



**BRUNO CUBATELI SANTOS BERNARDINO**

**INTERFACE NO SCILAB PARA PLOTAR AEROFÓLIOS A PARTIR  
DO CÓDIGO NACA DE 4 OU 5 DÍGITOS**

**Monografia apresentada à Universidade Federal  
do ABC como parte dos requisitos para  
aprovação na disciplina de Bases  
Computacionais da Ciência.**

**Professor: Harlen Costa Batagelo**

**Santo André - SP**

**2018**

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO.....</b>	<b>2</b>
<b>2</b>	<b>OBJETIVOS .....</b>	<b>3</b>
2.1	AEROFÓLIOS NACA .....	3
2.2	AEROFÓLIO NACA DE 4 DÍGITOS .....	4
2.3	AEROFÓLIO NACA DE 5 DÍGITOS .....	5
<b>3</b>	<b>JUSTIFICATIVAS .....</b>	<b>7</b>
<b>4</b>	<b>METODOLOGIA.....</b>	<b>7</b>
4.1	UTILIZAÇÃO DA EXTENSÃO GUI BUILDER .....	7
4.1.1	Interface Confeccionada Pelo GUI Builder .....	9
4.1.2	Código Gerado pelo GUI Builder .....	10
4.2	PROGRAMAÇÃO DAS FUNÇÕES .....	15
4.2.1	Definição das Línguas .....	15
4.2.2	Sinalização do Sistema de Numeração NACA .....	17
4.2.3	Checando as Entradas do Usuário .....	18
4.2.4	Calculando e Plotando as Coordenadas .....	20
4.2.5	Configuração do Botão para Limpar o Gráfico .....	25
4.2.6	Definindo a Exportação do Gráfico .....	26
4.2.7	Definindo um Botão para Sair da Interface.....	30
<b>5</b>	<b>RESULTADOS .....</b>	<b>30</b>
<b>6</b>	<b>DISCUSSÃO.....</b>	<b>35</b>
<b>7</b>	<b>CONCLUSÃO .....</b>	<b>37</b>
7.1	DIFICULDADES ENCONTRADAS .....	38
7.2	TRABALHOS FUTUROS .....	38
	<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>39</b>
	<b>APÊNDICE A .....</b>	<b>40</b>

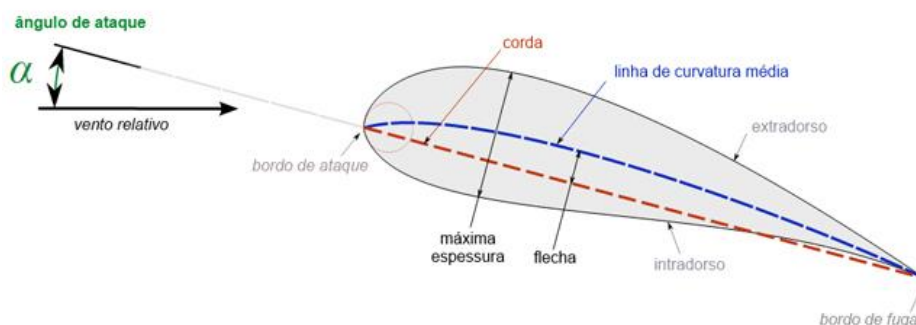
# 1 INTRODUÇÃO

Um dos atuais objetivos da engenharia aeroespacial e/ou aeronáutica é a construção de aeronaves cada vez mais eficientes, de modo a diminuir custos operacionais e reduzir o impacto ambiental causado por este meio de transporte.

Dentre as diversas áreas de estudo dentro destas engenharias, está a análise dos aerofólios, superfícies aerodinâmicas que produzem reações úteis, estas que são utilizadas por aviões e helicópteros. Nos aviões, as asas, superfícies de controle, estabilizadores e pás de uma hélice são exemplos de aerofólios. Já nas aeronaves de asas rotativas, como os helicópteros, são exemplos as pás do rotor, tanto principal como de cauda.

Um aerofólio, aplicado de diversas maneiras numa aeronave, é essencial para que esta possa voar. Porém, estudos complexos buscam analisa-lo de modo a torná-lo mais eficiente, ou seja, produza o máximo de sustentação possível com uma pequena quantidade de arrasto.

De modo a facilitar o entendimento do conceito de um aerofólio, a imagem abaixo ilustra os termos utilizados para discriminá-lo.



**Figura 1** – Discriminação e perfil de um aerofólio assimétrico

Durante este relatório, alguns destes termos serão necessários para a compreensão do funcionamento da interface. Estes conceitos e seus respectivos significados são (Homa, 2017):

- **Aerofólio simétrico:** aquele que pode ser dividido por uma linha reta em duas partes iguais;
- **Aerofólio assimétrico:** aquele que não pode ser dividido por uma linha reta em duas partes iguais;

- **Bordo de ataque:** extremidade dianteira do aerofólio;
- **Bordo de fuga:** extremidade traseira do aerofólio;
- **Extradorso:** superfície superior do aerofólio;
- **Intradorso:** superfície inferior do aerofólio;
- **Corda:** linha reta que liga o bordo de ataque ao bordo de fuga;
- **Linha de curvatura média:** linha que equidista do intradorso e do extradorso;
- **Vento relativo:** toda massa de ar que envolve o aerofólio em movimento, se opondo à sua trajetória de voo;
- **Ângulo de ataque:** ângulo entre a corda do aerofólio e o vento relativo.

## 2 OBJETIVOS

O objetivo geral deste trabalho é facilitar o estudo dos aerofólios a partir de uma interface simples que possibilite inserir códigos NACA e obter, em poucos segundos, sua plotagem em um gráfico.

### 2.1 AEROFÓLIOS NACA

Os aerofólios NACA foram desenvolvidos entre 1929 e 1947 no *NACA's Langley Field Laboratory*. Grande parte destes aerofólios foi baseada em descrições geométricas simples e, apesar dos aerofólios atuais serem feitos a partir de simulação computacional, os aerofólios NACA ainda são encontrados em aplicações aerodinâmicas (Russel E. Cummings, et al, 1995).

Estes aerofólios são construídos combinando-se um “envelope” de espessura com uma linha de curvatura média. Tantos os aerofólios de 4 dígitos como os de 5 dígitos baseiam-se nas seguintes equações:

$$\begin{aligned}x_s &= x - y_e(x) \sin \theta \\ y_s &= y_l(x) + y_e(x) \cos \theta\end{aligned}\tag{1}$$

e

$$\begin{aligned}x_i &= x + y_e(x) \sin \theta \\ y_i &= y_l(x) - y_e(x) \cos \theta\end{aligned}\tag{2}$$

Nas equações acima,  $y_e(x)$  representa a função da espessura,  $y_l(x)$  a função da linha de curvatura média, e  $\theta = \tan^{-1}\left(\frac{dy_l}{dx}\right)$ . Percebe-se ainda que as coordenadas definidas por essas equações estão divididas em superiores,  $(x_s, y_s)$ , e inferiores,  $(x_i, y_i)$ .

## 2.2 AEROFÓLIO NACA DE 4 DÍGITOS

O sistema de numeração para um aerofólio NACA de 4 dígitos é definido da seguinte forma:

### NACA MPXX

- M é o valor máximo da linha de curvatura média em centésimos da corda;
- P é a posição na corda onde ocorre a máxima curvatura em décimos da corda;
- XX é a máxima espessura,  $\frac{t}{c}$ , em porcentagem da corda.

Por exemplo, um aerofólio típico, de código NACA 2412, com corda de comprimento 1m, tem espessura de 0.12m e um valor máximo da linha de curvatura de 0.02m, a 0.4m do bordo de ataque.

A linha de curvatura média é dada por:

$$\begin{aligned} \frac{y_l}{c} &= \frac{M}{P^2} [2P(x/c) - (x/c)^2] \\ \frac{dy_l}{dx} &= \frac{2M}{P^2} [P - (x/c)] \end{aligned} \quad \text{para } \left(\frac{x}{c}\right) < P \quad (3)$$

e

$$\begin{aligned} \frac{y_l}{c} &= \frac{M}{(1-P)^2} [1 - 2P + 2P(x/c) - (x/c)^2] \\ \frac{dy_l}{dx} &= \frac{2M}{(1-P)^2} [P - (x/c)] \end{aligned} \quad \text{para } \left(\frac{x}{c}\right) \geq P \quad (4)$$

A distribuição da espessura de um aerofólio NACA de 4 dígitos é dada por:

$$\frac{y_e}{c} = \frac{t}{c} \left[ a_0 \sqrt{x/c} - a_1(x/c) - a_2(x/c)^2 + a_3(x/c) - a_4(x/c)^4 \right] \quad (5)$$

na qual  $a_0 = 1.4845, a_1 = 0.6300, a_2 = 1.7580, a_3 = 1.4215$   
e  $a_4 = 0.5075$  para bordo de ataque aberto/finito  
ou  $a_4 = 0.5180$  para bordo de ataque fechado/infinito.

Para facilitar a programação da interface, o comprimento da corda,  $c$ , utilizado será de 1 (m, pés, etc.).

### 2.3 AEROFÓLIO NACA DE 5 DÍGITOS

Os aerofólios NACA de 5 dígitos são uma extensão da série de 4 dígitos, já que o código de 4 dígitos não é suficiente para o estudo de curvatura máxima na parte extremamente frontal do aerofólio.

O sistema de numeração para um aerofólio NACA de 5 dígitos é definido da seguinte forma:

NACA LPQXX

- $L$  é a quantidade de curvatura, e o coeficiente de sustentação é  $\frac{3}{2}L$ , em décimos;
- $P$  é o designador da posição de máxima curvatura,  $x_f$ , onde  $x_f = P/2$  e  $P$  é dado em décimos da corda;
- Se  $Q = 0$ , então o aerofólio tem curvatura padrão e, se  $Q = 1$ , então o aerofólio tem curvatura “refletida”, o que não será estudado neste artigo;
- $XX$  é a máxima espessura em porcentagem da corda.

Por exemplo, um aerofólio NACA 23012, com corda de comprimento 1m, tem espessura de 0.12m, coeficiente de sustentação de 0.3, a posição de máxima curvatura está localizada a 0.15m do bordo de ataque e é um aerofólio de curvatura padrão.

A linha de curvatura média é dada por:

$$\frac{y_l}{c} = \frac{K_1}{6} \left[ (x/c)^3 - 3m(x/c)^2 + m^2(3-m)(x/c) \right] \quad \text{para } 0 \leq (x/c) \leq m \quad (6)$$

$$\frac{dy_l}{dx} = \frac{K_1}{6} \left[ 3(x/c)^2 - 6m(x/c) + m^2(3-m) \right]$$

e

$$\begin{aligned} \frac{y_l}{c} &= \frac{K_1}{6} m^3 [1 - (x/c)] \\ \frac{dy_l}{dx} &= -\frac{K_1}{6} m^3 \end{aligned} \quad \text{para } m < (x/c) \leq 1 \quad (7)$$

Na qual  $m$  está relacionado com a posição de curvatura máxima por:

$$x_f = m \left( 1 - \sqrt{\frac{m}{3}} \right) \quad (8)$$

E  $K_1$  é definido por:

$$K_1 = \frac{6C_l}{Q} = \frac{9L}{Q} \quad (9)$$

Na qual  $Q$  é definido por:

$$Q = \frac{3m - 7m^2 + 8m^3 - 4m^4}{\sqrt{m(1-m)}} - \frac{3}{2}(1-2m) \left[ \frac{\pi}{2} - \sin^{-1}(1-2m) \right] \quad (10)$$

Para reproduzir fielmente as coordenadas para aerofólios NACA de 5 dígitos, é necessário utilizar valores tabelados de  $m$ , o que simplifica, em parte, o código a ser escrito, já que as equações (8), (9) e (10) não precisarão ser utilizadas.

LPQ	$m$	$K_1$
210	0.058	361.4
220	0.126	51.65
230	0.2025	15.65
240	0.29	6.643
250	0.391	3.23

**Tabela 1** – Valores tabelados de  $m$  e  $K_1$  (ABBOTT e VON DOENHOFF, 1959)

A distribuição de espessura segue aquela descrita nos aerofólios NACA de 4 dígitos, equação (5).

Para facilitar a programação da interface, o comprimento da corda,  $c$ , utilizado será de 1 (m, pés, etc.).

### 3 JUSTIFICATIVAS

Como visto no capítulo anterior, plotagem de um aerofólio em um gráfico é definida por fórmulas complexas, o que desacelera a sua visualização e, consequentemente, sua análise. Este trabalho tem como objetivo facilitar tal processo, contribuindo para maior facilidade e agilidade na manipulação de códigos NACA e seu estudo.

Espera-se criar com este trabalho uma interface simples para manipulação de códigos NACA, mas também precisa, e que possa ser futuramente expandida para outros tipos de códigos NACA e até outros tipos de aerofólio.

### 4 METODOLOGIA

Para a construção de uma interface simplificada, que possa ser facilmente reprogramada de modo a atingir futuros objetivos, será utilizada a extensão GUI Builder, uma interface gráfica do usuário feita para facilitar a programação de interfaces gráficas no Scilab. É importante ressaltar que as versões utilizadas do Scilab e da extensão GUI Builder são, respectivamente, 5.5.2 e 3.0, ou seja, os códigos podem diferir de outras versões.

#### 4.1 UTILIZAÇÃO DA EXTENSÃO GUI BUILDER

A extensão GUI Builder pode ser instalada no Scilab a partir do seguinte comando:

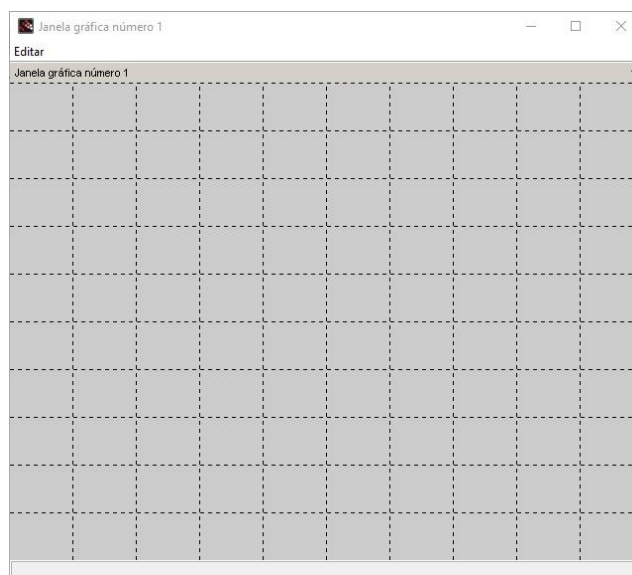
```
--> atomsInstall("guibuilder")
```

A extensão GUI Builder foi feita para facilitar a programação de interfaces gráficas no Scilab. Nas figuras abaixo, estão representadas as três janelas básicas que são utilizadas para a programação.

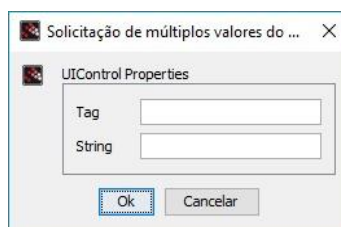




**Figura 2 – Janela gráfica primária**



**Figura 3 – Janela gráfica que simula a interface final**



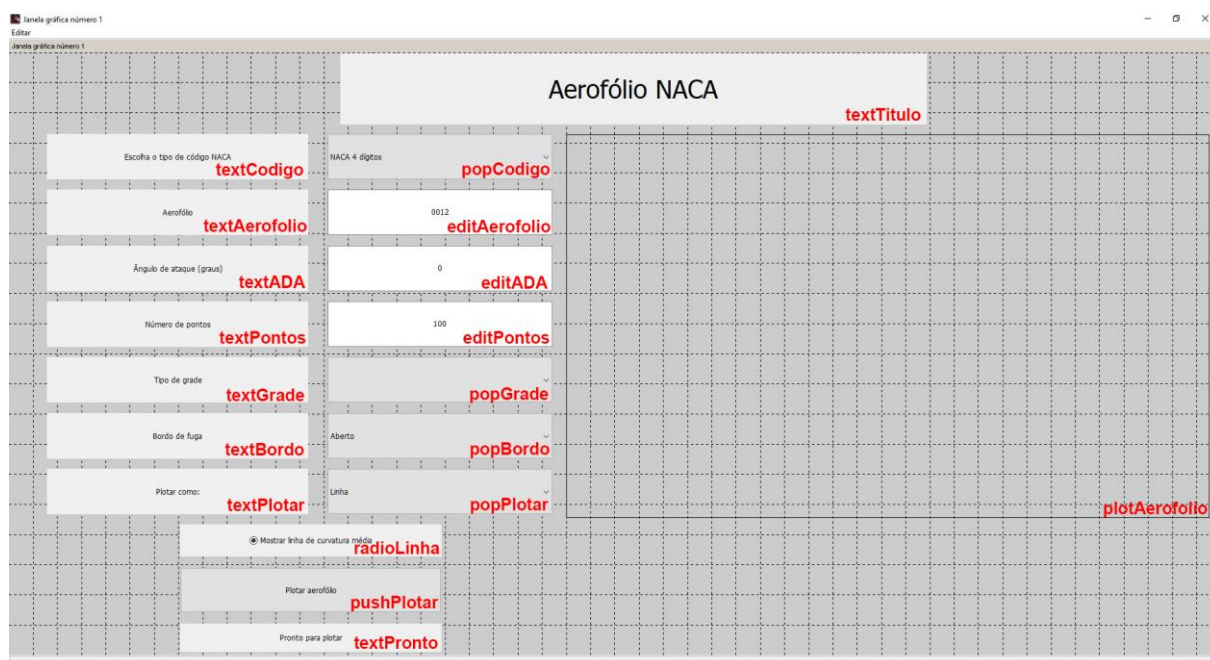
**Figura 4 – Etiqueta e linha da forma**

Na janela gráfica primária, é possível escolher o tipo de forma que será inserida na interface final (Figura 3). Na terceira janela (Figura 4), é possível inserir uma etiqueta, isto é, o código que será utilizado futuramente para

programar a interface, e uma linha, que podem ser tanto as opções a serem escolhidas pelo usuário quanto apenas um texto estático.

#### 4.1.1 Interface Confeccionada Pelo GUI Builder

Na imagem a seguir, estão destacadas em vermelho as etiquetas utilizadas para cada forma presente na interface.



**Figura 5** – Interface gráfica com descrição das etiquetas

Na tabela abaixo estão descritas as etiquetas, as linhas inseridas e as respectivas formas, que serão necessárias e importantes para a programação seguinte da interface. Ressalta-se que estas linhas iniciais poderão ser alteradas pelo usuário quando este selecionar outra língua ou, no caso específico de 'textPronto', de acordo com sua entrada de dados. O símbolo “|” é interpretado pelo GUI Builder como uma nova linha, ou seja, comando “enter”.

ETIQUETA (Tag)	LINHAS (String)	FORMA (Figure)
<b>textTitulo</b>	Aerofólio NACA	Text
<b>textLingua</b>	Escolha sua língua	Text
<b>textLanguage</b>	Choose your language	Text
<b>textLangue</b>	Choisissez votre langue	Text

<b>textCodigo</b>	Escolha o tipo de código NACA	Text
<b>textPontos</b>	Número de pontos	Text
<b>textGrade</b>	Tipo de grade	Text
<b>textBordo</b>	Bordo de fuga	Text
<b>textPlotar</b>	Plotar como:	Text
<b>popLingua</b>	Português English Français	Popupmenu
<b>popCodigo</b>	NACA 4 dígitos NACA 5 dígitos	Popupmenu
<b>editAerofolio</b>	0012	Edit
<b>editPontos</b>	100	Edit
<b>popGrade</b>	Uniforme Não uniforme	Popupmenu
<b>popBordo</b>	Aberto Fechado	Popupmenu
<b>popPlotar</b>	Linha Círculos Pontos	Popupmenu
<b>radioLinha</b>	Mostrar linha de curvatura média	Radiobutton
<b>pushPlotar</b>	PLOTAR AEROFÓLIO	Pushbutton
<b>pushLimpar</b>	LIMPAR GRÁFICO	Pushbutton
<b>textPronto</b>	Pronto para plotar	Text
<b>plotAerofolio</b>	*Não há	Axes
<b>textExportar</b>	Exportar como	Text
<b>popExportar</b>	PDF JPG PNG	Popupmenu
<b>pushExpotar</b>	EXPORTAR	Pushbutton
<b>pushSair</b>	SAIR	Pushbutton

Tabela 2 - Tags, strings e figures.

#### 4.1.2 Código Gerado pelo GUI Builder

O GUI Builder, após a criação da interface, gera um código de programação que pode ser decodificado por qualquer Scilab, mesmo que não tenha a extensão instalada. O código gerado pode ser observado abaixo:

```
f=figure('figure_position',[-8,-
8],'figure_size',[1936,1056],'auto_resize','on','background',[33],'figure_name','Aerofólio NACA');

delmenu(f.figure_id,gettext('File'))
delmenu(f.figure_id,gettext('?'))
delmenu(f.figure_id,gettext('Tools'))
toolbar(f.figure_id,'off')

handles.dummy = 0;
handles.textTitulo=icontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1],'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[60],'FontUnits','points','FontWe
ight','bold','ForegroundColor',[-1,-1,-
1],'HorizontalAlignment','center','ListboxTop',[1],'Max',[1],'Min',[0],'Position',[0.2737787,0.8813208,
```

```
0.4843633,0.1192453], 'Relief', 'default', 'SliderStep', [0.01, 0.1], 'String', 'Aerofólio
NACA', 'Style', 'text', 'Value', [0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'textTitulo', 'Callback', ''
)
```

```
handles.textCodigo=icontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize', [20], 'FontUnits', 'points', 'FontWe
ight', 'normal', 'ForegroundColor', [-1,-1,-
1], 'HorizontalAlignment', 'center', 'ListboxTop', [], 'Max', [1], 'Min', [0], 'Position', [0.0316462, 0.6998113,
0.2156981, 0.0754717], 'Relief', 'default', 'SliderStep', [0.01, 0.1], 'String', 'Tipo de código
NACA', 'Style', 'text', 'Value', [0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'textCodigo', 'Callback', ''
)
```

```
handles.textAerofolio=icontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize', [20], 'FontUnits', 'points', 'FontWe
ight', 'normal', 'ForegroundColor', [-1,-1,-
1], 'HorizontalAlignment', 'center', 'ListboxTop', [], 'Max', [1], 'Min', [0], 'Position', [0.0316462, 0.6072408,
0.2156981, 0.0754717], 'Relief', 'default', 'SliderStep', [0.01, 0.1], 'String', 'Código NACA do aerofólio
(MPXX)', 'Style', 'text', 'Value', [0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'textAerofolio', 'Callb
ack', ''
)
```

```
handles.textPontos=icontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize', [20], 'FontUnits', 'points', 'FontWe
ight', 'normal', 'ForegroundColor', [-1,-1,-
1], 'HorizontalAlignment', 'center', 'ListboxTop', [], 'Max', [1], 'Min', [0], 'Position', [0.0316462, 0.5120998,
0.2156981, 0.0754717], 'Relief', 'default', 'SliderStep', [0.01, 0.1], 'String', 'Número de
pontos', 'Style', 'text', 'Value', [0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'textPontos', 'Callback', ''
)
```

```
handles.textGrade=icontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize', [20], 'FontUnits', 'points', 'FontWe
ight', 'normal', 'ForegroundColor', [-1,-1,-
1], 'HorizontalAlignment', 'center', 'ListboxTop', [], 'Max', [1], 'Min', [0], 'Position', [0.0316462, 0.4195293,
0.2156981, 0.0754717], 'Relief', 'default', 'SliderStep', [0.01, 0.1], 'String', 'Tipo de
grade', 'Style', 'text', 'Value', [0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'textGrade', 'Callback', ''
)
```

```
handles.textBordo=icontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize', [20], 'FontUnits', 'points', 'FontWe
ight', 'normal', 'ForegroundColor', [-1,-1,-
1], 'HorizontalAlignment', 'center', 'ListboxTop', [], 'Max', [1], 'Min', [0], 'Position', [0.0316462, 0.3269589,
0.2156981, 0.0754717], 'Relief', 'default', 'SliderStep', [0.01, 0.1], 'String', 'Bordo de
fuga', 'Style', 'text', 'Value', [0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'textBordo', 'Callback', ''
)
```

```
handles.textPlotar=icontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize', [20], 'FontUnits', 'points', 'FontWe
ight', 'normal', 'ForegroundColor', [-1,-1,-
1], 'HorizontalAlignment', 'center', 'ListboxTop', [], 'Max', [1], 'Min', [0], 'Position', [0.0316462, 0.2343884,
0.2156981, 0.0754717], 'Relief', 'default', 'SliderStep', [0.01, 0.1], 'String', 'Plotar
como', 'Style', 'text', 'Value', [0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'textPlotar', 'Callback', ''
)
```

```
handles.pushPlotar=icontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-
1], 'Enable', 'on', 'FontAngle', 'normal', 'FontName', 'Tahoma', 'FontSize', [25], 'FontUnits', 'points', 'FontWe
ight', 'bold', 'ForegroundColor', [-1,-1,-
1], 'HorizontalAlignment', 'center', 'ListboxTop', [], 'Max', [1], 'Min', [0], 'Position', [0.0566462, 0.0711897,
0.1656981, 0.0829741], 'Relief', 'default', 'SliderStep', [0.01, 0.1], 'String', 'PLOTAR
AEROFÓLIO', 'Style', 'pushbutton', 'Value', [0], 'VerticalAlignment', 'middle', 'Visible', 'on', 'Tag', 'pushPlo
tar', 'Callback', 'pushPlotar_callback(handles)')
)
```

```
handles.textPronto=icontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[20], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','center','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.0416462,0.0059091,0.4056981,0.0481818], 'Relief','default','SliderStep',[0.01,0.1], 'String','Defina os parâmetros e clique em PLOTAR', 'AEROFÓLIO','Style','text','Value',[0], 'VerticalAlignment','middle','Visible','on','Tag','textPronto','Callback','')
```

```
handles.radioLinha=icontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[20], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','center','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.1416462,0.1637602,0.2156981,0.0535294], 'Relief','default','SliderStep',[0.01,0.1], 'String','Mostrar linha de curvatura média','Style','radiobutton','Value',[1], 'VerticalAlignment','middle','Visible','on','Tag','radioLinha','Callback','radioLinha_callback(handles)')
```

```
handles.editAerofolio=icontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[20], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','center','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.2634686,0.6072408,0.1856981,0.0754717], 'Relief','default','SliderStep',[0.01,0.1], 'String','0012','Style','edit','Value',[0], 'VerticalAlignment','middle','Visible','on','Tag','editAerofolio','Callback','editAerofolio_callback(handles)')
```

```
handles.popCodigo=icontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[20], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','center','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.2634686,0.6998113,0.1856981,0.0754717], 'Relief','default','SliderStep',[0.01,0.1], 'String','NACA 4 dígitos|NACA 5 dígitos','Style','popupmenu','Value',[1], 'VerticalAlignment','middle','Visible','on','Tag','popCodigo','Callback','popCodigo_callback(handles)')
```

```
handles.editPontos=icontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[20], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','center','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.2634686,0.5120998,0.1856981,0.0754717], 'Relief','default','SliderStep',[0.01,0.1], 'String','100','Style','edit','Value',[0], 'VerticalAlignment','middle','Visible','on','Tag','editPontos','Callback','editPontos_callback(handles)')
```

```
handles.popGrade=icontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[20], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','center','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.2634686,0.4195293,0.1856981,0.0754717], 'Relief','default','SliderStep',[0.01,0.1], 'String','Uniforme|Não uniforme','Style','popupmenu','Value',[1], 'VerticalAlignment','middle','Visible','on','Tag','popGrade','Callback','popGrade_callback(handles)')
```

```
handles.popBordo=icontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[20], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','center','ListboxTop',[], 'Max',[1], 'Min',[0], 'Position',[0.2634686,0.3269589,0.1856981,0.0754717], 'Relief','default','SliderStep',[0.01,0.1], 'String','Aberto|Fechado','Style','popupmenu','Value',[1], 'VerticalAlignment','middle','Visible','on','Tag','popBordo','Callback','popBordo_callback(handles)')
```

```
handles.popPlotar=icontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1],
'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[20],
'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','left','ListboxTop',[],'Max',[1],
'Min',[0],
'Position',[0.2634686,0.2343884,0.1856981,0.0754717],
'Relief','default','SliderStep',[0.01,0.1],
'String','Linha|Círculos|Pontos','Style','popupmenu','Value',[1],
'VerticalAlignment','middle','Visible','on','Tag','popPlotar','Callback','popPlotar_callback(handles)')
```

```
handles.pushSair=icontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1],
'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[25],
'FontUnits','points','FontWeight','bold','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','center','ListboxTop',[],'Max',[1],
'Min',[0],
'Position',[0.903125,0.0140086,0.0869792,0.0700431],
'Relief','default','SliderStep',[0.01,0.1],
'String','SAIR','Style','pushbutton','Value',[0],
'VerticalAlignment','middle','Visible','on','Tag','pushSair','Callback','pushSair_callback(handles)')
```

```
handles.pushLimpar=icontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1],
'Enable','off','FontAngle','normal','FontName','Tahoma','FontSize',[25],
'FontUnits','points','FontWeight','bold','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','center','ListboxTop',[],'Max',[1],
'Min',[0],
'Position',[0.2684686,0.0711897,0.1756981,0.0829741],
'Relief','default','SliderStep',[0.01,0.1],
'String','LIMPAR GRÁFICO','Style','pushbutton','Value',[0],
'VerticalAlignment','middle','Visible','on','Tag','pushLimpar','Callback','pushLimpar_callback(handles)')
```

```
handles.plotAerofolio=newaxes();handles.plotAerofolio.margins = [ 0 0 0 0];
handles.plotAerofolio.axes_bounds = [0.4601042,0.1368534,0.5305208,0.6346983];
```

```
handles.textLingua=icontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1],
'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[20],
'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','center','ListboxTop',[],'Max',[1],
'Min',[0],
'Position',[0.0316462,0.8446703,0.2156981,0.0354717],
'Relief','default','SliderStep',[0.01,0.1],
'String','Escolha sua língua','Style','text','Value',[0],
'VerticalAlignment','middle','Visible','on','Tag','textLingua','Callback','')
```

```
handles.textLanguage=icontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1],
'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[20],
'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','center','ListboxTop',[],'Max',[1],
'Min',[0],
'Position',[0.0316462,0.8146703,0.2156981,0.0354717],
'Relief','default','SliderStep',[0.01,0.1],
'String','Choose your language','Style','text','Value',[0],
'VerticalAlignment','middle','Visible','on','Tag','textLanguage','Callback','')
```

```
handles.textLangue=icontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1],
'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[20],
'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','center','ListboxTop',[],'Max',[1],
'Min',[0],
'Position',[0.0316462,0.7846703,0.2156981,0.0354717],
'Relief','default','SliderStep',[0.01,0.1],
'String','Choisissez votre langue','Style','text','Value',[0],
'VerticalAlignment','middle','Visible','on','Tag','textLangue','Callback','')
```

```
handles.popLingua=icontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1],
'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[20],
'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1],
'HorizontalAlignment','center','ListboxTop',[],'Max',[1],
'Min',[0],
'Position',[0.2634686,0.7846703,0.1856981,0.0954717],
'Relief','default','SliderStep',[0.01,0.1],
'String','Português|English|Français','Style','popupmenu','Value',[1],
'VerticalAlignment','middle','Visible','on','Tag','popLingua','Callback','popLingua_callback(handles)')
```



```
yle','popupmenu','Value',[1],'VerticalAlignment','middle','Visible','on','Tag','popLingua','Callback','popLingua_callback(handles)')
```

```
handles.textExportar=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[20], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','center','ListboxTop',[0], 'Max',[1], 'Min',[0], 'Position',[0.5770221,0.0800158,0.1270425,0.0594494], 'Relief','default','SliderStep',[0.01,0.1], 'String','Exportar como','Style','text','Value',[0], 'VerticalAlignment','middle','Visible','on','Tag','textExportar','Callback','')

```

```
handles.popExportar=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[20], 'FontUnits','points','FontWeight','normal','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','left','ListboxTop',[0], 'Max',[1], 'Min',[0], 'Position',[0.7082292,0.0800158,0.0370425,0.0594494], 'Relief','default','SliderStep',[0.01,0.1], 'String','PDF|JPG|PNG','Style','popupmenu','Value',[1], 'VerticalAlignment','middle','Visible','on','Tag','popExportar','Callback','popExportar_callback(handles)')
```

```
handles.pushExportar=uicontrol(f,'unit','normalized','BackgroundColor',[-1,-1,-1], 'Enable','on','FontAngle','normal','FontName','Tahoma','FontSize',[20], 'FontUnits','points','FontWeight','bold','ForegroundColor',[-1,-1,-1], 'HorizontalAlignment','center','ListboxTop',[0], 'Max',[1], 'Min',[0], 'Position',[0.6282292,0.0100158,0.0770425,0.0594494], 'Relief','default','SliderStep',[0.01,0.1], 'String','EXPORTAR','Style','pushbutton','Value',[0], 'VerticalAlignment','middle','Visible','on','Tag','pushExportar','Callback','pushExportar_callback(handles)')
```

Além do código acima, que determina todas as formas que serão mostradas ao usuário, o GUI Builder também gera funções para as *figures* que podem ser alteradas pelo usuário. Abaixo está uma parte dessas funções geradas, já que nem todas serão necessárias.

```
function popLingua_callback(handles)
```

```
////////
```

```
endfunction
```

```
function popCodigo_callback(handles)
```

```
////////
```

```
endfunction
```

```
function editAerofolio_callback(handles)
```

```
////////
```

```
endfunction
```

```
function editPontos_callback(handles)
```

```
////////
```

```
endfunction
```

```
function pushPlotar_callback(handles)
```

```
////////
```

```
endfunction
```

```
function pushLimpar_callback(handles)
```

```

////////
endfunction

function pushExportar_callback(handles)
////////
endfunction

function pushSair_callback(handles)
////////
endfunction

```

Cada dessas funções se relaciona com sua *figure* a partir de um *callback*, ou seja, toda vez que o usuário alterar ou clicar em uma das formas, o que estiver dentro de cada função (////////) será executado pelo Scilab. Cada uma dessas funções será detalhada em seguida.

## 4.2 PROGRAMAÇÃO DAS FUNÇÕES

Antes de iniciarmos a programar as funções, é necessário configurar o gráfico que será visto pelo usuário, já que a plotagem dos aerofólios NACA necessita de eixos com marcações espaçadas igualmente, de modo a não distorcer tais.

Para isso, iremos alterar algumas características do gráfico: *isoview*, que garante o espaçamento igual no gráfico; *tight\_limits*, que faz com que o gráfico tenha os limites laterais estabelecidos pelos *data\_bounds*. Abaixo está descrito o código utilizado para essas configurações.

```

handles.plotAerofolio.isoview = "on";
handles.plotAerofolio.tight_limits = "on";
handles.plotAerofolio.data_bounds = [0,-0.3;1,0.3];

```

O comando `handles.(tag).(propriedade)` é essencial para esta programação e será utilizada intensamente durante este trabalho. Este comando tem duas funções básicas: inserir ou alterar uma propriedade, como feito acima, ou retornar alguma propriedade, como um *string*.

### 4.2.1 Definição das Línguas

O usuário poderá, dentro da interface, alterar a língua em que deseja utilizá-la. Para isso, bastará que selecione Português, English ou Français na *figure* `popLingua`. Por conveniência, a interface iniciará na língua portuguesa. Para que a troca seja possível, deve-se configurar `function popLingua_callback(handles)` com uma simples estrutura condicional que leva em



consideração o valor de popLingua: 1 para Português, 2 para English e 3 para Français.

```
function popLingua_callback(handles)
```

```
    Lingua = handles.popLingua.Value;
```

```
    if Lingua == 1
```

```
        set(handles.textTitulo,'String','Aerofólio NACA');
        set(handles.textCodigo,'String','Tipo de código NACA');
        set(handles.textAerofolio,'String','Código NACA do aerofólio (MPXX)');
        set(handles.textPontos,'String','Número de pontos');
        set(handles.textGrade,'String','Tipo de grade');
        set(handles.textBordo,'String','Bordo de fuga');
        set(handles.textPlotar,'String','Plotar como');
        set(handles.pushPlotar,'String','PLOTAR AEROFÓLIO');
        set(handles.textPronto,'String','Defina os parâmetros e clique em PLOTAR AEROFÓLIO. ');
        set(handles.radioLinha,'String','Mostrar linha de curvatura média');
        set(handles.popCodigo,'String','NACA 4 dígitos|NACA 5
```

```
    dígitos');set(handles.popCodigo,'Value',[1]);
```

```
        set(handles.popGrade,'String','Uniforme|Não uniforme');set(handles.popGrade,'Value',[1]);
        set(handles.popBordo,'String','Aberto|Fechado');set(handles.popBordo,'Value',[1]);
        set(handles.popPlotar,'String','Linha|Círculos|Pontos');set(handles.popPlotar,'Value',[1]);
        set(handles.pushSair,'String','SAIR');
        set(handles.pushLimpar,'String','LIMPAR GRÁFICO');
        set(handles.textExportar,'String','Exportar como');
        set(handles.pushExportar,'String','EXPORTAR');
```

```
    elseif Lingua == 2
```

```
        set(handles.textTitulo,'String','NACA Airfoil');
        set(handles.textCodigo,'String','NACA Airfoil series');
        set(handles.textAerofolio,'String','NACA Airfoil number (MPXX)');
        set(handles.textPontos,'String','Number of grid points');
        set(handles.textGrade,'String','Grid type');
        set(handles.textBordo,'String','Trailing edge');
        set(handles.textPlotar,'String','Plot as');
        set(handles.pushPlotar,'String','PLOT AIRFOIL');
        set(handles.textPronto,'String','Choose your parameters and click on PLOT AIRFOIL. ');
        set(handles.radioLinha,'String','Show camber line');
        set(handles.popCodigo,'String','NACA 4-digit|NACA 5-
```

```
    digit');set(handles.popCodigo,'Value',[1]);
```

```
        set(handles.popGrade,'String','Uniform|Nonuniform');set(handles.popGrade,'Value',[1]);
        set(handles.popBordo,'String','Open|Closed');set(handles.popBordo,'Value',[1]);
        set(handles.popPlotar,'String','Line|Circles|Points');set(handles.popPlotar,'Value',[1]);
        set(handles.pushSair,'String','EXIT');
        set(handles.pushLimpar,'String','CLEAR GRAPH');
        set(handles.textExportar,'String','Export as');
        set(handles.pushExportar,'String','EXPORT');
```

```
    elseif Lingua == 3
```

```
        set(handles.textTitulo,'String','Profil NACA');
        set(handles.textCodigo,'String','Série NACA');
        set(handles.textAerofolio,'String','Code NACA (MPXX)');
        set(handles.textPontos,'String','Nombre de points de la grille');
        set(handles.textGrade,'String','Type de grille');
        set(handles.textBordo,'String','Bord de fuite');
        set(handles.textPlotar,'String','Tracer comme');
        set(handles.pushPlotar,'String','TRACER PROFIL');
        set(handles.textPronto,'String','Choisissez vos paramètres et cliquez sur TRACER PROFIL. ');
```

```

set(handles.radioLinha,'String','Montrer la ligne de cambrure');
set(handles.popCodigo,'String','NACA 4-chiffres|NACA 5-
chiffres');set(handles.popCodigo,'Value',[1]);
set(handles.popGrade,'String','Uniforme|Non uniforme');set(handles.popGrade,'Value',[1]);
set(handles.popBordo,'String','Ouvert|Fermé');set(handles.popBordo,'Value',[1]);
set(handles.popPlotar,'String','Ligne|Cercles|Points');set(handles.popPlotar,'Value',[1]);
set(handles.pushSair,'String','SORTIR');
set(handles.pushLimpar,'String','NETTOYER GRAPHIQUE');
set(handles.textExportar,'String','Exporter comment');
set(handles.pushExportar,'String','EXPORTER');
end
endfunction

```

No código acima, os comando do tipo `set(handles.(tag),'(propriedade)',(valor da propriedade))` tem a mesma função do código explicado no início desta seção.

Além desta alteração inicial, todos os textos alterados a partir daqui deverão levar em conta a língua previamente selecionada pelo usuário. Para isso, no início de cada função será definida a variável `Lingua = handles.popLingua.Value` e todas as alterações de textos terão a seguinte estrutura condicional:

```

if Lingua == 1
    /// Definição na língua portuguesa
elseif Lingua == 2
    /// Definição na língua inglesa
elseif Lingua == 3
    /// Definição na língua francesa
end

```

#### 4.2.2 Sinalização do Sistema de Numeração NACA

Neste código, toda vez que o usuário selecionar um tipo de código NACA, a forma `textAerofolio` será alterada ilustrando como é o código que o usuário deve digitar. Em seguida, o código utilizado para isso.

```

function popCodigo_callback(handles)

Codigo = handles.popCodigo.Value;
Lingua = handles.popLingua.Value;

if Codigo == 1
    if Lingua == 1
        set(handles.textAerofolio,'String','Código NACA do aerofólio (MPXX)');
    elseif Lingua == 2
        set(handles.textAerofolio,'String','NACA Airfoil number (MPXX)');
    elseif Lingua == 3
        set(handles.textAerofolio,'String','Code NACA (MPXX)');
    end
end

set(handles.editAerofolio,'String','0012');
else

```

```

if Lingua == 1
    set(handles.textAerofolio,'String','Código NACA do aerofólio (LPQXX)');
elseif Lingua == 2
    set(handles.textAerofolio,'String','NACA Airfoil number (LPQXX)');
elseif Lingua == 3
    set(handles.textAerofolio,'String','Code NACA (LPQXX)');
end

set(handles.editAerofolio,'String','21012');
end

endfunction

```

#### 4.2.3 Checando as Entradas do Usuário

Para evitar erros durante a execução e cálculo nas diferentes coordenadas, o programa será codificado de modo a evitar entradas diferentes do se espera.

##### 4.2.3.1 Checando o código inserido pelo usuário

Caso o usuário tenha escolhido trabalhar com o código de 4 dígitos, ele deverá e poderá plotar códigos que tenham a extensão de 4 dígitos, o que ocorrerá de forma análoga para o código de 5 dígitos. Para atingir esse objetivo, estruturas condicionais irão comparar a variável `Codigo = handles.popCodigo.Value` com a extensão da *string* digitada pelo usuário. No Scilab, a extensão de uma *string* pode ser obtida pelo comando `length` (variável = *string*).

```

function editAerofolio_callback(handles)

    Aerofolio = handles.editAerofolio.String ;
    Codigo = handles.popCodigo.Value;
    Lingua = handles.popLingua.Value;

    if Codigo == 1 & length(Aerofolio) == 4
        if Lingua == 1
            set(handles.textPronto,'String','Pronto para plotar. ');
        elseif Lingua == 2
            set(handles.textPronto,'String','Ready to plot. ');
        elseif Lingua == 3
            set(handles.textPronto,'String','Prêt à tracer. ');
        end

        set(handles.textPronto,'ForegroundColor',[0,0.7,0]);
        set(handles.pushPlotar,'Enable','on');
    end

    if Codigo == 1 & length(Aerofolio) ~= 4
        if Lingua == 1
            set(handles.textPronto,'String','Digite um código NACA de 4 dígitos. ');
        elseif Lingua == 2

```

```

    set(handles.textPronto,'String','Type a 4-digit NACA code. ');
elseif Lingua == 3
    set(handles.textPronto,'String','Tapez un code NACA 4-chiffres. ');
end

set(handles.textPronto,'ForegroundColor',[1,0,0]);
set(handles.pushPlotar,'Enable','off');
end

if Codigo == 2 & length (Aerofolio) == 5
    if Lingua == 1
        set(handles.textPronto,'String','Pronto para plotar. ');
    elseif Lingua == 2
        set(handles.textPronto,'String','Ready to plot. ');
    elseif Lingua == 3
        set(handles.textPronto,'String','Prêt à tracer. ');
    end

    set(handles.textPronto,'ForegroundColor',[0,0.7,0]);
    set(handles.pushPlotar,'Enable','on');
end

if Codigo == 2 & length (Aerofolio) ~= 5
    if Lingua == 1
        set(handles.textPronto,'String','Digite um código NACA de 5 dígitos. ');
    elseif Lingua == 2
        set(handles.textPronto,'String','Type a 5-digit NACA code. ');
    elseif Lingua == 3
        set(handles.textPronto,'String','Tapez un code NACA 5-chiffres. ');
    end

    set(handles.textPronto,'ForegroundColor',[1,0,0]);
    set(handles.pushPlotar,'Enable','off');
end

endfunction

```

#### 4.2.3.2 Checando o número de pontos no gráfico

Para plotar um aerofólio, faz sentido que o número de coordenadas seja no mínimo 3, caso contrário, o programa só ira plotar uma única linha. Nesta etapa, o comando `strtod(handles.(tag).String)` será essencial, já que é capaz de transformar o que foi digitado pelo usuário (*string*) em um número que pode ser utilizado pelo Scilab. Tal comando será muito utilizado para definir as diferentes variáveis para as equações posteriormente.

```

function editPontos_callback(handles)

Pontos = strtod(handles.editPontos.String);
Lingua = handles.popLingua.Value;

if Pontos >= 3 then
    if Lingua == 1
        set(handles.textPronto,'String','Pronto para plotar. ');
    end
end

```

```

elseif Lingua == 2
    set(handles.textPronto,'String','Ready to plot. ');
elseif Lingua == 3
    set(handles.textPronto,'String','Prêt à tracer. ');
end

set(handles.textPronto,'ForegroundColor',[0,0.7,0]);
set(handles.pushPlotar,'Enable', 'on');
else
    if Lingua == 1
        set(handles.textPronto,'String','Escolha um número de pontos maior que 2. ');
    elseif Lingua == 2
        set(handles.textPronto,'String','Choose a number of grid points greater than 2. ');
    elseif Lingua == 3
        set(handles.textPronto,'String','Choisissez un nombre de points de la grille supérieur à 2. ');
    end

    set(handles.textPronto,'ForegroundColor',[1,0,0]);
    set(handles.pushPlotar,'Enable', 'off');
end

endfunction

```

#### 4.2.4 Calculando e Plotando as Coordenadas

Ao clicar no *pushbutton* PLOTAR AEROFÓLIO, o Scilab ira computar todas as variáveis inseridas pelo usuário e então aplica-las nas equações mostradas anteriormente. Abaixo está o código criado para tal.

```

function pushPlotar_callback(handles)

    Lingua = handles.popLingua.Value;      // =1, português; =2, inglês; =3, francês

    /// Informando o usuário que o aerofólio está sendo plotado
    if Lingua == 1
        set(handles.textPronto,'String','Plotando... ');
    elseif Lingua == 2
        set(handles.textPronto,'String','Plotting... ');
    elseif Lingua == 3
        set(handles.textPronto,'String','Traçant... ');
    end

    set(handles.textPronto,'ForegroundColor',[-1,-1,-1]);

    /// Desativando o botão 'Plotar Aerofólio'

    set(handles.pushPlotar,'Enable','off');

    /// Definindo as variáveis inseridas pelo usuário

    Codigo = handles.popCodigo.Value;      // =1, NACA 4 dígitos; =2, NACA 5 dígitos
    Aerofolio = handles.editAerofolio.String; // Retira o código NACA (completo) inserido pelo
usuário
    Pontos = strtod(handles.editPontos.String); // Retorna o número de pontos a serem utilizados
no gráfico
    Grade = handles.popGrade.Value;      // =1, grade uniforme; =2, grade não uniforme

```

```

Bordo = handles.popBordo.Value;           // =1, aberto; =2, fechado
Plotarcomo = handles.popPlotar.Value;      // =1, linha; =2, círculos; =3, pontos
Linha = handles.radioLinha.Value;          // =0, não mostrar linha de curvatura média; =1,
mostrar linha de curvatura média

```

*/// Extraindo os valores do código NACA*

```

if Codigo == 1 then
    Mi = strtod(part(Aerofolio,[1]));
    Pi = strtod(part(Aerofolio,[2]));
    XXi = strtod(part(Aerofolio,[3,4]));
elseif Codigo == 2 then
    LPQ = strtod(part(Aerofolio,[1,2,3]));
    Qi = strtod(part(Aerofolio,[3]));
    XXi = strtod(part(Aerofolio,[4,5]));

```

*/// Para esta programação, não será possível utilizar Qi diferente de 0*

```

if Qi ~= 0 then
    if Lingua == 1
        set(handles.textPronto,'String','Utilize um código NACA de 5 dígitos com Q=0');
    elseif Lingua == 2
        set(handles.textPronto,'String','Use a 5-digit NACA code with Q=0');
    elseif Lingua == 3
        set(handles.textPronto,'String','Utilisez un code NACA 5-chiffres avec Q=0');
    end
    set(handles.textPronto,'ForegroundColor',[1,0,0]);
    set(handles.pushPlotar,'Enable','off');
end
end

```

*/// Transformando os valores extraídos para serem utilizados nas fórmulas*

```

if Codigo == 1 then
    M = Mi/100;
    P = Pi/10;
    t = XXi/100;
else
    t = XXi/100;
end

```

*/// Delimitando as constantes para cada tipo de código e tipo de bordo de fuga*

```

a0 = 1.4845;
a1 = 0.63;
a2 = 1.758;
a3 = 1.4215;

```

```

if Bordo == 1 then
    a4 = 0.5075;
else
    a4 = 0.5180;
end

```

*/// Constantes tabuladas para o código de 5 dígitos*

```

if Codigo == 2 then

```

```

if LPQ == 210
    m = 0.058
    K = 361.4
elseif LPQ == 220
    m = 0.126
    K = 51.65
elseif LPQ == 230
    m = 0.2025
    K = 15.65
elseif LPQ == 240
    m = 0.29
    K = 6.643
elseif LPQ == 250
    m = 0.391
    K = 3.23
end
end

/// Espaçamento dos pontos

if Grade == 1 then
    xl = linspace(0,1,Pontos);
elseif Grade == 2
    beta = linspace(0,%pi,Pontos);
    xl = (0.5*(1-cos(beta))); /// Equação para dar maior definição aos bordos de ataque e fuga
end

/// Linha de curvatura média

yl = ones(1, length(xl));
dyl_dxl = ones(1, length(xl));
theta=ones(1, length(xl));

/// Para código NACA de 4 dígitos

if Codigo==1 then
    for c=1:length(xl)
        if xl(c) >=0 & xl(c) < P
            yl(c) = (M/P^2).*((2*P*xl(c))-xl(c)^2);
            deff('dyl_dxl=f(yl,xl)','dyl_dxl=((2*M)/(P^2))*(P-xl)');
            dyl_dxl(c) = ode(0,0,xl(c),f);
        elseif xl(c) >= P & xl(c) <= 1
            yl(c) = (M/(1-P)^2).*(1-(2*P)+(2*P*xl(c))-(xl(c)^2));
            deff('dyl_dxl=f(yl,xl)','dyl_dxl=((2*M)/((1-P)^2))*(P-xl)');
            dyl_dxl(c) = ode(0,0,xl(c),f);
        end
        theta(c) = atan(dyl_dxl(c));
    end
end

/// Para código NACA de 5 dígitos

if Codigo==2 then
    for c=1:length(xl)
        if xl(c) >=0 & xl(c) <= m

```

```

    yl(c) = (K/6)*((xl(c))^3-3*m*(xl(c))^2+m^2*(3-m)*(xl(c)));
    deff('dyldxl=f(yl,xl)', 'dyldxl=(K/6)*(3*xl^2-6*m*xl+m^2*(3-m))');
    dyl_dxl(c) = ode(0,0,xl(c),f);
elseif xl(c) > m & xl(c) <= 1
    yl(c) = (K/6)*m^3*(1-xl(c));
    deff('dyldxl=f(yl,xl)', 'dyldxl=-(K/6)*m^3');
    dyl_dxl(c) = ode(0,0,xl(c),f);
end
theta(c)=atan(dyl_dxl(c));
end
end

/// Distribuição da espessura

ye = t.*(a0.*sqrt(xl) - a1.*xl - a2.*xl.^2 + a3.*xl.^3 - a4.*xl.^4);

/// Coordenadas superiores

xs = xl - ye.*sin(theta);
ys = yl + ye.*cos(theta);

/// Coordenadas inferiores

xi = xl + ye.*sin(theta);
yi = yl - ye.*cos(theta);

/// Plotar o aerofólio

xgrid;

if Plotarcomo == 1 then
    plotS = plot(xs,ys,'k-');
    plotI = plot(xi,yi,'k-');
    if Linha == 1 then
        plotL = plot(xl,yl,'r-');
    end
elseif Plotarcomo == 2
    plotS = plot(xs,ys,'ko');
    plotI = plot(xi,yi,'ko');
    if Linha == 1 then
        plotL = plot(xl,yl,'ro');
    end
elseif Plotarcomo == 3
    plotS = plot(xs,ys,'k. ');
    plotI = plot(xi,yi,'k. ');
    if Linha == 1 then
        plotL = plot(xl,yl,'r. ');
    end
end
end

/// Avisando o usuário que o aerofólio foi plotado

if Lingua == 1
    set(handles.textPronto,'String','Aerofólio plotado. Para deletar o aerofólio, clique em
LIMPAR GRÁFICO. ');
elseif Lingua == 2

```



```

set(handles.textPronto,'String','Airfoil plotted. To delete the airfoil, click on CLEAR GRAPH.');
```

```

elseif Lingua == 3
    set(handles.textPronto,'String','Profil tracé. Pour supprimer le profil, cliquez sur NETTOYER GRAPHIQUE.');
```

```

end

set(handles.pushLimpar,'Enable','on');
set(handles.textPronto,'ForegroundColor',[-1,-1,-1]);

// Ativando o botão 'PLOTAR AEROFÓLIO'

set(handles.pushPlotar,'Enable','on');
```

```
endfunction
```

De modo análogo às outras funções, deve-se inicialmente criar as variáveis que serão utilizadas tanto pelas equações quanto pelas estruturas condicionais. Dentre os comandos mais importantes nesta etapa estão:

- `part(Aerofolio,[x,(y),(z)])`, que seleciona o x-ésimo (y-ésimo e z-ésimo, ocasionalmente) da *string* que foi digitada pelo usuário e torna possível a transformação do código NACA nas variáveis M, P, XX, L e Q;
- `linspace(0,(1 ou %pi),Pontos)`, que cria as coordenadas `xl` dentro de um vetor;
- `ones(1, length(xl))`, que cria vetores preenchidos com 1 no tamanho do vetor de `xl`;
- `for c=1:length(xl)`, que possibilita o cálculo de `yl`, `dyl_dxl` e `theta` para cada componente do vetor, criando assim as coordenadas da linha de curvatura média (`xl,yl`);
- `deff('dyl_dxl=f(yl,xl)','dyl_dxl=...')`, que define as equações diferenciais em (4) e (6);
- `ode(0,0,xl(c),f)`, que faz o cálculo das equações diferenciais já definidas;

Na parte da função que define a plotagem dos pontos é necessário esclarecer dois fatores essenciais. O usuário tem a opção de escolher como vai plotar as coordenadas no gráfico, isto é, se utilizará pontos, linhas ou círculos. Para que isso seja possível a variável `Plotarcomo = handles.popPlotar.Value` (1 para linha, 2 para círculos e 3 para pontos) é adicionada à estruturas condicionais

que direcionam como as coordenadas serão plotadas ('-', linha; '.', pontos;'o', círculos). Além disso, o usuário pode escolher se plotará ou não a linha de curvatura média. Quando a variável `Linha = handles.radioLinha.Value` tem valor 0, o programa não plota a linha de curvatura, mas quando tem valor 1, ele plota.

Entre outras características deste código, é interessante notar que o botão PLOTAR AEROFÓLIO é desativado quando acionado e ativado quando a plotagem é finalizada, o que evita uma sobrecarga do Scilab. Outra característica interessante é que ao clicar o botão, o usuário é informado de que a interface está plotando o aerofólio, e quando os cálculos se encerram, é informado de que o aerofólio foi plotado, além de ter disponível o botão LIMPAR GRÁFICO

#### 4.2.5 Configuração do Botão para Limpar o Gráfico

O usuário tem a possibilidade de plotar diferentes aerofólios num mesmo gráfico para uma comparação, mas também pode plotar um de cada vez para uma visualização mais detalhada.

A configuração do botão LIMPAR GRÁFICO é relativamente simples. Numa explicação simplificada, ao clicar no botão, o sistema deleta todas as coordenadas que estão plotadas a partir do comando `delete(handles.plotAerofolio.children)`. Além disso, limpa todas as variáveis possivelmente armazenadas no Scilab. Ao limpar o gráfico, o usuário irá se deparar com a mesma mensagem do início do programa: “Defina os parâmetros e clique em PLOTAR AEROFÓLIO.

```
function pushLimpar_callback(handles)

    delete(handles.plotAerofolio.children);
    clear;
    clc;

    set(handles.pushLimpar,'Enable','off');
    set(handles.pushPlotar,'Enable','on');
    set(handles.textPronto,'ForegroundColor',[-1,-1,-1]);

    Lingua = handles.popLingua.Value;
    if Lingua == 1
        set(handles.textPronto,'String','Defina os parâmetros e clique em PLOTAR AEROFÓLIO.');
```

endfunction

#### 4.2.6 Definindo a Exportação do Gráfico

A partir desta interface, o usuário poderá exportar o gráfico de um aerofólio em três diferentes formatos: PDF, JPEG e PNG. Isto possibilitará que esse gráfico seja impresso, inserido em outros trabalhos ou ainda outros estudos relacionados à aerofólios.

O código da função é baseado naquele que calcula e plota o aerofólio. Porém, neste caso, é criada uma nova janela gráfica que contém apenas o gráfico, e esta é convertida para os formatos mencionados acima. Para isso, utiliza-se o comando `figure(5,'BackgroundColor',[1,1,1])`, que define uma nova janela gráfica com número de identificação 5 e cor de fundo branca. Assim como criou-se uma entidade gráfica para plotar o aerofólio na interface, ela será criada também nesta nova figura, a partir de comandos análogos.

**function** `pushExportar_callback(handles)`

Lingua = **handles**.popLingua.Value;      *// =1, português; =2, inglês; =3, francês*

*/// Criando uma nova janela*

```
e=figure(5,'BackgroundColor',[1,1,1]);
handles.plotExportar=newaxes(e);
handles.plotExportar.isoview = "on";
handles.plotExportar.tight_limits = "on";
handles.plotExportar.data_bounds = [0,-0.3;1,0.3];
handles.plotExportar.margins = [ 0 0 0 0];
handles.plotExportar.axes_bounds = [0.1,0.1,0.8,0.8];
```

*/// Informando o usuário que o aerofólio está sendo plotado*

```
if Lingua == 1
    set(handles.textPronto,'String','Exportando...');
elseif Lingua == 2
    set(handles.textPronto,'String','Exporting...');
elseif Lingua == 3
    set(handles.textPronto,'String','Exportant...');
end
```

```
set(handles.textPronto,'ForegroundColor',[-1,-1,-1]);
```

*/// Desativando o botão 'EXPORTAR'*

```
set(handles.pushExportar,'Enable','off');
```

*/// Definindo as variáveis inseridas pelo usuário*

Codigo = **handles**.popCodigo.Value;      *// =1, NACA 4 dígitos; =2, NACA 5 dígitos*

Aerofolio = **handles**.editAerofolio.String; // *Retira o código NACA (completo) inserido pelo usuário*

Pontos = strtod(**handles**.editPontos.String); // *Retorna o número de pontos a serem utilizados no gráfico*

Grade = **handles**.popGrade.Value; // =1, grade uniforme; =2, grade não uniforme

Bordo = **handles**.popBordo.Value; // =1, aberto; =2, fechado

Plotarcomo = **handles**.popPlotar.Value; // =1, linha; =2, círculos; =3, pontos

Linha = **handles**.radioLinha.Value; // =0, não mostrar linha de curvatura média; =1, mostrar linha de curvatura média

*/// Extraindo os valores do código NACA*

if Codigo == 1 then

Mi = strtod(part(Aerofolio,[1]));

Pi = strtod(part(Aerofolio,[2]));

XXi = strtod(part(Aerofolio,[3,4]));

elseif Codigo == 2 then

LPQ = strtod(part(Aerofolio,[1,2,3]));

Qi = strtod(part(Aerofolio,[3]));

XXi = strtod(part(Aerofolio,[4,5]));

*/// Para esta programação, não será possível utilizar Q diferente de 0*

if Qi ~= 0 then

if Lingua == 1

elseif Lingua == 2

elseif Lingua == 3

end

set(**handles**.textPronto,'String','Utilize um código NACA de 5 dígitos com Q=0.');

set(**handles**.textPronto,'ForegroundColor',[1,0,0]);

set(**handles**.pushPlotar,'Enable','off');

end

end

*/// Transformando os valores extraídos para serem utilizados nas fórmulas*

if Codigo == 1 then

M = Mi/100;

P = Pi/10;

t = XXi/100;

else

t = XXi/100;

end

*/// Delimitando as constantes para cada tipo de código e tipo de bordo de fuga*

a0 = 1.4845;

a1 = 0.63;

a2 = 1.758;

a3 = 1.4215;

if Bordo == 1 then

a4 = 0.5075;

else

a4 = 0.5180;

end

*/// Constantes tabuladas para o código de 5 dígitos*

```

if Codigo == 2 then
  if LPQ == 210
    m = 0.058
    K = 361.4
  elseif LPQ == 220
    m = 0.126
    K = 51.65
  elseif LPQ == 230
    m = 0.2025
    K = 15.65
  elseif LPQ == 240
    m = 0.29
    K = 6.643
  elseif LPQ == 250
    m = 0.391
    K = 3.23
  end
end

```

*/// Espaçamento dos pontos*

```

if Grade == 1 then
  xl = linspace(0,1,Pontos);
elseif Grade == 2
  beta = linspace(0,%pi,Pontos);
  xl = (0.5*(1-cos(beta)));
end

```

*/// Linha de curvatura média*

```

yl = ones(1, length(xl));
dyl_dxl = ones(1, length(xl));
theta=ones(1, length(xl));

```

*/// Para código NACA de 4 dígitos*

```

if Codigo==1 then
  for c=1:length(xl)
    if xl(c) >=0 & xl(c) < P
      yl(c) = (M/P^2).*((2*P*xl(c))-xl(c)^2);
      deff('dyl_dxl=f(yl,xl)','dyl_dxl=((2*M)/(P^2))*(P-xl)');
      dyl_dxl(c) = ode(0,0,xl(c),f);
    elseif xl(c) >= P & xl(c) <= 1
      yl(c) = (M/(1-P)^2).*(1-(2*P)+(2*P*xl(c))-xl(c)^2);
      deff('dyl_dxl=f(yl,xl)','dyl_dxl=((2*M)/((1-P)^2))*(P-xl)');
      dyl_dxl(c) = ode(0,0,xl(c),f);
    end
    theta(c) = atan(dyl_dxl(c));
  end
end

```

*/// Para código NACA de 5 dígitos*

```

if Codigo==2 then

```

```

for c=1:length(xl)
    if xl(c) >= 0 & xl(c) <= m
        yl(c) = (K/6)*((xl(c))^3-3*m*(xl(c))^2+m^2*(3-m)*(xl(c)));
        deff('dyldxl=f(yl,xl)','dyldxl=(K/6)*(3*xl^2-6*m*xl+m^2*(3-m))');
        dyl_dxl(c) = ode(0,0,xl(c),f);
    elseif xl(c) > m & xl(c) <= 1
        yl(c) = (K/6)*m^3*(1-xl(c));
        deff('dyldxl=f(yl,xl)','dyldxl=-(K/6)*m^3');
        dyl_dxl(c) = ode(0,0,xl(c),f);
    end
    theta(c)=atan(dyl_dxl(c));
end
end

```

*/// Distribuição da espessura*

```

ye = t.*(a0.*sqrt(xl) - a1.*xl - a2.*xl.^2 + a3.*xl.^3 - a4.*xl.^4);

```

*/// Coordenadas superiores*

```

xs = xl - ye.*sin(theta);
ys = yl + ye.*cos(theta);

```

*/// Coordenadas inferiores*

```

xi = xl + ye.*sin(theta);
yi = yl - ye.*cos(theta);

```

*/// Plotar o aerofólio*

```

xgrid;

if Plotarcomo == 1 then
    plotS = plot(xs,ys,'k-');
    plotI = plot(xi,yi,'k-');
    if Linha == 1 then
        plotL = plot(xl,yl,'r-');
    end
elseif Plotarcomo == 2
    plotS = plot(xs,ys,'ko');
    plotI = plot(xi,yi,'ko');
    if Linha == 1 then
        plotL = plot(xl,yl,'ro');
    end
elseif Plotarcomo == 3
    plotS = plot(xs,ys,'k. ');
    plotI = plot(xi,yi,'k. ');
    if Linha == 1 then
        plotL = plot(xl,yl,'r. ');
    end
end
end

```

*/// Exportar para o tipo de arquivo definido*

```

Arquivo = handles.popExportar.Value // =1, PDF; =2, JPG; =3, PNG

```

```

if Arquivo == 1

```

```

        xs2pdf(5,Aerofolio, 'landscape');
    elseif Arquivo == 2
        xs2jpg(5,Aerofolio, 0.95);
    elseif Arquivo == 3
        xs2png(5,Aerofolio);
    end

    /// Deletando a página criada

    delete(handles.plotExportar.children);
    delete(e);

    /// Informando que o aerofólio foi exportado

    if Lingua == 1
        set(handles.textPronto,'String','Aerofólio exportado. ');
    elseif Lingua == 2
        set(handles.textPronto,'String','Airfoil exported. ');
    elseif Lingua == 3
        set(handles.textPronto,'String','Profil exporté. ');
    end

    /// Ativando o botão "EXPORTAR"

    set(handles.pushExportar,'Enable','on');

endfunction

```

#### 4.2.7 Definindo um Botão para Sair da Interface

Por puros motivos estéticos e de curiosidade, foi inserido um botão que deleta a janela gráfica na qual está a interface, além de apagar todas as variáveis e limpar o console do Scilab.

```

function pushSair_callback(handles)

    delete(f);

    clear;

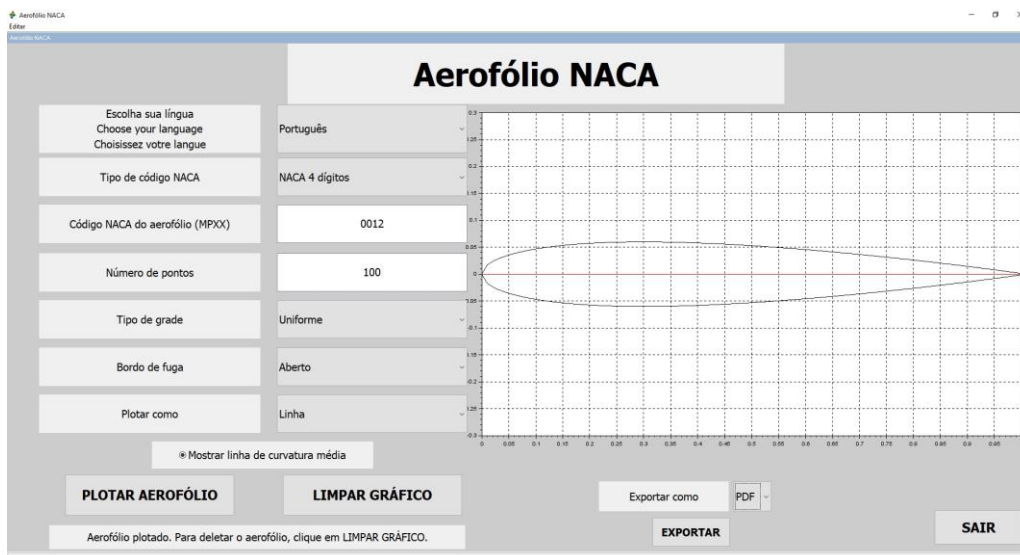
    clc;

endfunction

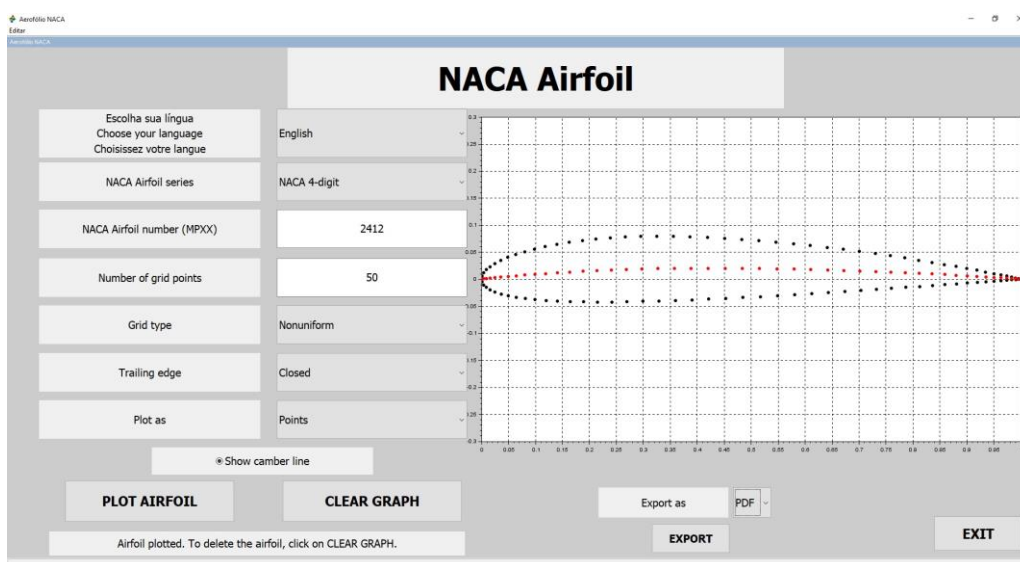
```

## 5 RESULTADOS

Os resultados obtidos são exatamente aqueles que eram esperados no início deste projeto. A partir de uma interface simples e descomplicada, o usuário pode plotar aerofólios que dependem de equações relativamente complexas em poucos segundos. Abaixo, diversas imagens representam os diferentes resultados obtidos a partir de diferentes entradas do usuário.

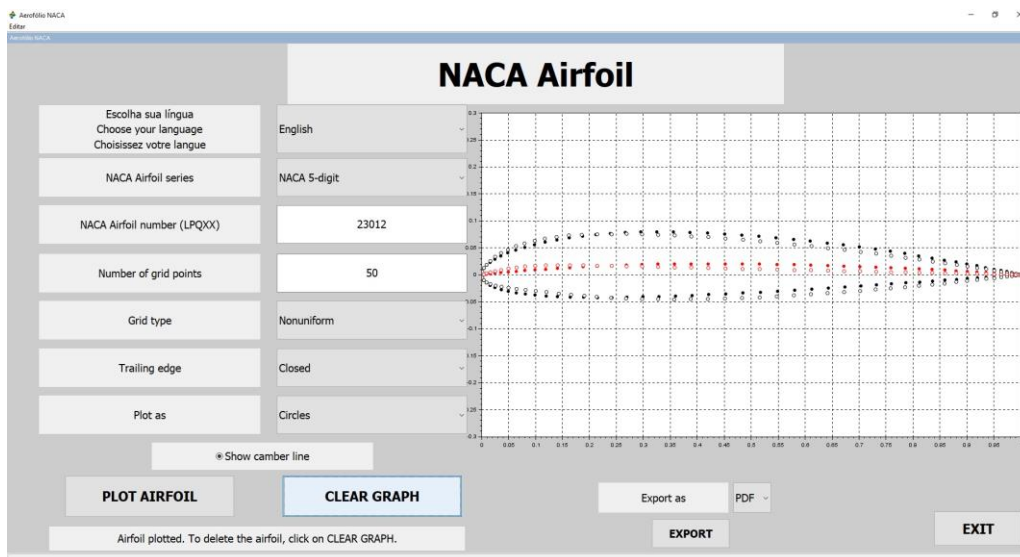


**Figura 6** – Gráfico do aerofólio NACA 0012 com 100 pontos uniformemente espaçados, bordo de fuga aberto e plotado em linhas a partir da interface em Português.

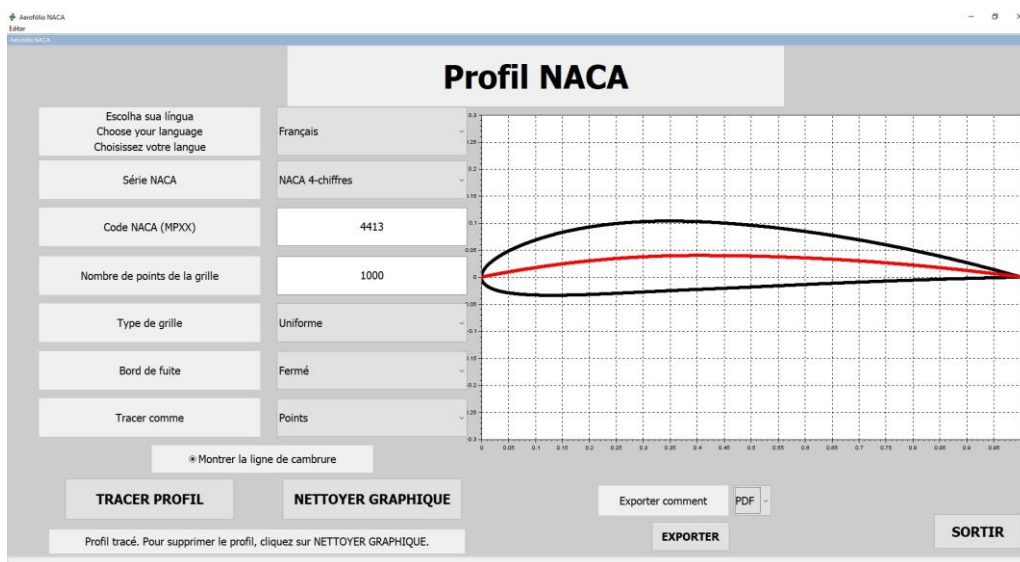


**Figura 7** – Gráfico do aerofólio NACA 2412 com 50 pontos não uniformes, bordo de fuga fechado e plotado em pontos a partir da interface em Inglês.

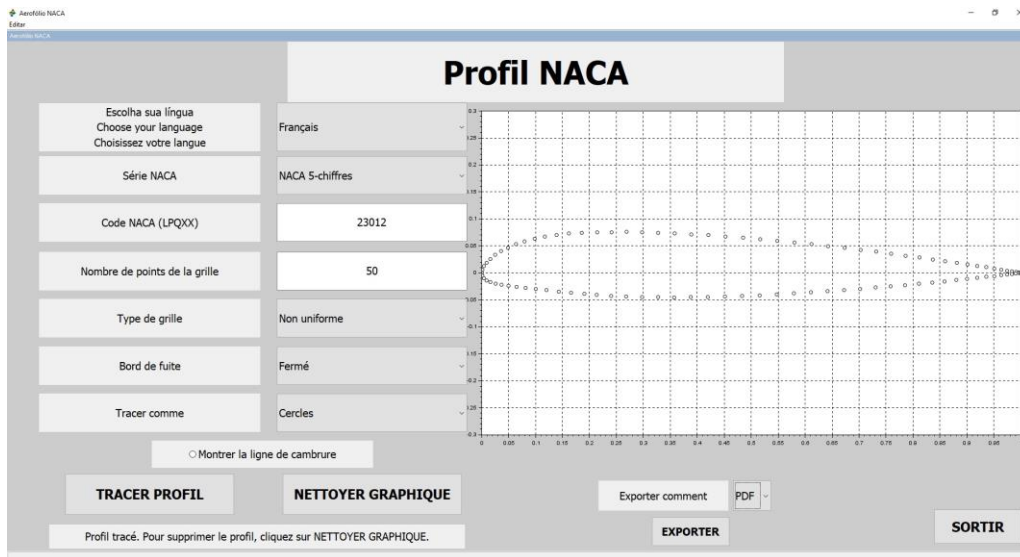




**Figura 8** – Gráfico dos aerofólios NACA 23012 e NACA 2412 com 50 pontos não uniformes, bordo de fuga fechado e plotado em círculos e pontos, respectivamente, a partir da interface em Inglês.

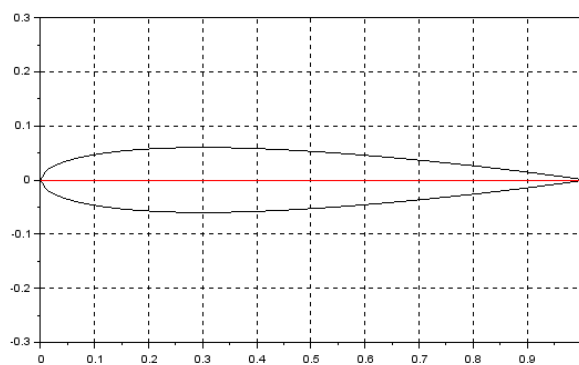


**Figura 9** – Gráfico do aerofólio NACA 4413 com 1000 pontos uniformemente espaçados, bordo de fuga fechado e plotado em pontos a partir da interface em Francês.

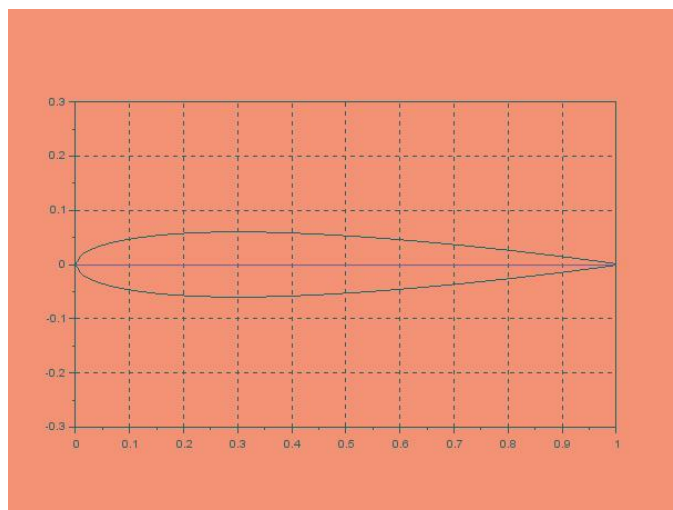


**Figura 10** – Gráfico do aerofólio NACA 23012 com 50 pontos não uniformes, bordo de fuga fechado e plotado em círculos a partir da interface em Francês.

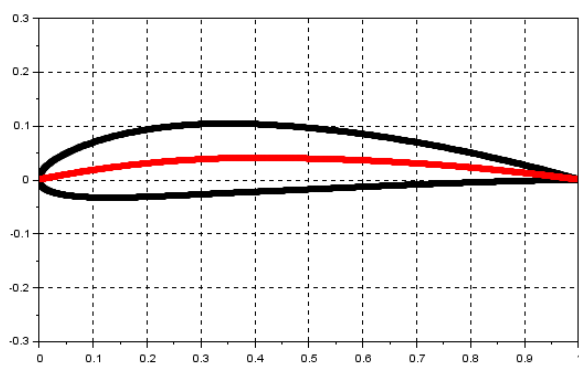
Além da possibilidade de visualizar os aerofólios dentro da interface e interagir com o gráfico, ele também pode exportá-los em arquivos PDF, JPG e PNG. A partir da interface, pode-se exportar arquivos PDF e PNG sem qualquer problema. Porém, os arquivos JPG ficam diferentes do que é esperado, o que parece ser um erro geral do Scilab. Abaixo, pode-se visualizar algumas das imagens geradas pela interface.



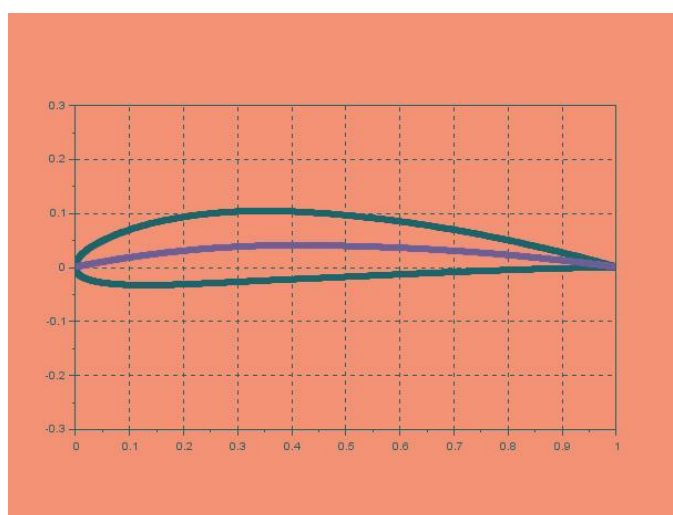
**Figura 11** – Aerofólio NACA 0012 no formato PNG.



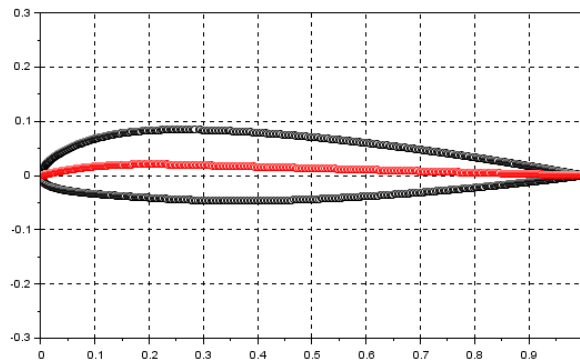
**Figura 12** – Aerofólio NACA 0012 no formato JPG.



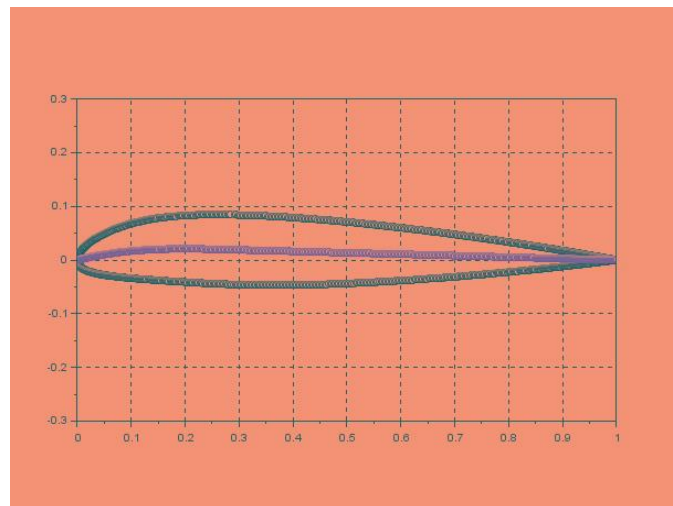
**Figura 13** – Aerofólio NACA 4413 no formato PNG.



**Figura 14** – Aerofólio NACA 4413 no formato JPG.



**Figura 15** – Aerofólio NACA 24013 no formato PNG.



**Figura 16** – Aerofólio NACA 24013 no formato JPG.

## 6 DISCUSSÃO

Os resultados obtidos são completamente coerentes com a literatura utilizada como fonte para este trabalho. O livro *Theory of Wing Sections* inclui um sumário com os dados e gráficos de alguns aerofólios NACA. Ao comparar-se os aerofólios obtidos a partir da interface com aqueles presentes no livro, percebe-se que são idênticos. Abaixo são mostradas algumas imagens de alguns dos aerofólios presentes no livro e, em seguida, os obtidos pela interface.

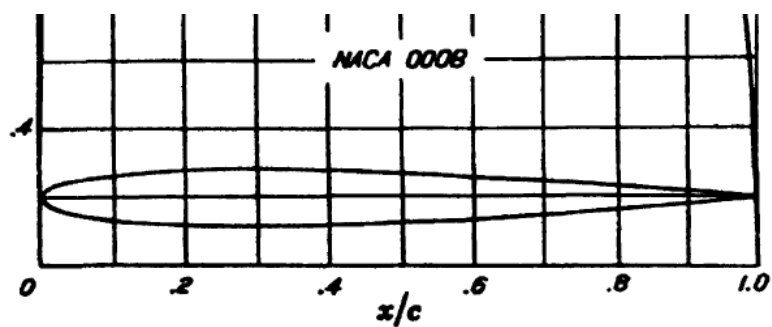


Figura 17 – Gráfico do aerofólio NACA 0008 apresentado em *Theory of Wing Sections*.

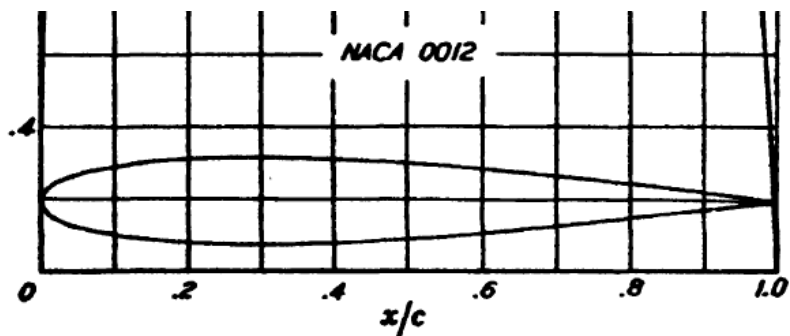


Figura 18 – Gráfico do aerofólio NACA 0012 apresentado em *Theory of Wing Sections*.

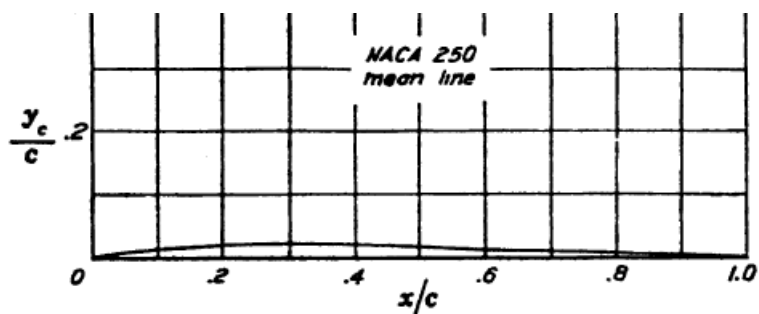


Figura 19 – Gráfico da linha de curvatura média para aerofólio NACA 5 dígitos com LPQ=250 apresentado em *Theory of Wing Sections*.

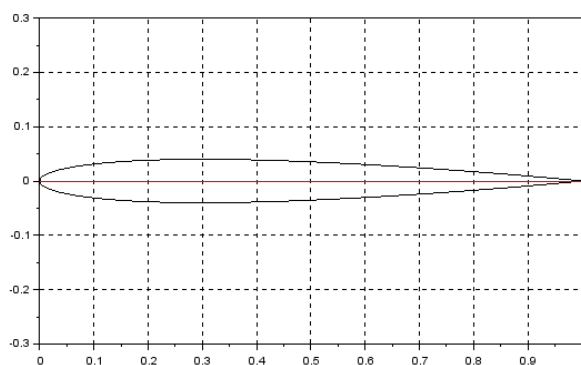
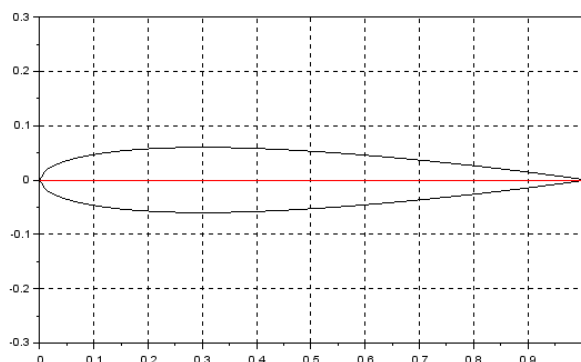
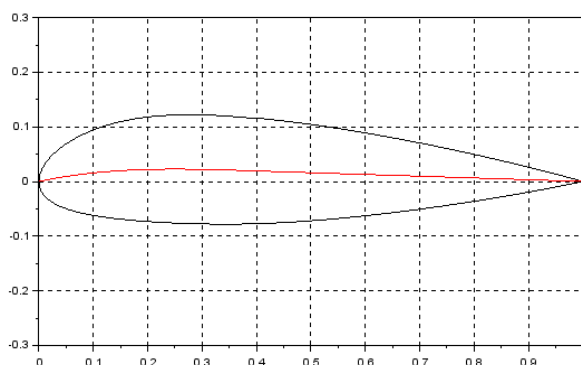


Figura 20 – Gráfico do aerofólio NACA 0008 obtido pela interface.



**Figura 21** – Gráfico do aerofólio NACA 0012 obtido pela interface.



**Figura 22** – Gráfico do aerofólio NACA 25020 obtido pela interface. A linha destacada em vermelho é a linha de curvatura média quando  $LPQ=250$  nos aerofólios NACA 5 dígitos.

## 7 CONCLUSÃO

A partir dos resultados e comparações com a literatura vigente, percebe-se que a interface, dentro das limitações do Scilab, é capaz de calcular e plotar sem qualquer problema as coordenadas de um aerofólio, cujas fórmulas são relativamente complexas e tornam sua visualização mais difícil.

No apêndice A está disponibilizada uma tabela com algumas aeronaves que utilizam aerofólios NACA em suas asas e os respectivos códigos e gráficos, estes obtidos pela interface aqui descrita.

## 7.1 DIFICULDADES ENCONTRADAS

Durante este trabalho foram várias as dificuldades encontradas. Primeiramente, para se aprender a utilizar a extensão GUI Builder é necessário dedicação e muita pesquisa sobre o assunto, já que não é algo trivial e não foi ensinado durante as aulas da disciplina Bases Computacionais da Ciência.

Outra grande dificuldade é trabalhar com *strings*, que devem sempre ser convertidas, e em certos casos divididas, para que possam ser utilizadas pelo programa.

É importante citar ainda a dificuldade em se utilizar equações diferenciais dentro do Scilab sem nunca ter visto algo do tipo. Aproveito este espaço para agradecer ao docente Harlen Costa Batagelo pela sua imensa ajuda nessa etapa crucial do projeto.

Por fim, destaco a complexidade em se trabalhar com figuras em geral, desde aquelas confeccionadas pela extensão GUI Builder até as que foram necessárias para a exportação do gráfico.

Apesar dessas dificuldades, o projeto foi concluído com sucesso após muita dedicação e estudo sobre os assuntos que este envolve.

## 7.2 TRABALHOS FUTUROS

Este projeto, por mais extenso que seja, está longe de ser completo. Dentre os trabalhos futuros que deseja-se realizar com ele está a possibilidade de exportar as coordenadas obtidas para que possam ser utilizadas em programas de simulação computacional e 3D, como o AutoCAD e CFD (*Computational fluid dynamics*). Além disso, pretende-se adicionar outros tipos aerofólios, como NACA 4 dígitos modificado, NACA 5 dígitos modificado e NACA 6 dígitos.

## REFERÊNCIAS BIBLIOGRÁFICAS

ABBOTT, Ira H; VON DOENHOFF, Albert E. **Theory of Wing Sections:** including a summary of airfoil data. New York: Dover Publications, 1959. 693 p.

CANAL PILOTO. **Teorias Rotativas 02.** Disponível em: <<http://canalpiloto.com.br/teorias-rotativas-02/>>. Acesso em: 10 ago. 2018.

CUMMINGS, Russel M, et al. Geometry for Aerodynamicists. In: CUMMINGS, Russel M, et al. **Applied Computational Aerodynamics: A Modern Engineering Approach.** New York: Cambridge University Press, 2015. Appendix A.

HOMA, Jorge. **Aerodinâmica e Teoria de Voo:** noções básicas. 36 ed. São Paulo: ASA, 2017.

JOSH THE ENGINEER. **Explained: NACA 4-Digit Airfoil MATLAB GUI.** Disponível em: <<http://www.joshtheengineer.com/2015/05/16/explained-naca-4-digit-airfoil-matlab-gui/>>. Acesso em: 10 ago. 2018.

SCILAB. **Ajuda do Scilab.** Disponível em: <[https://help.scilab.org/docs/5.5.2/pt\\_BR/index.html](https://help.scilab.org/docs/5.5.2/pt_BR/index.html)>. Acesso em: 10 ago. 2018.


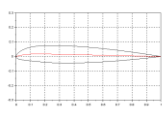
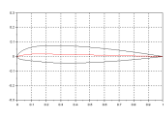

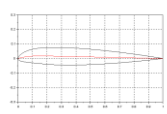
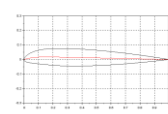

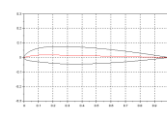

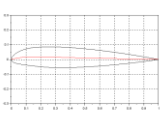
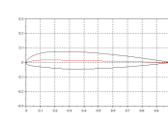

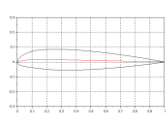
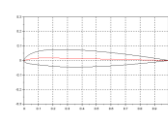

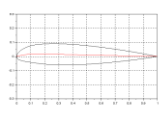
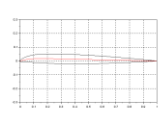

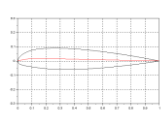
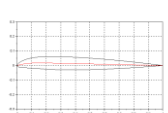

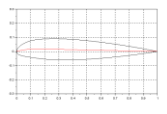
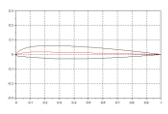

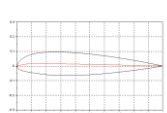
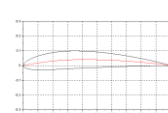

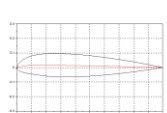
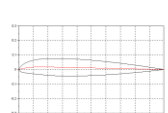

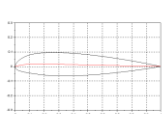
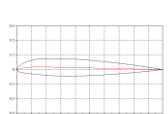

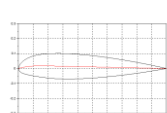
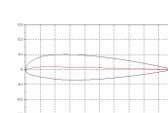
UIUC AIRFOIL DATA SITE. **The Incomplete Guide to Airfoil Usage.** Disponível em: <<http://m-selig.ae.illinois.edu/ads.html>>. Acesso em: 10 ago. 2018.


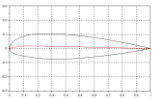
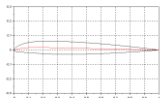

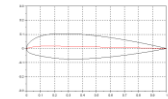
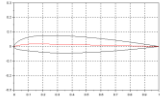

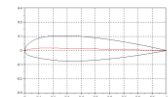
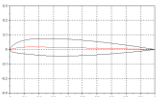

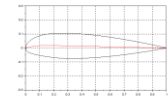
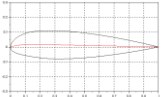

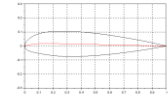
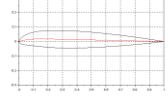

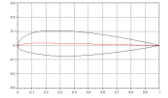
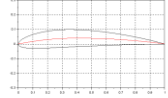
X-ENGINEER. **How to solve an ordinary differential equation (ODE) in Scilab.** Disponível em: <<https://x-engineer.org/graduate-engineering/programming-languages/scilab/solve-ordinary-differential-equation-ode-scilab/>>. Acesso em: 10 ago. 2018.



# APÊNDICE A

Aeronave		Raiz da asa (ou pás)	Ponta da asa (ou pás)	Fonte da imagem
	Aerospatiale AS-550	 0012	 0012	<a href="https://www.militaryaircraft.de/pictures/military/helicopter/AS-550/AS-550C2_Fennec_RIAT2005_001_800.jpg">https://www.militaryaircraft.de/pictures/military/helicopter/AS-550/AS-550C2_Fennec_RIAT2005_001_800.jpg</a>
	Agusta-Bell 212	 0012	 0006	<a href="https://afm.gov.mt/en/equipment/aircraft/agustabell212/Pages/212.aspx">https://afm.gov.mt/en/equipment/aircraft/agustabell212/Pages/212.aspx</a>
	Boeing 314 Clipper (C-98)	 0018	 0010	<a href="https://airandspace.si.edu/collection-objects/boeing-model-314-clipper-california-clipper-airlines-pan-american-airways-pan-1">https://airandspace.si.edu/collection-objects/boeing-model-314-clipper-california-clipper-airlines-pan-american-airways-pan-1</a>
	Convair 3 R3Y Tradewind	 1420	 4412	<a href="http://anigrand.com/AA4063_R3Y-2.htm">http://anigrand.com/AA4063_R3Y-2.htm</a>
	Supermarine Spitfire	 2213	 2209	<a href="https://en.wikipedia.org/wiki/Supermarine_Spitfire">https://en.wikipedia.org/wiki/Supermarine_Spitfire</a>
	Aeronca LC	 2218	 2202	<a href="https://www.john2031.com/aeronca/aeronca_lc_airplane/pictures_photos_information.html">https://www.john2031.com/aeronca/aeronca_lc_airplane/pictures_photos_information.html</a>
	Cessna 152	 2412	 2412	<a href="http://bodminairfield.com/product/cessna-152-flight-experience/">http://bodminairfield.com/product/cessna-152-flight-experience/</a>
	Aeronca 15-AC Sedan	 4412	 4412	<a href="https://www.flyer.co.uk/1947-aeronca-sedan-back-production/">https://www.flyer.co.uk/1947-aeronca-sedan-back-production/</a>
	AAI Shadow 200	 4415	 4415	<a href="https://www.suasnews.com/knowledge-centre/drone-systems/aai-rq-7-shadow-200/">https://www.suasnews.com/knowledge-centre/drone-systems/aai-rq-7-shadow-200/</a>
	Anderson Greenwood 14	 4418	 4418	<a href="http://www.airliners.net/photo/Untitled/Anderson-Greenwood-AG-14/2494686">http://www.airliners.net/photo/Untitled/Anderson-Greenwood-AG-14/2494686</a>
	Aeroprogress T-411 Aist-2	 23011	 23011	<a href="https://doc8643.com/aircraft/T411">https://doc8643.com/aircraft/T411</a>

	Aero Boero 115	 23012	 23012	<a href="http://www.airliners.net/photo/Aeroclube-do-Rio-Grande-do-Sul-ARGS/Aero-Boero-AB-115/815196">http://www.airliners.net/photo/Aeroclube-do-Rio-Grande-do-Sul-ARGS/Aero-Boero-AB-115/815196</a>
	Beagle D.5/180	 23012	 23012	<a href="https://abpic.co.uk/pictures/mo del/Beagle%20D.5x180%20Husky">https://abpic.co.uk/pictures/mo del/Beagle%20D.5x180%20Husky</a>
	Augusta 109	 23012	-	<a href="http://exclusiveaircraft.co.uk/helicopter-sales/agusta-109-elite-for-sale-">http://exclusiveaircraft.co.uk/helicopter-sales/agusta-109-elite-for-sale-</a>
	Cessna 501 Citation I	 23014	 23012	<a href="https://www.planespotters.net/photo/464373/n82aj-southern-aircraft-consultancy-cessna-501-citation-isp">https://www.planespotters.net/photo/464373/n82aj-southern-aircraft-consultancy-cessna-501-citation-isp</a>
	Cessna 551 Citation II	 23014	 23012	<a href="https://www.planespotters.net/photo/149572/d-irup-private-cessna-551-citation-isp">https://www.planespotters.net/photo/149572/d-irup-private-cessna-551-citation-isp</a>
	Aichi D3A1	 23015	 23007	<a href="http://www.airpages.ru/eng/img/d3a1_2.shtml">http://www.airpages.ru/eng/img/d3a1_2.shtml</a>
	Goodyear FG-1 Corsair	 23015	 23009	<a href="http://www.vintagewings.ca/Aircraft/tabid/66/articleType/ArticleView/articleid/8/The-Robert-Hampton-Gray-Goodyear-FG-1D-Corsair.aspx">http://www.vintagewings.ca/Aircraft/tabid/66/articleType/ArticleView/articleid/8/The-Robert-Hampton-Gray-Goodyear-FG-1D-Corsair.aspx</a>
	Piper PA-46 Malibu	 23015	 23009	<a href="https://barrieaircraft.com/photo/piper-pa-46-malibumalibumirage-02.html">https://barrieaircraft.com/photo/piper-pa-46-malibumalibumirage-02.html</a>
	AerMacchi AM-3	 23016	 4412	<a href="http://www.flugzeuginfo.net/acdata_php/acdata_am3_en.php">http://www.flugzeuginfo.net/acdata_php/acdata_am3_en.php</a>
	Embraer 110 Bandeirante	 23016	 23012	<a href="https://www.planespotters.net/photo/370477/pj-vic-insel-air-embraer-emb-110-bandeirante">https://www.planespotters.net/photo/370477/pj-vic-insel-air-embraer-emb-110-bandeirante</a>
	Embraer 121 Xingu	 23016	 23012	<a href="https://www.rockaircraft.com/2016/02/emb-121-xingu-o-primeiro-executivo.html">https://www.rockaircraft.com/2016/02/emb-121-xingu-o-primeiro-executivo.html</a>
	Grumman G-111 Albatross	 23017	 23017	<a href="http://www.airliners.net/photo/Chalk-s-International-Airlines/Grumman-G-111-Albatross/1206102">http://www.airliners.net/photo/Chalk-s-International-Airlines/Grumman-G-111-Albatross/1206102</a>

	Abrams Explorer	 23018	 23009	<a href="https://airandspace.si.edu/collection-objects/abrams-explorer">https://airandspace.si.edu/collection-objects/abrams-explorer</a>
	Armstrong Whitworth AW.650 Argosy	 23018	 23012	<a href="http://www.airwar.ru/enc/craft/argosy.html">http://www.airwar.ru/enc/craft/argosy.html</a>
	Beech 200 Super King Air (C-12)	 23018	 23012	<a href="https://www.airteamimages.com/beechnair-king-air_N57FT_-_private_145316_large.html">https://www.airteamimages.com/beechnair-king-air_N57FT_-_private_145316_large.html</a>
	Cessna 441 Conquest II	 23018	 23019	<a href="http://www.airliners.net/photo/Untitled/Cessna-441-Conquest-II/1587732">http://www.airliners.net/photo/Untitled/Cessna-441-Conquest-II/1587732</a>
	Embraer 120 Brasilia	 23018	 23012	<a href="https://pt.wikipedia.org/wiki/Ficheiro:Embraer_C-97_Brasilia_EMB-120RT_Lofting.jpg">https://pt.wikipedia.org/wiki/Ficheiro:Embraer_C-97_Brasilia_EMB-120RT_Lofting.jpg</a>
	Lockheed 1049 Constellation	 23018	 4412	<a href="https://www.air-and-space.com/Lockheed%20Constellation%20survivors.htm">https://www.air-and-space.com/Lockheed%20Constellation%20survivors.htm</a>