sentido de giro:** como su nombre lo dice, el usuario seleccionará el sentido de giro de la hélice deseado para su diseño (*horario* o *antihorario*).

- c. **Parámetros constructivos:** Está conformado por el número de palas para el cual se diseñará la hélice (2, 3 o 4). En la casilla peso, éste equivale al peso de la aeronave más el peso de los pasajeros. La variable *Área* del ala es del todo intuitiva, sin embargo, en caso de estarse diseñando la hélice para un *glider*, el área del ala es igual al aérea del paracaídas.
- d. **Parámetros geométricos.** En este módulo el usuario podrá especificar si tiene restricciones dimensionales en cuyo caso deberá seleccionar la opción de diámetro fijo y proporcionar el valor de dicho diámetro; en caso contrario, podrá seleccionar la opción de diámetro variable, donde el valor de diámetro suministrado equivaldría al primer valor de la iteración realizada por el software para el cálculo del diámetro que mejor se ajuste a las condiciones dadas.

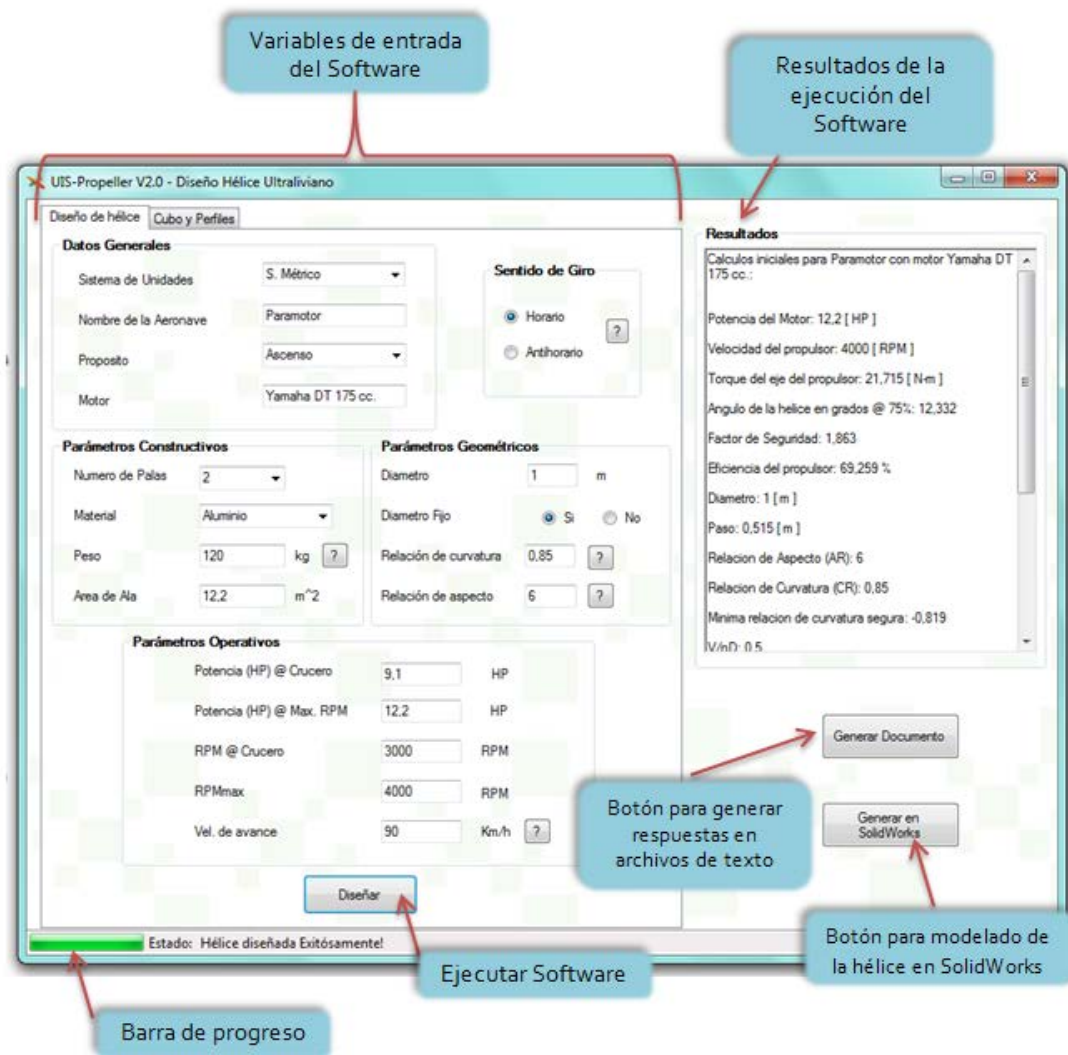
En este módulo, el usuario deberá suministrar el valor de la relación de aspecto AR (figura 56a) y relación de curvatura CR (figura 56b). Dando clic en los signos de interrogación ubicados frente a estas opciones, se tendrá acceso a la siguiente información:

Figura 56. Ventanas de ayuda del software (AR y CR)



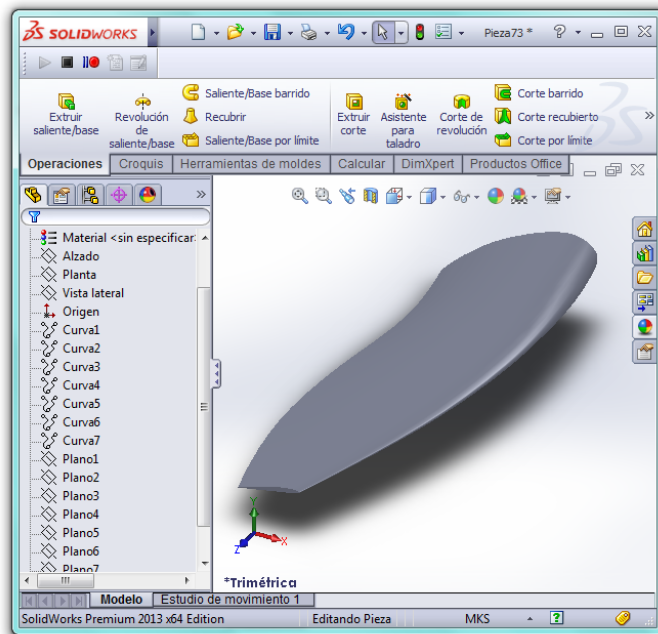
- e. **Parámetros operativos.** El usuario ingresará las especificaciones de *potencia*, *rpm* y *velocidad de avance* bajo la cual desea que opere la hélice diseñada.
- f. **Ejecución del Software.** La ejecución del software se ordenará al dar clic sobre el botón *Diseñar* ubicado en la parte inferior de la pantalla de inicio de esta función.

Figura 57. Descripción de la pantalla de inicio de la función de diseño de hélices para ultralivianos



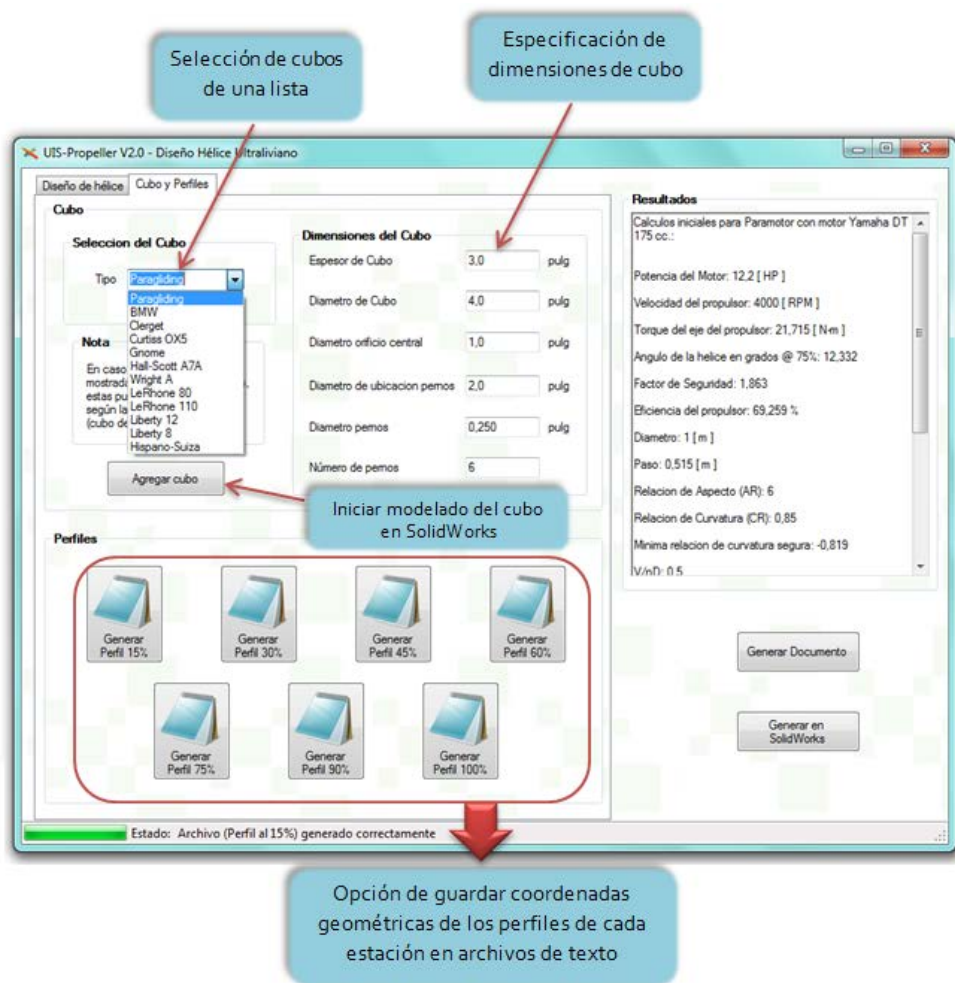
- g. **Resultados.** Una vez ejecutado el software, los resultados obtenidos serán mostrados en el panel “Resultados”, ubicado en la parte derecha de la pantalla de inicio (figura 57).
- h. **Botón Generar Documento.** Éste botón es habilitado una vez ejecutado el software, y su propósito es generar un documento de texto con los resultados obtenidos del proceso de diseño (figura 57). Dicho archivo será guardado en el directorio: *C: Users\Public\Documents*; con el nombre “Resultados del proceso de diseño Ultraliviano”
- i. **Botón generar en SolidWorks.** Al igual que el anterior, este botón es habilitado una vez realizado el proceso de diseño. Cliqueando sobre él, *UIS-Propeller* ejecutará el modelado de la hélice en SolidWorks 2013 (figura 58).

Figura 58. Pala de hélice, generada en SolidWorks 2013



5.2.2 Pantalla de cubo y perfiles. Una vez finalizado el proceso de diseño, los botones ubicados en el panel perfiles de la pestaña cubo y perfiles son habilitados, estos generan archivos de texto con las coordenadas geométricas de los perfiles calculados en cada estación de la hélice (figura 59), siendo estas ubicadas al 15%, 30%, 45%, 60%, 75%, 90% y 100% del radio de la hélice. Al igual que el archivo de texto generado para los resultados, estos son guardados en el directorio *C:\Users\Public\Documents*; con los nombres "Ultraliviano coordenadas Perfil al 15%" y así sucesivamente para cada una de las estaciones de la hélice.

Figura 59. Descripción de la pantalla cubo y perfiles de la función de diseño de hélices para ultralivianos

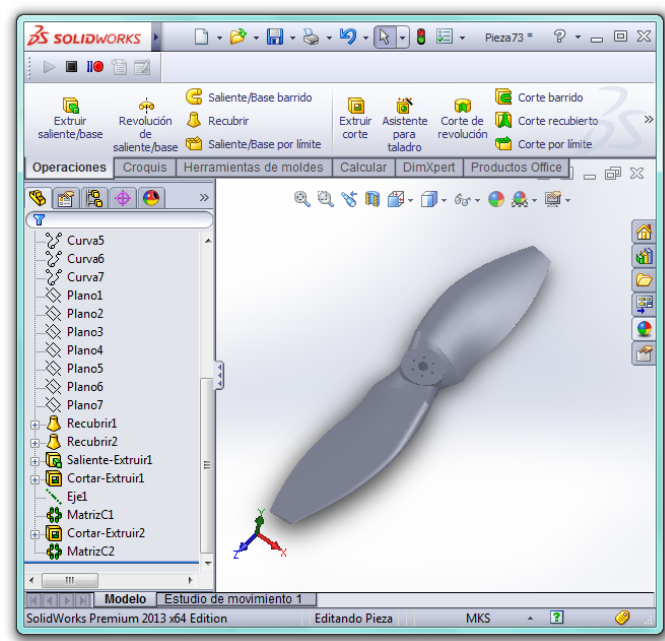


Por otro lado, el listado y las cajas de texto ubicadas en el módulo *Cubo*, son habilitados siempre y cuando la hélice haya sido modelada en SolidWorks. En este módulo, el usuario podrá seleccionar el perfil deseado de una lista o suministrar las dimensiones deseadas según sea conveniente. En este apartado, es importante tomar en consideración que el cubo de la hélice depende del motor al cual será instalada.

Una vez establecido el cubo de la hélice, cliqueando el botón agregar cubo, iniciará el proceso de construcción del cubo de la hélice en SolidWorks 2013 (figura 59).

NOTA: Es importante **no** cerrar la ventana de SolidWorks con la hélice modelada inicialmente, ya que el cubo de esta, será modelado sobre el mismo archivo (figura 60).

Figura 60. Resultado final de la ejecución de la función de diseño de hélices para ultralivianos

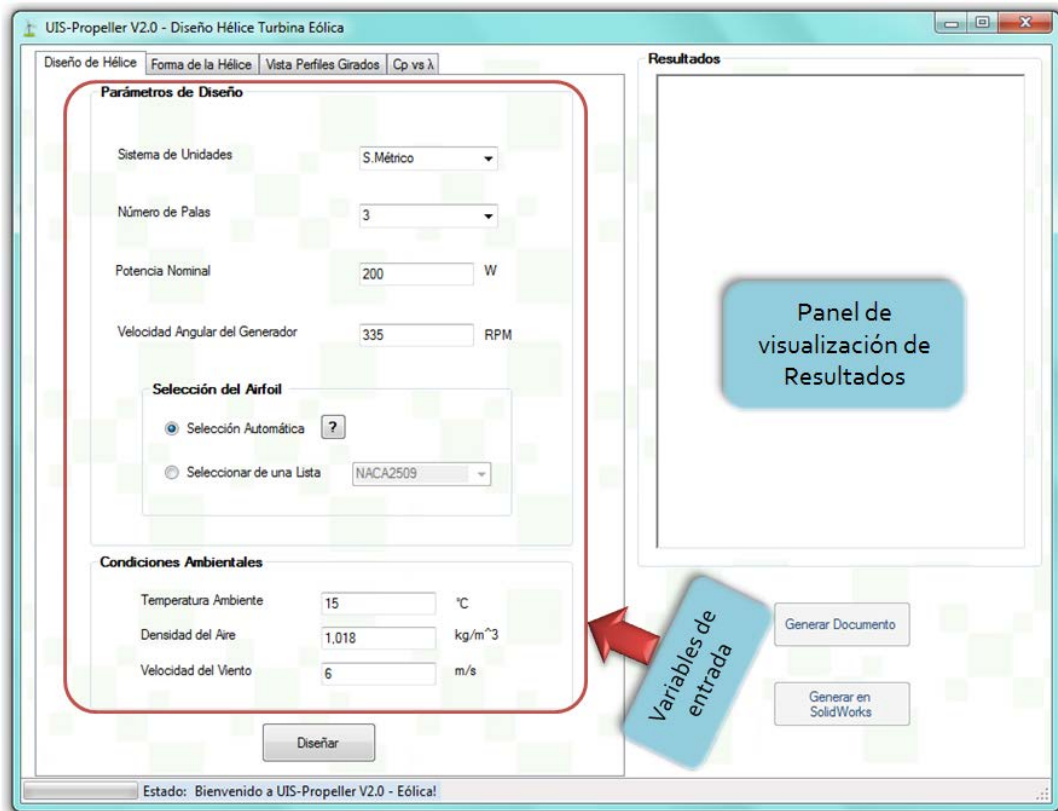


5.3 FUNCIÓN DE DISEÑO DE HÉLICES PROPULSADAS (AEROGENRADORES)

5.3.1 Pantalla de inicio. Esta puede ser vista en la figura 61, de donde puede apreciarse que se encuentra dividida en dos partes básicas: Variables de entrada al software y datos de salida (*Resultados*). A continuación serán descritos cada uno de sus componentes.

- a. **Parámetros de diseño:** hace referencia a los requerimientos del usuario, tales como el sistema de unidades deseado (*Americano* o *Métrico*), el número de palas (*1, 2* o *3*), la potencia nominal que requiere generar y las revoluciones del generador que tenga bajo su poder o desee adquirir.

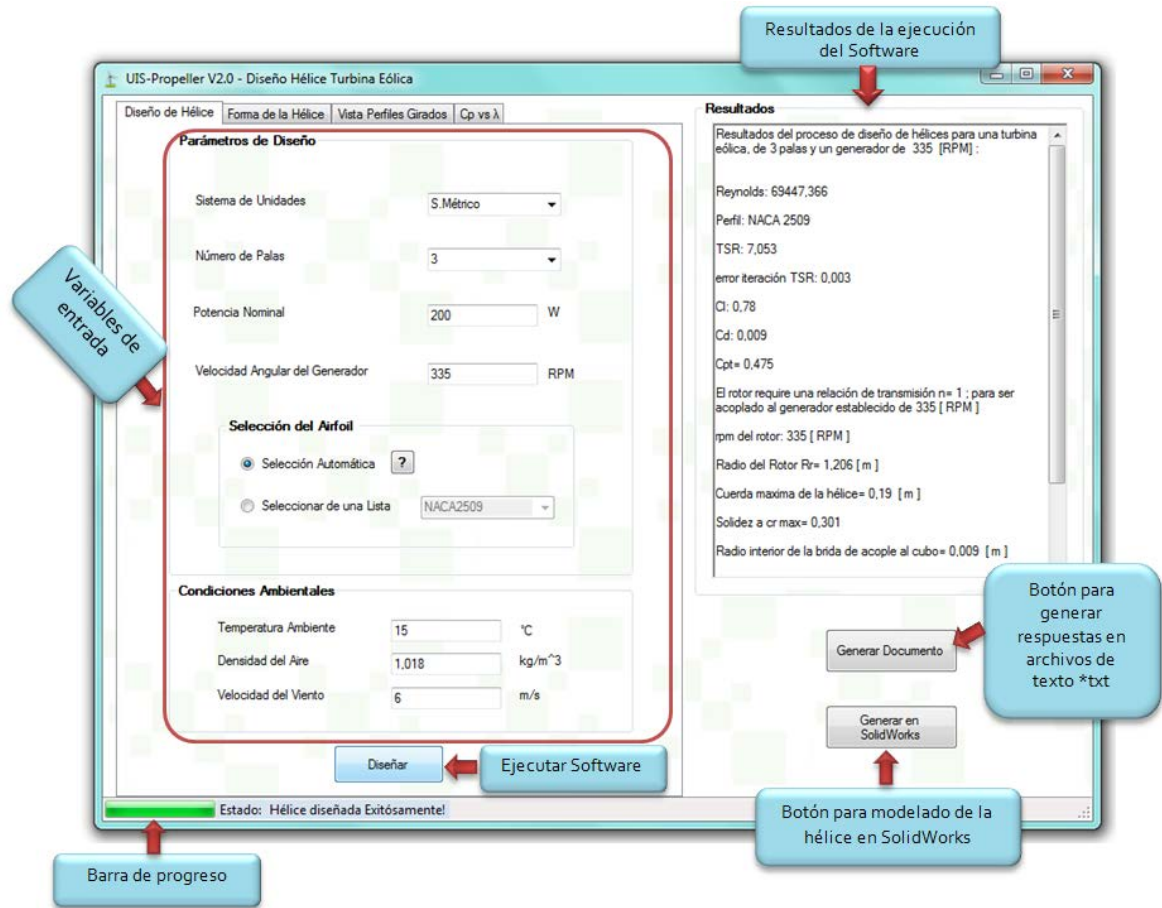
Figura 61. Pantalla de inicio función de diseño de hélices propulsadas



En la figura 61 vemos también un panel llamado selección de perfil, en este el usuario podrá seleccionar el perfil de su preferencia de una lista de perfiles NACA (2509; 4412; 4415; 63-215; 63(4)-221; 63(2)-415; 63(4)-421; 64(2)-415; 64(4)-421; 65(2)-415 y 65(4)-421), o, en caso de que el usuario no esté seguro de cual perfil elegir, podrá seleccionar la casilla de selección automática i.e. el software realizará el proceso de selección de perfil(4.1.2.2) .

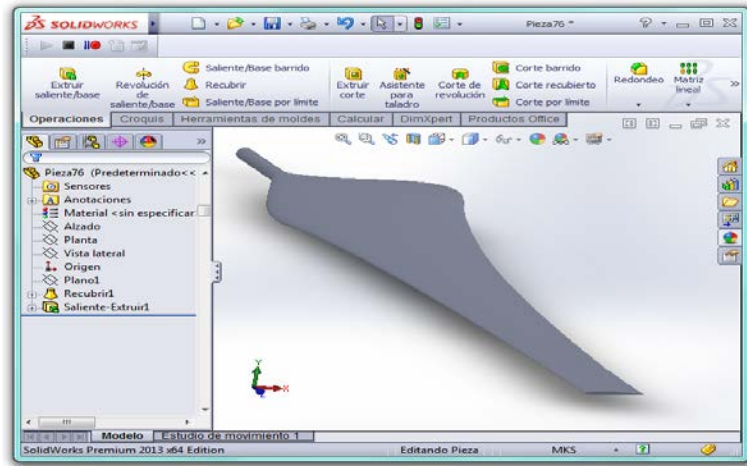
- b. **Condiciones ambientales.** Como su nombre lo indica, en este panel el usuario debe introducir los valores de *temperatura*, *densidad* y *velocidad del viento* del lugar donde desea instalar el aerogenerador.
- c. **Ejecución del Software.** La ejecución del software se ordenará al dar clic sobre el botón *Diseñar* ubicado en la parte inferior de la pantalla de inicio.
- d. **Resultados.** Una vez ejecutado el software, los resultados obtenidos serán mostrados en el panel "*Resultados*", ubicado en la parte derecha de la pantalla de inicio (figura 62).
- e. **Botón Generar Documento.** Al igual que en la función del diseño de hélices propulsoras, para este propósito, sólo se habilitará este botón una vez ejecutado el software, generando un documento de texto con los resultados obtenidos del proceso de diseño (figura 62). El archivo será guardado en el directorio: *C: Users\Public\Documents*; con el nombre "*Resultados del proceso de diseño Eólica*"

Figura 62. Descripción de la pantalla de inicio de la función de diseño de hélices para aerogeneradores



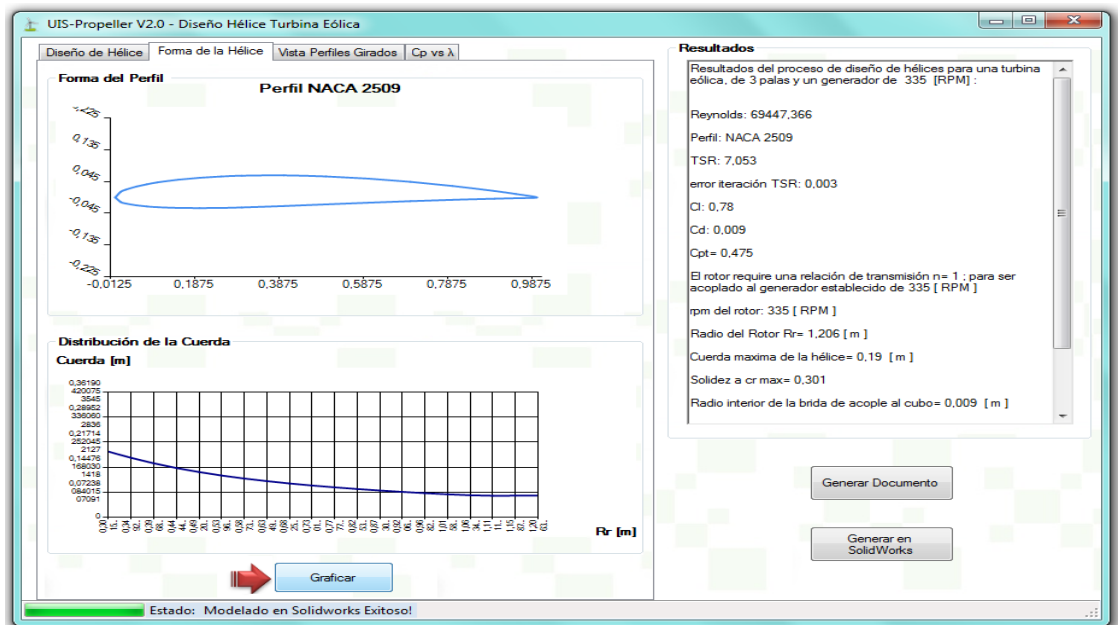
- f. **Botón generar en SolidWorks.** Se habilitará una vez realizado el proceso de diseño. Al dar clic sobre él, *UIS-Propeller* ejecutará el modelado de la hélice en SolidWorks 2013 (figura 63).

Figura 63. Modelado de la hélice resultante de la ejecución de la función de diseño de hélices para aerogeneradores



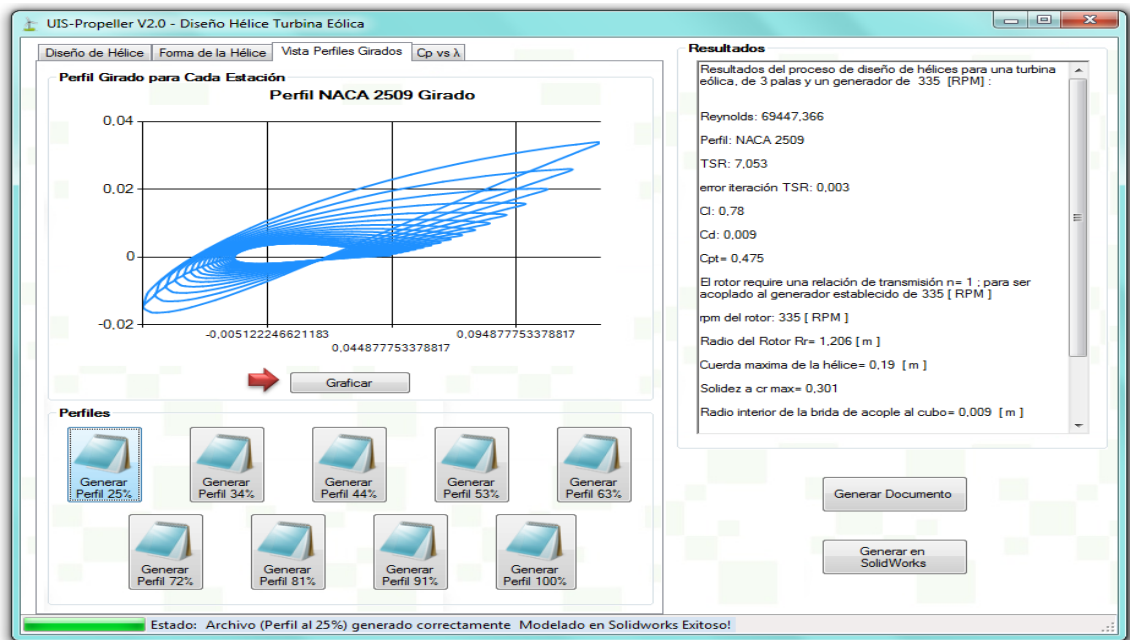
5.3.2 **Módulo forma de la hélice.** En esta pestaña, al clicar el botón *Graficar*, el usuario puede ver la forma del perfil empleado, al igual que la distribución de la cuerda a lo largo de la hélice i.e. curva cr vs r/R (figura 64).

Figura 64. Módulo forma de la hélice de la función de diseño de hélices para aerogeneradores



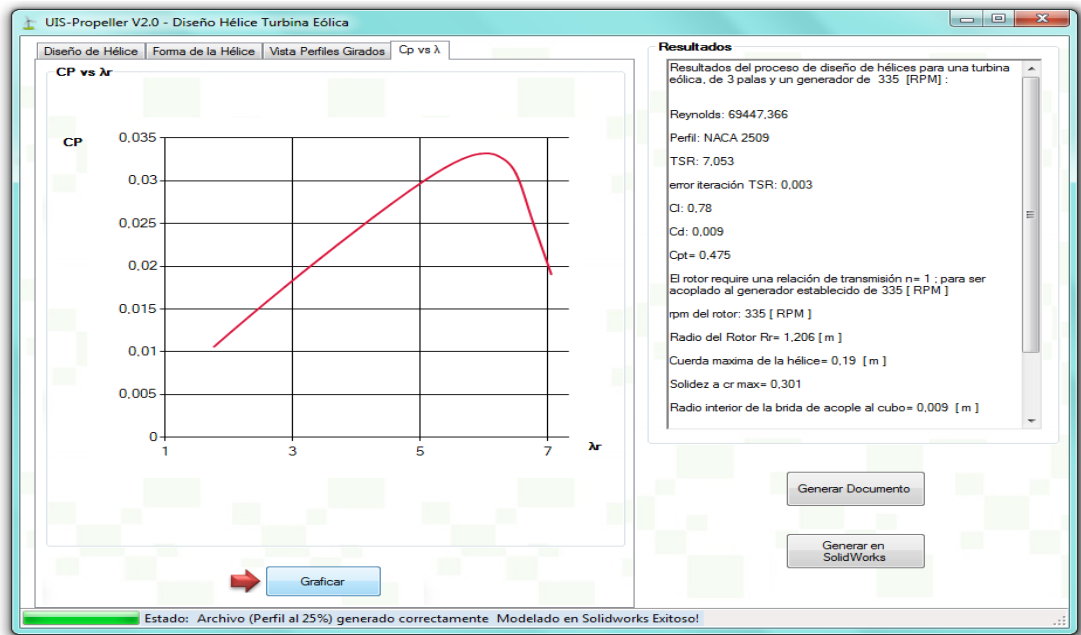
5.3.3 Módulo vista de perfiles girados. En este módulo, el usuario podrá graficar los perfiles girados a su ángulo de calado β_i , adicional a esto, también podrá obtener como documentos de texto i.e. extensión *.txt, las coordenadas geométricas de los perfiles ubicados al 25%, 34%, 44%, 53%, 63%, 72%, 81%, 91% y 100% del radio del rotor. Dichos archivos serán guardados en el directorio *C: Users\Public\Documents*; con los nombres “Eólica Coordenadas Perfil al 25%” y así sucesivamente para cada una de las estaciones mencionadas (figura 65).

Figura 65. Módulo vista de perfiles girados de la función de diseño de hélices para aerogeneradores



5.3.4 Módulo de desempeño C_p vs λ . Para finalizar, la nueva función de UIS-Propeller permite al alcance de un clic obtener la gráfica del Coeficiente de potencia (C_p) en función de la relación de velocidad de punta (TSR o λ) de la hélice diseñada (figura 66).

Figura 66. Módulo de desempeño C_p vs λ de la función de diseño de hélices para aerogeneradores



5.4 ABOUTBOX

El software posee un AboutBox, en el cual éste es descrito de forma general (figura 67). Para acceder a este, se debe dar clic izquierdo sobre el ícono de notificación que aparece en la barra de tareas al ejecutar el software (figura 68).

Figura 67. AboutBox de *UIS-Propeller V2.0*

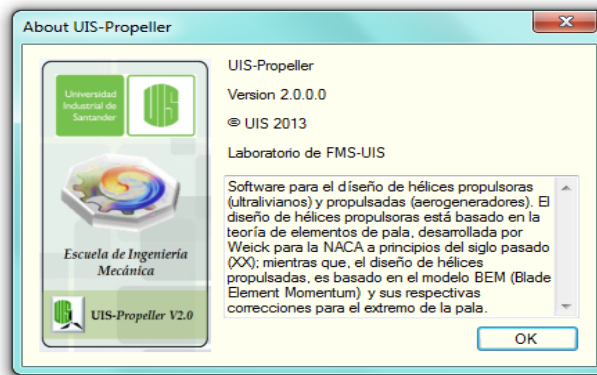


Figura 68. Ícono de notificación de *UIS-Propeller* en la barra de tareas, acceso al AboutBox



6. CONCLUSIONES

- ▶ La segunda versión del software *UIS-Propeller* para el diseño de hélices de ultralivianos (propulsoras) y turbinas eólicas (propulsadas) fue desarrollada, bajo la plataforma Visual Studio 2010, en lenguaje Visual Basic (VB.NET).
- ▶ Durante el proceso de actualización de la función de diseño de hélices para ultraliviano del Software, se llevaron a cabo verificaciones de las diferentes ecuaciones utilizadas en el proceso de diseño de hélices para cada uno de los propósitos de vuelo (*Ascenso, Estándar, Crucero y Velocidad*), con el fin de garantizar la confiabilidad del software.
- ▶ Se encontró que el software no realizaba la verificación del mínimo error de iteración permitido para el caso de diámetro variable, así mismo no se daba al usuario el valor de éste nuevo diámetro i.e. las coordenadas geométricas de la hélice eran calculadas con el valor de diámetro proporcionado para la primera iteración. Como es natural, estas discrepancias fueron corregidas en la nueva versión del software.
- ▶ Mediante la implementación de bucles de programación, se logró una reducción del 83% de las líneas de código programadas en la versión 1.0 del software i.e. las 14593 líneas programadas en la versión 1.0 se redujeron a 2473 para esta nueva versión, mejorando con ello los tiempos de respuesta de la función de diseño de hélices para ultralivianos .
- ▶ Una nueva estrategia para el modelado de hélices y construcción del cubo para ultralivianos fue ideada, con lo que se dio solución a las múltiples interferencias

geométricas generadas en la etapa de modelado de la hélice en SolidWorks de la versión 1.0.

- ▶ Se programaron API's de integración con SolidWorks 2013, estableciendo con ello la comunicación entre *UIS-Propeller V2.0* y el software CAD, a través de la cual modelar las hélices diseñadas.
- ▶ Una nueva función para el diseño de hélices para aerogeneradores fue adicionada a *UIS-Propeller*. El fundamento teórico de esta función es el modelo BEM (*Blade Element Momentum*) y los modelos de pérdidas en el extremo de la hélice pertinentes a un número finito de palas.
- ▶ Una opción de selección automática de perfil aerodinámico para turbinas eólicas fue adicionado a *UIS-Propeller V2.0*, facilitando de este modo el diseño de la hélice a usuarios con pocos (o ningún) conocimientos aerodinámicos.
- ▶ La confiabilidad de la nueva función de diseño de hélices para aerogeneradores, fue comprobada durante una etapa de pruebas. En esta etapa se comparó la hélice obtenida a través de la ejecución de *UIS-Propeller* bajo ciertos requerimientos de diseño, con una hélice previamente diseñada (en este proyecto la hélice tomada como referencia fue la diseñada por Beltrán L. *et al* en su proyecto de grado).

7. RECOMENDACIONES

- ▶ Para nuevas versiones del software, se recomienda incluir al módulo de diseño de hélices para aerogeneradores, otros tipos de perfiles aerodinámicos, tales como la familia “S” de perfiles desarrollados por NREL³⁷ exclusivamente para turbinas eólicas de eje horizontal (HAWTs)³⁸.
- ▶ Sabiendo que las regiones interna y externa de la pala requieren diferentes especificaciones de elevación (lift), para nuevos desarrollos del software, se recomienda implementar un proceso de selección de perfil por separado para cada región, mejorando con ello el desempeño de la hélice i.e. la captación de energía disponible en el viento³⁹.
- ▶ En versiones posteriores de *UIS-Propeller*, como acción de mejora a la función de diseño de hélices para vehículos ultralivianos, podría ser agregada una opción de cálculo de hélices de paso variable, las cuales presentan mejor desempeño durante el régimen operacional.

³⁷ National Renewable Energy Laboratory (NREL)

³⁸ <http://wind.nrel.gov/airfoils/AirfoilFamilies.html>

³⁹ HABALI, S.M y SALEH, I.A, 2000.

BIBLIOGRAFÍA

- ▶ ABBOTT, Ira H.; VON DOENHOFF, Albert E. and STIVERS, Louis S. SUMMARY OF AIRFOIL DATA. Washington, United State: NACA Report 824, 1945.

- ▶ ALDANA ZAMBRANO, Cristian y GRANADOS SANMIGUEL, Sergio. Herramienta Software para cálculo y diseño de hélices en aviación experimental basada en la teoría de elementos aerodinámicos y manufactura experimental de hélice. Bucaramanga, Colombia: Universidad Industrial de Santander, 2009.

- ▶ BELTRÁN DUARTE, Luis Gabriel y FIGUEROA MADERO, Juan Carlos. Diseño de un sistema eólico de baja potencia para la zona aledaña al municipio de Gameza - Boyacá. Bucaramanga, Colombia: Universidad Industrial de Santander, 2011, pp. 77-122.

- ▶ DIXON, S.L. Fluid Mechanics and Thermodynamics of Turbomachinery. 5th Edition Elsevier [online]. 1998 [cited 17 of August 2012], pp. 323-362. Available from internet:
<http://www.knovel.com/web/portal/browse/display?_EXT_KNOVEL_DISPLAY_bookid=2071&VerticalID=0>

- ▶ GOMEZ R., William; APERADOR C., William y DELGADO T. Emilio. Desarrollo de rotores para turbina eólica de 200 vatios mediante la implementación de tecnología local. Ingeniería y Universidad [online]. 2011, vol. 15 no. 2 [citado en 13 de Marzo de 2012], pp. 303-318. Disponible en

internet:

<<http://revistas.javeriana.edu.co/index.php/IyU/article/view/2027>>.

- ▶ HABALI, S.M y SALEH, I.A. Local design, testing and manufacturing of small mixed airfoil wind turbine blades of glass fiber reinforced plastics. Part I: Design of the blade and root. Energy Conversion and Management [online]. 2000, no. 41 [cited 13 de March 2012], pp. 249-280. Available from internet: <<http://www.sciencedirect.com.ezproxy.lib.ou.edu/science/article/pii/S019689049900103X>>.
- ▶ HAU, E. WIND TURBINES FUNDAMENTALS, TECHNOLOGIES, APLICATIONS, ECONOMICS. 2nd Edition. Berlin, Germany: Springer, 2006, pp. 91-145.
- ▶ JACOBS, E. WARD, K. and PINKERTON R. THE CHARACTERISTICS OF 78 RELATED AIRFOIL SECTIONS FROM TESTS IN THE VARIABLE-DENSITY WIND TUNNEL. Washington DC, United States: NACA Report 460, 1935.
- ▶ JHA, A.R. WIND TURBINE TECHNOLOGY. Florida, United States: Taylor and Francis Group, 2011, pp. 79-109.
- ▶ MANWELL, J.F. MCGOWAN, J.G. and ROGERS, A.L. WIND ENERGY EXPLAINED. West Sussex, England: John Wiley & Sons, LTD 2002, pp. 124-130.
- ▶ SHEN, W. Z. MIKKELSEN, R. And SØRENSEN, J. N. Tip Loss Corrections for Wind Turbine Computations. Wiley Interscience [online]. 2005, no. 8 [cited 10

of November 2012], pp. 464-474. Available from internet:
<<http://onlinelibrary.wiley.com/doi/10.1002/we.153/pdf>>

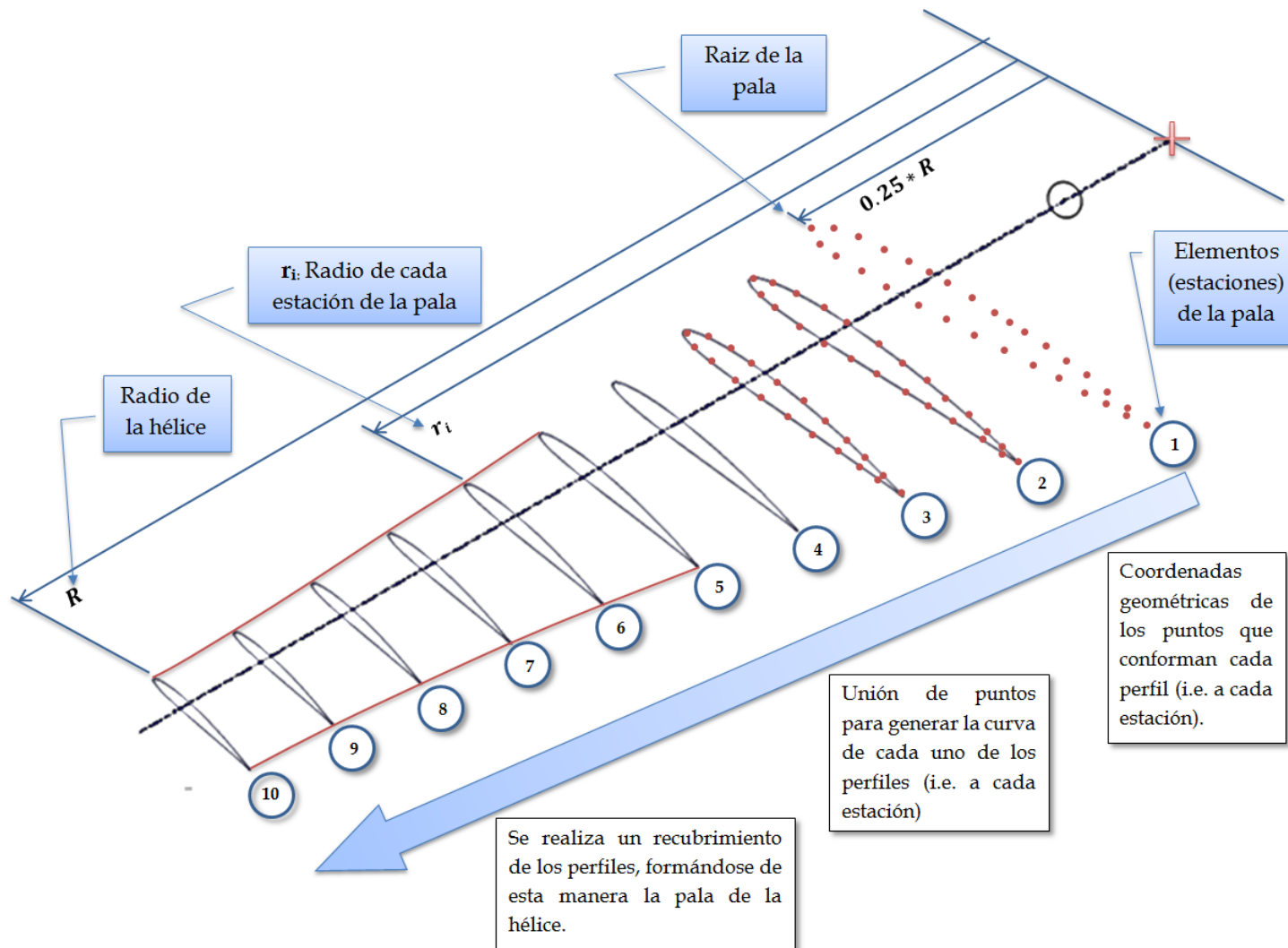
- ▶ WEICK, F. E. Practical Application of the blade element theory - I. Washington, United State: NACA, 1926.
- ▶ WEICK, F. E. Practical Application of the blade element theory - II. . Washington, United State: NACA, 1926.
- ▶ WEICK, F. E. Practical Application of the blade element theory - III. . Washington, United State: NACA, 1926.
- ▶ WEICK, F. E. Practical Application of the blade element theory - IV. . Washington, United State: NACA, 1926.

WEBGRAFÍA

- ▶ http://www.ae.illinois.edu/m-selig/ads/coord_database.html
- ▶ <http://www.mh-aerotoools.de/airfoils/javafoil.htm>
- ▶ <http://wind.nrel.gov/airfoils/AirfoilFamilies.html>

ANEXOS

ANEXO A. Explicación esquemática de la metodología para el modelado de la pala en SolidWorks



ANEXO B. Proceso de diseño de hélices para ultralivianos¹

1. PROCESO DE DISEÑO DE LA HÉLICE

El proceso de diseño de la hélice se basa en la teoría de elementos aerodinámicos, descrita anteriormente. En esta teoría se desarrolla todo el diseño correspondiente a la hélice mediante un paso a paso, en donde cada uno de estos tiene un significado físico y resulta indispensable para la consecuente ejecución del diseño. Dentro de esta teoría se distinguen 2 grandes bloques de procedimientos de diseño que nosotros distinguimos como diseño geométrico y diseño complementario.

1.1 DISEÑO GEOMÉTRICO

El diseño geométrico consiste en una serie de pasos ejecutados con el fin de lograr un dimensionamiento general de la hélice, mediante la determinación del diámetro y del paso.

1.1.1 Datos de entrada para el diseño geométrico. Los datos de entrada utilizados para desarrollar el diseño geométrico son en su mayoría los ingresados por el usuario en la ventana de interfaz del programa.

Inicialmente se debe ingresar a Datos Generales, donde se debe escoger tanto el tipo de unidades en las que se quiere trabajar (Sistema Métrico o Sistema

¹ ALDANA, Cristian y GRANADOS, Sergio. HERRAMIENTA SOFTWARE PARA CÁLCULO Y DISEÑO DE HÉLICES EN AVIACIÓN EXPERIMENTAL BASADA EN LA TEORÍA DE ELEMENTOS AERODINÁMICOS Y MANUFACTURA EXPERIMENTAL DE HÉLICE. Bucaramanga, Colombia: Universidad Industrial de Santander, 2009, pp. 19-41.

Americano) como el propósito para el cual se quiere la hélice, entre estos se tienen Ascenso, Estándar, Crucero y Velocidad.

Seguido a esto es necesario introducir dentro de Datos Constructivos el número de palas de la hélice, donde nos encontramos con las opciones de 2, 3 y 4 palas.

Ahora nos encontramos con los Datos Operativos, considerados indispensables y en torno a los cuales gira el diseño de la hélice. Estos datos son: Potencia de crucero (BHP), Potencia a máxima velocidad (HP_{max}), Velocidad angular de crucero (RPM), Velocidad angular a máxima velocidad (RPM_{max}) y Velocidad de avance. En este último sus unidades variarían según el sistema elegido, Km / h para el sistema métrico y Mph para el sistema americano.

Por último necesitamos los datos geométricos, como el diámetro de hélice, que para el caso de desarrollar un diseño con restricciones de diámetro este será el definitivo, mientras para un diseño con cierta flexibilidad de espacio donde va ubicada la hélice, este solo será un diámetro inicial o aproximado. Además, se requiere del ingreso de la relación de curvatura de la pala de la hélice como un todo (CR) la cual dependerá del material con el cual se vaya a fabricar la hélice. Este término representa una relación del espesor de la pala completa con el espesor de una pala estándar, siendo esta constante a lo largo de la hélice. Entonces, si la curva se incrementa en un 10% en cada uno de los puntos, la relación de curvatura será de 1.1. Para el caso de propulsores de madera encontramos relaciones de curvatura entre 1.0 y 1.4; y para propulsores de metal entre 0.8 y 0.95.

Adicional a esto se requiere de la relación de aspecto (AR), la cual varía especialmente según el número de palas. Sin embargo también se debe tener en cuenta el diámetro de la hélice y el máximo espesor de pala. En el programa se

sugieren algunos valores predeterminados de AR según el número de palas. Tenemos que para 2 palas podemos tener valores entre 6.0 y 6.66; para 3 palas de 4.0 y para 4 palas de 3.0.

Figura 1. Ventana interfaz del software UIS – Propeller v.1.0.

The screenshot shows the 'UIS-Propeller v.1.0' software interface. It has a blue title bar and a main window with a light beige background. At the top, there are two tabs: '1. Calculos Iniciales' (selected) and '2. Cubo y Perfiles'. The interface is divided into several sections:

- Datos Generales:** Includes dropdown menus for 'Unidades' (set to 'S. Metrico'), 'Aeronave' (set to 'Paramotor'), 'Proposito' (set to 'Ascenso'), and 'Motor' (set to 'Yamaha DT 175 cc.').
- Datos Constructivos:** Includes dropdown for 'Numero de Palas' (set to '2'), 'Material' (set to 'Aluminio'), and input fields for 'Peso' (120 kg) and 'Area de Ala' (12.2 m²).
- Datos Operativos:** Includes input fields for 'Potencia (HP) @ Crucero' (9.1 HP), 'Potencia (HP) @ Max. RPM' (17 HP), 'RPM @ Crucero' (3000 RPM), 'RPMmax' (7000 RPM), and 'Vel. de avance' (90 Km/h).
- Datos Geométricos:** Includes input for 'Diametro' (1.10 m), radio buttons for 'Diametro Fijo' (set to 'Si'), and input fields for 'Relación de curvatura' (0.85) and 'Relación de aspecto' (6).
- Sentido de Giro:** Includes radio buttons for 'Horario' and 'Antihorario' (set to 'Antihorario').

At the bottom right, there are three buttons: 'Diseñar', 'Generar Documento', and 'Generar en Solidworks'. The status bar at the bottom left shows 'Estado: Listo'.

Con estos valores ingresados al programa se puede comenzar la ejecución o desarrollo del mismo, lo cual a continuación explicaremos detalladamente.

Todos los cálculos realizados a continuación se basan en una serie de artículos de la N.A.C.A. (National Advisory Committee for Aeronautics) llamados “Propeller Design, Practical Application of the Blade Element Theory I, II, III y IV”.

1.1.2 Cálculo del coeficiente de velocidad - potencia para hélices de $AR= 6$ y $CR= 1$. Para llevar a cabo el cálculo del coeficiente de velocidad-potencia para hélices de $AR= 6$ y $CR= 1$, se deben realizar cálculos anteriores a este.

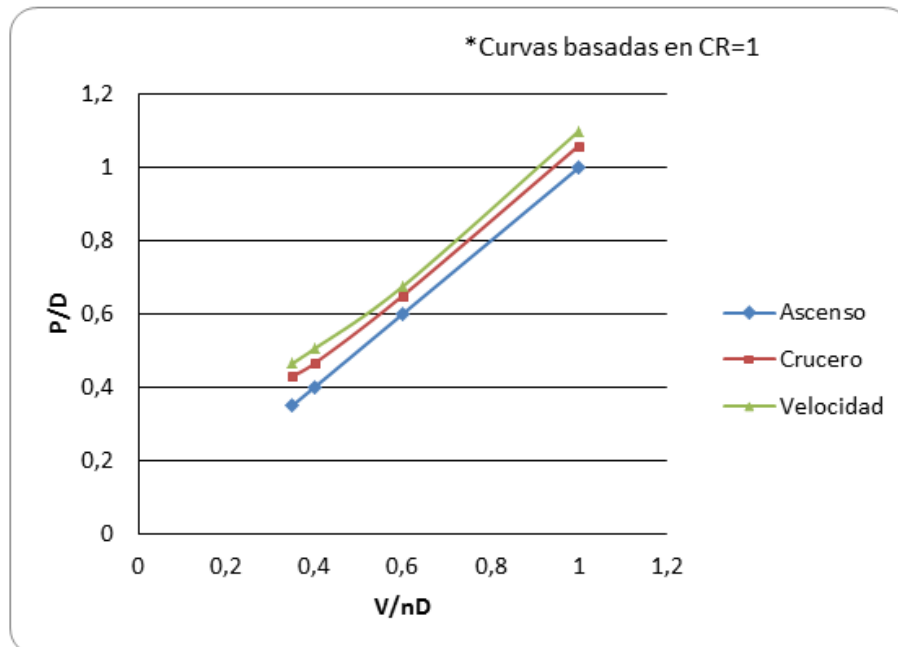
Las condiciones de operación de cualquier propulsor dependen de la velocidad de avance, de las revoluciones y del diámetro de este. Estas variables son agrupadas en un coeficiente adimensional llamado función de deslizamiento,

$$\frac{V}{nD} = 88 * \frac{MPH}{(RPM * D)}$$

Donde la velocidad de avance es *MPH*, las revoluciones *RPM* y el diámetro *D*.

Este coeficiente es usado para realizar el cálculo de la relación Paso-Diámetro Aproximado (P/D_{aprox}), además del propósito para el cual va a ser utilizada la hélice, tales como Ascenso, Crucero y Velocidad, y de la relación de curvatura (*CR*) ingresada por el usuario. Para hallar esta relación fue necesario convertir la figura 2 en una ecuación para cada uno de los propósitos.

Figura 2. Combinaciones de P/D y V/nD para Ascenso, Crucero y Velocidad.



Fuente: (Weick, 1926)

Las ecuaciones conseguidas fueron:

▶ Ascenso

$$\frac{P}{D} \text{ approx} = \frac{V}{nD} - \frac{(CR - 1)}{10}$$

▶ Crucero

$$\frac{P}{D} \text{ approx} = -0.1035 \left(\frac{V}{nD}\right)^3 + 0.3642 \left(\frac{V}{nD}\right)^2 + 0.6396 \left(\frac{V}{nD}\right) + 0.1606 - \frac{CR - 1}{10}$$

▶ Velocidad

$$\frac{P}{D} \text{ approx} = -0.1184 \left(\frac{V}{nD}\right)^3 + 0.4977 \left(\frac{V}{nD}\right)^2 + 0.4892 \left(\frac{V}{nD}\right) + 0.2352 - \frac{CR - 1}{10}$$

Ahora teniendo $\frac{P}{D} \text{ approx}$ y $\frac{V}{nD}$ se halla $1-s$, donde s es el deslizamiento nominal pero que por motivos de diseño y análisis se ha encontrado más conveniente usar $1-s$.

$$1 - s = \frac{\frac{V}{nD}}{\frac{P}{D} \text{ approx}}$$

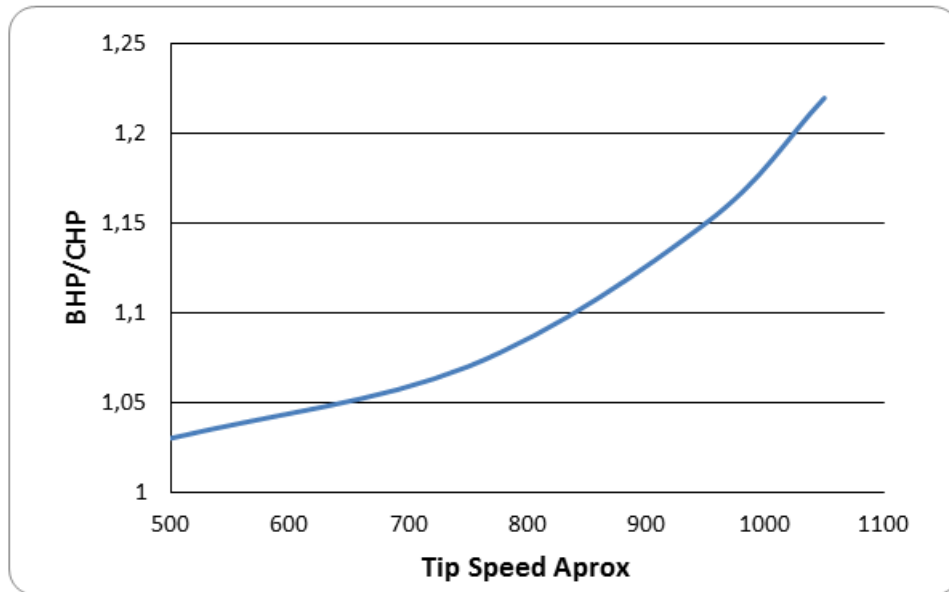
Paralelamente se puede calcular la velocidad máxima aproximada utilizando las revoluciones y el diámetro de la hélice.

$$\text{Tipspeedapprox} = 0.052356 * \text{RPM} * D$$

Con este valor se halla la relación entre la potencia al freno o crucero del motor (BHP) y la potencia considerando el factor de forma, el cual depende de la

velocidad máxima, de la interferencia del fuselaje y de la distorsión de la hélice, pero que para el caso nuestro va a tener un valor de 1 porque se trabajan hélices Navales Estándar. Para hallar esta relación fue necesario convertir la figura 3 en una ecuación.

Figura 3. Factor de potencia para hélices.



Fuente: (Weick, 1926)

La ecuación conseguida fue la siguiente:

$$\frac{BHP}{CHP} = 1.7424 * 10^{-12} * Tipspeedaprox^4 - 4.742 * 10^{-9} * Tipspeedaprox^3 + 5.21 * 10^{-6} * Tipspeedaprox^2 - 2.482 * 10^{-3} * Tipspeedaprox + 1.452$$

Con esto se puede ahora calcular la potencia al torque (QHP) mediante la siguiente expresión,

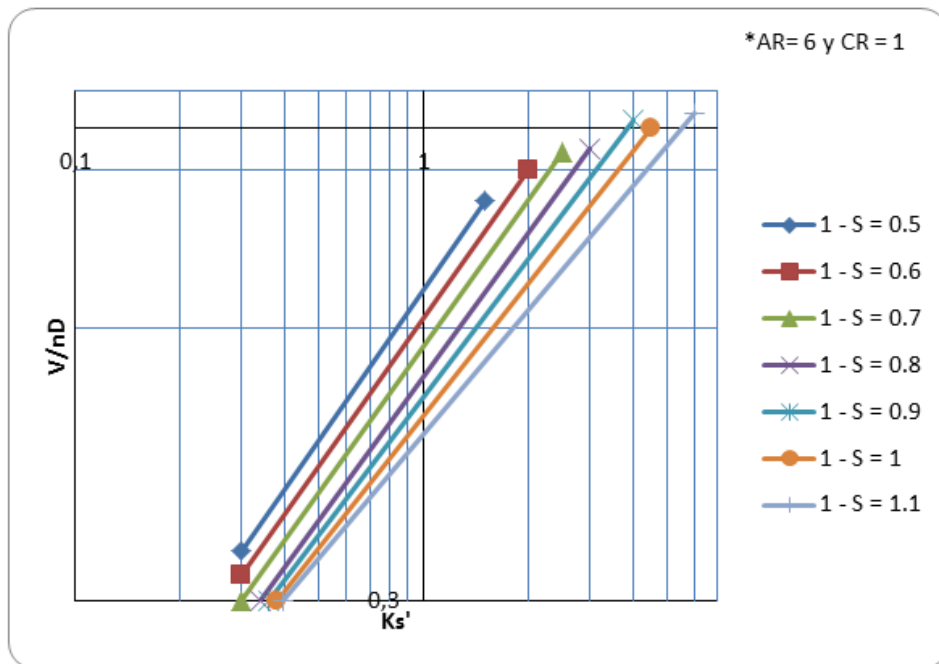
$$QHP = \frac{BHP}{FormFactor * \left(\frac{BHP}{CHP}\right)}$$

Seguido a esto se halla el coeficiente de velocidad - potencia (K_s) mediante la siguiente ecuación,

$$K_s = 0.325 * \sqrt{\frac{MPH^5}{QHP * RPM^2}}$$

Después de todos los cálculos anteriores se hallara el coeficiente de velocidad y potencia para hélices de $AR=6$ y $CR=1$ (K_s'), el cual se halla mediante la figura 4 que tiene en el eje de ordenadas la función de deslizamiento (V/nD) y el deslizamiento nominal ($1-s$), y en el eje de abscisas el coeficiente de velocidad - potencia K_s' .

Figura 4. Coeficiente de velocidad - potencia para análisis de hélices.



Fuente: (Weick, 1926)

Sin embargo, para el desarrollo en nuestro software se transformó esta grafica a una iteración, la cual toma rangos de valores del deslizamiento nominal y a cada

uno de estos se aplican distintas ecuaciones que representan la relación de la función de deslizamiento (V/nD) y el deslizamiento nominal ($1-s$) con el coeficiente de velocidad - potencia Ks' .

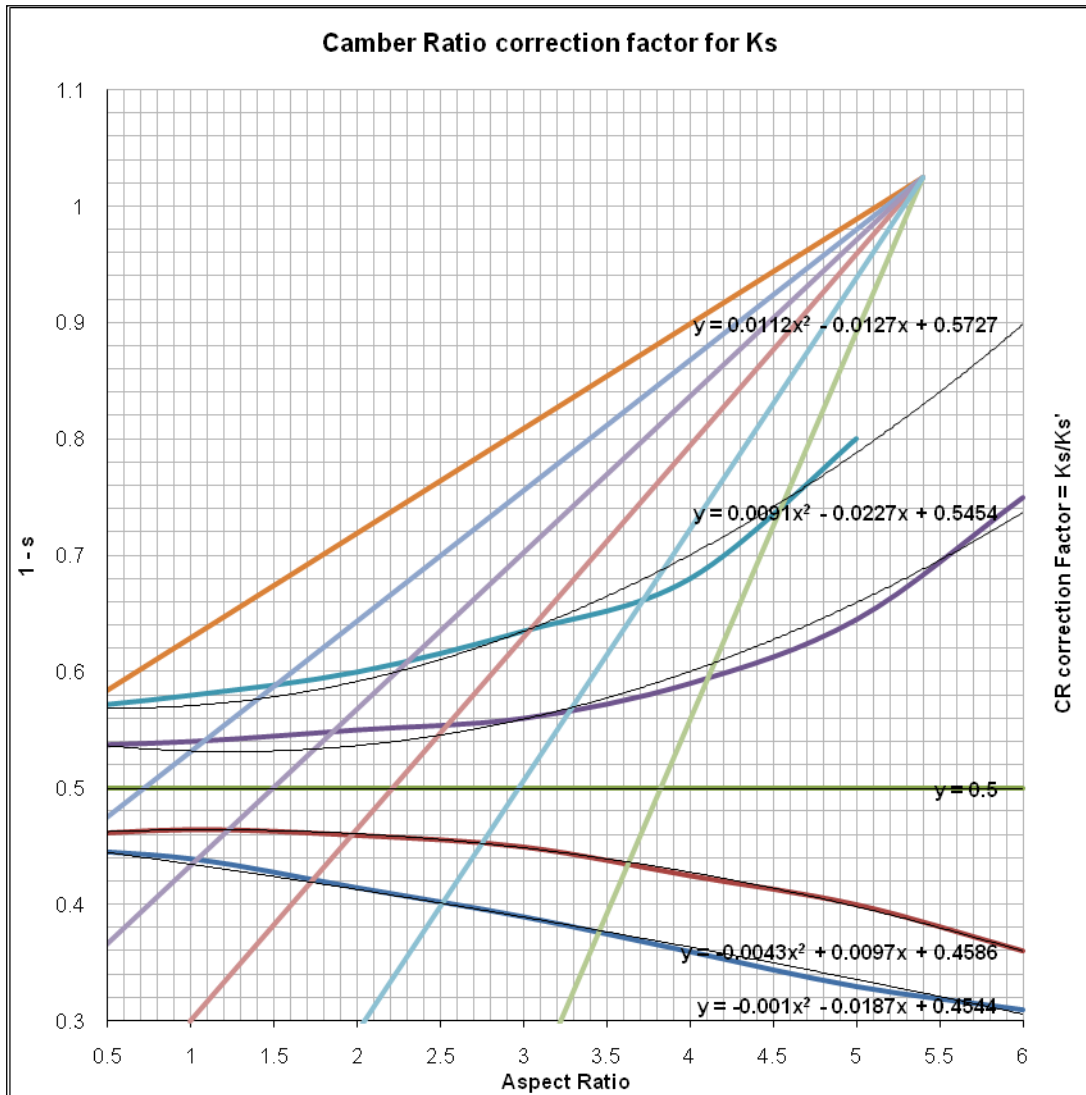
1.1.3 Cálculo de la relación de curvatura corregida. Siguiendo con el desarrollo del diseño, el siguiente paso resulta ser el cálculo de la relación o diferencia entre el Ks y el Ks' , es decir, entre el coeficiente de velocidad - potencia y el coeficiente de velocidad - potencia para una hélice de $AR= 6$ y $CR= 1$. Esta relación da como resultado la corrección total, es decir, la corrección de la relación de aspecto (AR) por la corrección de la relación de curvatura (CR).

$$AR_{\text{corregido}} * CR_{\text{corregido}} = \frac{K_s}{K'_s}$$

Seguido a esto se halla la corrección de la relación de curvatura ($CR_{\text{corregido}}$), la cual se halla mediante la figura 5 que tiene en el eje de ordenadas el deslizamiento nominal ($1-s$), la relación Paso - Diámetro aproximado (P/D_{aprox}) y la relación de curvatura (CR), y en el eje de abscisas la corrección de la relación de curvatura ($CR_{\text{corregido}}$).

Sin embargo, para el desarrollo en nuestro software se transforma esta grafica a una iteración que se divide en 2 bloques. El primero consiste en tomar rangos de valores de la relación Paso - Diámetro aproximado (P/D_{aprox}) y relacionarlos mediante ecuaciones con el deslizamiento nominal ($1-s$). El segundo bloque iterativo relaciona los resultados obtenidos del primer bloque con rangos de la relación de curvatura (CR), que al final me dará la corrección de la relación de curvatura ($CR_{\text{corregido}}$).

Figura 5. (1 - s) Vs. Relación de aspecto.



Fuente: (Weick, 1926)

1.1.4 Cálculo del paso de la hélice. Para dar por finalizado el diseño geométrico se debe calcular el paso (P) de la hélice, el cual brinda un dimensionamiento inicial del propulsor junto con el diámetro.

Para llegar a esto se calcula primero la corrección de la relación de aspecto ($AR_{\text{corregido}}$) mediante la siguiente expresión:

$$AR_{\text{corregido}} = \frac{AR_{\text{corregido}} * CR_{\text{corregido}}}{CR_{\text{corregido}}}$$

Continuando con los cálculos se halla una relación de aspecto final dada por la corrección de la relación de aspecto desarrollada anteriormente, mediante la siguiente fórmula:

$$ARDFijo = 3.8354 * AR_{\text{corregido}}^3 - 5.2327 * AR_{\text{corregido}}^2 + 11.61 * AR_{\text{corregido}} - 4.2082$$

Ahora se debe calcular la relación Paso - Diámetro (P/D) mediante la función de deslizamiento (V/nD) y el deslizamiento nominal ($1-s$), utilizando la siguiente expresión:

$$\frac{P}{D} = \frac{\frac{V}{nD}}{1-s}$$

Para finalizar con el cálculo del paso de la hélice (P),

$$P = D * \frac{P}{D}$$

Con el valor del paso de la hélice conocido se puede seguir con el diseño de este dispositivo, ahora a un nivel más profundo y específico, conocido como diseño complementario.

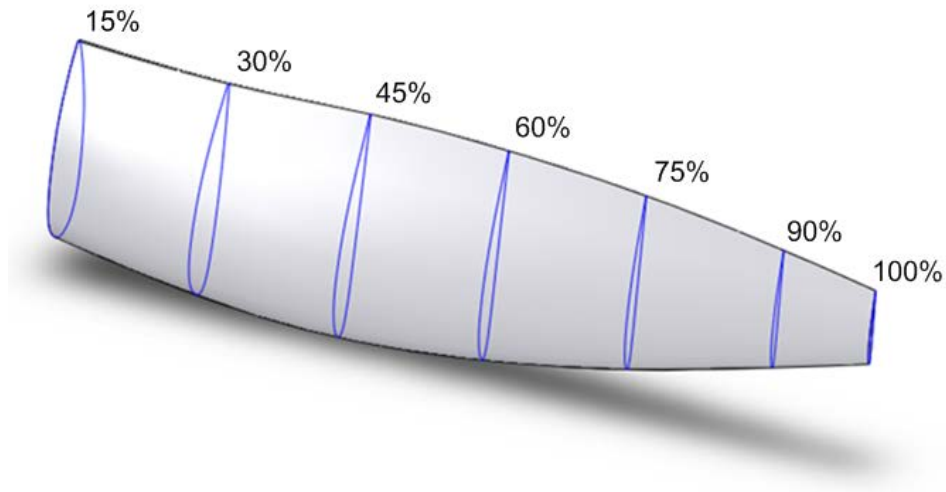
1.2 DISEÑO COMPLEMENTARIO

El desarrollo de este segmento de diseño de la hélice incluye expresiones que dan un dimensionamiento total de la hélice, generalmente por secciones o estaciones a

lo largo de la hélice. Entre estas expresiones se encuentran pasos, ángulos y puntos de cada una de las estaciones o secciones de la hélice.

La hélice se subdivide en 8 estaciones, al 0%, 15%, 30%, 45%, 60%, 75%, 90% y 100%. Donde al 0% se encuentra el centro del cubo de la hélice, al 15% el primer perfil de la hélice como tal y al 100% el perfil final de la hélice. Para un mejor entendimiento véase la figura 6.

Figura 6. Hélice dividida en siete estaciones.

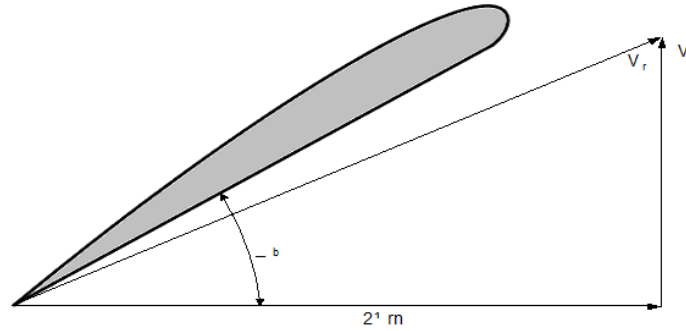


1.2.1 Cálculo de pasos para cada estación. Para llevar a cabo el cálculo de los pasos para cada una de las estaciones y propósitos de la hélice, se utiliza la sección al 75% del radio de la hélice ya que en esta se logran las curvas máximas de torque y empuje, mientras hay una relación constante entre esta estación y el área total de la curva de empuje o torque.

Además de trabajar solo con la estación al 75% del radio de la hélice, se utiliza la relación de aspecto (AR), el diámetro de la hélice en pies y en pulgadas y el paso.

El primero de los términos a calcular es el ángulo de la pala (φ_B), mostrado en la figura 7 y el cual se determina mediante la siguiente expresión:

Figura 7. Representación del ángulo de pala (φ_B).

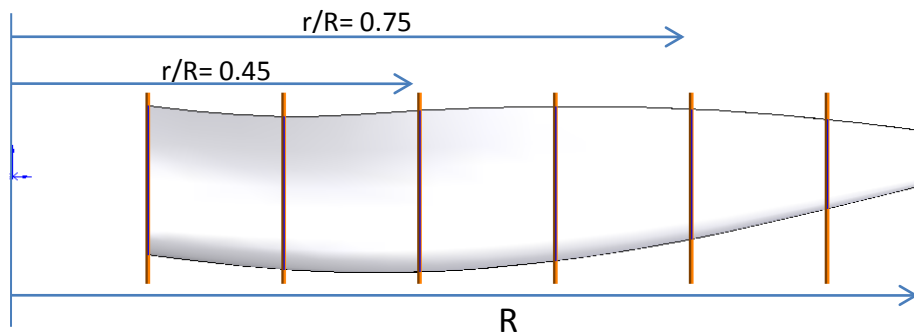


Fuente: (Weick, 1926)

$$\varphi_B = \tan^{-1} \frac{P}{2\pi r}$$

Donde r es el radio o distancia a cualquier estación de la pala. Este término se encuentra representado en la figura 8.

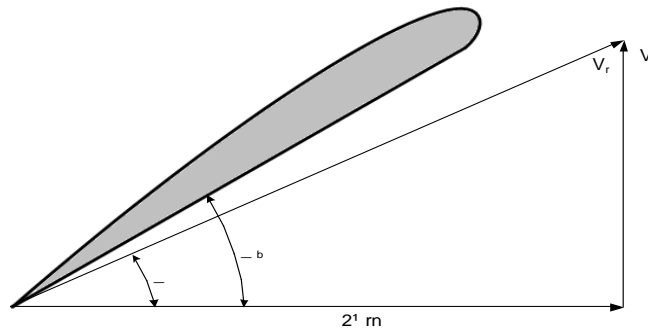
Figura 8. Representación del radio de la pala.



Luego se debe calcular el ángulo que hace la pala con un plano perpendicular al eje de la hélice (φ), mostrado en la figura 16 y desarrollado mediante la siguiente ecuación:

$$\varphi = \tan^{-1} \frac{PPS}{2\pi r * RPS}$$

Figura 9. Representación del ángulo (φ).



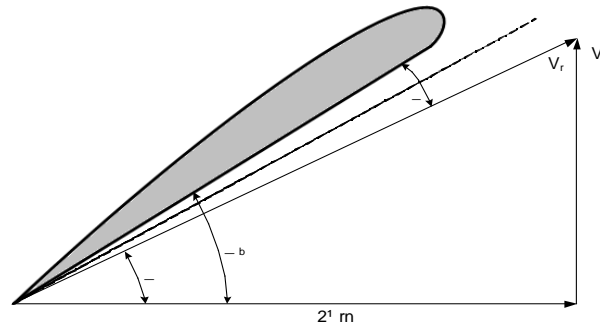
Fuente: (Weick, 1926)

Donde PPS es la velocidad de avance (MPH) en pies por segundo y RPS son las revoluciones máximas desarrolladas por el motor de la hélice en revoluciones por segundo.

Seguido a esto se halla el ángulo de ataque aparente de la sección (α), mostrado en la figura 10 mediante una simple resta entre el ángulo de la pala (φ_B) y el ángulo que hace la pala con un plano perpendicular al eje de la hélice (φ).

$$\alpha = \varphi_B - \varphi$$

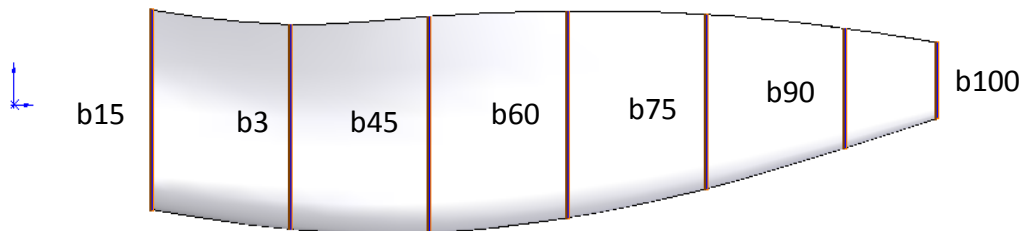
Figura 10. Representación del ángulo de ataque aparente de la sección (a).



Fuente: (Weick, 1926)

Con esto terminamos el direccionamiento de la pala de la hélice, para continuar con el cálculo de los anchos de la pala (b) en cada una de las estaciones, representados en la figura 11.

Figura 11. Representación de los anchos de pala.



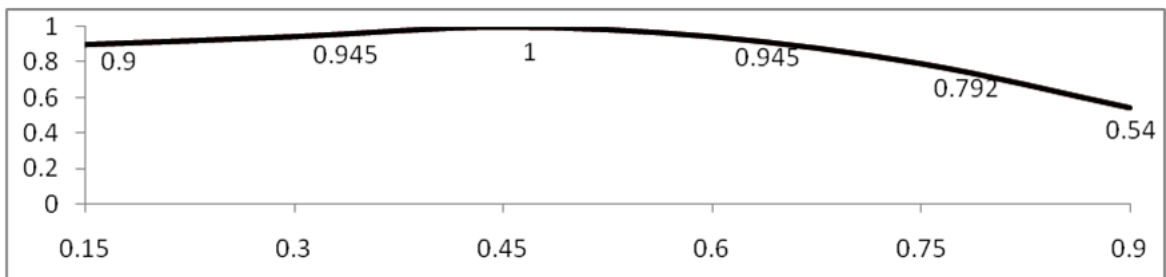
Estos cálculos se realizan basándose en el ancho de la pala al 45% mediante la siguiente expresión:

$$b_{45} = \frac{D}{AR * NoPalas}$$

Donde D es el diámetro de la hélice en pulgadas y $Nopalas$ es el numero de palas de la hélice.

Este término nos permite calcular el valor de los anchos de pala de cada una de las demás estaciones, multiplicando este por distintas constantes dependiendo del ancho de la sección que se quiera calcular. Estas constantes pueden ser observadas en la figura 12.

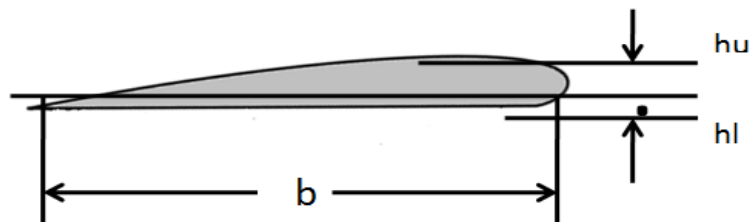
Figura 12. Hélice para propulsor de forma estándar (Standard Navy Plan Form).



Fuente: (Weick, 1926)

Siendo estos términos conocidos, se puede pasar a calcular la máxima curvatura superior (hu) e inferior (hl) para cada una de las estaciones. El significado físico de estas expresiones se muestra en la figura 13.

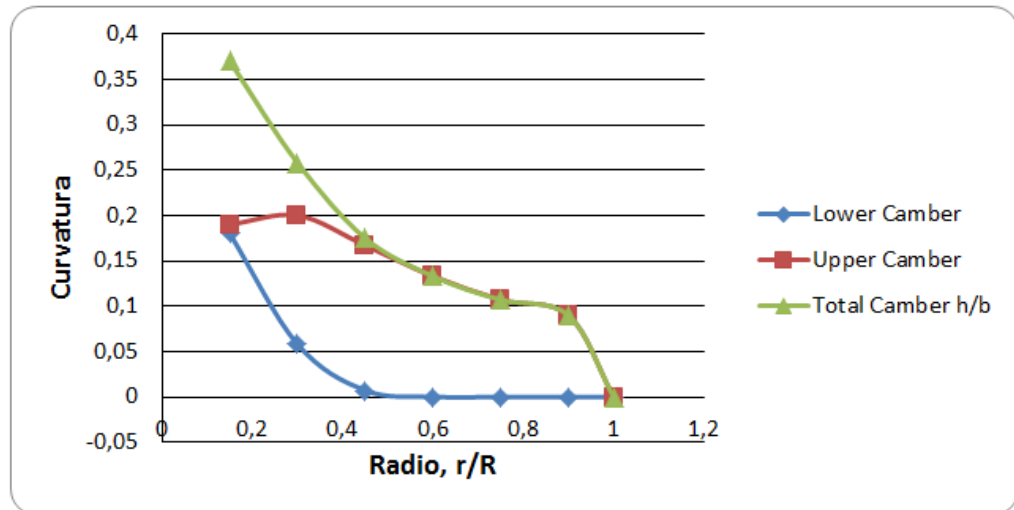
Figura 13. Representación de la curvatura superior e inferior.



Fuente: (Weick, 1926)

La máxima curvatura superior (hu) e inferior (hl) para cada estación depende directamente de la relación de curvatura (CR), del ancho de pala (b) de la sección correspondiente y de una constante particular para cada una de las estaciones y cuya representación se puede observar en la figura 14.

Figura 14. Curvas de curvaturas para un $CR=1$.



Fuente: (Weick, 1926)

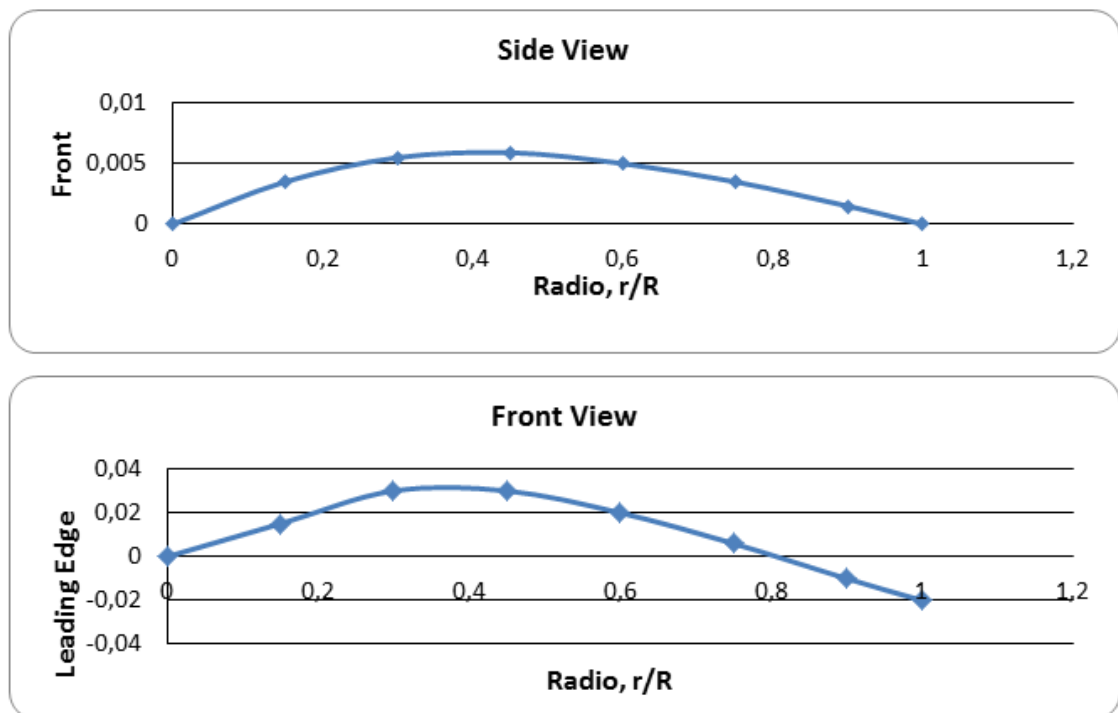
Ahora es necesario hallar los centros de gravedad (CG) de los perfiles de cada una de las estaciones, para los 2 ejes en los que se encuentre representado este. En nuestro caso fueron el eje X y el Y .

Los centros de gravedad de cada una de las secciones en el eje X (CG_x) dependerán de los anchos de pala (b) correspondientes y de una constante particular para cada una de estas derivadas de la forma del perfil de la estación.

Por otro lado, los centros de gravedad en el eje Y (CG_Y) obedecerán a los valores de la máxima curvatura superior (hu) e inferior (hl) correspondientes a cada una de las secciones y a unas constantes que son función de la forma del perfil de la estación. Como complemento a estos centros de gravedad en el eje X y Y, se deben hallar otros centros de gravedad para la vista frontal ($PFCG$) y lateral ($PLCG$) de cada una de las estaciones de la hélice.

Estos dependerán del radio de la hélice y de unas constantes representadas en la figura 15.

Figura 15. Trayectoria de los centros de gravedad de las secciones.

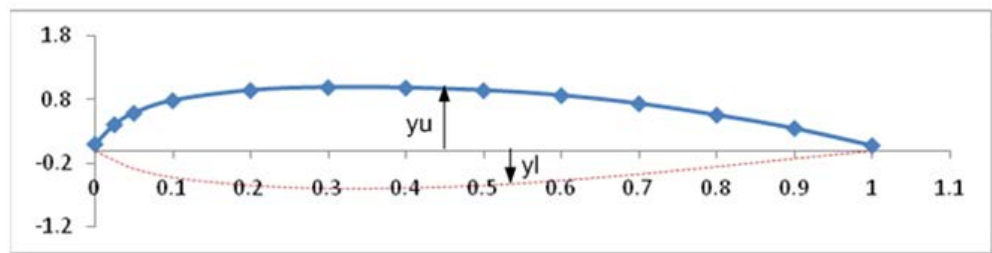


Fuente: (Weick, 1926)

A continuación se hallan los puntos que forman los perfiles normales para cada una de las estaciones. Estos perfiles son denominados normales debido a que no se

encuentran en su posición final, no tienen influencia de ningún tipo de ángulo que los gire respecto a un plano de referencia. Estos puntos definen la forma del perfil, más no su ubicación. Para dar una idea de lo que representan los puntos que se hallan en la dirección X como eje central y en las direcciones Y superior (y_u) e inferior (y_l), se puede observar la figura 16.

Figura 16. Sección de pala estándar de la Marina R.A.F. No. 6 Modificada.



x	0	0.025	0.05	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
Y(x)	0.1	0.41	0.59	0.79	0.95	0.998	0.99	0.95	0.87	0.74	0.56	0.35	0.077

Fuente: (Weick, 1926)

Los puntos en el eje X para cada una de las estaciones dependerán del ancho de pala (b) y del centro de gravedad en X, correspondiente a cada sección. Mientras que los puntos en la ordenada Y superior (y_u) e inferior (y_l) obedecerán a la máxima curvatura superior (h_u) e inferior (h_l), respectivamente, además de los centros de gravedad en Y, correspondientes a cada estación.

Como ejemplo de las ecuaciones utilizadas para determinar estos puntos iniciales, se muestra a continuación un grupo de expresiones que definen la ubicación de un punto en el eje X y en las direcciones Y superior (y_u) e inferior (y_l) para la estación al 15%.

$$X_{15} = b_{15} * 0.01 - CG_{X15}$$

$$yu_{15} = hu_{15} * 0.25 - CG_{Y15}$$

Para continuar con el cálculo de los pasos para cada estación se debe determinar la velocidad para un mejor ángulo de ascenso (V_x).

La velocidad para un mejor ángulo de ascenso (V_x) depende del peso que tiene el dispositivo de vuelo completo (*Peso*), que para nuestro caso será el paramotor, y del área del ala (*Wing area*) que será el área del paracaídas totalmente abierto², ambos en unidades métricas. La ecuación utilizada para llevar a cabo este cálculo es la siguiente:

$$V_x = \sqrt{\frac{Peso}{WingareaS * 16}} * 3.6$$

Con los términos anteriores conocidos es posible determinar el paso para cada una de las estaciones y propósitos de vuelo correspondientes. Estas ecuaciones se encuentran relacionadas entre sí y dependen del paso de la hélice (P), de la velocidad angular máxima (RPM_{max}) y de la velocidad para un mejor ángulo de ascenso (V_x).

1.2.2 Cálculo de ángulo de pala para cada estación. El cálculo del ángulo de pala (φ_B) para cada estación es un procedimiento similar al realizado con los pasos de cada una de las estaciones. Así como se definió el paso en cada uno de los perfiles correspondientes a cada estación, se debe hacer lo mismo con el ángulo de pala. En

² www.nova-wings.com

la figura 7 se observa la representación del ángulo de pala para una estación determinada.

La expresión utilizada para calcular el ángulo de pala al 15% es la siguiente:

$$\varphi_{B15} = \frac{Paso_{15}}{D * \pi * 0.15}$$

Las ecuaciones manejadas para las demás estaciones varían al utilizar el paso y la constante correspondiente a la estación.

1.2.3 Cálculo de puntos de cada estación. Finalmente para terminar con nuestro diseño se deben hallar los puntos que formaran cada estación. Tenemos un total de 28 puntos para cada una de las secciones, los primeros 14 conformando la curva superior y los otros 14 la curva inferior. Cada uno de estos puntos tienen su respectiva ubicación en el eje X y el Y.

Estos puntos dependen de los puntos hallados anteriormente que formaban el perfil normal y del ángulo de pala (φ_B) para cada estación, además de estar relacionados entre sí.

Como ejemplo de las ecuaciones utilizadas para determinar los puntos, se muestra el siguiente par de expresiones que define un punto de la curva superior del perfil de la estación al 15%.

$$XU_{15} = \frac{\cos \left[\left(\tan^{-1} \frac{yu_{15}}{X_{15}} \right) - \varphi_{B15} \right] * X_{15}}{\cos \left[\left(\tan^{-1} \frac{yu_{15}}{X_{15}} \right) \right]}$$

$$YU_{15} = \tan \left[\left(\tan^{-1} \frac{yu_{15}}{X_{15}} \right) - \varphi_{B15} \right] * XU_{15}$$

Los puntos generados en el eje X se deben girar según el centro de gravedad de la vista frontal y los generados en el eje Y se giran según el centro de gravedad de la vista lateral. A continuación se muestran las ecuaciones correspondientes.

$$XU_{R15} = XU_{15} - PFCG_{15}$$

$$YU_{R15} = YU_{15} + PLCG_{15}$$

Por último se tiene en cuenta el sentido de giro de la hélice. Para un giro antihorario los puntos hallados inmediatamente anterior se mantienen iguales, mientras que para un giro horario (pala ataca el aire en sentido contrario) los puntos cambian su ubicación con respecto a un plano XY, al ser multiplicados por -1.

ANEXO C. Nomenclatura utilizada por Weick en el diseño de hélices para ultralivianos

D	Diámetro de la Hélice [Ft]
P	Paso geométrico de la hélice [Ft]
r	Radio de alguna de sección de la hélice [Ft]
b	Ancho de alguna sección de la hélice [Ft]
c	Máximo ancho de la hélice [Ft]
B	Número de palas de la hélice
Q	Torque de la hélice [Ft-lb]
T	Empuje de la hélice [lb]
Qc	Coficiente de torque en cualquier sección de la hélice
Tc	Coficiente de empuje en cualquier sección de la hélice
V	Velocidad de avance [Ft/s]
MPH	Velocidad de avance [mi/h]
n	Revoluciones por segundo de la hélice
N	Revoluciones por minuto de la hélice
BHP	Potencia de freno del motor [HP]
CHP	Potencia calculada considerando el factor de forma. Difiere del BHP Dependiendo del Tip speed, la interferencia del fuselaje, y la distorsión de la hélice.
THP	Potencia de Empuje [HP]
QHP	Potencia de Torque [HP]
η	Eficiencia
$\varphi\beta$	Ángulo pala en grados
φ	Ángulo que el paso de la pala hace con el plano perpendicular del eje de la hélice [atan($V_{rn}/2\pi$)]
α	Ángulo de ataque aparente de la sección = $\varphi\beta - \varphi$
ϵ	Corrección de interferencia para el ángulo de ataque
α'	Ángulo de ataque corregido = $\alpha - \epsilon$
CL	Coficiente de elevación absoluto de la superficie de sustentación a el ángulo de ataque α
δCL	Corrección de interferencia para el coeficiente de elevación
C'L	Coficiente de elevación corregido = $CL - \delta CL$

L/D	Relación elevación/arrastre de la sección de perfil aerodinámico
hU	Máxima curvatura superior de la sección
hU/hL	Relaciones estándar de la pala
hL	Máxima curvatura inferior de la sección
h/b	Relación de curvatura de cualquier sección
γ	Ángulo entre la fuerza resultante en la pala del elemento y una línea perpendicular a V_r
ρ	Densidad de la masa de aire. Ésta debe ser tomada como 0.00237 para el nivel del mar y una atmósfera estándar
Tip speed	La distancia recorrida por el extremo de la hélice por unidad de tiempo en un plano de rotación $\text{Tip speed} = \pi n D = 0.0524 n D$ [Ft/seg]
AR	Relación de aspecto $AR = D/c_b$
CR	Curvatura estándar de la pala de la hélice como un todo. Esta es la relación de espesor de la pala completa de la hélice respecto de la de una pala estándar, la variación del espesor a lo largo del radio es la misma para todas las palas.
CP	Coefficiente de potencia de la hélice (adimensional)
K_s	Coefficiente de velocidad-potencia
V/nD	Función de deslizamiento
s	Deslizamiento nominal de una hélice basado en la relación $1-s = V/nP$
P/D	Relación paso-diámetro
$K's$	Coefficiente de velocidad-potencia para hélices de $AR=6$ y $CR=1$

Fuente: (Weick, 1926)

ANEXO D. Programación de API's de integración SolidWorks en lenguaje VB para el modelado de hélices para ultraliviano

(Tomado de *UIS-Propeller V2.0*)

```
' =====
' SOLIDWORKS APIS
' =====
' Utilización de APIS para el modelado de la hélice SolidWorks:

' NOTA: para el modelado exitoso de la hélice es necesario ir menú Proyecto,
agregar referencia, pestaña COM y luego las librerías de SolidWorks necesarias,
si no se sabe cuáles agregar, se pueden incluir todas las que hagan relación a
SolidWorks y eliminar posteriormente las que no se estén utilizando.

' Declaración de variables de SolidWorks:
' -----
ToolStripStatusLabel2.Text = "Modelando en Solidworks"

Dim swApp As SldWorks.SldWorks
Dim PART As SldWorks.ModelDoc2
swApp = New SldWorks.SldWorks()
swApp = GetObject("", "SldWorks.Application")
swApp.Visible = True
swApp.NewPart()
PART = swApp.ActiveDoc

' Procedimiento para establecer el sistema de unidades a ser utilizado por
SolidWorks No es necesario realizar la conversión de unidades ya que los códigos
utilizados en los macros operan en el sistema métrico de unidades, por tanto
SolidWorks realiza automáticamente la conversión al establecer el sistema de
unidades en el cual se desea mostrar el sólido.
Select Case Unidades
    Case 0 'S. Métrico
PART.Extension.SetUserPreferenceInteger(swUserPreferenceIntegerValue_e.swUnitSys
tem, 0, swUnitSystem_e.swUnitSystem_MKS)
```

```

        Case 1 'S. Americano
PART.Extension.SetUserPreferenceInteger(swUserPreferenceIntegerValue_e.swUnitSystem, 0, swUnitSystem_e.swUnitSystem_IPS)

PART.Extension.SetUserPreferenceInteger(swUserPreferenceIntegerValue_e.swUnitsLinearFractionDenominator, 0, 0)

PART.Extension.SetUserPreferenceToggle(swUserPreferenceToggle_e.swUnitsLinearFeetAndInchesFormat, 0, False)

PART.Extension.SetUserPreferenceInteger(swUserPreferenceIntegerValue_e.swUnitsDualLinearFractionDenominator, 0, 0)

PART.Extension.SetUserPreferenceToggle(swUserPreferenceToggle_e.swUnitsDualLinearFeetAndInchesFormat, 0, False)
End Select

' Barra de progreso en cero
' -----

ToolStripProgressBar1.Value = 0

'Modelado de los perfiles en cada estación de la pala

For i As Double = 1 To nel
    ToolStripProgressBar1.Value = ToolStripProgressBar1.Value + 2
    If i = 2 Or i = 3 Then
        nc = ncc ' los perfiles a 30% y 45% tienen 28 puntos
    Else
        nc = 26 ' las demás estaciones tienen 27 puntos
    End If
    PART.InsertCurveFileBegin()
    For j As Double = 0 To nc
        PART.InsertCurveFilePoint(0.0254 * XXULg(i, j), 0.0254 * YYULg(i, j), rc(i))
    Next

```

```

PART.InsertCurveFileEnd()
Next

'Insertar planos en cada estacion
planor = {"", "Plano1", "Plano2", "Plano3", "Plano4", "Plano5", "Plano6",
"Plano7"}
curvar = {"", "Curva1", "Curva2", "Curva3", "Curva4", "Curva5", "Curva6",
"Curva7"}

For i = 1 To nel
    ToolStripProgressBar1.Value = ToolStripProgressBar1.Value + 8
    If i = 2 Or i = 3 Then
        nc = ncc ' los perfiles a 30% y 45% tienen 28 puntos
    Else
        nc = 26 ' las demás estaciones tienen 27 puntos
    End If
    PART.Extension.SelectByID2("Alzado", "PLANE", 0, 0, 0, True, 0, Nothing, 0)
    PART.CreatePlaneAtOffset3(rc(i), False, True)
    PART.ClearSelection2(True)
    PART.Extension.SelectByID2(planor(i), "PLANE", XXULg(i, 13) * 0.0254, -
YYULg(i, 13) * 0.0254, rc(i), False, 0, Nothing, 0)
    '-----
    PART.InsertSketch2(True)
    PART.Extension.SelectByID2(curvar(i), "REFERENCECURVES", XXULg(i, 14) *
0.0254, YYULg(i, 14) * 0.0254, rc(i), False, 0, Nothing, 0)
    PART.SketchUseEdge2(False)
    PART.ClearSelection2(True)
    PART.CreateLine2(XXULg(i, 0) * 0.0254, YYULg(i, 0) * 0.0254, rc(i), XXULg(i,
nc) * 0.0254, YYULg(i, nc) * 0.0254, rc(i))
    PART.ClearSelection2(True)
    PART.InsertSketch2(True)
Next

```

' Selección de curvas para el recubrimiento

```
- PART.Extension.SelectByID2("Croquis1", "SKETCH", XXULg(1, 14) * 0.0254, -  
  YYULg(1, 14) * 0.0254, rc(1), True, 1, Nothing, 0)  
- PART.Extension.SelectByID2("Croquis2", "SKETCH", XXULg(2, 14) * 0.0254, -  
  YYULg(2, 14) * 0.0254, rc(2), True, 0, Nothing, 0)  
- PART.Extension.SelectByID2("Croquis3", "SKETCH", XXULg(3, 14) * 0.0254, -  
  YYULg(3, 14) * 0.0254, rc(3), True, 0, Nothing, 0)  
- PART.Extension.SelectByID2("Croquis4", "SKETCH", XXULg(4, 14) * 0.0254, -  
  YYULg(4, 14) * 0.0254, rc(4), True, 0, Nothing, 0)  
- PART.Extension.SelectByID2("Croquis5", "SKETCH", XXULg(5, 14) * 0.0254, -  
  YYULg(5, 14) * 0.0254, rc(5), True, 0, Nothing, 0)  
- PART.Extension.SelectByID2("Croquis6", "SKETCH", XXULg(6, 14) * 0.0254, -  
  YYULg(6, 14) * 0.0254, rc(6), True, 0, Nothing, 0)  
- PART.Extension.SelectByID2("Croquis7", "SKETCH", XXULg(7, 14) * 0.0254, -  
  YYULg(7, 14) * 0.0254, rc(7), True, 0, Nothing, 0)  
- PART.ClearSelection2(True)  
- PART.Extension.SelectByID2("Croquis1", "SKETCH", XXULg(1, 14) * 0.0254, -  
  YYULg(1, 14) * 0.0254, rc(1), False, 1, Nothing, 0)  
- PART.Extension.SelectByID2("Croquis2", "SKETCH", XXULg(2, 14) * 0.0254, -  
  YYULg(2, 14) * 0.0254, rc(2), True, 1, Nothing, 0)  
- PART.Extension.SelectByID2("Croquis3", "SKETCH", XXULg(3, 14) * 0.0254, -  
  YYULg(3, 14) * 0.0254, rc(3), True, 1, Nothing, 0)  
- PART.Extension.SelectByID2("Croquis4", "SKETCH", XXULg(4, 14) * 0.0254, -  
  YYULg(4, 14) * 0.0254, rc(4), True, 1, Nothing, 0)  
- PART.Extension.SelectByID2("Croquis5", "SKETCH", XXULg(5, 14) * 0.0254, -  
  YYULg(5, 14) * 0.0254, rc(5), True, 1, Nothing, 0)  
- PART.Extension.SelectByID2("Croquis6", "SKETCH", XXULg(6, 14) * 0.0254, -  
  YYULg(6, 14) * 0.0254, rc(6), True, 1, Nothing, 0)  
- PART.Extension.SelectByID2("Croquis7", "SKETCH", XXULg(7, 14) * 0.0254, -  
  YYULg(7, 14) * 0.0254, rc(7), True, 1, Nothing, 0)  
- PART.FeatureManager.InsertProtrusionBlend(False, True, False, 1, 6, 6, 1, 1,  
  True, True, False, 0, 0, 0, True, True, True)
```

```

'procedimiento para ocultar los planos, croquis y curvas de referencia
croquisr = {"", "Croquis1", "Croquis2", "Croquis3", "Croquis4", "Croquis5",
"Croquis6", "Croquis7"}

For i = 1 To nel
PART.Extension.SelectByID2(planor(i), "PLANE", 0, 0, 0, False, 0, Nothing, 0)
PART.BlankRefGeom()
PART.Extension.SelectByID2(croquisr(i), "SKETCH", 0, 0, 0, False, 0, Nothing, 0)
PART.BlankSketch()
PART.Extension.SelectByID2(curvar(i), "REFERENCECURVES", 0, 0, 0, False, 0,
Nothing, 0)
PART.BlankRefGeom()
Next
ToolStripProgressBar1.Value = 100

PART.ShowNamedView2("*Trimétrica", 8)
PART.ViewZoomtofit2()
PART.ViewDisplayShaded()

ToolStripProgressBar1.Value = 100


If ToolStripProgressBar1.Value = 100 Then
ToolStripStatusLabel2.Text = "Modelado en SolidWorks Exitoso!"
End If
' Una vez se ha construido la hélice en SolidWorks, se activan las opciones para
construcción del cubo
ComboBoxCubo.Enabled = True
TextBoxespesorcubo.Enabled = True
TextBoxdiametrocubo.Enabled = True
TextBoxdiametrocentral.Enabled = True
TextBoxdiametroapernado.Enabled = True
TextBoxdiametroperno.Enabled = True
TextBoxNumeroPernos.Enabled = True
ButtonCubo.Enabled = True
End Sub

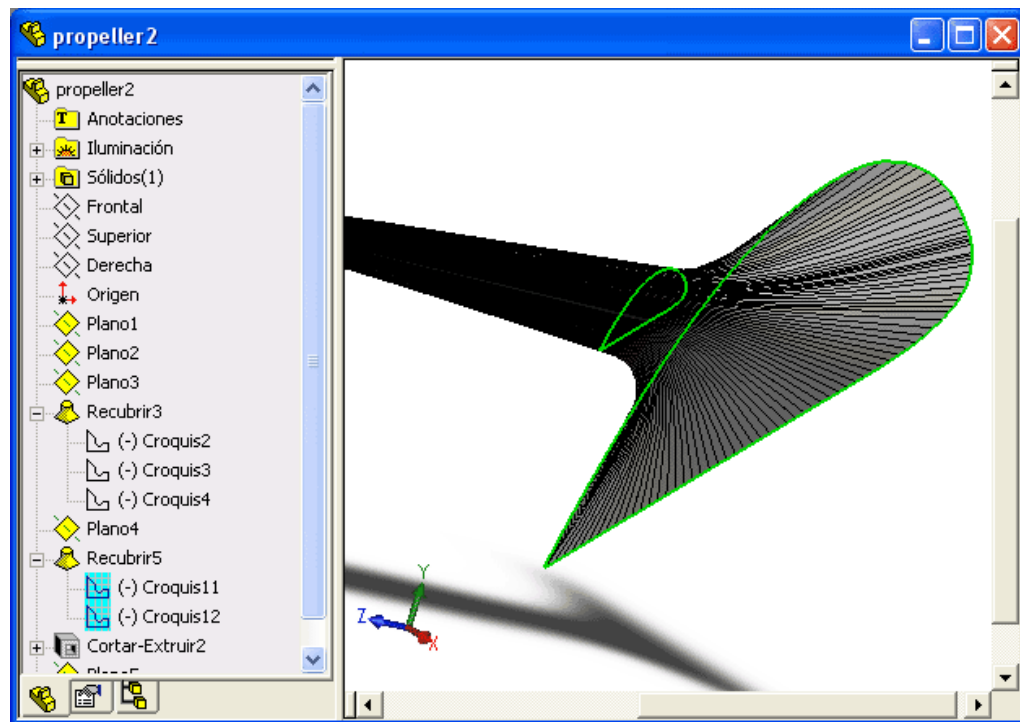
```

**ANEXO E. Estrategia de modelado del cubo de la hélice implementada en UIS-
*Propeller V1.0***

Modelado del cubo de la hélice en SolidWorks.

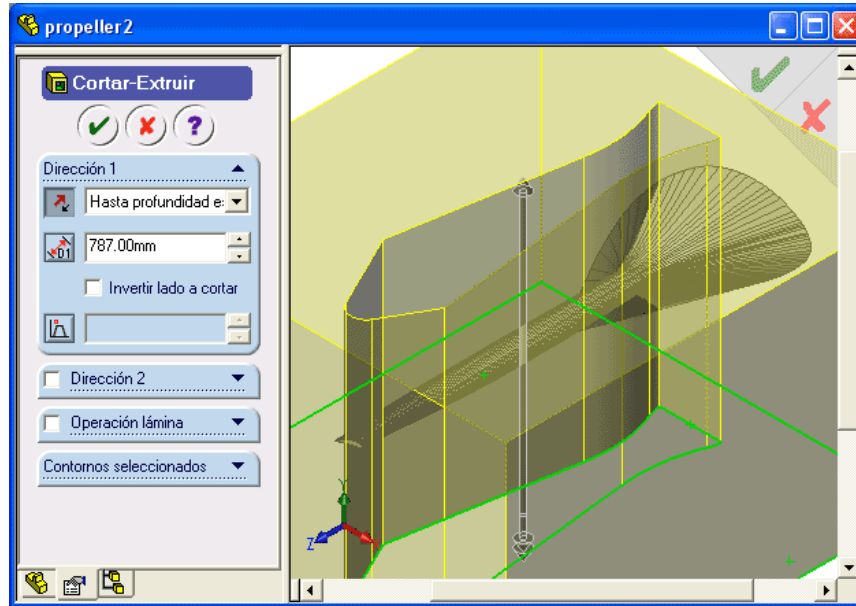
*Wooden propeller style modeling tutorial for CNC machining finality.
Jose Luis Cortés, 2004.*

Duplicar el primer boceto al plano frontal y hacerlo a 5 veces de escala con respecto al tamaño original. Luego usar la herramienta Recubrir  para hacer la raíz de la hélice.

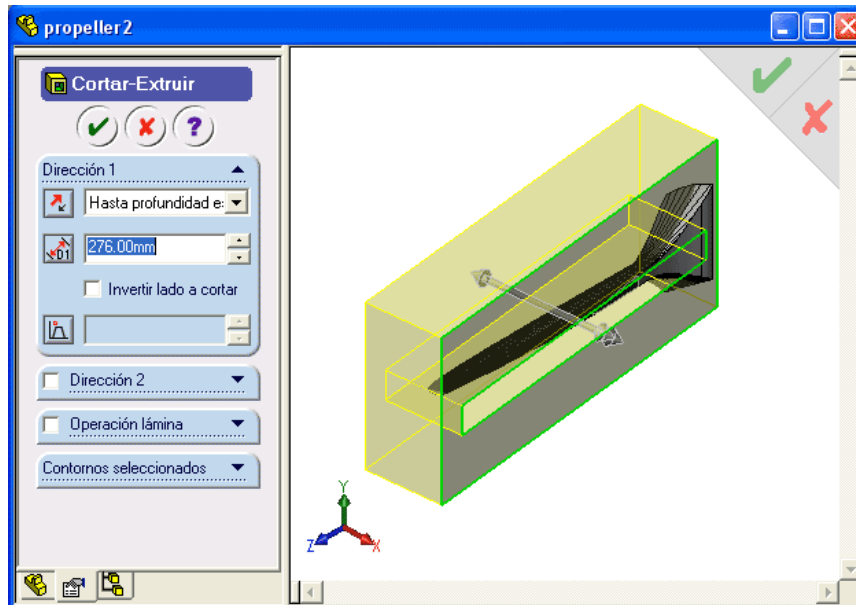


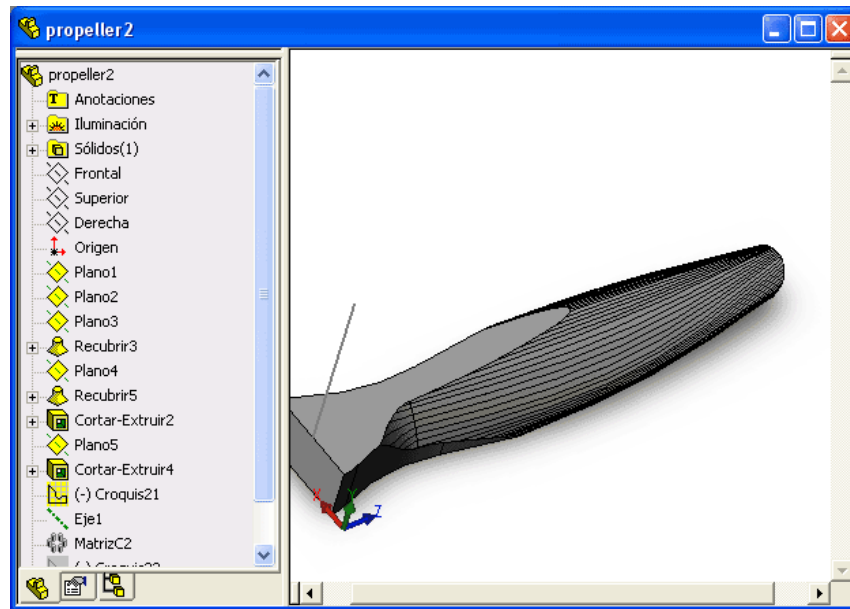
Recortar la raíz a la forma definitiva.


Se dibuja la matriz de corte deseada en la vista superior.

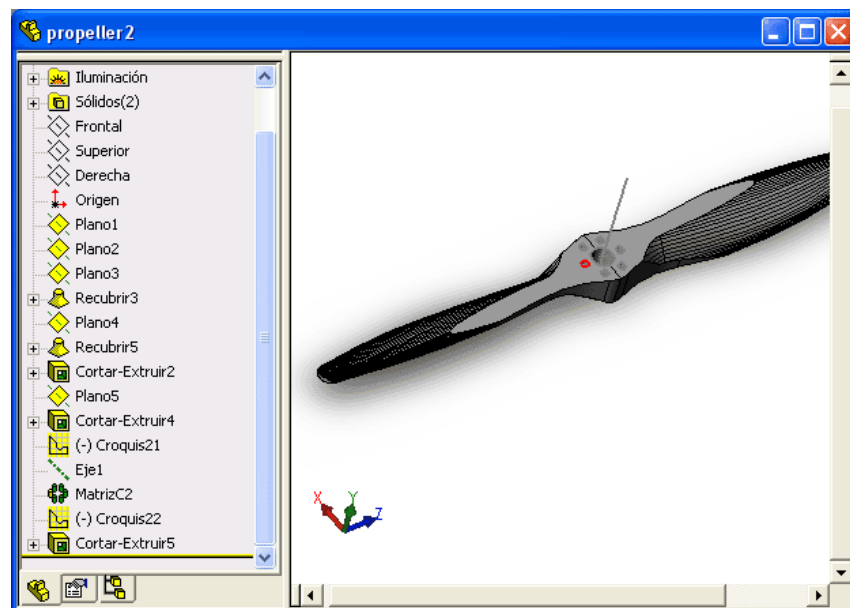


Repetir la misma operación con el plano lateral. Esta operación dará la forma final de la hélice.





La parte es terminada con una matriz circular . Tenemos una bella es hélice.



ANEXO F. Programación de API's de integración SolidWorks en lenguaje VB para la construcción del cubo de hélices para ultraliviano

(Tomado de UIS-Propeller V2.0)

' Construcción del cubo en SolidWorks

```
Dim EspesorCubo As Double
Dim DiametroCubo As Double
Dim Diametrocentro As Double
Dim Diametroperno As Double
Dim Diametroapernado As Double
Dim NumeroPernos As Double

Dim swApp As SldWorks.SldWorks
Dim PART As SldWorks.ModelDoc2
swApp = New SldWorks.SldWorks
swApp.Visible = True
PART = swApp.ActiveDoc

EspesorCubo = Cdbl(TextBoxespesorcubo.Text)
DiametroCubo = Cdbl(TextBoxdiametrocubo.Text)
Diametrocentro = Cdbl(TextBoxdiametrocentral.Text)
Diametroperno = Cdbl(TextBoxdiametroperno.Text)
Diametroapernado = Cdbl(TextBoxdiametroapernado.Text)
NumeroPernos = Cdbl(TextBoxNumeroPernos.Text)

ToolStripProgressBar1.Value = 0
ToolStripStatusLabel2.Text = "Adicionando Cubo a la hélice"
```

' Extensión de la pála hasta el eje central de la hélice

```
PART.Extension.SelectByID2("Alzado", "PLANE", 0, 0, 0, False, 0,
Nothing, 0)
PART.SketchManager.InsertSketch(True)
PART.ClearSelection2(True)
```

```

PART.SketchManager.CreateEllipse(0, 0, 0, (DiametroCubo / 2) * 0.0254,
0, 0, 0, (EspesorCubo / 2) * 0.0254, 0)
PART.ClearSelection2(True)
PART.SketchManager.InsertSketch(True)

ToolStripProgressBar1.Value = ToolStripProgressBar1.Value + 10

PART.ClearSelection2(True)
PART.Extension.SelectByID2("Croquis1", "SKETCH", 0, 0, 0, True, 0,
Nothing, 0)
PART.Extension.SelectByID2("Croquis1", "SKETCH", 0, 0, 0, True, 0,
Nothing, 0)
PART.Extension.SelectByID2("Croquis8", "SKETCH", (DiametroCubo / 2) *
0.0254, 0, 0, True, 0, Nothing, 0)
PART.ClearSelection2(True)
PART.Extension.SelectByID2("Croquis1", "SKETCH", 0, 0, 0, False, 1,
Nothing, 0)
PART.Extension.SelectByID2("Croquis8", "SKETCH", (DiametroCubo / 2) *
0.0254, 0, 0, True, 1, Nothing, 0)
PART.FeatureManager.InsertProtrusionBlend(False, True, False, 1, 0, 0,
1, 1, True, True, False, 0, 0, 0, True, True, True)

```

```

ToolStripProgressBar1.Value = ToolStripProgressBar1.Value + 20

```

' Construcción de cubo de la hélice

```

PART.Extension.SelectByID2("Planta", "PLANE", 0, 0, 0, False, 0,
Nothing, 0)
PART.SketchManager.InsertSketch(True)
PART.ClearSelection2(True)
PART.SketchManager.CreateCircle(0.0#, 0.0#, 0.0#, (DiametroCubo / 2) *
0.0254, 0, 0.0#)
PART.ClearSelection2(True)
PART.Extension.SelectByID2("Arc1", "SKETCHSEGMENT", 0, 0, 0, False, 0,
Nothing, 0)

```

```

PART.FeatureManager.FeatureExtrusion(False, False, False, 0, 0,
(EspesorCubo / 2) * 0.0254, (EspesorCubo / 2) * 0.0254, False, False, False,
False, 0, 0, False, False, False, False, True, True, True)
PART.SelectionManager.EnableContourSelection = False
'-----
ToolStripProgressBar1.Value = ToolStripProgressBar1.Value + 10

```

' Orificio eje central

```

PART.Extension.SelectByID2("Planta", "PLANE", 0, 0, 0, False, 0,
Nothing, 0)
PART.SketchManager.InsertSketch(True)
PART.ClearSelection2(True)
PART.SketchManager.CreateCircle(0.0#, 0.0#, 0.0#, (Diametrocentro / 2) *
0.0254 * Math.Cos(Math.PI / 12), 0, (Diametrocentro / 2) * 0.0254 *
Math.Sin(Math.PI / 12))
PART.ClearSelection2(True)
PART.SketchManager.InsertSketch(True)
PART.ClearSelection2(True)
PART.Extension.SelectByID2("Croquis10", "SKETCH", 0, 0, 0, False, 0,
Nothing, 0)
PART.FeatureManager.FeatureCut(False, False, False, 1, 1, 0.01, 0.01,
False, False, False, False, 0, 0, False, False, False, False, True, True)
PART.SelectionManager.EnableContourSelection = False
'-----
ToolStripProgressBar1.Value = ToolStripProgressBar1.Value + 10

```

' Matriz circular Número de palas de la hélice

```

PART.Extension.SelectByID2("Alzado", "PLANE", 0, 0, 0, True, 0, Nothing, 0)
PART.Extension.SelectByID2("Vista lateral", "PLANE", 0, 0, 0, True, 0,
Nothing, 0)
PART.InsertAxis2(True)

```

```

PART.Extension.SelectByID2("Cortar-Extruir1", "SOLIDBODY", 0, 0, 0,
True, 0, Nothing, 0)
PART.ClearSelection2(True)
PART.Extension.SelectByID2("Eje1", "AXIS", 0, 0, 0, False, 1, Nothing,
0)
PART.Extension.SelectByID2("Cortar-Extruir1", "SOLIDBODY", 0, 0, 0,
True, 256, Nothing, 0)
PART.FeatureManager.FeatureCircularPattern2(NumerodePalasB, (2 * Math.PI
/ NumerodePalasB), False, "NULL", False)

ToolStripProgressBar1.Value = ToolStripProgressBar1.Value + 20

' Orificios para los pernos

PART.InsertSketch2(True)
PART.Extension.SelectByID2("", "FACE", 0, 0, (EspesorCubo / 2) * 0.0254,
False, 0, Nothing, 0)
PART.ClearSelection2(True)
PART.SketchManager.CreateCircle((Diametroapernado / 2) * 0.0254, 0, 0,
((Diametroapernado / 2) * 0.0254 + (Diametroperno / 2) * 0.0254), 0, 0)
PART.InsertSketch2(True)
PART.ClearSelection2(True)
PART.Extension.SelectByID2("Croquis11", "SKETCH", 0, 0, 0, False, 0,
Nothing, 0)
PART.FeatureManager.FeatureCut(True, False, False, 1, 0, 0.01, 0.01,
False, False, False, False, 0, 0, False, False, False, False, True, True)

ToolStripProgressBar1.Value = ToolStripProgressBar1.Value + 10

PART.SetUserPreferenceToggle(7, True)
PART.Extension.SelectByID2("", "AXIS", 0, 0, 0, True, 0, Nothing, 0)
PART.ClearSelection2(True)
PART.Extension.SelectByID2("Cortar-Extruir2", "BODYFEATURE", 0, 0, 0,
False, 4, Nothing, 0)
PART.Extension.SelectByID2("", "AXIS", 0, 0, 0, True, 1, Nothing, 0)

```

```
PART.FeatureManager.FeatureCircularPattern2(NumeroPernos, 2 * (Math.PI)
/ NumeroPernos, False, "NULL", True)
```

```
ToolStripProgressBar1.Value = 100
```

```
If ToolStripProgressBar1.Value = 100 Then
```

```
    ToolStripStatusLabel2.Text = "Cubo adicionado Exitosamente!"
```

```
End If
```

```
PART.ShowNamedView2("Isometric", 7)
```

```
PART.ViewZoomtofit2()
```

```
' Ocultar los ejes generados en cada uno de los orificios del cubo de la hélice
```

```
PART.SetUserPreferenceToggle(5, False)
```

```
PART.SetUserPreferenceToggle(4, False)
```

```
PART.SetUserPreferenceToggle(7, False)
```

```
PART.SetUserPreferenceToggle(6, False)
```

```
PART.SetUserPreferenceToggle(195, False)
```

```
PART.SetUserPreferenceToggle(197, False)
```

```
PART.SetUserPreferenceToggle(219, False)
```

```
PART.SetUserPreferenceToggle(196, False)
```

```
PART.SetUserPreferenceToggle(13, False)
```

```
PART.SetUserPreferenceToggle(19, False)
```

```
End Sub
```

ANEXO G. Nomenclatura de variables empleadas por *UIS-Propeller* en el diseño de hélices para aerogeneradores de eje horizontal

1. Determinación de parámetros básicos			
Variable	Nomenclatura	Unidades	Nombre
Pe	Pe	[W]	Potencia nominal
ρ	Rho	[kg/m ³]	Densidad del aire
η	ng	[-]	Rendimiento de la turbina
Vw	Vw	[m/s]	Velocidad del viento
Ω	rpm	[rpm]	Velocidad angular en revoluciones por minuto
Ω_{rad}	rads	[1/s]	Velocidad angular en radianes por segundo
B	B	[-]	Número de palas
Cp	Cp	[-]	Coefficiente de potencia
λ_d	TSRd	[-]	Rangos de valores de relación de velocidad de punta en función del
λ_{up}	TSRup	[-]	número de palas
λ_c	TSRc	[-]	Relación de velocidad de punta calculada en la iteración
λ	TSR	[-]	Relación de velocidad de punta
n	n	[-]	relación de transmisión
Rr	Rr	[m]	Radio del rotor
error	er	[-]	error de iteración valor de Cp
2. Definición de la forma de la Pala			
2.1 Cálculo del Número de Reynolds			
Variable	Nomenclatura	Unidades	Nombre
r_root	rroot	[m]	Radio de la raíz de la pala
Vr_root	Vwroot	[m/s]	Velocidad del viento en la raíz de la hélice
ϕ_{rad_root}	phirroot	[rad]	Ángulo de flujo en la raíz de la hélice
c_root	croot	[m]	Cuerda en la raíz de la pala
T	Ta	[°C]	Temperatura Ambiente
T	TaK	[K]	Temperatura Ambiente en grados Kelvin
T_0	Tr	[K]	Temperatura de referencia
Cs	Cs	[K]	Constante de Sutherland
μ_o	Muo	[Pa*s]	Viscosidad de referencia a Temperatura de referencia
μ	Mu	[Pa*s]	Viscosidad dinámica
v	Nu	[m ² /s]	Viscosidad cinemática
Rer	Re	[-]	Número de Reynolds
2.2 Selección del Perfil			

Variable	Nomenclatura	Unidades	Nombre
a2509	a2509		Relación (Clmax/Clmin) a un número de Reynolds específico
b2509	b2509		Relación (L/D)min a un número de Reynolds específico
α 2509	alpha2509		Ángulo de cero Elevación (lift) para el perfil
pun2509	pun2509		puntuación obtenida por el perfil
a4412	a4412		Relación (Clmax/Clmin) a un número de Reynolds específico
b4412	b4412		Relación (L/D)min a un número de Reynolds específico
α 4412	alpha4412		Ángulo de cero Elevación (lift) para el perfil
pun4412	pun4412		puntuación obtenida por el perfil
a4415	a4415		Relación (Clmax/Clmin) a un número de Reynolds específico
b4415	b4415		Relación (L/D)min a un número de Reynolds específico
α 4415	alpha4415		Ángulo de cero Elevación (lift) para el perfil
pun4415	pun4415		puntuación obtenida por el perfil
a63215	a63215		Relación (Clmax/Clmin) a un número de Reynolds específico
b63215	b63215		Relación (L/D)min a un número de Reynolds específico
α 63215	alpha63215		Ángulo de cero Elevación (lift) para el perfil
pun63215	pun63215		puntuación obtenida por el perfil
a63415	a63415		Relación (Clmax/Clmin) a un número de Reynolds específico
b63415	b63415		Relación (L/D)min a un número de Reynolds específico
α 63415	alpha63415		Ángulo de cero Elevación (lift) para el perfil
pun63415	pun63415		puntuación obtenida por el perfil
a63421	a63421		Relación (Clmax/Clmin) a un número de Reynolds específico
b63421	b63421		Relación (L/D)min a un número de Reynolds específico
α 63421	alpha63421		Ángulo de cero Elevación (lift) para el perfil
pun63421	pun63421		puntuación obtenida por el perfil
a64415	a64415		Relación (Clmax/Clmin) a un número de Reynolds específico
b64415	b64415		Relación (L/D)min a un número de Reynolds específico
α 64415	alpha64415		Ángulo de cero Elevación (lift) para el perfil
pun64415	pun64415		puntuación obtenida por el perfil
a64421	a64421		Relación (Clmax/Clmin) a un número de Reynolds específico
b64421	b64421		Relación (L/D)min a un número de Reynolds específico
α 64421	alpha64421		Ángulo de cero Elevación (lift) para el perfil
pun64421	pun64421		puntuación obtenida por el perfil
a65415	a65415		Relación (Clmax/Clmin) a un número de Reynolds específico
b65415	b65415		Relación (L/D)min a un número de Reynolds específico
α 65415	alpha65415		Ángulo de cero Elevación (lift) para el perfil
pun65415	pun65415		puntuación obtenida por el perfil

a65421	a65421		Relación (Clmax/Clmin) a un número de Reynolds específico
b65421	b65421		Relación (L/D)min a un número de Reynolds específico
αo65421	alpha65421		Ángulo de cero Elevación (lift) para el perfil
pun65421	pun65421		puntuación obtenida por el perfil
perfil1	perfil1		Valores proceso de selección de perfil
perfil2	perfil2		
perfil3	perfil3		
perfil4	perfil4		
perfil5	perfil5		
perfil6	perfil6		
perfil7	perfil7		
perfil8	perfil8		
perfil9	perfil9		
perfil10	perfil10		
perfil	perfil		
punperfil1	punperfil1		
punperfil2	punperfil2		
punperfil3	punperfil3		
punperfil4	punperfil4		
punperfil5	punperfil5		
punperfil6	punperfil6		
punperfil7	punperfil7		
punperfil8	punperfil8		
punperfil9	punperfil9		
punperfil10	punperfil10		
punperfil	punperfil		
2.3 Establecimiento de condiciones iniciales de diseño			
Variable	Nomenclatura	Unidades	Nombre
α	alpha	[-]	Ángulo de ataque
Cldis	Cldis	[-]	Coeficiente de elevación de diseño
2.4 Estimación de la forma del perfil			
Variable	Nomenclatura	Unidades	Nombre
nel	nel	[-]	número de elementos de pala
rh	rh	[m]	radio hasta el cual llega la raíz de la pala
inc	inc	[m]	distancia entre cada elemento de pala
Cl1	Cl1	[-]	Coeficiente de elevación a Re=100000

Cd1	Cd1	[-]	Coeficiente de arrastre a Re=100000
Cl4	Cl4	[-]	Coeficiente de elevación a Re=400000
Cd4	Cd4	[-]	Coeficiente de arrastre a Re=400000
Cl50	Cl50	[-]	Coeficiente de elevación a Re=5000000
Cd50	Cd50	[-]	Coeficiente de arrastre a Re=5000000
Cl	Cl	[-]	Coeficiente de Elevación al Reynolds calculado
Cd	Cd	[-]	Coeficiente de arrastre al Reynolds calculado
r	r	[m]	radio de elemento de pala
λ_r	TSRr	[-]	Relación de velocidad de punta del elemento de pala
a	a	[-]	Coeficiente de inducción axial
ap	ap	[-]	Coeficiente de inducción tangencial
difa	difa	[-]	Error de cálculo de coeficiente de inducción axial
difaa	difaa	[-]	Error de cálculo de coeficiente de inducción tangencial
ϕ	phi	[-]	Ángulo de flujo del elemento de pala
F	F	[-]	Función de Glauert
g	g	[-]	Coeficiente g para la función de Glauert modificada
F1	F1	[-]	Función de Glauert modificada
Ctn	Ctn	[-]	Coeficiente de fuerza tangencial
Cn	Cn	[-]	Coeficiente de fuerza normal
cr	cr	[m]	cuerda del elemento de pala
σ	solidez	[-]	Solidez del elemento de pala
Y1	Y1	[-]	Coeficiente Y1 para el cálculo del coeficiente de inducción axial
Y2	Y2	[-]	Coeficiente Y2 para el cálculo del coeficiente de inducción tangencial
ac	ac	[-]	Coeficiente de inducción axial calculado
aac	aac	[-]	Coeficiente de inducción tangencial calculado
acr	acr	[-]	Coeficiente de inducción axial crítico
CT	CT	[-]	Coeficiente de empuje
$\Delta\lambda_r$	deltaTSR	[-]	Diferencia de TSR
ϕ_r	phir	[-]	Ángulo de flujo del elemento de pala en grados
β	Beta	[-]	Ángulo de giro del elemento de pala
Cpt	Cpt	[-]	Coeficiente de potencia total de la hélice

ANEXO H. Valores de las propiedades empleadas en proceso de selección del perfil.

N° Reynolds	Perfil NACA	Clmax/Cdmin	(L/D)_Clmax	(L/D)max	Clmax	α_o
100000	2509	83.11	10.21	57.07	0.94	-2
100000	4412	76.11	27.21	41.87	1.54	-3.9
100000	4415	77.77	17.37	37.78	1.79	-3.8
100000	63-215	60.75	8.22	31.06	1.14	-5.69
100000	63(4)-221	64.08	9.54	31.14	1.51	-5.69
100000	63-415	85.37	14.51	54.48	1.20	-5.69
100000	63(4)-421	72.33	16.22	35.23	1.75	-5.69
100000	64(2)-415	61.23	12.74	39.99	1.72	-6.19
100000	64(4)-421	60.99	9.49	33.17	1.49	-6.19
100000	65(2)-415	59.16	10.10	38.18	1.16	-6.84
100000	65(4)-421	58.86	9.70	35.99	1.42	-6.84
400000	2509	100.32	12.47	77.89	0.94	-2
400000	4412	148.61	30.57	51.24	1.61	-3.9
400000	4415	102.02	28.36	51.30	1.87	-3.8
400000	63-215	89.97	10.19	41.68	1.14	-5.69
400000	63(4)-221	110.67	19.76	41.40	1.66	-5.69
400000	63-415	149.69	16.74	99.53	1.20	-5.69
400000	63(4)-421	116.82	22.55	46.26	1.78	-5.69
400000	64(2)-415	110.43	36.36	53.70	1.40	-6.19
400000	64(4)-421	119.10	22.31	51.55	1.72	-6.19
400000	65(2)-415	95.46	12.65	54.17	1.16	-6.84
400000	65(4)-421	116.22	25.83	52.72	1.66	-6.84
1000000	2509	130.26	43.89	83.41	1.16	-2
1000000	4412	161.97	37.76	78.44	1.63	-3.9
1000000	4415	107.11	35.54	53.65	1.90	-3.8
1000000	63-215	113.26	28.15	49.63	1.32	-5.69
1000000	63(4)-221	129.67	23.98	45.19	1.67	-5.69
1000000	63-415	115.62	43.47	55.10	1.44	-5.69
1000000	63(4)-421	133.31	27.24	50.26	1.79	-5.69
1000000	64(2)-415	125.56	42.91	55.74	1.41	-6.19
1000000	64(4)-421	141.69	26.93	49.78	1.73	-6.19
1000000	65(2)-415	122.85	39.05	51.22	1.39	-6.84
1000000	65(4)-421	132.09	30.51	50.23	1.67	-6.84
5000000	2509	146.43	56.13	97.53	1.17	-2
5000000	4412	187.34	50.28	86.88	1.66	-3.9
5000000	4415	109.61	45.24	62.58	1.93	-3.8
5000000	63-215	125.53	41.72	68.98	1.36	-5.69
5000000	63(4)-221	132.53	32.83	55.47	1.70	-5.69
5000000	63-415	128.14	60.07	75.13	1.46	-5.69
5000000	63(4)-421	140.42	37.39	62.28	1.82	-5.69
5000000	64(2)-415	126.66	59.88	72.90	1.43	-6.19
5000000	64(4)-421	134.33	36.97	60.32	1.75	-6.19
5000000	65(2)-415	121.59	53.36	72.85	1.41	-6.84
5000000	65(4)-421	125.35	41.49	62.64	1.69	-6.84

Fuente: <http://www.mh-aerotoools.de/airfoils/javafoil.htm>; NACA, reportes 460 y 824

ANEXO I. Ecuaciones de C_{lmax}/C_{lmin} y $(L/D)_{Clmax}$ a $Re \leq 400000$ y $400000 < Re \leq 5000000$ para cada tipo de perfil

NOTA: Para Valores de $Re > 5000000$, será seleccionado el perfil NACA 64(2)415, el cual es utilizado en turbinas eólicas de gran tamaño.

Perfil NACA	Rangos de Re	Ecuaciones
2509	$Re \leq 400000$	$\left(\frac{C_{lmax}}{C_{dmin}}\right)_{Rei} = 83,11 + 5,74 \times 10^{-5} * (Rei - 100000)$ $\left(\left(\frac{L}{D}\right)_{Clmax}\right)_{Rei} = 10,21 + 7,53 \times 10^{-6} * (Rei - 100000)$
	$400000 < Re \leq 5000000$	$\left(\frac{C_{lmax}}{C_{dmin}}\right)_{Rei} = 100,32 + 1,00 \times 10^{-5} * (Rei - 400000)$ $\left(\left(\frac{L}{D}\right)_{Clmax}\right)_{Rei} = 12,47 + 9,49 \times 10^{-6} * (Rei - 400000)$
4412	$Re \leq 400000$	$\left(\frac{C_{lmax}}{C_{dmin}}\right)_{Rei} = 76,11 + 2,42 \times 10^{-4} * (Rei - 100000)$ $\left(\left(\frac{L}{D}\right)_{Clmax}\right)_{Rei} = 27,21 + 1,12 \times 10^{-5} * (Rei - 100000)$
	$400000 < Re \leq 5000000$	$\left(\frac{C_{lmax}}{C_{dmin}}\right)_{Rei} = 148,61 + 8,42 \times 10^{-6} * (Rei - 400000)$ $\left(\left(\frac{L}{D}\right)_{Clmax}\right)_{Rei} = 30,57 + 4,28 \times 10^{-6} * (Rei - 400000)$
4415	$Re \leq 400000$	$\left(\frac{C_{lmax}}{C_{dmin}}\right)_{Rei} = 77,77 + 8,08 \times 10^{-5} * (Rei - 100000)$ $\left(\left(\frac{L}{D}\right)_{Clmax}\right)_{Rei} = 17,37 + 3,66 \times 10^{-5} * (Rei - 100000)$
	$400000 < Re \leq 5000000$	$\left(\frac{C_{lmax}}{C_{dmin}}\right)_{Rei} = 102,02 + 1,65 \times 10^{-6} * (Rei - 400000)$ $\left(\left(\frac{L}{D}\right)_{Clmax}\right)_{Rei} = 28,36 + 3,67 \times 10^{-6} * (Rei - 400000)$

63-215	Re≤400000	$\left(\frac{Cl_{max}}{Cd_{min}}\right)_{Rei} = 60,75 + 9,74 \times 10^{-5} * (Rei - 100000)$ $\left(\left(\frac{L}{D}\right)_{Cl_{max}}\right)_{Rei} = 8,22 + 6,57 \times 10^{-6} * (Rei - 100000)$
	400000<Re≤5000000	$\left(\frac{Cl_{max}}{Cd_{min}}\right)_{Rei} = 89,97 + 7,73 \times 10^{-6} * (Rei - 400000)$ $\left(\left(\frac{L}{D}\right)_{Cl_{max}}\right)_{Rei} = 10,19 + 6,85 \times 10^{-6} * (Rei - 400000)$
63(4)-221	Re≤400000	$\left(\frac{Cl_{max}}{Cd_{min}}\right)_{Rei} = 64,08 + 1,55 \times 10^{-4} * (Rei - 100000)$ $\left(\left(\frac{L}{D}\right)_{Cl_{max}}\right)_{Rei} = 9,54 + 3,41 \times 10^{-5} * (Rei - 100000)$
	400000<Re≤5000000	$\left(\frac{Cl_{max}}{Cd_{min}}\right)_{Rei} = 110,67 + 4,75 \times 10^{-6} * (Rei - 400000)$ $\left(\left(\frac{L}{D}\right)_{Cl_{max}}\right)_{Rei} = 19,76 + 2,84 \times 10^{-6} * (Rei - 400000)$
63-415	Re≤400000	$\left(\frac{Cl_{max}}{Cd_{min}}\right)_{Rei} = 85,37 + 2,14 \times 10^{-4} * (Rei - 100000)$ $\left(\left(\frac{L}{D}\right)_{Cl_{max}}\right)_{Rei} = 14,51 + 7,43 \times 10^{-6} * (Rei - 100000)$
	400000<Re≤5000000	$\left(\frac{Cl_{max}}{Cd_{min}}\right)_{Rei} = 149,69 - 4,68 \times 10^{-6} * (Rei - 400000)$ $\left(\left(\frac{L}{D}\right)_{Cl_{max}}\right)_{Rei} = 16,74 + 9,42 \times 10^{-6} * (Rei - 400000)$
63(4)-421	Re≤400000	$\left(\frac{Cl_{max}}{Cd_{min}}\right)_{Rei} = 72,33 + 1,48 \times 10^{-4} * (Rei - 100000)$ $\left(\left(\frac{L}{D}\right)_{Cl_{max}}\right)_{Rei} = 16,22 + 2,11 \times 10^{-5} * (Rei - 100000)$
	400000<Re≤5000000	$\left(\frac{Cl_{max}}{Cd_{min}}\right)_{Rei} = 116,82 + 5,13 \times 10^{-6} * (Rei - 400000)$ $\left(\left(\frac{L}{D}\right)_{Cl_{max}}\right)_{Rei} = 22,55 + 3,23 \times 10^{-6} * (Rei - 400000)$

64(2)-415	Re≤400000	$\left(\frac{Cl_{max}}{Cd_{min}}\right)_{Rei} = 61,23 + 1,64 \times 10^{-4} * (Rei - 100000)$ $\left(\left(\frac{L}{D}\right)_{Cl_{max}}\right)_{Rei} = 12,74 + 7,87 \times 10^{-5} * (Rei - 100000)$
	400000<Re≤5000000	$\left(\frac{Cl_{max}}{Cd_{min}}\right)_{Rei} = 110,43 + 3,53 \times 10^{-6} * (Rei - 400000)$ $\left(\left(\frac{L}{D}\right)_{Cl_{max}}\right)_{Rei} = 36,36 + 5,11 \times 10^{-6} * (Rei - 400000)$
64(4)-421	Re≤400000	$\left(\frac{Cl_{max}}{Cd_{min}}\right)_{Rei} = 60,99 + 1,94 \times 10^{-4} * (Rei - 100000)$ $\left(\left(\frac{L}{D}\right)_{Cl_{max}}\right)_{Rei} = 9,49 + 4,27 \times 10^{-5} * (Rei - 100000)$
	400000<Re≤5000000	$\left(\frac{Cl_{max}}{Cd_{min}}\right)_{Rei} = 119,10 + 3,31 \times 10^{-6} * (Rei - 400000)$ $\left(\left(\frac{L}{D}\right)_{Cl_{max}}\right)_{Rei} = 22,31 + 3,19 \times 10^{-6} * (Rei - 400000)$
65(2)-415	Re≤400000	$\left(\frac{Cl_{max}}{Cd_{min}}\right)_{Rei} = 59,16 + 1,21 \times 10^{-4} * (Rei - 100000)$ $\left(\left(\frac{L}{D}\right)_{Cl_{max}}\right)_{Rei} = 10,10 + 8,50 \times 10^{-6} * (Rei - 100000)$
	400000<Re≤5000000	$\left(\frac{Cl_{max}}{Cd_{min}}\right)_{Rei} = 95,46 + 5,68 \times 10^{-6} * (Rei - 400000)$ $\left(\left(\frac{L}{D}\right)_{Cl_{max}}\right)_{Rei} = 12,65 + 8,85 \times 10^{-6} * (Rei - 400000)$
65(4)-421	Re≤400000	$\left(\frac{Cl_{max}}{Cd_{min}}\right)_{Rei} = 58,86 + 1,91 \times 10^{-4} * (Rei - 100000)$ $\left(\left(\frac{L}{D}\right)_{Cl_{max}}\right)_{Rei} = 9,70 + 5,38 \times 10^{-5} * (Rei - 100000)$
	400000<Re≤5000000	$\left(\frac{Cl_{max}}{Cd_{min}}\right)_{Rei} = 116,22 + 1,98 \times 10^{-6} * (Rei - 400000)$ $\left(\left(\frac{L}{D}\right)_{Cl_{max}}\right)_{Rei} = 25,83 + 3,40 \times 10^{-6} * (Rei - 400000)$

ANEXO J. Referencia bibliográfica de las ecuaciones utilizadas en *UIS-Propeller*

No.	Ecuación	Referencia
1	$D_{calculado} = \frac{88 * MPH}{N * V/nD}$	Weick, F. E., 1926 (paper III)
2	$Error \rightarrow D_{aprox} - D_{calc} \leq 0.05 * D_{aprox}$	Weick, F. E., 1926 (paper III)
3	$\lambda = \Omega * \frac{R}{Vw}$	HAU, E., 2006, pp. 94
4	$Cp_{B=1} = 0.0002 * \lambda^4 - 0.0098 * \lambda^3 + 0.1855 * \lambda^2 - 1.4537 * \lambda + 4.1834$	Regresiones de las curvas de la figura 31 (HAU, E., 2006, pp. 98)
5	$Cp_{B=2} = -2E - 05 * \lambda^5 + 0.001 * \lambda^4 - 0.0202 * \lambda^3 + 0.1902 * \lambda^2 - 0.7529 * \lambda + 1.1724$	
6	$Cp_{B=3} = -0.0017 * \lambda^4 + 0.044 * \lambda^3 - 0.4368 * \lambda^2 + 1.952 * \lambda - 2.8464$	
7	$R = \sqrt{\frac{2 * Pe}{Cp * \eta * \pi * \rho * Vw^3}}$	DIXON, S. L., 1998, pp. 337
8	$\lambda_c = \frac{\Omega_{rad} * R}{Vw * n}$	HAU, E., 2006, pp. 94
9	$\Omega r = \frac{\Omega}{n}$	--
10	$error = \lambda - \lambda_c $	BELTRAN, L. G. <i>et al</i> , 2011
11	$Re = \frac{Vr * c}{\nu}$	MANWELL, J.F <i>et al</i> , 2002, pp.96

12	$r_{root} = 0.25 * Rr$	DIXON, S. L., 1998, pp. 337
13	$C_{root} = \frac{8 * \pi * r_{root} * (1 - \cos(\varphi_{root}))}{B * Cl_{dis}}$	MANWELL, J.F <i>et al</i> ,2002, pp.126
14	$\varphi_{root} = \sin^{-1} \left(\frac{2 * Vw}{3 * Vr_{root}} \right)$	MANWELL, J.F <i>et al</i> ,2002, pp.111
15	$Vr_{root} = Vw + \Omega r * r_{root}$	HABALI, S. M. <i>et al</i> , pp. 267
16	$v = \frac{\mu}{\rho}$	MANWELL, J.F <i>et al</i> ,2002, pp.97
17	$\mu = \mu_0 \left[\frac{T_0 + Cs}{Ta + Cs} \right]^{\frac{3}{2}}$	Sutherland Equation: http://en.wikipedia.org/wiki/Viscosity
18	$Ta[K] = Ta[^\circ C] + 273.5$	http://en.wikipedia.org
19	$\alpha_0 = \alpha_i - \frac{57.3}{2 * \pi} * Cl_i$	ABBOTT, I. <i>et al</i> , NACA Report 824, 1945, pp. 14
22	$\alpha = \varphi - (\beta + \theta_p)$	MANWELL, J.F <i>et al</i> ,2002, pp.127
26	$\lambda r_i = \lambda \left(\frac{r_i}{Rr} \right)$	MANWELL, J.F <i>et al</i> ,2002, pp.125
27	$\varphi_i = \tan^{-1} \left(\frac{1 - a_i}{\lambda r_i * (1 + a'_i)} \right)$	MANWELL, J.F <i>et al</i> ,2002, pp.128

28	$F_i = \frac{2}{\pi} * \cos^{-1} \left(e^{-\frac{B}{2} * \left(\frac{Rr-r_i}{r_i * \sin(\varphi_i)} \right)} \right)$	
29	$F1_i = \frac{2}{\pi} * \cos^{-1} \left(e^{-g_i * \frac{B}{2} * \left(\frac{Rr-r_i}{r_i * \sin(\varphi_i)} \right)} \right)$	SHEN, W. Z. <i>et al</i> , 2005, pp. 464-465
30	$g = e^{-0.125(B * \lambda r_i - 21)} + 0.1$	
31	$Ctn_i = Cl * \sin(\varphi_i) - Cd * \cos(\varphi_i)$	
32	$Cn_i = Cl * \cos(\varphi_i) + Cd * \sin(\varphi_i)$	HABALI, S. M. <i>et al</i> , pp. 257
33	$cr_i = \frac{8 * \pi * r_i * (1 - \cos(\varphi_i))}{B * Cl}$	MANWELL, J.F <i>et al</i> , 2002, pp.126
34	$\sigma_i = \frac{B * cr_i}{2 * \pi * r_i}$	
35	$ac_i = \frac{2 + Y1_i - \sqrt{4 * Y1_i * (1 - F_i) + Y1_i^2}}{2 * (1 + F_i * Y1_i)}$	SHEN, W. Z. <i>et al</i> , 2005, pp. 465
36	$a'c_i = \frac{1}{\frac{(1 - a_i * F_i) * Y2 - 1}{1 - a_i}}$	
37	$Y1 = 4 * F * \frac{\sin(\varphi_i)^2}{\sigma r_i * Cn_i * F1_i}$	
38	$Y2 = \frac{4 * F_i * \sin(\varphi_i) * \cos(\varphi_i)}{\sigma r_i * Ctn_i * F1_i}$	
39a	$si \ a \leq \frac{1}{3} \quad C_{T_i} = 4 * ac_i * F_i * (1 - ac_i * F_i)$	

39b	$si a > \frac{1}{3} \quad C_{T_i} = 4 * [acr^2 * F_i^2 + (1 - 2 * acr * F_i) * ac_i * F_i]$	
42	$Cp_i = \frac{8 * \Delta\lambda}{\lambda^2} * \left(F_i * \sin(\varphi_i)^2 (\cos(\varphi_i) - \lambda r_i * \sin(\varphi_i)) * (\sin(\varphi_i) + \lambda r_i * \cos(\varphi_i)) * \left(1 - \left(\frac{Cd}{Cl} \right) * \cot(\varphi_i) \right) * \lambda r_i^2 \right)$	
43	$\Delta\lambda = \frac{\lambda - \lambda_{root}}{nel}$	MANWELL, J.F <i>et al</i> ,2002, pp.129
44	$Cp_{total} = \sum_{i=1}^{20} Cp_i$	