

# amiet\_tools: um pacote em Python para predição de ruído de interação turbulência-aerofólio

## EM CONSTRUÇÃO

Fabio Casagrande Hirono  
fchirono@gmail.com

Janeiro 2021

### Resumo

O pacote `amiet_tools` (AmT), escrito em linguagem Python, é uma implementação do modelo analítico de Amiet [1] para predição de ruído de interação turbulência-aerofólio, com extensões. As funções permitem o cálculo do “salto” de pressão na superfície do aerofólio (i.e. a distribuição da fonte acústica) gerado em resposta à turbulência incidente, e do campo acústico radiado por esta interação. A turbulência incidente pode ser uma única rajada senoidal, ou uma soma de rajadas incoerentes com amplitudes definidas por um espectro energético. Outras funções incluem o modelamento de efeitos de convecção e refração na propagação sonora para modelar medições acústicas realizadas em túneis de vento abertos ou fechados. Uma publicação de referência está disponível [2], e o pacote pode ser encontrado através do link [https://github.com/fchirono/amiet\\_tools](https://github.com/fchirono/amiet_tools).

## 1 Introdução

Ruído de aerofólio: importância, mecanismos físicos, modelo analítico.

Referências teóricas:

- R. Amiet, “*Acoustic radiation from an airfoil in a turbulent stream*”, Journal of Sound and Vibration, Vol. 41, No. 4:407–420, 1975 [1];
- G. Reboul, “*Modélisation du bruit à large bande de soufflante de turboréacteur*”, PhD Thesis, Laboratoire de Mécanique des Fluides et d’Acoustique - École Centrale de Lyon, Lyon - France, 2010 [3];
- M. Roger, “*Broadband noise from lifting surfaces: Analytical modeling and experimental validation*”, in: R. Camussi (Ed.), “*Noise Sources in Turbulent Shear Flows: Fundamentals and Applications*”, Springer-Verlag, 2013 [4];
- L. de Santana, “*Semi-analytical methodologies for airfoil noise prediction*”, PhD Thesis, Faculty of Engineering Sciences - Katholieke Universiteit Leuven, Leuven, Belgium, 2015 [5];
- F. Casagrande Hirono, “*Far-Field Microphone Array Techniques for Acoustic Characterisation of Aerofoils*”, PhD Thesis, Institute of Sound and Vibration Research, University of Southampton, Southampton - UK, 2018 [6].

## 2 Usando o pacote

### 2.1 Requerimentos

O pacote `amiet_tools` (abreviado por “AmT”) foi desenvolvido em Python 3.x, e tem como dependências os pacotes `numpy` e `scipy`. Ambos os pacotes estão inclusos na “*Anaconda Python Distribution*”<sup>1</sup>, uma distribuição Python gratuita e de código aberto. A distribuição Anaconda é usada para desenvolver o pacote AmT, e é a distribuição recomendada para usar AmT.

### 2.2 Pseudocódigo

O pseudocódigo usado para calcular o espectro cruzado  $S_{\Delta p \Delta p'}$  da pressão de superfície do aerofólio e/ou o espectro cruzado  $S_{pp'}$  do ruído radiado é mostrado abaixo, retirado de [2]. A maior parte das instruções mostradas abaixo possuem funções Python equivalentes ou métodos associados no pacote AmT, e não precisam ser implementadas pelo usuário final. Para mais detalhes sobre o uso destas funções, veja a documentação e os scripts de exemplo no pacote.

---

**Algorithm 1** Pseudocódigo para o modelo de predição de ruído de interação turbulência-aerofólio do pacote `amiet_tools`

---

```
1: Definir as variáveis do ambiente de teste:  $c_0, \rho_0, p_{ref}, U_x, \overline{w^2}, \Lambda, z_{sl}, M_x, \beta$ 
2: Definir as variáveis da geometria do aerofólio:  $b, d, N_x, N_y, \Delta x_s, \Delta y_s$ 
3: Calcular a amostragem da corda do aerofólio  $x_s[n]$ ,  $n \in [1, \dots, N_x]$ 
4: Calcular a amostragem da envergadura do aerofólio  $y_s[n]$ ,  $n \in [1, \dots, N_y]$ 
5: Definir as coordenadas dos observadores  $\mathbf{r}_m = (x_m, y_m, z_m)$ ,  $m \in [1, \dots, M]$ 
6: if usando correção de camada cisalhante then
7:   for all pares  $(m, n)$  de pontos observador-aerofólio  $(\mathbf{r}_m | \mathbf{r}_s)$  do
8:     Calcular os pontos de cruzamento da camada cisalhante  $\mathbf{r}_l(m, n)$  e tempos de propagação  $\tau_{sm}(m, n)$ 
9:   end for
10: end if
11: for cada frequência  $f \in [f_{min}, \dots, f_{max}]$  do
12:   Calcular  $k_0, \kappa_\chi, k_\psi^{crit}$ 
13:   Calcular valores de números de onda de rajada  $k_\psi \in [-k_\psi^{max}, k_\psi^{max}]$ 
14:   Calcular intervalo de amostragem dos números de onda de rajada  $\Delta k_\psi$ 
15:   Calcular espectro de energia dos números de onda de rajada  $\Phi_{ww}(k_\psi)$ 
16:   Calcular matriz  $\partial G(\mathbf{r}_m | \mathbf{r}_s) / \partial z_s$  para todos os pares  $(m, n)$  de pontos observador-aerofólio;
17:   Inicializar  $S_{\Delta p \Delta p'} \leftarrow 0$ 
18:   Inicializar  $S_{pp'} \leftarrow 0$ 
19:   for cada rajada  $k_\psi \in [-k_\psi^{max}, k_\psi^{max}]$  do
20:      $w(k_\psi) \leftarrow \sqrt{\Phi_{ww}(k_\psi)}$ 
21:     Calcular  $\Delta p(x_s, y_s, k_\psi)$  usando  $w(k_\psi)$ 
22:      $S_{\Delta p \Delta p'}(\mathbf{r}_s, \mathbf{r}'_s) \leftarrow S_{\Delta p \Delta p'}(\mathbf{r}_s, \mathbf{r}'_s) + [\Delta p(x_s, y_s, k_\psi) \cdot \Delta p^*(x'_s, y'_s, k_\psi)] \cdot U_x \cdot \Delta k_\psi$ 
23:   end for
24:    $S_{pp'}(\mathbf{r}_m, \mathbf{r}_{m'}) \leftarrow 4\pi [S_{\Delta p \Delta p'} \cdot (\Delta x_s \cdot \Delta y_s) \cdot (\Delta x'_s \cdot \Delta y'_s)] \cdot [\partial G(\mathbf{r}_m | \mathbf{r}_s) / \partial z_s] \cdot [\partial G(\mathbf{r}_{m'} | \mathbf{r}'_s) / \partial z_s]$ 
25: end for
```

---

<sup>1</sup>Disponível em <https://www.anaconda.com/products/individual>.

## 2.3 Inserção de dados usando classes

O pacote `amiet_tools` utiliza classes - i.e. estruturas de abstração usadas em programação orientada a objetos - para armazenar dados relacionados aos diferentes aspectos da simulação. As três classes utilizadas em AmT estão descritas abaixo, seguidas de uma lista de seus atributos:

- Classe `TestSetup`:
  - `TestSetup.c0`: velocidade do som em um meio em repouso  $c_0$  (em m/s);
  - `TestSetup.rho0`: densidade do ar  $\rho_0$  (em kg/m<sup>3</sup>);
  - `TestSetup.p_ref`: pressão acústica de referência  $p_{ref} = 20\mu\text{Pa}$  RMS;
  - `TestSetup.Ux`: velocidade média do escoamento  $U_x$  (em m/s);
  - `TestSetup.turb_intensity`: intensidade da turbulência  $\overline{w^2}/U_x$ ;
  - `TestSetup.length_scale`: escala de comprimento de turbulência  $\Lambda$  (em m);
  - `TestSetup.z_sl`: altura da camada cisalhante  $z_{sl}$  (em m);
  - `TestSetup.flow_dir`: caractere único indicando a direção do escoamento (i.e. ‘x’ para indicar escoamento na direção  $+x$ );
  - `TestSetup.dipole_axis`: caractere único indicando a direção do eixo dos dipolos (i.e. ‘z’ para dipolos apontando “para cima”, na direção  $+z$ );
  - `TestSetup.Mach`: número de Mach do escoamento  $M_x = U_x/c_0$ ;
  - `TestSetup.beta`: fator de Prandtl-Glauert  $\beta = \sqrt{1 - M_x^2}$ .
- Classe `AirfoilGeom`:
  - `AirfoilGeom.b`: semi corda do aerofólio  $b = c/2$  (em m);
  - `AirfoilGeom.d`: semi envergadura do aerofólio  $d = L/2$  (em m);
  - `AirfoilGeom.Nx`: número de amostras sobre a corda  $N_x$ , amostragem não-uniforme;
  - `AirfoilGeom.Ny`: número de amostras sobre a envergadura  $N_y$ , amostragem uniforme;
  - `AirfoilGeom.dx`: intervalos de amostragem sobre a corda  $\Delta x_s$  (calculado internamente);
  - `AirfoilGeom.dy`: intervalo de amostragem sobre a envergadura  $\Delta y_s$  (calculado internamente).
- Classe `FrequencyVariables`:
  - `FrequencyVariables.freq`: frequência  $f$  (em Hz);
  - `FrequencyVariables.k0`: número de onda acústico  $k_0 = 2\pi f/c_0$ ;
  - `FrequencyVariables.Kx`: número de onda de rajada na direção da corda  $\kappa_x = \omega/U_x$ ;
  - `FrequencyVariables.Ky_crit`: número de onda crítico de rajada na direção da envergadura  $k_\psi^{crit}$ .

Uma instância de cada classe é criada para armazenar os valores das variáveis relacionados às condições de um experimento e para passá-las às diferentes funções no pacote. Valores numéricos podem ser atribuídos diretamente através de um script em Python, ou podem ser lidos em um arquivo externo através das funções `loadTestSetup` e `loadAirfoilGeom`.

Uma quarta classe chamada `MicArrayCsmHDF5` também está inclusa no pacote, e permite a leitura e escrita de dados de CSM de arranjos de microfones no formato HDF5 adotado pela comunidade “Array Methods” [7, 8] para compartilhar dados de “beamforming”.

## Referências

- [1] R. K. Amiet. Acoustic radiation from an airfoil in a turbulent stream. *Journal of Sound and Vibration*, 41, No. 4:407–420, 1975.
- [2] Fabio Casagrande Hirono, Phillip Joseph, and Filippo Fazi. An open-source implementation of analytical turbulence-airfoil interaction noise model. In *26th AIAA/CEAS Aeroacoustics Conference*, 2020. URL <https://doi.org/10.2514/6.2020-2544>. AIAA Paper 2020-2544.
- [3] Gabriel Reboul. *Modélisation du bruit à large bande de soufflante de turboréacteur*. PhD Thesis, Laboratoire de Mécanique des Fluides et d’Acoustique - École Centrale de Lyon, Lyon - France, 2010.
- [4] Michel Roger. Broadband noise from lifting surfaces: Analytical modeling and experimental validation. In Roberto Camussi, editor, *Noise Sources in Turbulent Shear Flows: Fundamentals and Applications*. Springer-Verlag Wien, 2013.
- [5] Leandro de Santana. *Semi-analytical methodologies for airfoil noise prediction*. PhD Thesis, Faculty of Engineering Science - Katholieke Universiteit Leuven, Leuven - Belgium, 2015.
- [6] Fabio Casagrande Hirono. *Far-Field Microphone Array Techniques for Acoustic Characterisation of Aerofoils*. PhD Thesis, Institute of Sound and Vibration Research - University of Southampton, Southampton - UK, 2018.
- [7] Christopher Bahr, William Humphreys Jr., Daniel Ernst, Thomas Ahlefeldt, Carsten Spehr, Antonio Pereira, Quentin Leclerc, Christophe Picard, Ric Porteous, Danielle Moreaux, Jeffrey Fischer, and Con Doolan. A comparison of microphone phased array methods applied to the study of airframe noise in wind tunnel testing. In *23rd AIAA/CEAS Aeroacoustics Conference*, 2017. AIAA Paper AIAA 2017-3718.
- [8] Ennes Sarradj, Gert Herold, Pieter Sijtsma, Roberto Merino-Martinez, Anwar Malgoezar, Mirjam Snellen, Thomas Geyer, Christopher Bahr, Ric Porteous, Danielle Moreau, and Con Doolan. A microphone array method benchmarking exercise using synthesized input data. In *23rd AIAA/CEAS Aeroacoustics Conference*, 2017. AIAA Paper AIAA 2017-3719.