von Karman Institute for Fluid Dynamics

Lecture Series 2025-01

MEASUREMENT, MODELLING AND SIMULATIONS TECHNIQUES FOR NOVEL AIRCRAFT ARCHITECTURES

November 4-6, 2024

ADVANCED PROCESSING TECHNIQUES AND DATA ASSIMILATION

R. Camussi & U. Iemma
Università Roma Tre DICITA, Italy

1. Introduction

These notes report an overview of advanced methodologies used for the post processing of data in aeroacoustics including novel approaches for data assimilation.

The signal-processing world is wide and complex, therefore we will limit to describe some specific features that have been applied within the framework of the ENODISE project. The word "Advanced" usually refers to methodologies that go beyond the standard Fourier analysis or the space-time correlations. In this Lecture the focus is upon the so-called Time-Frequency decomposition techniques with particular emphasis towards the Wavelet transform. A technique called Matching Pursuit, novel in the field of aeroacoustics, is also described as an alternative tool for efficient time-frequency or space-wavenumber decomposition. Examples related to propeller noise are presented as well in the final part of the notes.

For the sake of conciseness, no background concepts related to Fourier series or Fourier transform (including power spectra or coherence) are provided therein. We remind to specialized textbooks for the details about these arguments.

2. Advanced signal processing based on time-frequency decompositions

2.1 Heisenberg principle and time-frequency representation

Signal processing often consists in representing a complex signal through a proper combination of simpler functions. This projection is carried out with the scope of extracting details that are not known or achievable by the simple observation of the signal itself in the physical domain.

This approach requires the definition of an analysing function that is chosen on the basis of the information that we wish to extract from the signal.

Consider a mono-dimensional case with independent variable x. For a given signal $u(x)$ we can define an analysing function $\Theta(x, p_n)$ and project $u(x)$ over Θ through a convolution product:

$$T_u(p_n) = \int u(x)\ \Theta^*(x, p_n)\, dx$$

$T_u(p_n)$ is the transform of the signal and p_n are the parameters of the transform.

$\Theta(x, p_n)$ is selected on the basis of the required information to be extracted from $u(x)$. The most common choices are the following:

- *Shannon decomposition*: the purpose is to extract information as much as possible detailed in the physical domain. This objective can be accomplished by selecting the analysing function as a Dirac pulse in the physical space at x_0, that is:

$\Theta(x, x_0) = \delta(x - x_0), p_n = x_0$

This implies that the analysing function is constant in the Fourier domain:

$\widehat{\Theta}(k, x_0) = \text{cost}$

with k representing wavenumbers.

- *Fourier decomposition*: the purpose is to extract information as much as possible detailed in the Fourier domain. This objective can be achieved by selecting the analysing function as a Dirac pulse in the Fourier space at a certain wavenumber $\pm k_0$ that corresponds to a trigonometric function with wavenumber k_0 .

  Formally, this procedure reads as follows:

  $\Theta(x, k_0) = \frac{1}{\sqrt{2\pi}} e^{-ik_0 x}, \widehat{\Theta}(k, k_0) = \delta(k - k_0), p_n = k_0$

The Shannon decomposition provides the maximum accuracy Δx in x whereas the accuracy Δk in k is completely loss. Therefore, with the Shannon decomposition:

$$\Delta x \to 0, \qquad \Delta k \to \infty$$

This representation is sketched in Figure 1.

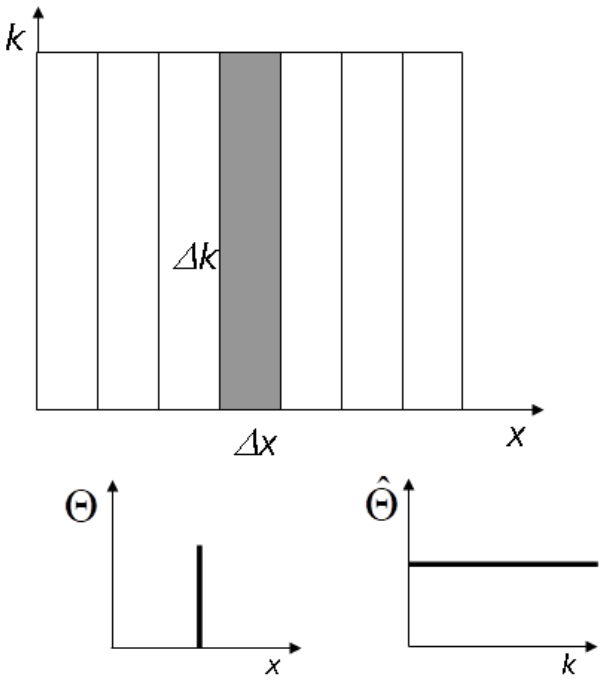


Figure 1: A simplified scheme illustrating the Shannon decomposition. The upper plot denotes the resolution in the *x*-*k* space whereas the lower plots are the analyzing function in the physical and transformed domain.

The opposite happens for the Fourier decomposition. The accuracy is optimal in k and is completely lost in x:

$$\Delta x \to \infty, \qquad \Delta k \to 0$$

This representation is sketched in Figure 2.

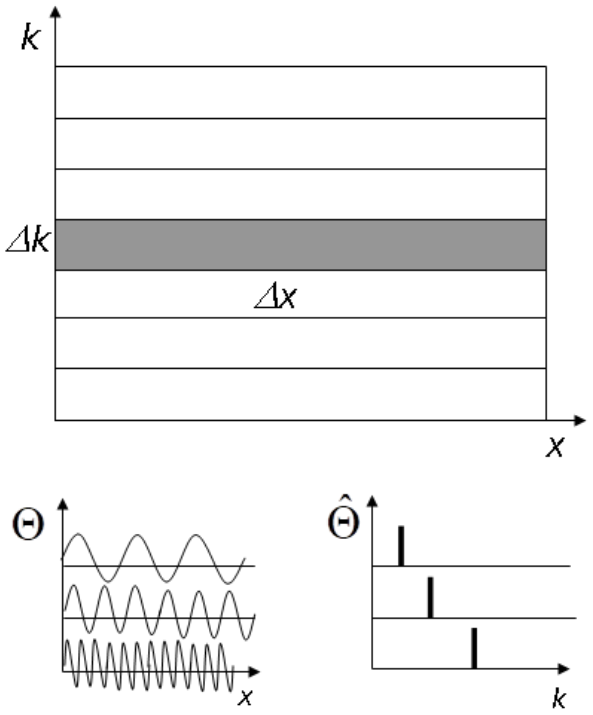


Figure 2: Same as Figure 1 but for the Fourier decomposition.

The ideal transform should provide an optimal resolution in both the physical and the Fourier space, that is:

$$\Delta x \to 0, \qquad \Delta k \to 0$$

But this is impossible! The constrain comes from the Heisenberg uncertainty principle originally providing a prescription for the position and momentum of a quantum object (Heisenberg, [1]). It is not possible to obtain a perfect resolution simultaneously in the physical and the transformed space or, quoting Cohen [2], "one can't simultaneously have good frequency resolution and good time resolution".

A compromise can be achieved by considering analysing functions that have a "good" resolution in both the physical and Fourier transformed space. This is the case of the space-wavenumber of time-frequency representation. In the following sections three data processing techniques providing such a representation are briefly worked out.

To the purpose of correlating the theory with the examples that will be presented in Sec. 2.5, hereinafter we will refer to time-frequency representations rather than space-wavenumbers.

This means that we will consider only temporal signals, time series, represented in the Fourier domain in terms their frequency content.

2.2 The Short-Time Windowed Fourier Transform

The elementary functions employed in the Fourier Transform (FFT) consist of trigonometric functions that extend from negative infinite time to positive infinite time. They are not associated with a particular instant in time and thus they are not suited to retrieve signal components localized in time (see Figure 2).

The temporal dynamics of a signal can be retrieved by applying the FFT to a properly windowed portion of the signal providing the so-called Short Time Windowed Fourier Transform (STWFT). This procedure is usually used for the extraction of narrow-band frequency content in non-stationary or noisy signals. The basic idea of STWFT is to split the initial signal into small time windows through a proper windowing and employ the FFT to each time segment for expressing the variation in signal frequency content over the time segment or block. It provides constant absolute bandwidth analysis to identify harmonic components and offers constant resolution in two-dimensional representation, regardless of the actual frequency.

In other words, instead of processing the entire signal at once, STWFT takes the Fourier transform on a block-by-block basis. Therefore, the resulting Fourier transform can be thought of as a frequency representation of the signal during the time period covered by the data block.

For a given signal *u(t)* in the time domain, the mathematical equation representing the short time Fourier Transform is given by:

$$STWFT(f, mT) = \int_{-\infty}^{+\infty} u(t) W^*(t - mT)\, e^{-i2\pi f t} dt$$

Where $W^*(t - mT)$ is the complex conjugate of a properly translated window function $W(t)$ that balances the time and frequency resolutions. The smaller the time duration of $W(t)$ the better the time resolution (poorer frequency resolution).

In order to improve the temporal resolution, the blocks could be overlapped. The percentage of overlap between each block is determined by the time sampling step T and the length of the analysis window function $W(t)$.

In 1946, Dennis Gabor, a Hungarian-born British physicist, suggested expanding a signal into a set of functions that are concentrated in both the time and frequency domains [3]. These functions are basically obtained as in the STWFT procedure by considering a Gaussian form of the window $W(t)$. More details will be given below when presenting the Matching Pursuit technique (Sec. 2.4), here we just underline that the only difference between STWFT and Gabor

transform lies in the use of Gabor functions to set up an expansion series that can be thought as the inverse of the STWFT.

Figure 3 reports a scheme of the STWFT. This figure highlights the main limitation of this approach that is related to the use of a constant window for each frequency to be resolved. Therefore, the accuracy of obtaining frequency information is limited by the length of the window compared to the duration of the block. The size of the window determines the accuracy of time. The larger the window, the poorer time resolution but the better the frequency resolution and vice-versa.

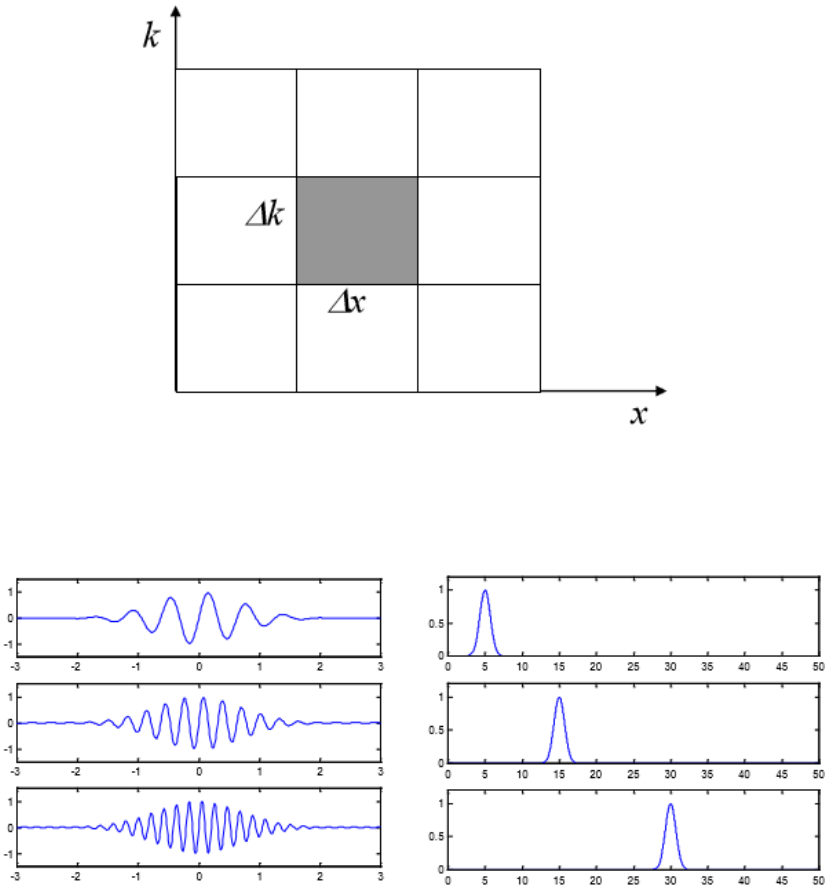


Figure 3: A scheme of the STWFT. The resolution is good in both the physical and transformed space but the window length strongly affects the accuracy of the achieved transform.

This limitation might be overcome by increasing the resolution when high frequencies (or small scales) have to be observed. This consists in setting up a so-called “mathematical microscope” whose magnification factor varies for different frequencies. This is the case of the Wavelet transform that is described in the next section.

2.3 Wavelet transform

The first mention of wavelets appeared in an appendix to the thesis of Alfred Haar (1909, PhD thesis supervised by Hilbert). The Haar wavelet has *compact support* but it is not continuously differentiable. In the 1930s several groups working independently researched the representation of functions using *scale-varying* basis functions. Between 1960 and 1980 G. Weiss and R. Coifman introduced the concept of *atoms* as the simplest elements of a function. In 1980 Grossman (a physicist) and Morlet (an engineer) defined wavelets in the context of quantum physics. In 1985 Stephan Mallat worked on digital signal processing introducing the concepts of quadrature mirror filters, pyramid algorithms and othonormal wavelet basis. Successively, Meyer constructed the first *non-trivial* Wavelets and I. Daubechies constructed a set of wavelet orthonormal basis functions.

According to the discussion of Sec. 2, the wavelet transform accomplish the need of increasing the resolution to the scope of analysing the small scales and a sketch of the wavelet properties is given in Figure 4.

In the wavelet transform, the analysing function is the following:

$$\Theta(x, x_0, \lambda) = \frac{1}{\sqrt{\lambda}} \psi\left(\frac{x - x_0}{\lambda}\right)$$

Where ψ(x) is the so-called mother wavelet.

The transform is obtained by the covolution with the dilated and translated counterpart of the mother wavelet:

$$T_u(x_0, \lambda) = \int u(x) \, \frac{1}{\sqrt{\lambda}} \psi^*\left(\frac{x - x_0}{\lambda}\right) dx$$

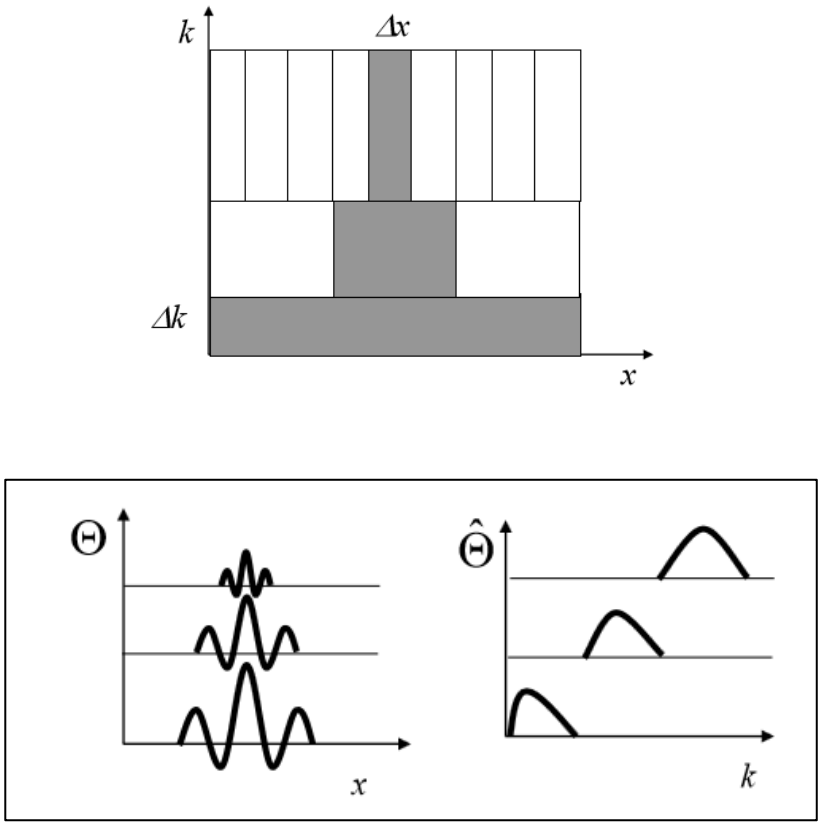


Figure 3: A scheme of the Wavelet transform. The resolution increases when k increases (small scales) according to the different support of the analyzing function.

To be consistent with the results of the following sections, we consider a time dependent arbitrary signal *f(t)* that is projected onto the basis of compact support functions obtained by dilation and translation of the mother wavelet defined in the time domain. In this decomposition, the wavelet coefficients are usually denoted as $w(r,t)$. They are functions of the time *t* and of the frequency that corresponds to the inverse of the resolution *s=1/r*. The wavelet decomposition reads as follows:

$$w(s,t) = C_{\Psi}^{-1/2} r_{\square}^{-1/2} \int_{\infty}^{\infty} \Psi^{*}\left(\frac{t-\tau}{r}\right) f(\tau) d\tau$$

where $C_{\Psi}^{-1/2}$ denotes a coefficient which accounts for the mean value of □*(t)*, and the integral represents a covolution between *f(t)* and the dilated and translated complex conjugate counterpart of □*(t)*.

The mother wavelet is a compact support function that has to satisfy some mathematical constraints that are not presented therein for the sake of conciseness. Furthermore, the wavelet kernel can be real or complex, discrete and orthogonal and, per each type of wavelet, many

different families exist. We remind to specialized texts for the details, comprehensive reviews about the wavelet theory and its mathematical background can be found in many reference papers or books (see e.g. Mallat 1999, Chui, 1992, Daubechies 1992). It should be stressed that fast and efficient wavelet transform algorithms have been developed in the past so that, nowadays, the cpu time required for the wavelet transform is comparable to that needed for a standard FFT and wavelet toolbox are present in most of the software packages used for data processing.

To the extent of application, wavelet analysis has been extensively used to treat random data representing physical quantities such as velocity, vorticity or pressure, obtained from both numerical simulations and experimental investigations of turbulent flows. Early applications in turbulence can be found in and e.g. Bacry, 1989; Meneveau 1991, Farge 1992. In many of those and successive applications, the use of wavelet transform was motivated by the need of separating local events from an apparently nearly Gaussian turbulent background.

Once the wavelet transform of a signal of interest is available, several quantities can be computed depending on the purpose of the analysis and a brief overview is given in the following.

A relevant outcome of the wavelet transform is the wavelet scalogram, given by the square of the wavelet coefficients $w^2(s,t)$. The scalogram represents a direct link between Wavelet and Fourier transform because is provides a decomposition of the energy, or Fourier modes, onto the (s,*t*) plane. As reported e.g. in Farge (1992), Guj and Camussi (1999) and Micci et al. (2022), the scalogram represents a localized counterpart of the standard Fourier spectrum that can be recovered by a simple integration in time.

One of the most interesting features that justifies the use of wavelet transform is intermittency. In a statistical sense, intermittency is intended as a sequence of quiescent phases interrupted by active events inducing a non-homogeneous distribution of energy in time. According e.g. to Farge (1992) and Camussi and Guj (1997) a wavelet based quantity able to identify intermittent events is the so-called Local Intermittency Measure (LIM), a normalized version of the wavelet scalogram. The formal definition of LIM is the following:

$$LIM(s,t) = \frac{w^2(s,t)}{\langle w^2(s,t)\rangle_t}$$

where the symbol $\langle \dots \rangle_t$ denotes averaging in time. The condition LIM> 1 identifies regions where the energy is larger than the mean and thus permits the localization of relevant, in terms

of energy content, events in time and scale. To the purpose of identifying events related to intermittency, Camussi and Bogey (2021) suggested that a more suited indicator is the LIM squared, LIM^2 (hereafter denoted as LIM2):

$$LIM2(s,t) = \frac{w^4(s,t)}{\langle w^2(s,t)\rangle_t^{\,2}}$$

Indeed, LIM2 can be interpreted as a time-scale (or time-frequency) dependent measure of the FF of the input signal.

The LIM2 can be averaged in time in order to retrieve a depiction of the standard Flatness Factor (the normalized fourth order centered statistical moment) as a function of frequency. This quantity is defined as Wavelet Flatness Factor, WFF (Camussi and Bogey, 2021):

$$WFF(s) = \langle LIM2(s,t)\rangle_t = \frac{\langle w^4(s,t)\rangle_t}{\langle w^2(s,t)\rangle_t^{\,2}}$$

it provides an efficient way to discover which is the most intermittent frequency contained by a signal. To the extent of the conditional statistics, LIM2 represents a convenient tool to extract those features contributing to the deviation from Gaussianity. Indeed, LIM2 is equal to 3 if the probability distribution is Gaussian, consequently the condition LIM2 > 3 identifies only those events contributing to the departure from a Gaussian distribution of the wavelet coefficients.

This indicator can then be adopted to track efficiently intermittent events.

2.4 A novel approach: the Matching Pursuit decomposition

The Matching Pursuit technique (MP) provides, through an iterative procedure, a decomposition of the given signal as a linear combination of a specified family of functions. The method was firstly presented by Mallat & Zhang (1993) and successively applied in several fields out of aeroacoustics, including neuroscience (e.g. Ray et al., 2008 and 2008b), medicine (e.g. Durka, 2007, Jabbari & Ghassemian, 2011), astrophysics (e.g. Lachowicz and Done, 2010), audio and music processing (Cho and Kuo, 2010, Zantalis, 2016), geophysics (e.g. Li and Zhang, 2017).

The motivation in the use of MP lies in the need of extending as much as possible, the set of *simple* functions that can be used to reconstruct a *complex* signal and thus to provide an optimal non-linear approximation by adaptively choosing the proper basis depending on the signal nature. The idea is to construct a best basis, not necessarily orthogonal, that approximates

efficiently the signal of interest. The MP algorithm iteratively constructs an approximation by selecting a set of elementary waveforms, called *atoms*, chosen from an over-complete set of functions, called *dictionary Dc*, in such a way so that they represent at the best, the inherent signal structures.

A crucial part of MP is the creation of the dictionary which directly affects the results of the decomposition and subsequently the quality of the signal reconstruction. Under the framework of the time-frequency decomposition, the waveforms belonging to the dictionary are usually selected as time-frequency atoms obtained as dilations, translations, and modulations of a single window function. According to the formalism introduced by Mallat & Zhang (1993), the window function in the temporal domain *t* can be denoted as follows:

$$g_\gamma(t) = \frac{1}{\sqrt{s}} g\left(\frac{t-u}{s}\right) e^{i\xi t}$$

Where g is the index that represents the set of parameters (s, u, x) of the element selected within the set of functions of the dictionary *Dc*, $\frac{1}{\sqrt{s}}$ is a normalizing factor, *u* and *s* give the translation and dilation factors respectively and *x* is the frequency.

The family of *g(t)* functions is generated by considering the set of parameters (s_n, u_n, x_n) that are varied independently from each other. The functions composing the dictionary are therefore identified by two indexes and are denoted as g_{gn}. In this way, the dictionary is extremely redundant and contains three parameters that can be selected appropriately to provide the best approximation. This method results to be more flexible than the traditional signal processing procedures applied in time-frequency decompositions, i.e. the Short Time Windowed Fourier Transform and the Wavelet Transform (WT). With reference to the definition of $g_\gamma(t)$, STWFT corresponds to fixing the scale *s* to a certain value, whereas in the WT, the frequency parameter x_n is forced to be inversely proportional to the scale s_n. The MP projection provides an over-complete dictionary containing a wider range of time and frequency support functions allowing, for instance, to represent efficiently signals that are extended in time but narrow in frequency or vice-versa.

In many applications, the dictionary is composed of Gabor atoms that provide optimal joint time–frequency localization. The Gabor dictionary contains functions obtained by the combination of a Gaussian envelope, having variable time location and time extension, with a cosine function, having variable frequency.

The Gabor function can be formalized as follows:

$$g_\gamma(t) = K_\gamma(t) \cdot G_\gamma(t) = K_\gamma(t) \cdot exp\left[-\pi\left(\frac{t-u}{s}\right)^2\right]\cos(2\pi\xi t + \varphi)$$

Where the factor $K_\gamma(t)$ represents the energy content of the atom i.e.:

$$K_\gamma(t) = \left(\int_{-\infty}^{\infty} G_\gamma(t)^2 \; dt\right)^{-1/2}$$

The MP algorithm (described in details in e.g. Mallat and Zhang, 1993, Ray et al., 2008a), is based on an iterative process that maximizes the correlation between the signal reconstructed through the combination of the atoms belonging to the dictionary and the original signal. The algorithm iteratively selects the atoms that provide the best approximation of a given signal *x*. At the first step, the atom g_{g0} is selected from the dictionary *Dc* to provide the highest correlation, i.e. the highest absolute inner product with the signal *x*. In the successive steps, the waveform $g_{g,n}$ is matched with the residual obtained by subtracting to the signal the result of the previous iteration.

The iterative process therefore starts with the selection of the atom within the dictionary that best describes the whole signal and its projection onto the signal is subtracted from it in order to achieve a residual. The best description is achieved by computing the correlation (or the inner product) and thus by searching the atom the best correlates with the original signal. Therefore, for a given signal *x*, at the first step the dictionary *Dc* is scanned to search for the atom $g_{g,1}$ for which the correlation or inner product $\left|\langle x, g_{\gamma,1}\rangle\right|$ is as large as possible. The residual $R^n x$ at the first step is obtained by the following relationship:

$$x = \left|\langle x, g_{\gamma,1}\rangle\right| + R^1 x$$

At the successive step, the atom $g_{\gamma,2}$ is identified in *Dc* by finding the maximum correlation with $R^1 x$.

The procedure is then repeated iteratively with the residual replacing the signal.

The algorithm can then be formalized by the following three steps:

$$\begin{cases} R^0 x = x \\ R^i x = \langle R^i x, g_{\gamma,i}\rangle g_{\gamma,i} + R^{i+1} x \\ g_{\gamma,i} = \arg max_{g_{\gamma,i'} \in D} \langle R^i x, g_{\gamma,i'}\rangle \end{cases}$$

Thus, the standard MP algorithm starts with an initial approximation that is empty and a residual that equals the original signal. At each iteration it subtracts from the residual the atom that has the highest correlation with it and adds that atom to the approximation. Continuing in this procedure, MP builds-up the signal model one atom at a time and at each iteration. It selects the atom that absorbs most of the energy from the residual signal since the identification is based on the maximum of the correlation function. It has been demonstrated (e.g. Davis et al., 1994) that the residual goes to zero when the number of iterations goes to infinity. In practice, iterations are stopped when the desired level of accuracy is reached, in terms of the number of extracted atoms or, as in the present case, in terms of the energy ratio between the original signal and the current residual.

We remind to the literature for further details about convergence (Mallat & Zhang, 1993) and possible variants of the MP procedure (see e.g. Durka et al, 2001, Kus et al., 2013, Różański, 2020).

2.5 An example of application: propeller noise

In this section an explanatory application of the techniques described above is presented. The reported results have been obtained under the framework of the ENODISE project and part of them have been reported in a published paper (Meloni et al., 2023).

The motivation behind the investigation briefly worked out in the following, is related to the observation that propeller noise generally exhibits a rich mixture of tonal and broadband components that are known to be related to different physical mechanisms. A separation between these two components is therefore often essential to identify and possibly control the different noise sources.

More specifically, the deterministic harmonic noise is originated by the blade loading and thickness contributions, which are determined by the rotational motion of the object. Therefore, propeller harmonic components are indeed expected to be integer multiples of the fundamental frequency related to the engine rotational speed. On the other hand, the broadband component is a random nonperiodic contribution either determined by the trailing-edge noise scattering of the blade, interactions with the airframe or by other stochastic phenomena related to complex mechanisms, including turbulence ingestion or blade-vortex interactions, which are typically

weakly correlated with the rotor revolutions per minute (RPM). It is important to point out that the different nature of the tonal/broadband noise components, leads to different acoustic directivity in the far field and to different dependencies upon the most important governing parameters (e.g. Reynolds number, advance ratio). Therefore, the investigation of each noise component independently from the other is essential for setting up reliable predictive models as well as for assessing noise-mitigation strategies. We remind to the other Notes of this Lecture Series for more details on the nature of propeller noise.

How time-frequency decompositions can be useful in this context?

There are several reasons justifying the use of time-frequency decomposition techniques. Some of them are the following:

- Ideally, the tonal components are perfectly stable in time (constant rotor speed and absence of the so-called clock noise), however, this ideal situation is rarely met in practice
- The number of tonal components may be large and, more importantly, they are not infinitely extended in time
- Most of the methodologies developed so far provide the reconstruction of the two components only in the spectral domain.

In order to overcome these limitations, a wavelet-based separation technique has been developed and the main features of the procedure are briefly worked out in the following. Interesting results are achieved as well by the application of the Matching Pursuit decomposition presented above. A few preliminary examples are also reported in the next section.

2.5.1 The wavelet approach

As pointed out above, the tones are characterized by having deterministic and persistent characteristics, while the broadband counterpart has a random behavior. This implies that the cross-correlation between two tonal signals is characterized by a non-vanishing oscillatory trend whereas the equivalent for two random components is a vanishing function. This principle is the basis of the wavelet-based separation algorithm. The core of the procedure is the iterative process that is schematically summarized below whereas the details can be found in Meloni et al. (2023):

- Using a first guess threshold, the set of wavelet coefficients obtained by the discrete wavelet transform of a given signal is separated into two sets. On the basis of the

threshold selection, it is assumed that the coefficients exceeding the threshold are related to the tonal component and those having magnitude lower than the threshold are considered representative of the broadband part. The two signal components are reconstructed in the physical domain by the inverse wavelet transform

- This separation is applied to two consecutive microphones. The cross-correlation between the presumed tonal components extracted from two microphones is computed and its non-vanishing nature is verified
- If the cross-correlation vanishes the threshold value is updated and the process is repeated

Results are presented in Meloni et al. (2023) and are not reported here for brevity. We just underline that the procedure works very well even though some residual tones at high frequencies are present in the reconstructed background random signal.

A similar procedure has been implemented through the application of the Matching Pursuit approach described in the previous section. The dictionary is built with the purpose of extracting the tonal component from the given signal and the algorithm applied is provided under the Matlab framework through the function matchingPursuit.m and the basis elements correspond to the discrete cosine transform-II orthonormal basis. This transform, despite the FFT, uses real numbers and is extensively used in image compression algorithms.

An example of the achieved results is given in Figure 5 where it is shown that the method is able to extract efficiently the tonal component and thus to separate the random content of the signal.

Further analysis on this topic are currently underway to assess the procedure and apply it to the ENODISE database.

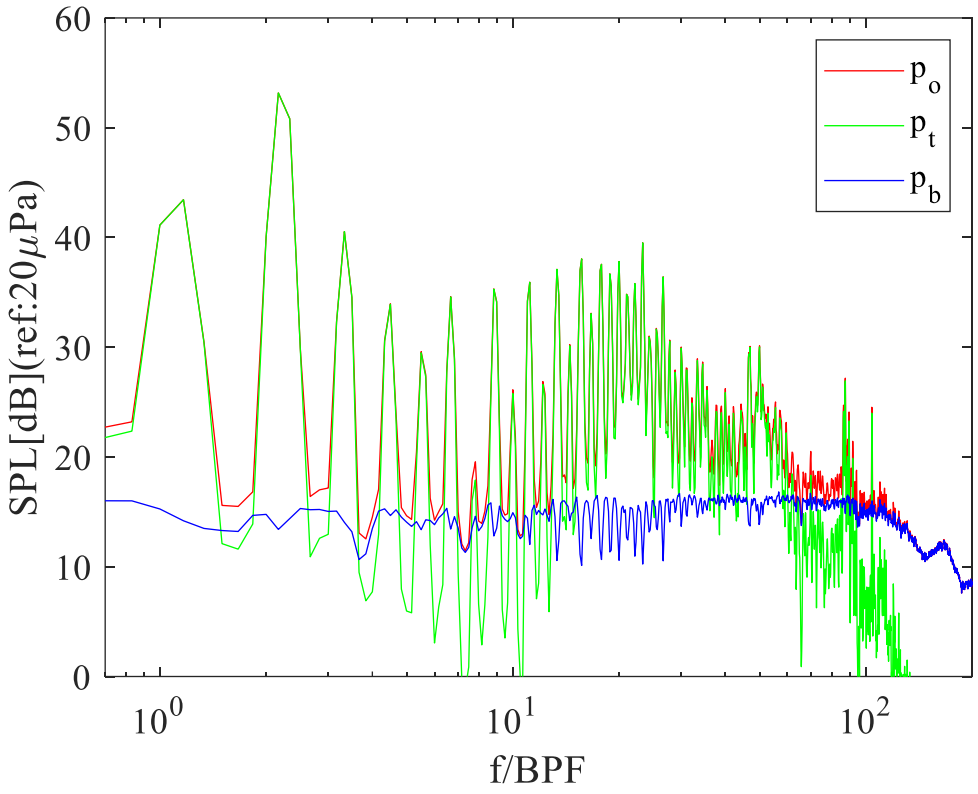


Figure 4: An example reporting spectra obtained from the application of the Matching Pursuit separation procedure. *po* indicates the original signal, p_t the tonal component, p_b the background random component.

3. Data assimilation and surrogate modelling

According to its standard definition, data assimilation refers to all the methods aiming at combining experimental data with computational models and simulation-based results, with the main objective of improving the accuracy of the theoretical/numerical models by extracting information from the data as they become available. From this point of view, the post-processing techniques described in the previous sections can be considered as a fundamental ingredient to gain insight of the physics underlying the observations.

In the present section, we want to briefly outline a class of dynamically adaptive techniques for the development of surrogate models from the available data. Indeed, key aspects of any data assimilation strategy are the capacity to be updated and improved by incorporating new data in a dynamic fashion, and the possibility to quantify the uncertainties of measures and simulations, in order to enhance the models and, possibly, maximise the amount of information extracted. The techniques that will be presented in the following push these two requirements a little bit further, with the purpose of identifying the regions of the variables/parameters space where the

information in the available data is lacking and, thus, where additional simulations and/or experiments represent the most profitable investment of additional resources. The approach to achieve this relies on the following assumptions.

I. The analytical structure used to model the data is built in a statistical way, by defining the relevant parameters as stochastic events with a specific Probability Density Function (PDF).

II. The quantification of the uncertainty of the model's response for a set of variables/parameters not included in the data, is assumed to be a measure of the lack of information at that specific point of the design space.

III. The identification of the maximum uncertainty indicates the region of the space where the lack of information is higher, and thus where new simulations or experiments can improve at most the model itself.

The techniques that will be presented are based on stochastic radial basis functions (SRBF) and dynamically adaptive artificial neural networks (ANN), in the optics of comparing an approach using an evolution of a standard data interpolation technique with one relying on the adaptive version of a well-assessed machine-learning tool.

3.1 Adaptive SRBF-based metamodeling

The basic idea behind the RBF formulation (Hardy [29]) is that we look for a function s that interpolates the data in the points where the target response is known, *i.e.* it satisfies the conditions $s(x_j) = f_j$, $1 \leq j \leq N$, being x_j the j-th training sample and $f_j = f(x_j)$ the function value. Given a training set (TS) of pairs $(\xi_i, f(\xi_i))$ for $i = 1, N_T$, a simple interpolation $\check{f}$ of the data using RBF has the form

$$\check{f}(\boldsymbol{\xi}) = \sum_{j=1}^{N_T} w_j \, \varphi(|\boldsymbol{\xi} - \boldsymbol{\xi}_j|)$$

where $\varphi(r)$ is called the *kernel* of the representation. The condition for the surrogate model to reproduce the training set yields the system of equations for the unknowns w_j, and solving the inverse linear problem

$$\mathbf{w} = \mathbf{A}^{-1} \mathbf{f}$$

where the distance matrix elements are $A_{ij} = \varphi(\xi_i, \xi_j)$ and $f_j = f(\xi_j)$. Therefore, our approximation is the linear combination of shifts of a function, which is radially symmetric with respect to the given norm about their *centres*, or *knots*, where the function is known. The matrix **A** is symmetric for sets of distinct data sites and the solution to the fitting problem exists and is unique, if and only if **A** is non–singular.

The objective of a dynamic metamodelling approach is not the simple interpolation of data using fthe equations above, but instead obtaining an optimal training set with the minimum number of points needed to predict the unknown response $f(\xi)$ with the desired accuracy. Therefore, the design space is no longer sampled *a priori* employing a static Design–of–Experiment (DoE) technique but instead the TS is dynamically updated where it is more useful, trying to use the minimum number of simulations to represent the function accurately. In this approach, the RBF metamodel is reinterpreted in a statistical fashion introducing a stochastic version of the RBF kernel $\varphi(r, \epsilon)$, obtained from the variation of its hyperparameter ε subject to a probability density function distribution $P(\varepsilon)$. The advantage of a stochastic RBF formulation is the evaluation of the local metamodel uncertainty, which is used to drive the improvement of the *TS* with new samples in the more uncertain regions. The prediction is made on the so–called validation set, which is an ensemble of points (N_{val}) where the output function is known, left out from the training set to validate the efficiency of the surrogate model. The prediction $\hat{f}$ obtained through the stochastic metamodel is thus defined as the expected value (EV) of the metamodels obtained varying the hyperparameter in the range $[\epsilon_{min}; \epsilon_{max}]$

$$\hat{f}(\boldsymbol{\xi}) = EV\left[\sum_{j=1}^{N_T} w_j\, \varphi(|\boldsymbol{\xi} - \boldsymbol{\xi}_j|, \epsilon)\right], \varepsilon \in P(\varepsilon)$$
$$= \int_{\epsilon_{min}}^{\epsilon_{max}} \hat{f}(\boldsymbol{\xi}, \varepsilon) P(\varepsilon) \mathrm{d}\varepsilon$$

The uncertainty $U_{\hat{f}}(\boldsymbol{\xi})$ is quantified as the difference of the relevant *p*–quantiles of the predictions obtained varying the ε parameter so as to have a 95%–confidence band. Adopting a dynamic update of the metamodel based on the above mentioned stochastic formulation $\hat{f}_S(\boldsymbol{\xi})$, at each iteration of the algorithm, the *TS* is updated with an additional sample located where the metamodel uncertainty is maximum. An appropriate choice of the stochastic parameter allows the conjecture that higher values of *U* correspond to a local lack of knowledge about the target response dynamics. The flowchart of the dynamic adaptive algorithm is depicted in Fig. 5.

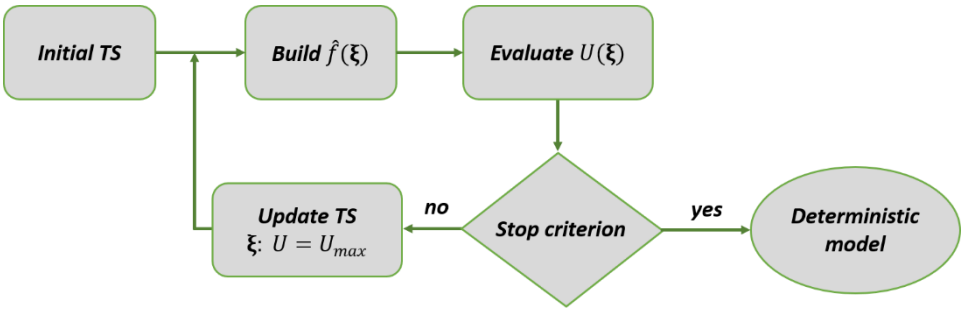


Figure 5: Flowchart of the dynamic adaptive metamodel update

Starting from an initial *TS*, the metamodel is built and evaluated for the validation set, and the local value of its uncertainty $U(\xi)$ is calculated. If the maximum value of the uncertainty is above a certain threshold ϵ, the *TS* is updated by adding a new sample $(\xi, f(\xi))$, which is obtained exploring the variables domain by means of a genetic algorithm [30]. Therefore, the objective function for the search algorithm is the maximum uncertainty in prediction of the current surrogate model. It is worth noting that this procedure allows for an estimate of the local quality of the surrogate model without additional calls to the expensive simulation models. The adaptive sampling can follow different approaches depending on the scope of the analysis. In context of uncertainty quantification, a global approximation of the function is required, so the sampling is usually driven by the overall model variance [31]. Whereas, for optimization purposes, the aim is at improving the accuracy of the model in regions of the design space that seems promising, thus focusing on local model improvement. The accuracy of the interpolating function $\hat{f}(\xi)$ for $\xi \notin TS$ strongly depends on the characteristics of the RBF kernel $\varphi(|\xi - \xi j|) = \varphi(r)$ and, of course, on the properties of the *TS*. training set (number of points and their distribution) and the distance matrix condition number, so that the well-poseness cannot be ascribed only to the kernel function but to the combination of these factors. Thus, for each kernel, there is a best choice for the kernel hyperparameters that depends on the available training set. Even though several approaches, such as stabilisation techniques, have been applied in literature to overcome the issue of ill–conditioned distance matrix [32], choosing the correct hyperparameters is often the most challenging activity of RBF methods and strongly influences the accuracy of the surrogate model. Many authors dealt with the problem of identifying the best shape parameters for classical kernel functions [33,34,35,36,37,38,39]. Dealing with non-conventional basis functions and adaptive schemes, for which the training set is updated throughout the process, the mentioned empirical formulas cannot be applied directly. Furthermore, thanks to the development of high computational power, iterative approaches have

been formulated to determine the best hyperparameters. Despite the high computation demand, the method of cross-validation is widely used in the statistics literature. In this method, the optimal hyperparameter values are found minimising a cost function given by the (least–square) error for a sequence of partial fits where a sub–set of centres are left out for the prediction (a series of *test sets* which in batches span all the N_T points). A special implementation of cross–validation is the leave–one–out cross–validation (LOOCV) where, in round, one point of the training set is left out from the fit until all points are used as test points, which is more robust for small TS. Consequently, the evaluation metric is based on the Euclidean norm of the vector $E = E\,[E_1, \ldots, E_N\,]^T$, where N is the total number of training points available. LOOCV forms the basis of the algorithm proposed by Rippa in for choosing an optimal value of the shape parameter in the setting of scattered data interpolation with RBFs, requiring less computation effort. Even though this simplification introduces an error, Rippa showed that this error is sufficiently constant over the range of variation, so that the minimum coincides with that obtained using the classical LOOCV approach.

Performing hyperparameters tuning means that the model used for the approximation is identified within the analysis process to maximise its fitting capability and accuracy. This strategy can be further applied to the choice of the kernel itself, and often these approaches are coupled. A loop on the stochastic (ε) and the auxiliary hyperparameters is hence added to the process comparing the error norms for different values of the hyperparameters, choosing the optimal ones that yield the minimum error norm. Adding this further step, the adaptive algorithm reported in Fig. 4 becomes the one depicted in Fig. 6.

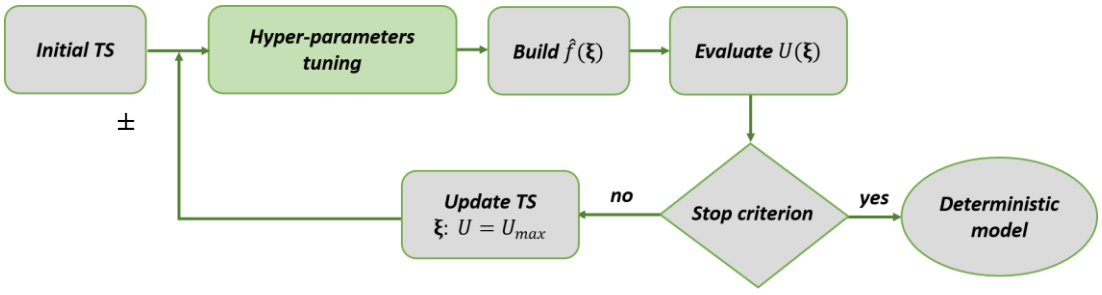
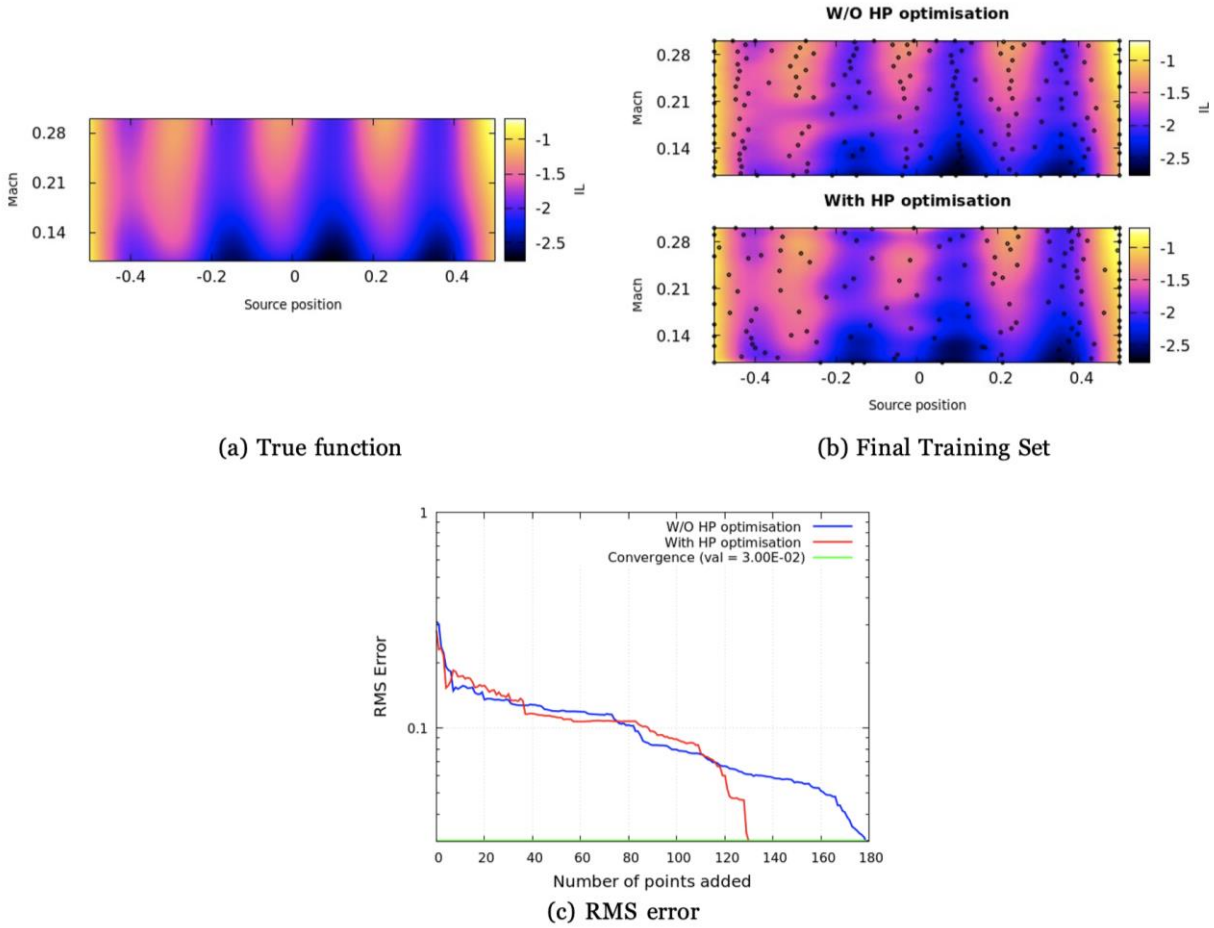


Figure 6: Flowchart of the dynamic adaptive metamodel update with kernel hyperparameters tuning.

It's worth noting that both the auxiliary and the stochastic hyperparameters have been optimised in a deterministic way, *i.e.* through the standard RBF formulation. To inject uncertainty in the system, which is used to detect the next point to be added, the stochastic hyperparameter is varied in a range given by 15% of the optimal value. The stochastic hyperparameters are sampled uniformly in their range, using a *Latin hypercube* sampling method (LHC-SM) to

guarantee such a uniform distribution with a number of samples that depends on the problem dimensionality so as to guarantee the statistic convergence. The method presented was used to build a surrogate model of noise shielding for the application case of an airfoil in a uniform flow impinged by an incident field generated by a point source monopole. The acoustic describer for the noise shielding effect characterisation is the Insertion Loss (IL), spatially integrated over a linear array of receivers. The design space bounds are $-0.5 < \mathbf{x}_S < 0.5$ (leading edge and trailing edge of the airfoil) and $0.1 < \mathbf{x}_M < 0.3$. The comparison between metamodels obtained with and without hyperparameter optimisation (Fig. 7) shows how the proposed tuning strategy significantly reduces the number of points in the final training set.



(a) True function (b) Final Training Set

(c) RMS error

Figure 7: SRBF metamodels with and without hyperparameter tuning, freq. band = 630 Hz.

3.2 Active metamodelling using Neural Networks

The artificial neuron is a mathematical structure which accepts several inputs and a single output. For the generic *k*-th node of a multilayer dense structure, the output is given by

$$a_k = f_l\left(\sum_{j=1}^{n} w_{jk}x_j + b_k\right)$$

being f_l the activation function of the l-th layer, w_{jk} the synaptic weight of the j-th input to the k-th node, x_j the node input and b_k the bias, an additional parameter required to adjust the output. Let consider a complex multilayer structure with $L = Q + 2$ layers, *i.e.* the input layer, Q hidden layers and the output layer. In compact form, the output of the l-th layer, with 2 can be written as it follows

$$\mathbf{a}_l = \mathbf{f}_l(\mathbf{W}_{l-1}\mathbf{a}_{l-1} + \mathbf{b}_l)$$

so that the inputs $\mathbf{x} \equiv \mathbf{a}_1$ and the outputs $\mathbf{y} \equiv \mathbf{a}_L$. The selection of weights matrices $\{\mathbf{W}_l\}_{l=1}^{L-1}$ and biases

vectors $\{\mathbf{b}_l\}_{l=2}^{L}$ can be performed by means of the so–called *training with backpropagation*. Within the

training process, a cost function has to be computed at the t-th epoch, so that its gradient (evaluated

backpropagating the error information layer by layer) can be used to update weight and bias values. Weights and biases at the t-th epoch are updated used the *learning rate*, a free parameter selected by the designer to control the change in weights and biases based on the error gradient components. As a result of the training process (described in Sec. 3.2.1), the model can be completely identified by the parameters vector $\Theta = \{\mathbf{f}_l, \mathbf{W}_{l-1}, \mathbf{b}_l\}^L$, where $L = Q + 2$ and Q is the number of hidden layers. The network parameters can be divided in two classes (*training* and *architectural*) depending on the fact that are involved or not in the final model: a synthesis of the parameters is presented in Tab. 1.

Table 1: Training parameters and architectural parameters.

Class	**Parameter**	**Description**
Training	$\mathbf{W}_0$	Weights initial values
	$\mathbf{b}_0$	Biases initial values Number of
	T	training epochs
	B	Batch size
	η	Learning rate
Architectural	$\mathbf{f}$	Activations
	Q	

	N_{h_q}	Number of hidden layers Neurons per hidden layer

Training parameters The training parameters of Tab. 6 affect the learning capability of the network. Their choice is rather critical because has a great influence on the success or failure of the ANN model. The following rules have been implemented to address a data–informed tuning of the training parameters. The initialisation of weights matrices and bias vectors is a crucial aspect in the model performances. The initial values $\mathbf{W}_0$ and $\mathbf{b}_0$ are random numbers with a normal distribution between 0.0 and 1.0 (generated with a fixed seed). The number of training epochs is selected based on an early stop criterion, built on the basis of a *runtime* monitoring of different performance metrics (it can be demonstrated that the analysis of a single loss metric may lead to conflicting results): specifically, absolute error, squared error and squared logarithmic error on the training set are analysed in both their mean and maximum values. In addition, to avoid overfitting, the validation loss is also analysed. Although capable to train the network with an arbitrary batch size, the implementation performs a short test training with the SGD (stochastic gradient descent), the FCG (full-batch gradient descend) and the MGD (mini-batch gradient descend), thus the complete training is performed using the batch size value that returns the lower training loss within the test. It is known that low values of learning rate could result in a slow training, whereas high values of learning rate can generate instabilities in the training (and validation) loss dynamics: a learning rate update, based on the loss metrics acceleration values (computed with the finite differences), is used to mitigate such effects.

Architectural parameters It is known that the ANN fitting capability is sensitive to the architectural parameters, and the set of topologies capable of correctly fitting the training data is strongly problem– dependent: unfortunately there is no specific scheme to find the optimal network configuration. Thus, to find the network architecture (in terms of number of hidden layers Q, neurons per layer N_{h_q} and activations $\mathbf{f}$) that provides a response that meets the designer needs, the procedure is depicted in Fig. 16 is used. The activation functions combination (for hidden and output layers) is selected on the basis of a preliminary tournament of the combinations between $\boldsymbol{\varphi}_h$ (activations for the hidden layers) and $\boldsymbol{\varphi}_o$ (activations for the output layer), choosen by the designer. A factorial experiment involving two factors (with n_h and n_o levels respectively) is built: each experiment provides an observation related to a different combination of factors and levels. The initial tournament implies the additional computational

cost, entirely balanced with faster network training, since the activation functions are tailored to the specific problem. The deterministic optimisation algorithm develeped is now briefly described. Figure 16 shows that after the activations selection the current network is trained. The network training is stopped when the training loss meets the designer's need, and in this case the model can be considered as final, meaning that the network response is capable to correctly describe the dynamics of the data. It may occur that the training is stopped when no improvements are foreseen, based on the observation of the performance metrics: in this case, a new topology is built and the network training starts again. The optimisation algorithm adds new layers nodes, choosing the solution that gives the best performance in terms of training loss. Each iteration of the algorithm developed in this work mainly consists in the following five phases: the number of hidden layers is selected between a certain number of combination based on the problem size; the number of nodes of each layer is optimised using a coordinate descent method, to find the minimum of the training loss; all the $n_s = 3^Q$ topologies (with Q the number of layers) neighboring the local optimum are tested; the topologies built mutating the worst ones (in terms of nodes per layer) are analysed; new topologies are added and tested, on the basis of the current loss distribution. If a new optimum is not found, the algorithm starts over and with a new layers initialisation is adopted in the first step.

Uncertainty function and adaptive sampling

The ANN response is assumed to be the expectation of a uniform probability $U_j(\mathbf{x})$ = unif $a_j(\mathbf{x})$; $b_j(\mathbf{x})$ with $\{a_j(\mathbf{x}) < b_j(\mathbf{x})\}$. The hypothesis of continuous uniform distribution yields to the following system of equations

$$\begin{cases} g_j(\mathbf{x}) = \frac{1}{2}\left[a_j(\mathbf{x}) + b_j(\mathbf{x})\right] \\ \sigma_j(\mathbf{x}) = \sqrt{\frac{1}{12}\left[b_j(\mathbf{x}) - a_j(\mathbf{x})\right]^2} \end{cases}$$

being $\sigma_j = \sqrt{\mathrm{Var}[Uj(\mathbf{x})]}$ the standard deviation. The above equation can be solved for the unknowns $a_j(\mathbf{x})$ and $b_j(\mathbf{x})$, as follows

$$\begin{bmatrix} a_j(\mathbf{x}) \\ b_j(\mathbf{x}) \end{bmatrix} = g_j(\mathbf{x}) + \begin{bmatrix} \sqrt{3} \\ -\sqrt{3} \end{bmatrix} \sigma_j(\mathbf{x})$$

The metamodel global uncertainty $U(\mathbf{x})$ is evaluated as the normalised norm on the R^N Euclidean space

$$U(\mathbf{x}) = \sqrt{\frac{1}{N}\sum_{j=1}^{N} u_j^2(\mathbf{x})}$$

being $u_j(\mathbf{x})$ the magnitude of the uncertainty function related to the j-th component of the modelled function. The adaptive sampling scheme aims at driving the designer towards the choice of new training points, and is here addressed based on the definition of the global uncertainty function $U(\mathbf{x})$. Specifically, the new training point to be analysed corresponds to the maximum value of $U(\mathbf{x})$ as it follows

$$\mathbf{x}_{new} = \arg\max_{\mathbf{x}\in\mathcal{D}} [U(\mathbf{x})]$$

so that the size of the training set is systematically increased, and a new network can be built.

3.3 Results and discussion

The results presented here are related to a data–driven model of the jet–plate interaction. The domain variables are the radial distance of the plate *H/D*, the axial distance *x/D* and the nozzle pressure ratio *NPR*. The measurements provide two spectra in 1/12th octave band approximation related to the SPL and the coherence γ. Such a problem can be translated into a network with 4 input neurons and 2 output neurons (in addition to a unknown hidden topology), and Mn_ω points (being M the number of experiments and n_ω the number of frequencies) are used to build the model. The data were used to create a database consisting of 7884 elements which define the four variables (*H/D*, *x/D*, *NPR* and i_ω) with associated two function values (SPL and γ): the dataset has been randomly shuffled, and 5913 elements were used as training set, whereas 1971 as validation set. The model has the characteristics reported in Tab. 2.

Table 2: ANN metamodel final configuration.

• Final topology	• 4/42/46/49/2
• Hidden layer activation	• *gaussian*
• Output layer activation	• *tanh*
• $RMSE_T$	• 1.25%
• $RMSE_V$	• ∼3%

The results show that the ANN model is fully capable to describe the overall dynamics of the data, as depicted in Fig. 8: furthermore, the generalizing property of the neural networks ensures the regu- larization of some noisy phenomena (especially for the coherences), typical of experimental data despite the post processing.

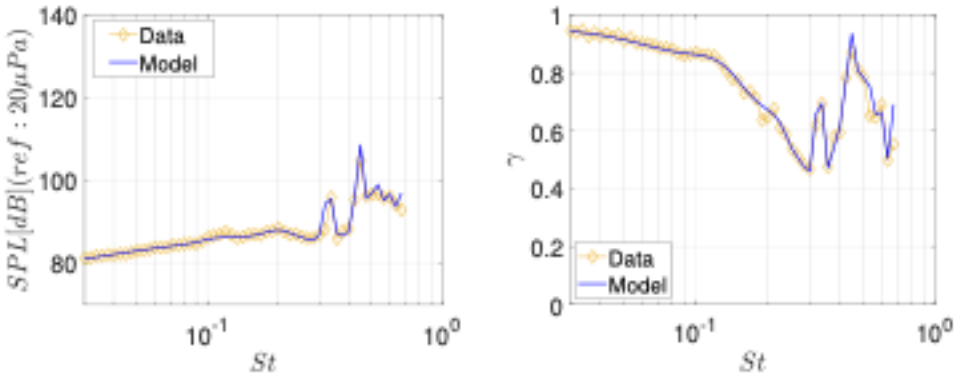


Figure 8: Comparison between ANN metamodel response and original data in terms of SPL (left) and coherence (right) at H/D=0.75, x/D=3 and NPR=2.6.

It is worth noting that the model response is associated with a quantification of its uncertainty, to provide an estimate of the prediction reliability: the spatial correlation is used to derive the uncertainty

function. Figure 9 shows the map of the model response, obtained on a structured grid composed of about 80000 prediction points.

Results in Fig. 9 look consistent with the literature and the physics: the ANN model global uncertainty combines the dynamic properties of the uncertainties related to each component of the vector function, and provides information on the domain regions where the knowledge of the data should be improved.

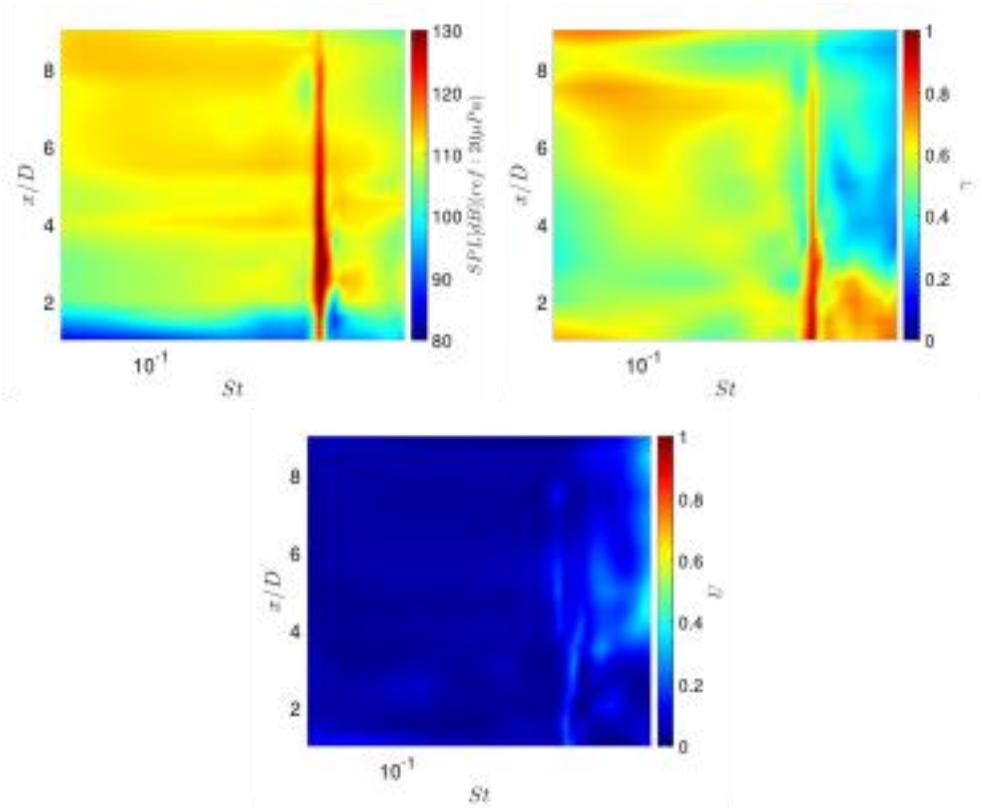


Figure 9: ANN model response for SPL (left) and coherence (right) with the global uncertainty function (bottom) at H/D=0.75 and NPR=2.55

4. References

1. Heisenberg, W. 1927, “On the precepted content of quantic theoretical kinematics and mechanics”, Zeischrift für Physik, 43, 172-198.
2. Cohen, L. 1995, “Time–frequency Analysis, Signal Processing Series, vol. 1406. Prentice Hall PTR, Englewood Cliffs, NJ. p. 48–95.
3. Gabor, D. 1946, ‘Theory of communication”, 1946, IET, vol. 93, no. 111, pp. 429-457.

4. Grossman, A. and Morlet, J. 1984, “Decomposition of Hardy Functions into Square Integrable Wavelets of Constant Shape“ SIAM Journal on Mathematical Analysis, Vol. 15, Iss. 4 (1984)
5. Mallat, S. A 1989, “Theory for multiresolution signal decomposition: The wavelet representation”. IEEE Trans. Pattern Anal. Mach. Intell., 11, 674–693.
6. Chui, C.K., 1992, “An Introduction to Wavelets”, Academic Press, Boston.
7. Daubechies, I. 1992, “Ten Lectures on Wavelets”. SIAM.
8. Bacry, E. Arneodo, A. Frisch, U., Gagne, Y. and Hopfinger, E. 1989, “Wavelet analysis of fully developed turbulence data and measurement of the scaling exponents”. In Proc. Turbulence 89: Organized Structures and Turbulence in Fluid Mechanics, Grenoble (France), (ed. M. Lesieur and O. Metais), pp. 203-215. Kluwer.
9. Meneveau, C. 1991, “Analysis of turbulence in the orthonormal wavelet representation”, Journal of Fluid Mechanics 23, 469-520.
10. Farge, M. 1992, “Wavelet transforms and their applications to turbulence”, Annual Review of Fluid Mechanics 24, 395-457.
11. Micci, G.L., Camussi, R., Meloni, S., Bogey, C. 2022, “Intermittency and stochastic modeling of low- and high-Reynolds-number compressible jets”, AIAA Journal 60, 1983–1990.
12. Guj, G. and Camussi, R. 1999, “Statistical analysis of local turbulent energy fluctuations”, Journal of Fluid Mechanics 382, 1–26.
13. Camussi, R. and Guj, G. 1997, “Othonormal wavelet decomposition of turbulent flows: intermittency and coherent structures”, Journal of Fluid Mechanics 348: 177-199.
14. Camussi, R., Bogey, C. 2021, “Intermittent statistics of the 0-mode pressure fluctuations in the near field of mach 0.9 circular jets at low and high Reynolds numbers”, Theoretical and Computational Fluid Dynamics 35, 229–247.
15. S. G. Mallat and Z. Zhang. 1993. Matching pursuits with time-frequency dictionaries. IEEE Transaction on Signal Processing 41, 3397–3415.
16. Ray S, Crone NE, Niebur E, Franaszczuk PJ, Hsiao SS 2008, “Neural correlates of high-gamma oscillations (60–200 Hz) in macaque local field potentials and their potential implications in electrocorticography”, Journal of Neuroscience 28: 11526–11536.
17. Ray, S., Hsiao, S.S., Crone, N.E., Franaszczuk, P.J., Niebur, E. 2008b, “Effect of stimulus intensity on the spike-local field potential relationship in the secondary somatosensory cortex”, Journal of Neuroscience 28, 7334–7343.

18. Durka, P. 2007 “Matching pursuit and unification in EEG analysis”, Artech House, Norwood, MA.
19. Jabbari, S. and Ghassemian, H., 2011 “Modeling of heart systolic murmurs based on multivariate matching pursuit for diagnosis of valvular disorders”, Computers in Biology and Medicine 41, 802–811.
20. Lachowicz, P. and Done, C. 2010, “Quasi-periodic oscillations under wavelet microscope: the application of Matching Pursuit algorithm”, Astronomy & Astrophysics 515, A65.
21. Cho, N. and Kuo, C.C.J. 2010 “Sparse representation of musical signals using source-specific dictionaries”, Signal Processing Letters, IEEE 17, 913–916.
22. Zantalis D., 2016, “Guided Matching Pursuit and its Application to Sound Source Separation”, PhD Thesis, University of New York.
23. Li, C. and Zhang, F., 2017, “Matching pursuit parallel decomposition of seismic data“, Computer and Geoscience 104, 54-61.
24. Davis, G.M., Mallat, S.G. and Zhang, Z. 1994 “Adaptive time-frequency decompositions”, Optical Engineering 33, 2183–2191.
25. Durka, P., Dobiesław, I., and Katarzyna, J. B. 2001, “Stochastic Time–Frequency Dictionaries for Matching Pursuit”, IEEE Transaction on Signal Processing 49, 507-510.
26. Kus, R., Różański, P.T., Durka, P. 2013, “Multivariate matching pursuit in optimal Gabor dictionaries: theory and software with interface for EEG/MEG via Svarog”, BioMedical Engineering OnLine 2013, 12:94.
27. Różański, P.T., 2020 “Effects of envelope and dictionary structure on the performance of matching pursuit », IET Signal Processing, Wiley.
28. S Meloni, S., De Paola E., Grande, E., Ragni, D , Stoica, L.G., Di Marco, A., Camussi, R. 2023 “A wavelet-based separation method for tonal and broadband components of low Reynolds-number propeller noise” Measurement Science and Technology 34, 044007.
29. Rolland L. Hardy. Multiquadric equations of topography and other irregular surfaces. Journal of Geophysical Research (1896-1977), 76(8):1905–1915, 1971.
30. D.L. Carroll. Chemical Laser Modeling with Genetic Algorithms. *AIAA Journal*, 34(2):338–346, 1996.
31. Liang Zhao, K. K. Choi, and Ikjin Lee. Metamodeling Method Using Dynamic Kriging for Design Optimization. *AIAA Journal*, 49(9):2034–2046, sep 2011.

32. B. Fornberg, E. Larsson, and G. Wright. A new class of oscillatory radial basis functions. *Comput. Math. Appl.*, 51(8):1209, 1222, 2006.
33. Shmuel Rippa. An algorithm for selecting a good value for the parameter c in radial basis function interpolation. *Advances in Computational Mathematics*, 11:193–210, 1999.
34. C. Roque and A. J. Ferreira. Numerical experiments on optimal shape parameters for radial basis functions. *Numerical Methods for Partial Differential Equations: An International Journal*, 26(3), 2010.
35. Michael Mongillo. Choosing Basis Functions and Shape Parameters for Radial Basis Function Meth- ods. *SIAM Undergraduate Research Online*, 4:190–209, 2011.
36. E.J. Kansa and R.E. Carlson. Improved accuracy of multiquadric interpolation using variable shape parameters. *Computers & Mathematics with Applications*, 24(12):99–120, 1992.
37. Gregory E. Fasshauer and Jack G. Zhang. On choosing "optimal" shape parameters for RBF ap- proximation. *Numerical Algorithms*, 45(1-4):345–368, 2007.
38. Scott A. Sarra and Derek Sturgill. A random variable shape parameter strategy for radial basis function approximation methods. *Engineering Analysis with Boundary Elements*, 33(11):1239–1245, 2009.
39. E Acar. Optimizing the shape parameters of radial basis functions: An application to automobile crashworthiness. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 224