*get*WAKE

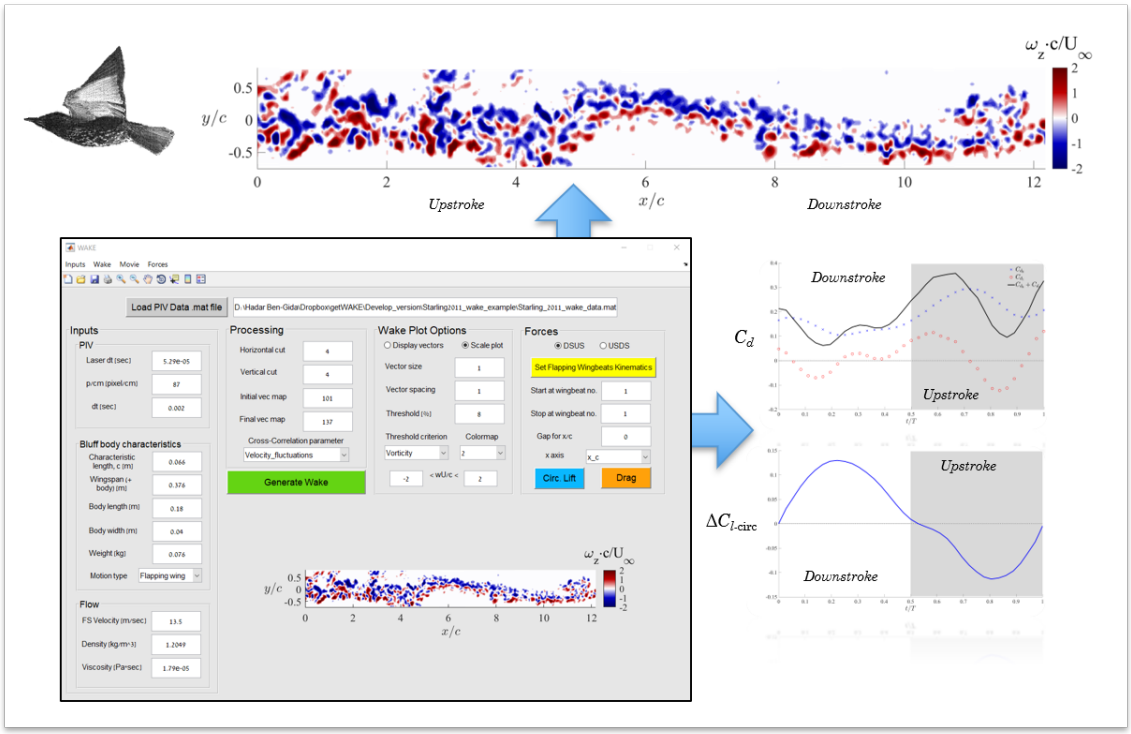
A MATLAB Graphical Interface Package for Bluff Bodies Wake Reconstruction and Forces Estimation from Particle Image Velocimetry Measurements

User's Manual

Ver. 1.7

Dr. Hadar Ben-Gida

June 25, 2020



Contents

1 Introduction

*get*WAKE v1.6 is a collection of Matlab subroutines and GUI for post-processing of time-resolved wake data (velocity vector maps) measured using Particle Image Velocimetry (PIV), and analyzed by OpenPIV (or other) software. *get*WAKE accepts the PIV wake data as .mat files extracted from the OpenPIV spatial analysis toolbox. *get*WAKE allows the user to re-construct a full unsteady wake image from a set of PIV images recorded behind stationary/moving bluff bodies using a cross-correlation algorithm. The *get*WAKE can also be used for estimation of drag and cumulative circulatory lift forces from the wake data, thus enabling the user to estimate the loads exerted on the body generating the wake. To cite this software, please use the following: Ben-Gida, Hadar; Gurka, Roi; Liberzon, Alex (2020): OpenPIV - getWAKE Matlab Toolbox. *figshare*. Software. https://figshare.com/articles/OpenPIV_-_WAKE-GUI_Matlab_ToolBox/12331007. The source code is available for download in the following GitHub page: https://github.com/OpenPIV/getWAKE. Contributions to the current version are more than welcome.

The aim of this document is to deliver a detailed description of the *get*WAKE Matlab Toolbox, along with a tutorial. The PIV data used as an example in the tutorial presented here was extracted from wake measurements behind a freely flying Starling (*Sturnus vulgaris*) in a wind-tunnel [1–3].

The next sections will refer to the main parameters in the *get*WAKE software, wake processing elements and characterization features, output files and post-processing of the wake to estimate forces.

2 Support

The *get*WAKE can run on Matlab 2015a-2020a versions (cross-platform support). The tutorial presented here was executed on Matlab 2020a (64bit Windows 10 OS, PC).

3 Graphical User Interface (GUI)

3.1 Program execution

The *get*WAKE program starts in Matlab from the command window, by executing the following:

```
WAKE
```

Once executed, the Graphical User Interface (GUI) of the *get*WAKE is opened, as depicted in Fig. 1.

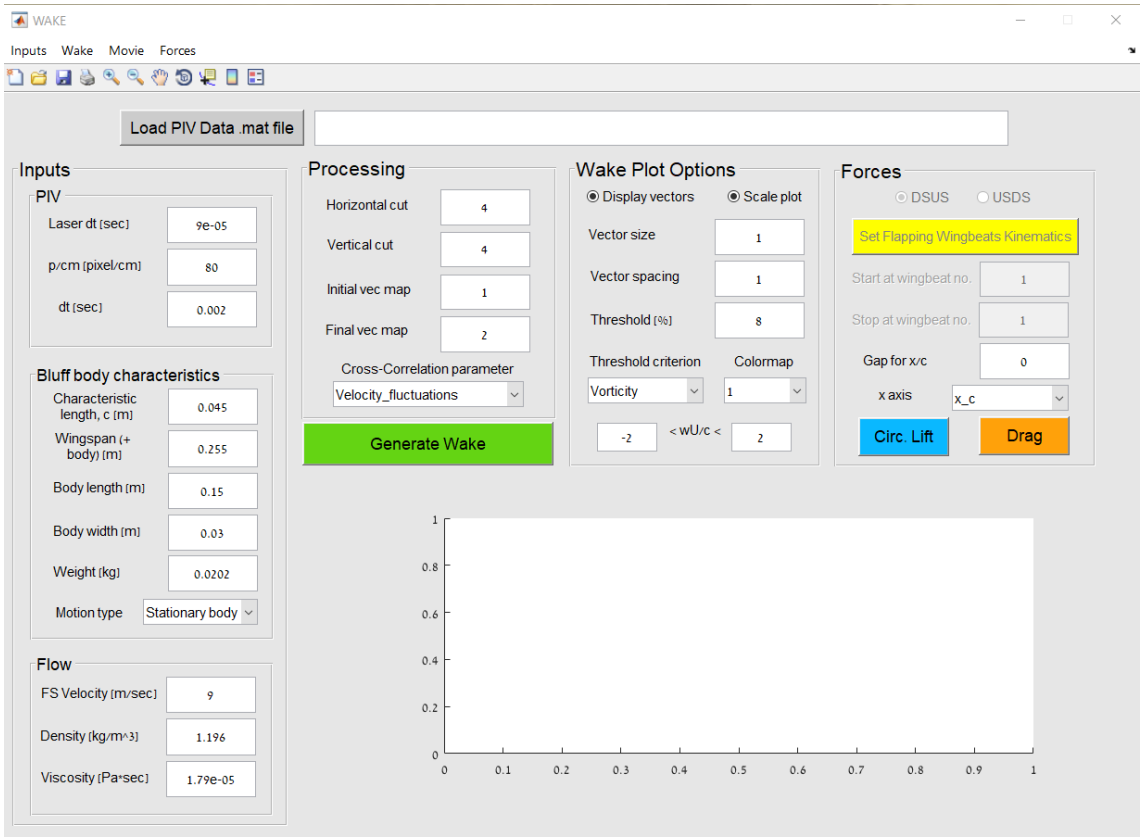


Figure 1: First window of the *get*WAKE

3.2 GUI overview

The *get*WAKE has of several sections (as depicted in Fig. 2):

- **Upper menu bar**: consists of four drop-down menus ("Inputs", "Wake", "Movie" and "Forces"), each has several selectable options that will be explained later in this tutorial.
- **Editor menu**: consists of several tools for editing the GUI objects.
- **PIV Data .mat file path**: placed on the upper middle GUI region and represent the current .mat file path loaded in the GUI (by using the "Load PIV Data .mat file" button) for the wake post analysis. This .mat file consists of the PIV velocity vector maps, in a format exported by the OpenPIV spatial analysis toolbox.
- **"Inputs"**: includes input parameters for the wake analysis in terms of the PIV apparatus, bluff body characteristics and flow conditions.
- **"Processing"**: consists of parameters relevant for generating the complete wake image from the various instantaneous PIV wake images.
- **"Wake Plot Options"**: includes plotting parameters for controlling the presentation of the complete wake image, shown in the lower GUI region.

- **"Forces"**: includes parameters relevant for the estimation of forces from the wake.
- **Wake figure region**: placed on the bottom GUI region, where the wake image is generated by the GUI.

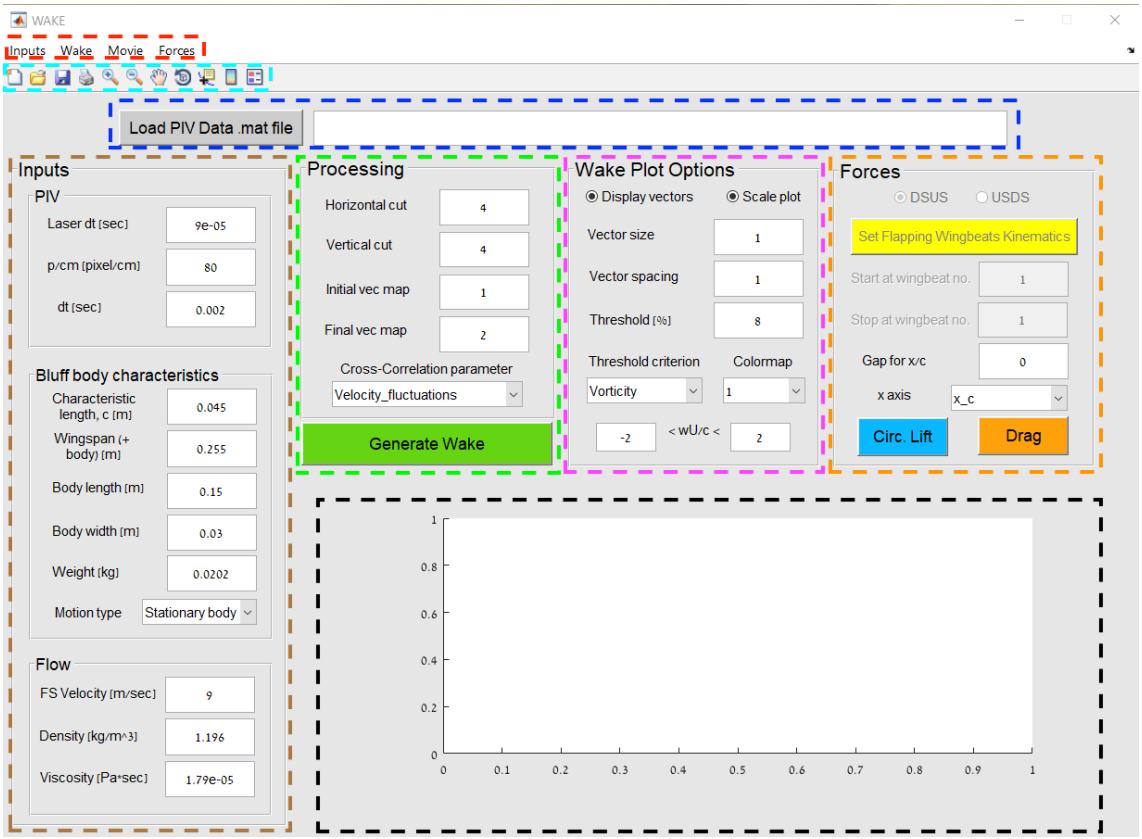


Figure 2: *get*WAKE sections

3.3 Load PIV wake data

The first step is to load the PIV velocity vector maps as a .mat file format. Such .mat file is available by exporting PIV velocity vector maps from the OpenPIV spatial analysis toolbox For more information on the export process and the parameters exist in the exported .mat file that is required as an input for the *get*WAKE, the reader is referred to the following webpage tutorial. The procedure, in short, is as follows: in the OpenPIV spatial analysis toolbox, on the top left menu bar, under the drop-down menu "File", choose one of the load options for the PIV velocity vector maps. Once loaded, export the velocity vector maps by choosing the "Export to MAT file" option (under the drop-down menu "File").

To load the .mat file in the *get*WAKE, click the "Load PIV Data .mat file" button located on the top left region of the GUI, within the PIV Data .mat file path section (see Fig. 2). Once clicked, an explorer window will open ("*select the flow data .mat file*"), from which the user can choose the .mat file containing the PIV

velocity vector maps (see Fig. 3). In this tutorial, we will choose a .mat file entitled "*Starling_2011_wake_data.mat*", which contains velocity vector maps measured in the wake behind a freely flying Starling (*Sturnus vulgaris*) in a wind tunnel. The wake data shown here corresponds to four flapping flight wingbeats performed consecutively by the bird. Each wingbeat is composed of two phases: a downstroke phase, followed by an upstroke phase; with transition in between.

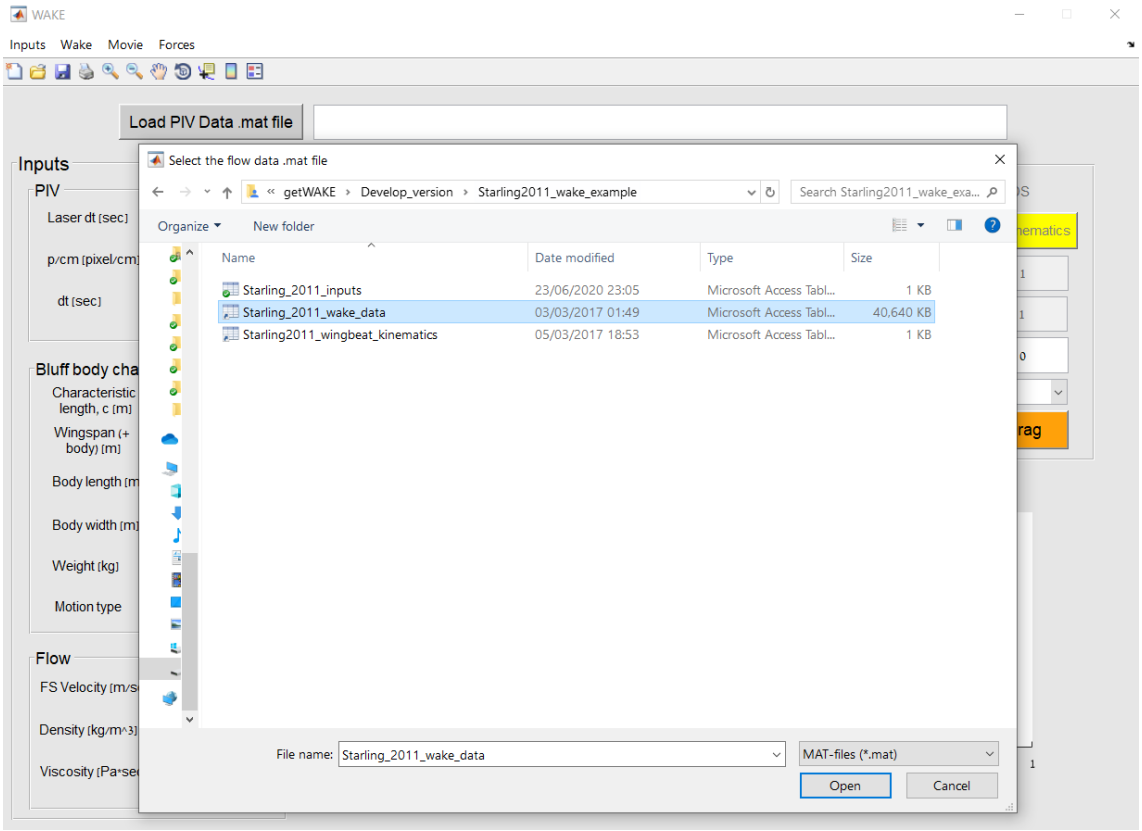


Figure 3: Explorer window for selecting the PIV velocity vecotr maps as .mat file

3.4 Load epxerimental inputs data

On the top left menu bar, under the drop-down menu "Inputs", the option "Load inputs" is available (see Fig. 4), either this option or its shortcut Ctrl+L opens a window where the user can select the experimental inputs data file (.mat format) to be loaded (see Fig. 5). In this tutorial, we will choose a .mat file entitled "*Starling_2011_inputs.mat*", which contains the required experimental inputs for analyzing the wake measured behind the bird. The inputs data .mat file includes information regarding all the parameters within the "Inputs" section in the GUI. The inputs data .mat file can be exported from the GUI by editing the various parameters within the GUI's "Inupts" section in the GUI, and then on the top left menu bar, under the drop-down menu "Inputs", click the option "Save inputs" (see Fig. 4). Either this option or its shortcut Ctrl+I will open a window where the user can export the experimental input data file as a .mat file.

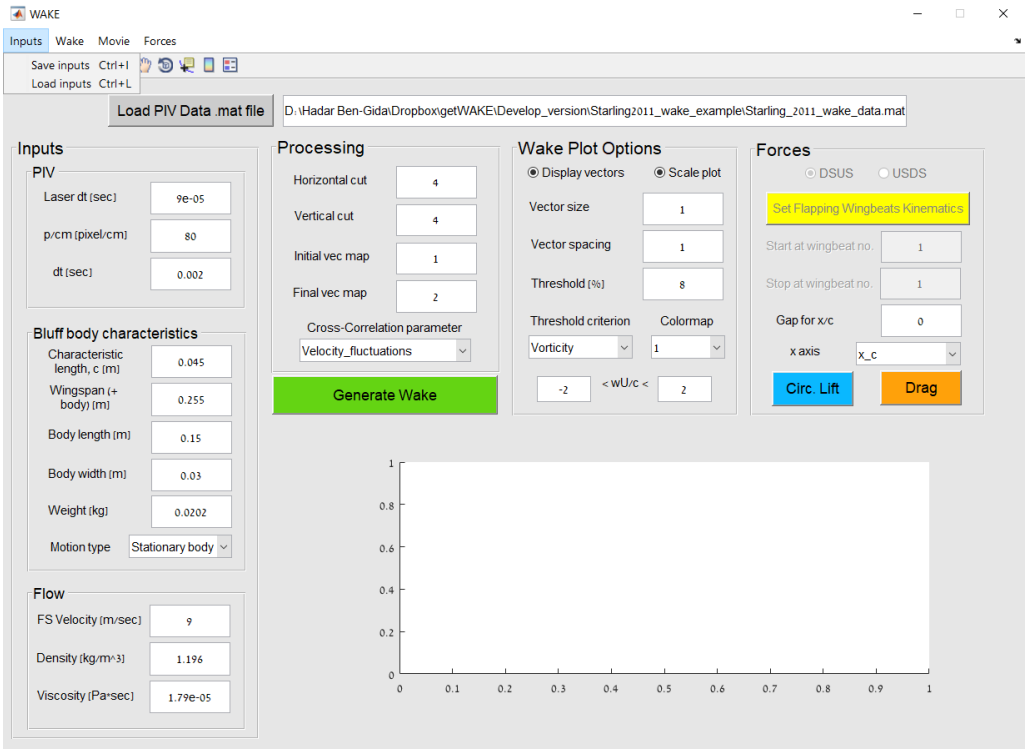


Figure 4: Inputs drop-down menu

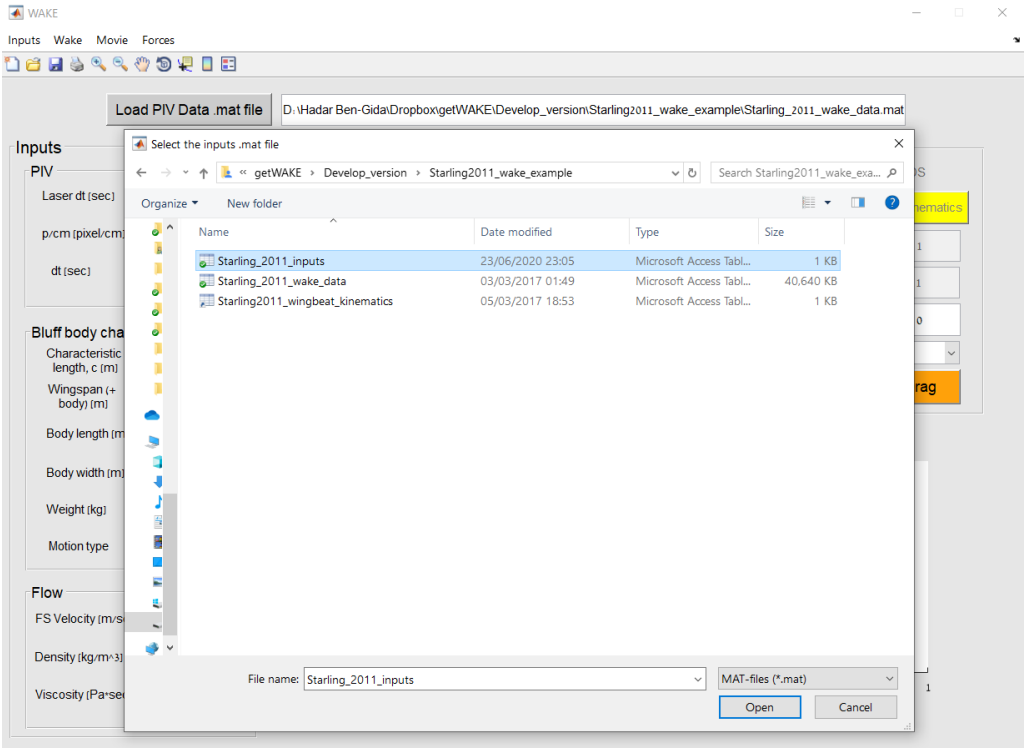


Figure 5: Explorer window for selecting the experimental input data .mat file

The parameters listed in the GUI's "Inputs" section and included in the experimental inputs data .mat file are as follows:

- **Laser dt [sec]**: time difference between the PIV laser's two pulses used to obtain one PIV velocity map from a pair of PIV images in [sec].
- **p/cm [pixel/cm]**: pixel-to-centimeter scaling ratio of the PIV velocity maps.
- **dt [sec]**: time difference between two consecutive PIV velocity maps in [sec].
- **Characteristic length [m]**: body's characteristic length in [m], used for normalizing the physical length scales.
- **Wingspan (+ body) [m]**: body's wingspan length, including the width of the body in [m].
- **Body length [m]**: streamwise length of the body in [m].
- **Body width [m]**: lateral length (width) of the body in [m].
- **Weight [kg]**: body's weight in [kg].
- **Motion type**: determines the motion type conducted by the bluff body. There are two options: i) "*Stationary body*" - for which the body generated the wake is stationary, or ii) "*Flapping wing*" - for which the body generated the wake is moving in a somewhat cyclic manner that resembles that of flapping wingbeats, consisting of three phases; i.e., downstroke, transition and upstroke.
- **FS Velocity [m/sec]**: Freestream velocity in [m/sec].
- **Density [kg/m^3]**: fluid density in [kg/m^3].
- **Viscosity [kg]**: fluid dynamic viscosity in [Pa $\cdot$ sec].

3.5 Load flapping wingbeats kinematics database

In case the body performs a flapping wing type motion (consisting of downstroke, transition and upstroke phases), such as the one performed by a flapping wing or bird, the user can set the kinematics database of the flapping wingbeats by clicking the "Set Flaping Wingbeats Kinematics" button in the right side of the GUI, within the "Forces" section (see Fig. 6). Please note that the "Set Flaping Wingbeats Kinematics" button will only be available if the "*Flapping wing*" motion type is selected for the body.

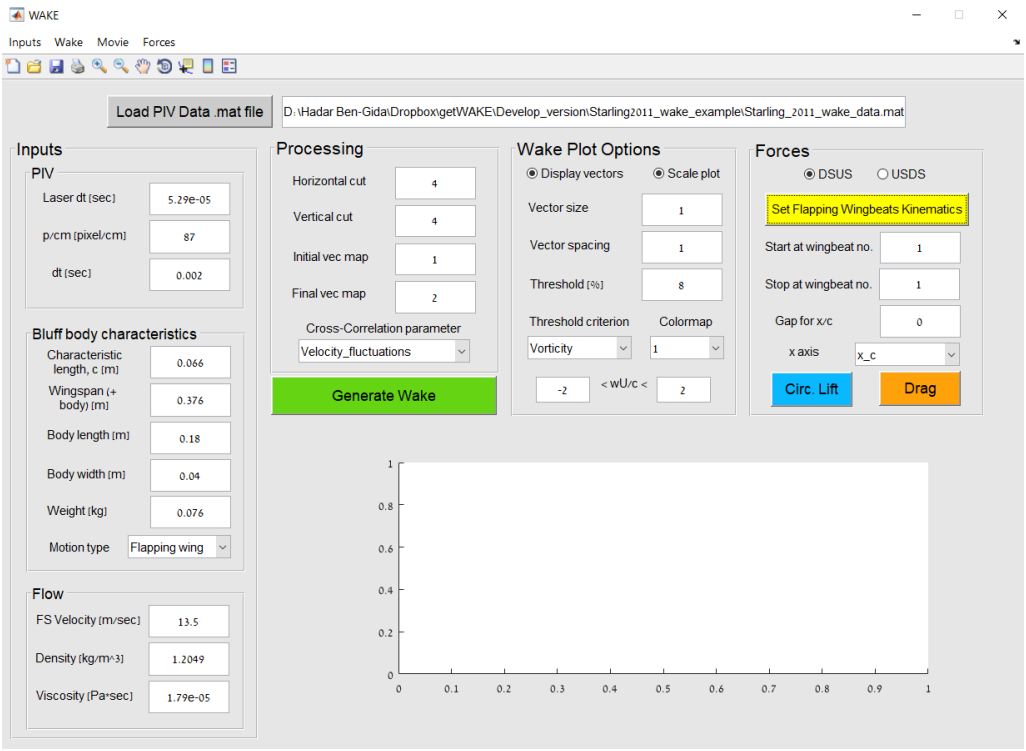


Figure 6: Set flapping wingbeats kinematics database button

This action will open the "Set Kinematics Info" window (see Fig. 7), in which the user can edit, load, export and save the flapping wingbeats kinematics information of the body in terms of the downstroke, transition and upstroke phases. Each row in the database correspond to a certain wingbeat performed by the moving body. The numbers entered underneath each column (downstroke | transition | upstroke) corresponds to the PIV velocity vector map number in the wake data lodaed to the *get*WAKE that correlates with the specific phase.

For example, assume we loaded 175 PIV vector maps describing a moving body's wake (as a .mat file, by clicking the "Load PIV Data .mat file" button in the GUI), where wake image no. 12-58 correlate with the first wingbeat, wake image no. 58-101 correlate with a second consecutive wingbeat, wake image no. 101-137 correlate with a third consecutive wingbeat and wake image no. 137-174 correlate with a fourth consecutive wingbeat. For the first wingbeat, wake image no. 12 marks the start of its downstroke phase, wake image no. 34 marks the end of the downstroke phase and wake image no. 58 marks the end of the upstroke phase. For the second

wingbeat, wake image no. 58 marks the start of its downstroke phase, wake image no. 78 marks the end of the downstroke phase and wake image no. 101 marks the end of the upstroke phase. For the third wingbeat, wake image no. 101 marks the start of its downstroke phase, wake image no. 119 marks the end of the downstroke phase and wake image no. 137 marks the end of the upstroke phase. For the fourth wingbeat, wake image no. 137 marks the start of its downstroke phase, wake image no. 155 marks the end of the downstroke phase and wake image no. 174 marks the end of the upstroke phase. Therefore, one has to edit the kinematics database table as depicted in Fig. 8; adding/removing wingbeats in the table can be done by clicking the "Add Wingbeat"/"Remove Wingbeat" button. Here, the kinematics database presented in Fig. 8 corresponds to four consecutive flapping flight wingbeats of the Starling.

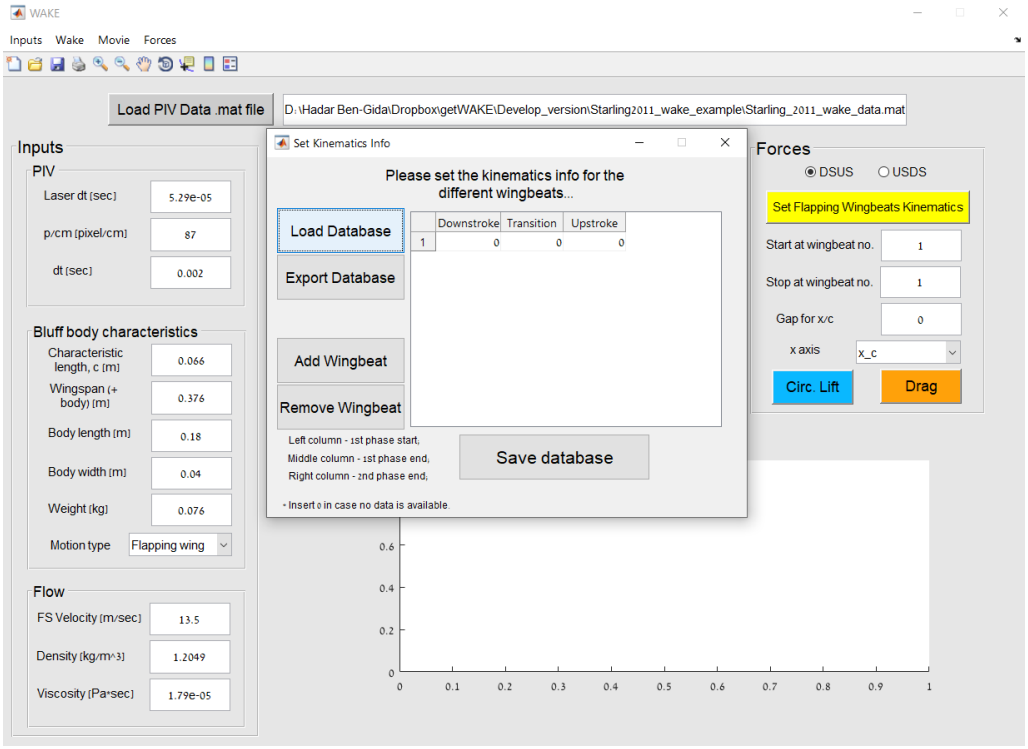


Figure 7: Set Kinematics information window

After finish editing the kinematics databse table, the user can save the database in the *get*WAKE workspace for further analysis, by clicking the "Save database" button in the "Set Kinematics Info" window. The user can also export the kinematics database as a .mat file for future usage by clicking the "Export Databse" button in the "Set Kinematics Info" window. This .mat file format can be re-loaded to the *get*WAKE workspace by clicking the "Load Database" button in the "Set Kinematics Info" window. Please be sure to click the "Save database" button after finish editing the kinematics database and before exiting the "Set Kinematics Info" window, so the database will be available in the *get*WAKE workspace.

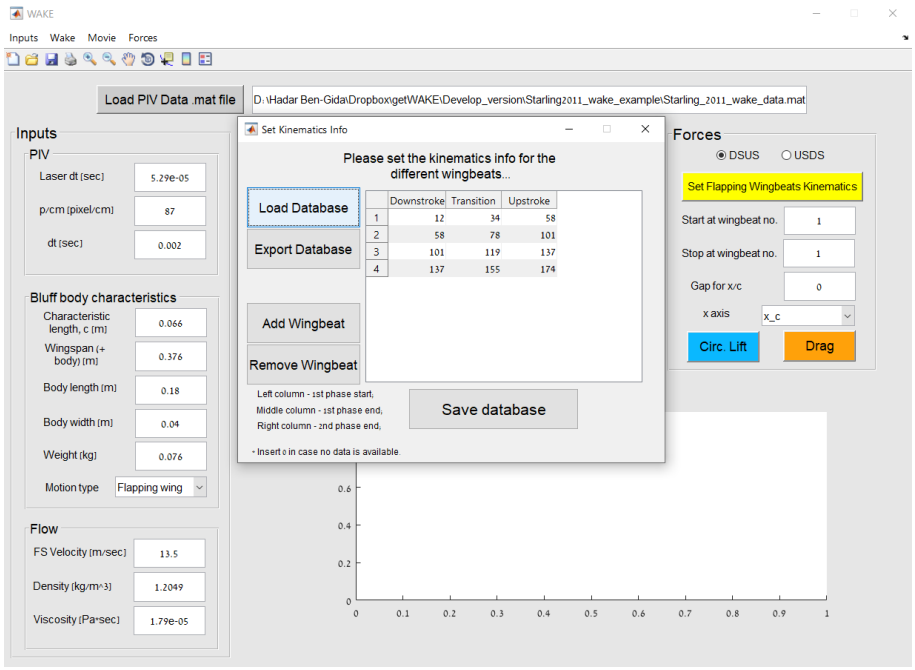


Figure 8: Kinematics database table filled for four consecutive flapping flight wingbeats of a Starling

3.6 Processing the PIV wake data

After importing the PIV wake data, experimental inputs data and kinematics DB (database), one can analyze a sequence of instantaneous wake images (time-resolved PIV velocity maps) and compose a complete wake image, depicting the flow structures as they evolve downstream the source. In the "Processing" section, we enter the following parameters for processing the wake (see Fig. 9):

- **Horizontal cut**: number of vector columns to remove from both the left and rights edges of each instantaneous PIV velocity map, before processing the wake images in the sequence (*default* value is 4).
- **Vertical cut**: number of vector columns to remove from both the top and bottom edges of each instantaneous PIV velocity map, before processing the wake images in the sequence (*default* value is 4).
- **Initial vec map**: initial PIV wake image number in the wake images sequence from the loaded wake data .mat file (*default* value is 1).
- **Final vec map**: final PIV wake image number in the wake images sequence from the loaded wake data .mat file (*default* value is 1).
- **Cross-Correlation parameter**: menu bar for defining the flow parameter according to which the cross-correlation algorithm will perform for each pair of consecutive wake images to reconstruct the complete wake image. Two flow parameters are available for the cross-correlation algorithm: "*velocity fluctuations*" or "*velocity*"; i.e., (u', v') or (u, v). (*default* value is "*velocity fluctuations*").

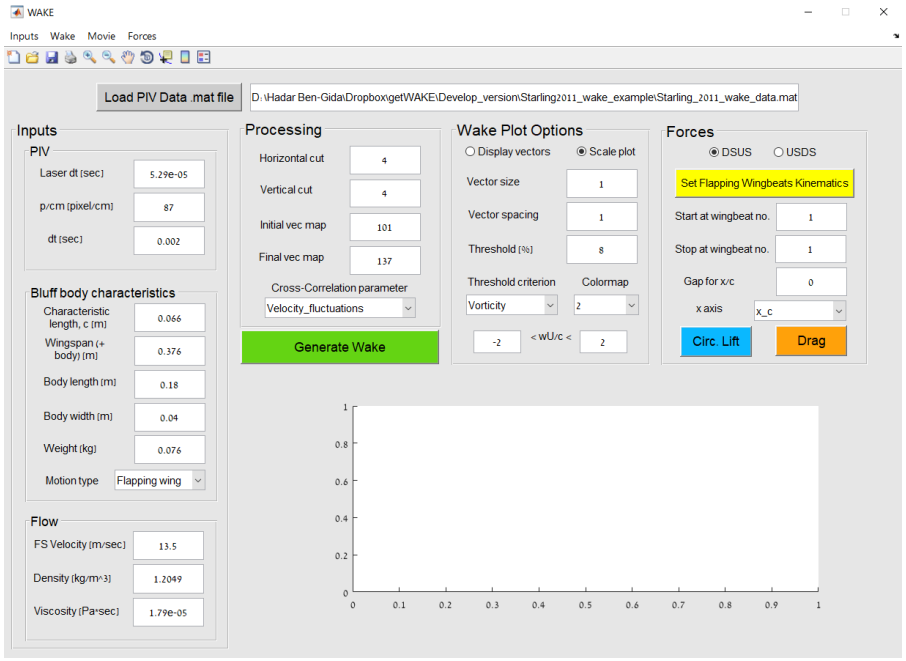


Figure 9: Parameters for processing the PIV wake images

After defining the various parameters above, the user can now generate the complete wake image. This is done by pressing the "Generate Wake' push button. Pressing this button will perform the main computation of generating the complete wake from the sequence of wake images chosen by the user and according to the selected processing parameters. The progress of the computation process is shown in a new open window (see Fig. 10). When the computation completed the complete wake will be plotted in the "Wake figure region" (see Fig. 11).

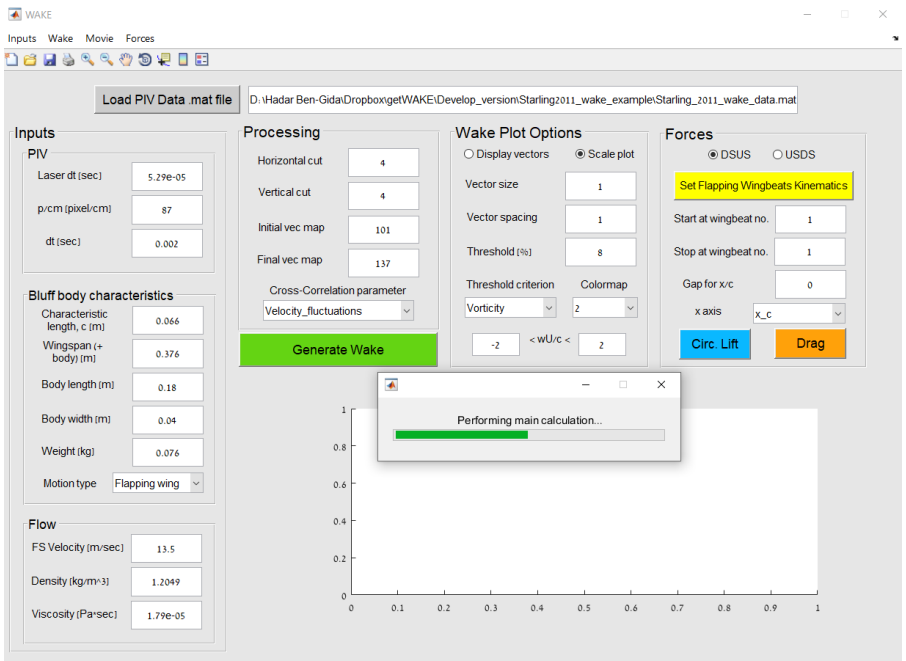


Figure 10: Window showing the progress of the complete wake computation

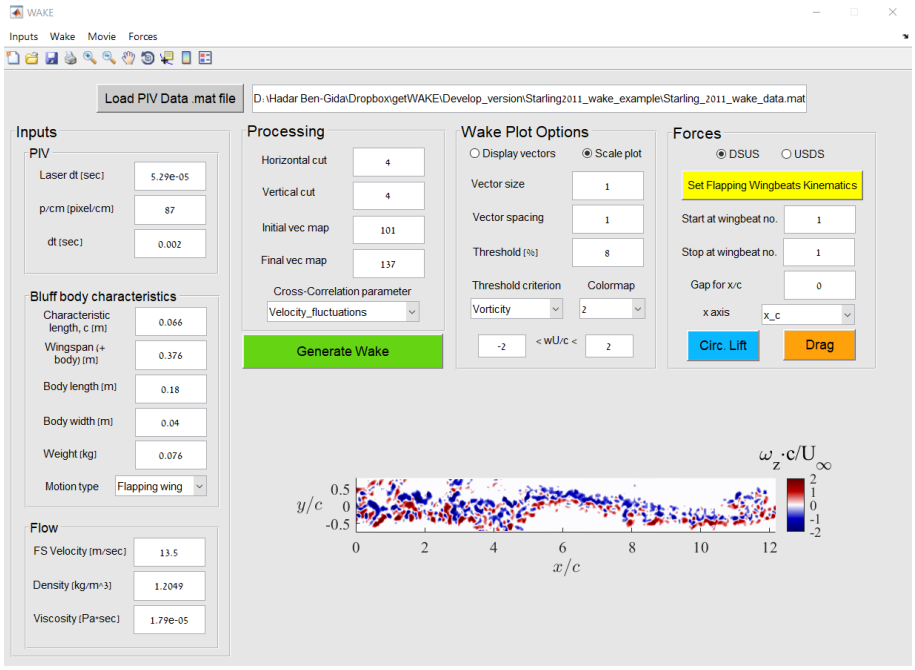


Figure 11: The complete wake image plotted in the "Wake figure region"

3.7 Wake plot options

Once a complete wake image was generated, the user can edit the plot from the "Wake Plot Options" section, by using the following parameters (see Fig. 11):

- **Display vectors**: on/off button for plotting the velocity vectors in the complete wake image (*default* status is *on*).
- **Scale plot**: on/off button for equaling the scale of the complete wake image axes (*default* status is *on*).
- **Vector size**: vectors size plotted in the complete wake image (*default* value is 1).
- **Vector spacing**: spacing between the velocity vectors plotted in the complete wake image (*default* value is 1).
- **Threshold**: threshold percentage value [%] for the vortices in the complete wake image. The vortices visible in the complete wake image will be those with vorticity/swirl (depends on the threshold criterion set by the user) value that exceeds the threshold percentage value from the maximum absolute vorticity/swirl value in the wake image; e.g., assuming the user choose the vorticity as the threshold criterion, if the maximum absolute vorticity in the complete wake image is 5 sec^{-1} and the threshold vorticity percentage set by the user is 8%, then, vortices with absolute vorticity value under 0.4 sec^{-1} will not be shown in the complete wake image (*default* value is 8).

- **Threshold criterion**: the flow quantity according to which a threshold is applied on the complete wake image. Two options are available (*default* parameter is "*vorticity*"):
  - vorticity - $\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$
  - swirl - **Im** $\left[\sqrt{\frac{1}{4}\left(\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}\right)^2+\left(\frac{\partial u}{\partial y}\frac{\partial v}{\partial x}\right)-\left(\frac{\partial u}{\partial x}\frac{\partial v}{\partial y}\right)}\right]$
- **Colormap**: color map for the complete wake image. Five color maps are available in total (*default* value is 1).
- $< \boldsymbol{\omega U/c} <$: range of normalized vorticity values in the color bar of the plotted wake image (*default* values are -2 to 2).

Please note that any editing done in the "Wake Plot Options" section will immediately affect the complete wake image, re-plotting it.

On the top left menu bar, under "Wake", several more options exist. The user can open the wake image in a new figure window for better visualization and inspection of the flow structures in the wake by clicking the option "Open wake in a new window" or using the shortcut Ctrl+P (see Fig. 12). The complete wake image can also be exported as a figure file by clicking the option "Save wake plot" or using the shortcut Ctrl+S (see Fig. 12). Vorticity values computed in the complete wake image can be exported as a .mat. file for further analysis by clicking the option "Save wake vorticity map" or using the shortcut Ctrl+V (see Fig. 12).

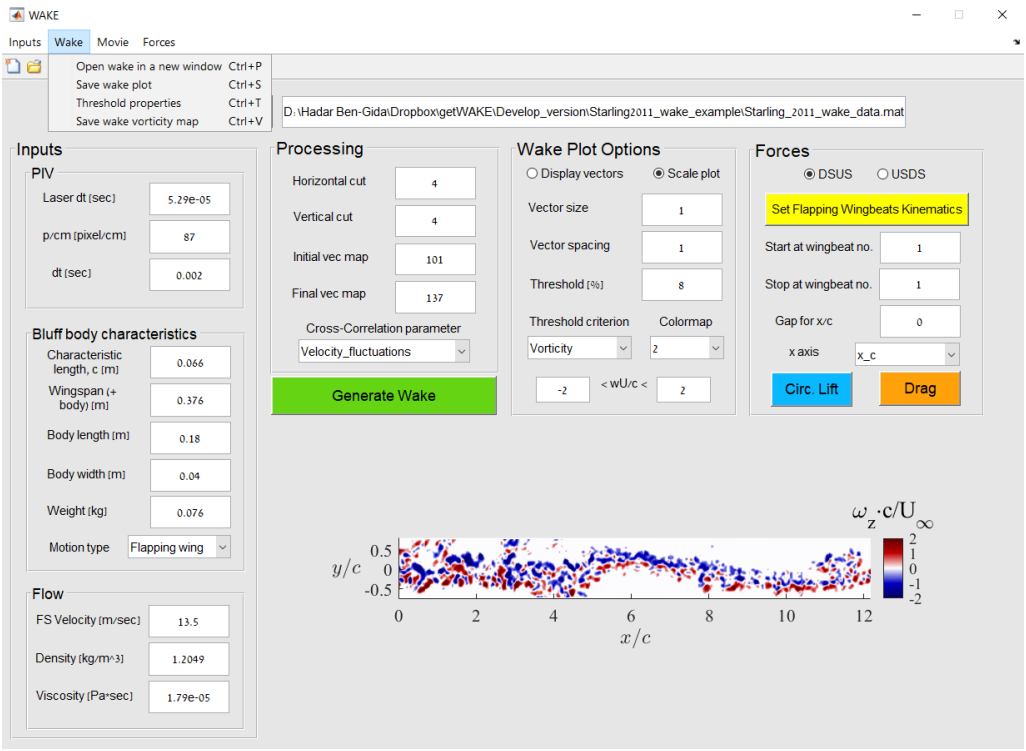


Figure 12: Options in the "Wake" menu bar

The user can edit the properties of the threshold utilized by clicking the option "Threshold properties" or using the shortcut Ctrl+T (see Fig. 12). This will open a new window entitled "Threshold properties" (see Fig. 13).

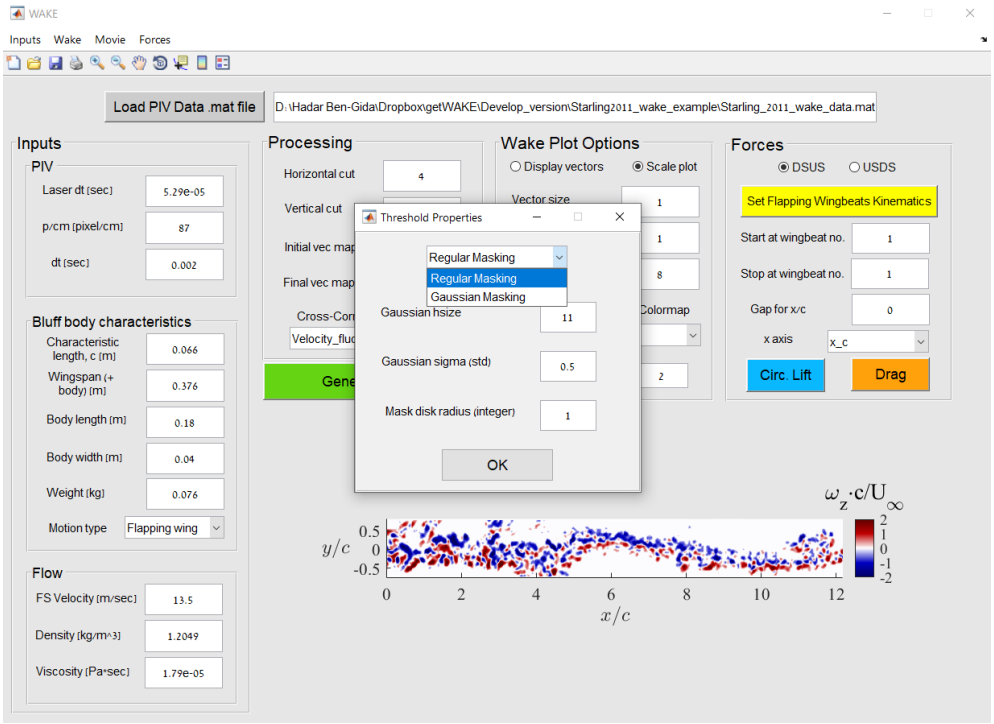


Figure 13: Wake threshold properties

Two masking options exist for the threshold to be applied on the complete wake image (*default* option is "*Regular Masking*"):

- "Regular Masking" - applies the threshold directly on the wake data array with no smoothing.
- "Gaussian Masking" - applies a 2D rotationally symmetric Gaussian lowpass filter with size that is indicated in "Gaussian hsize" and standard deviation that is indicated in "Gaussian sigma (std)". The radius of the masking Gaussian disk is indicated by "Mask disk radius (integer)" in terms of array elements.

When finished, clicking the "OK" button will confirm the changes done.

The user can also export a movie of the PIV wake data sequence chosen in the "Processing" section. On the top left menu bar, under the drop-down menu "Movie", the option "Save movie" is available (see Fig. 14), either this option or its shortcut Ctrl+M will export a movie of the PIV wake data sequence. The properties of the exported movie can be determined from the "Movie properties" option under the drop-down menu "Movie". Either this option or its shortcut Ctrl+N will open a new window entitled "Movie properties" (see Fig. 15), in which the user can set the frames per second rate in the exported movie.

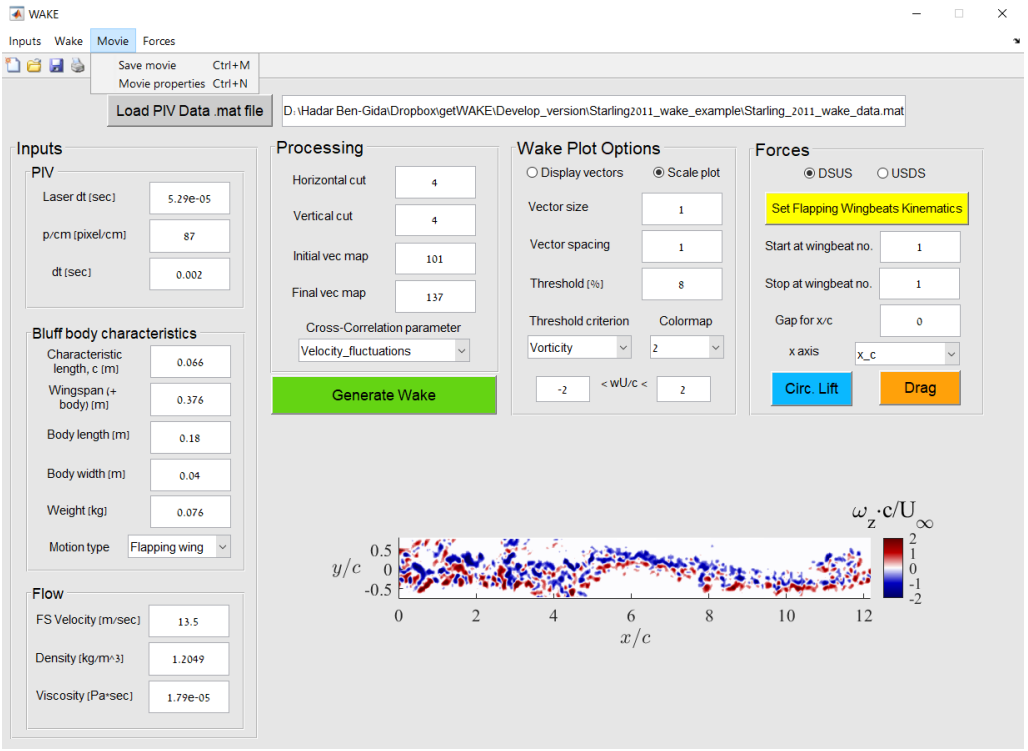


Figure 14: Options in the "Movie" menu bar

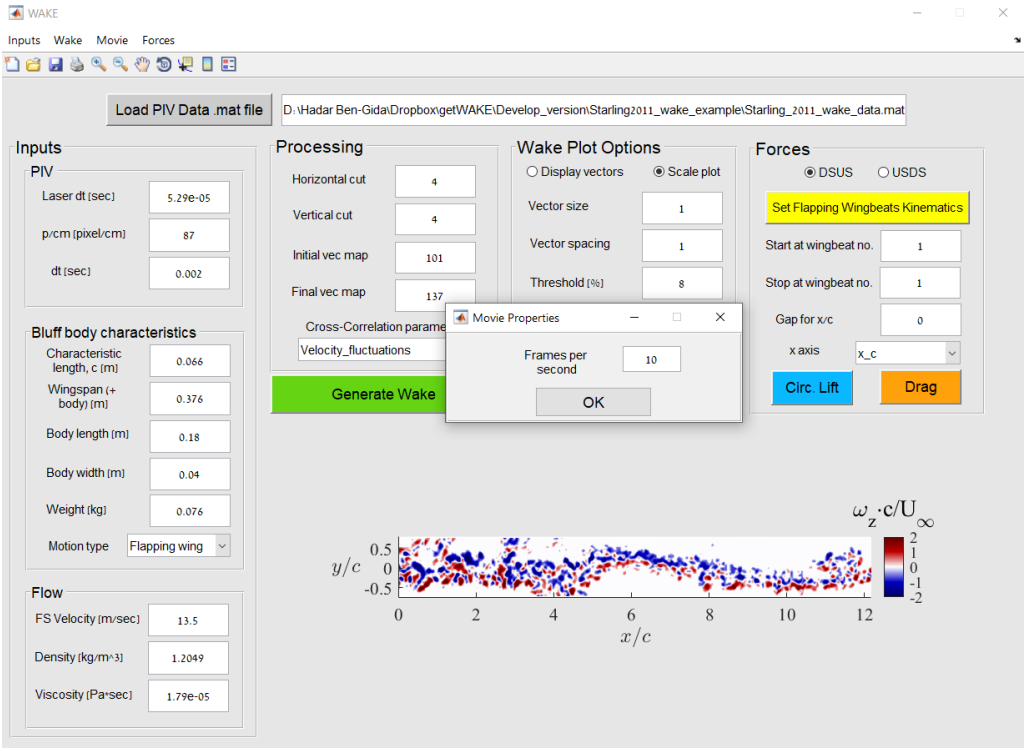


Figure 15: Movie properties

3.8 Forces estimation

The user can estimate the variation of drag and cumulative circulatory lift throughout the wake, as a function of time or streamwise distance (x/c). Several parameters are availabel in the "Forces" section (see Fig. 16):

- **DSUS**: on/off button for specifying that the flapping wingbeats are characterized first with a downstroke phase that is followed with an upstroke phase (*default* status is *on*). Please note that this push button will only be available if the "*Flapping wing*" motion type is selected for the body.
- **USDS**: on/off button for specifying that the flapping wingbeats are characterized first with an upstroke phase that is followed with a downstroke phase (*default* status is *off*). Please note that this push button will only be available if the "*Flapping wing*" motion type is selected for the body.
- **Gap for x/c**: setting an artificial normalized streamwise extent shift, after which the forces estimation will be plotted; e.g., if set to 5, the forces plots will not start from $x/c = 0$, but from $x/c = 5$. This parameter is only applicable for plotting the forces vs. the streamwise distance, x/c (*default* value is 0).
- **Set Flapping Wingbeats Kinematics**: push button (see Section 3.5); when clicked will open a new window entitled "Set Kinematics Info" for setting the kinematics database of the various wingbeats. The phases definition in this database will determine how the forces variation throughout the wake will be presented for the various wingbeat cycles. Please note that this push button will only be available if the "*Flapping wing*" motion type is selected for the body.
- **Start at wingbeat no.**: initial wingbeat no. from which the forces variation plot will start, according to the kinematics database (*default* value is 1). Please note that this push button will only be available if the "*Flapping wing*" motion type is selected for the body.
- **Stop at wingbeat no.**: final wingbeat no. up to which the forces variation will be presented, according to the kinematics database (*default* value is 1). Please note that this push button will only be available if the "*Flapping wing*" motion type is selected for the body.
- **x axis**: define the x axis in the plots of the forces variation throughout the wake. Two options are available (*default* option is "x_c"):
  - "x_c" - plots the forces estimation vs. the normalized streamwise distance of the wake, x/c.
  - "t_T" - plots the forces estimation vs. the normalized time, t/T; where T is the wingbeat total time period.

For the wake behind a stationary body (when the "*Stationary body*" option is selected in "Motion type"), the variation of the drag and cumulative circulatory lift will be plotted throughout the wake maps sequence, as determined from "Initial vec map"

and "Final vec map". For the wake behind a body that performs a flapping wing motion (when the "*Flapping wing*" option is selected in "Motion type"), the variation of the drag and cumulative circulatory lift will be plotted for the various wingbeat cycles, as defined in "Set Flapping Wingbeats Kinematics" and selected by "Start at wingbeat no." and "Stop at wingbeat no.". Please note, the user can plot the forces variation for multiple wingbeats ("Start at wingbeat no."<"Final at wingbeat no.") or for a single wingbeat ("Start at wingbeat no."="Final at wingbeat no."). If choosing to plot the forces variation for multiple wingbeats, the x axis can only be depicted in terms of absolute time, t, and not the normalized time, t/T; as the time period, T, can be different for multiple wingbeats.

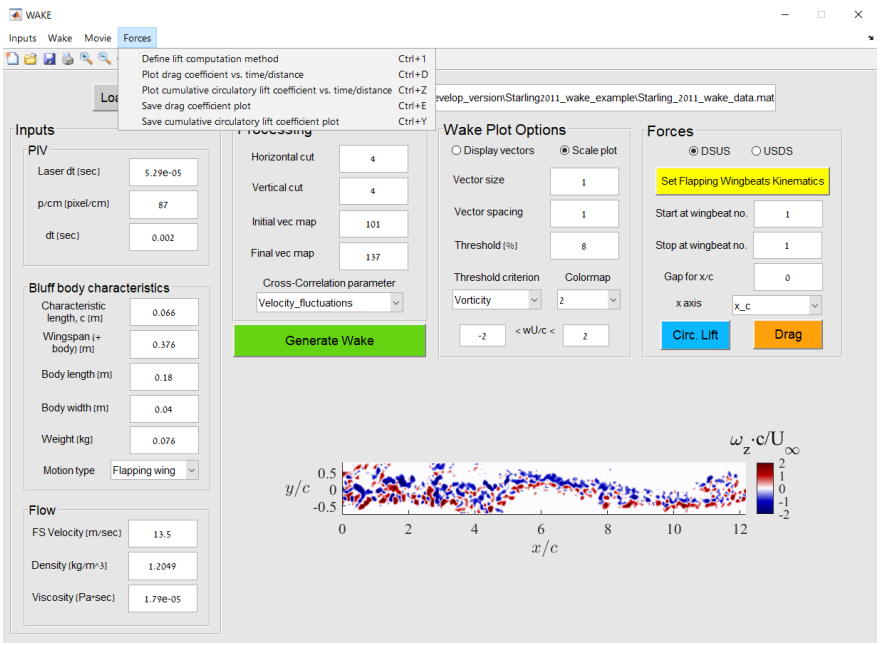


Figure 16: Options in the "Forces" menu bar

Ploting the variation of forces along the wake is available on the top left menu bar, under the drop-down menu "Forces". The option "Plot cumulative circulatory lift coefficient vs time/distance", its shortcut Ctrl+Z or the push button "Circ. Lift" (see Fig. 16) will plot the variation in the cirulatory lift coefficient $\Delta C_{l_{circ}}$ vs. time (see Fig. 17) or x/c (see Fig. 18), depending on the "x axis" parameter. The option "Save cumulative circulatory lift coefficient plot" or its shortcut Ctrl+Y will export the variation in the circulatory lift coefficient as an image file. Fig. 19 depicts the time variation of the cumulative circulatory lift coefficient for four consecutive Starling wingbeats. The option "Plot drag coefficient vs time/distance", its shortcut Ctrl+D or the push button "Drag" (see Fig. 16) will plot the drag coefficient C_d vs. time or x/c, depending on the "x axis" parameter (see Fig. 20). The option "Save drag coefficient plot" or its shortcut Ctrl+E will export the drag coefficient figure as an image file. Please note, the white shaded region in the above plots corresponds to the downstroke phase in the wingbeat, whereas the gray shaded region corresponds to the upstroke phase. C_{d_0} and C_{d_1} are the steady and unsteady components of the drag coefficient, respectively.

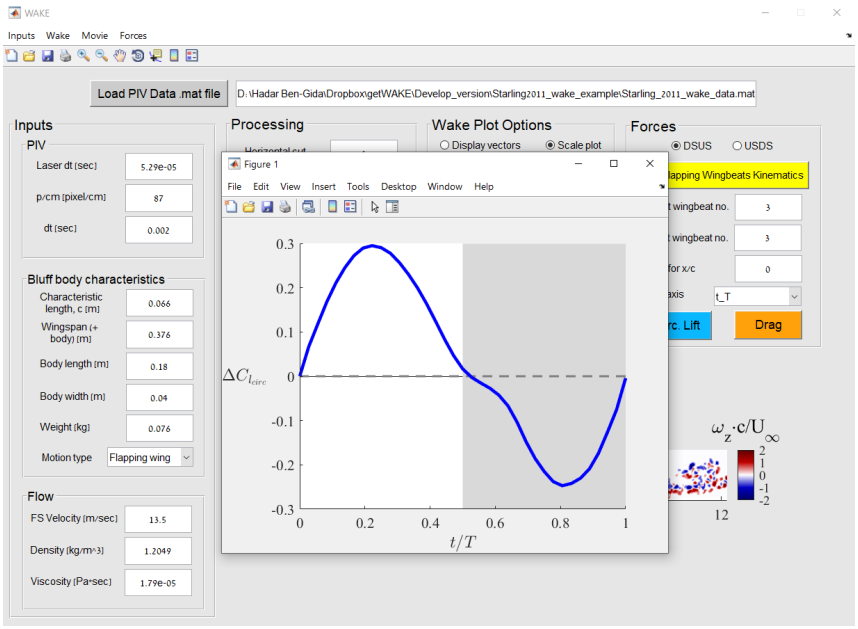


Figure 17: Variation of the cumulative circulatory lift coefficient with t/T for the 3rd Starling's wingbeat

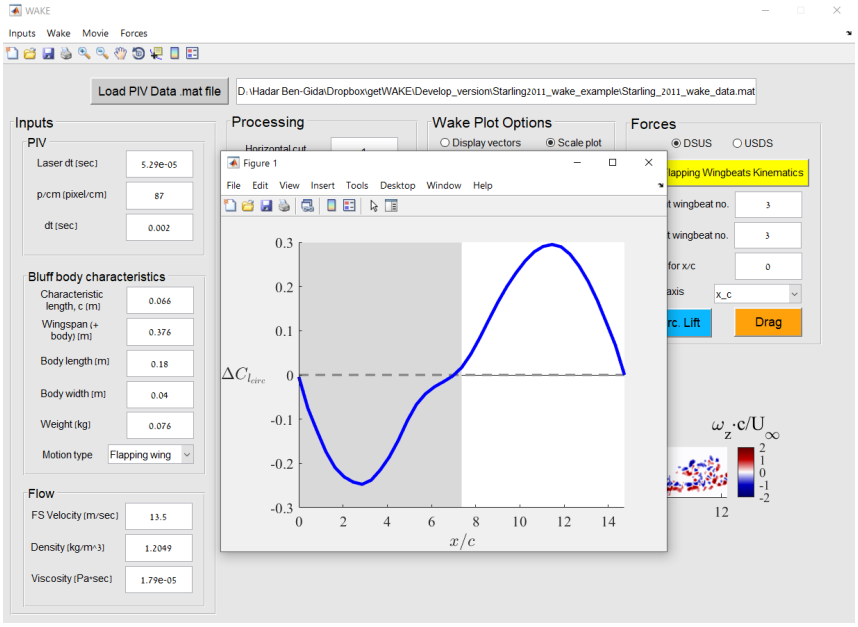


Figure 18: Variation of the cumulative circulatory lift coefficient with x/c for the 3rd Starling's wingbeat

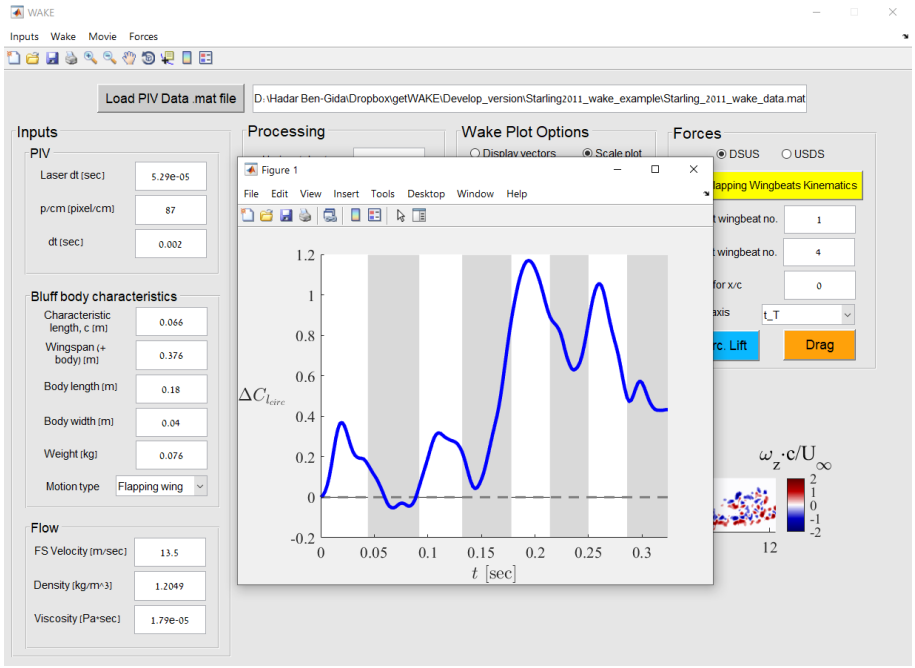


Figure 19: Time variation of the cumulative circulatory lift coefficient for four consecutive Starling wingbeat

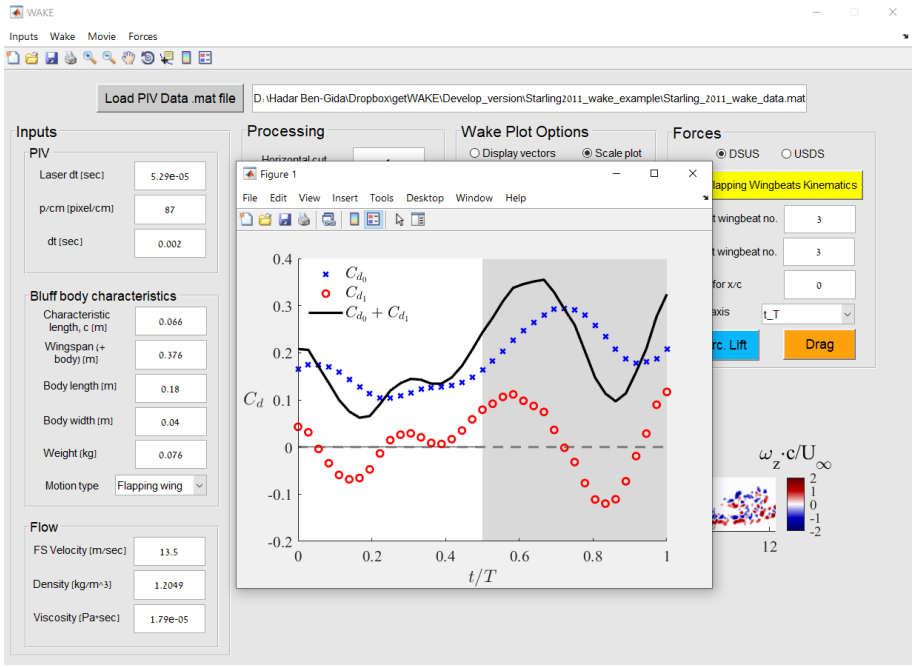


Figure 20: Variation of the drag coefficient with t/T for the 3rd Starling's wingbeat

The method for computing the cumulative circulatory lift throughout the wake can be determined from the "Define lift computation Method" option under the drop-down menu "Forces". Either this option or its shortcut Ctrl+1 will open a new window entitled "Lift Computation Method" (see Fig. 21), where the user can set the method for computing the lift. A total of four options are available (*default* method is "*Panda & Zaman (1994), based on PIV individual maps (threshold applied)*"):

- **Panda & Zaman (1994), based on PIV individual maps (threshold applied)**: computing the cumulative circulatory lift coefficient, based on Panda and Zaman [4], from individual PIV maps data across the wake (after applying the threshold).
- **Panda & Zaman (1994), based on generated wake**: computing the cumulative circulatory lift coefficient, based on Panda and Zaman [4], from the complete wake image data.
- **Panda & Zaman (1994), based on generated wake (threshold applied)**: computing the cumulative circulatory lift coefficient, based on Panda and Zaman [4], from the complete wake image data (after applying the threshold).
- **Cumulative sum**: computing the cumulative circulatory lift coefficient from a direct summation of the circulation values of individual PIV maps along the wake (with the threshold applied).

For more information regarding the above options, see Section 5.2. Further information regarding the drag coefficient computation is given in Section 5.1.

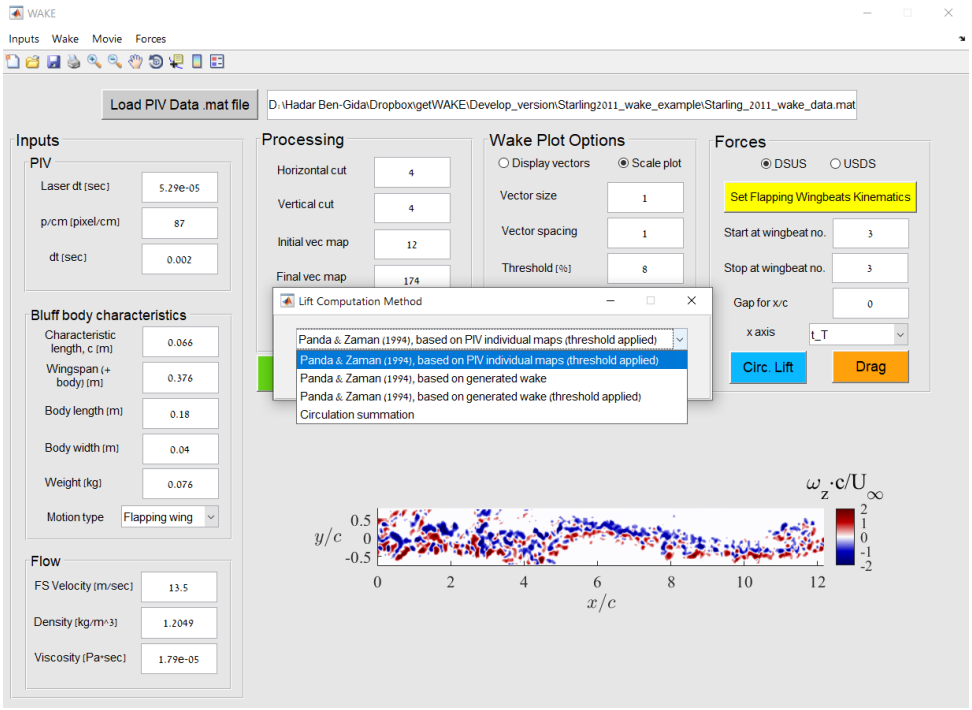


Figure 21: Cumulative circulatory lift computation methods

4 Wake Reconstruction

The main motivation in the *get*WAKE Matlab Toolbox is to describe, visually, the wake evolution behind flapping birds or bodies undergoing unsteady motion to observe the formation of flow features as they shed downstream [1–3, 5, 6]. A detailed wake characterization can help to investigate the flow dynamics experienced by the body. Utilization of a long-duration time-resolved PIV system enables the wake measurement behind flying birds and bodies for a relatively long time (or distance) and with high spatial and temporal resolution. The wake reconstruction (see an example wake measured behind a flapping Starling in Fig. 22), which is the main output of this toolbox, is based on PIV images taken from a stationary camera yielding Eulerian observation of the flow field behind the body. For the wake reconstruction, we assume the bird/body's position did not change much relative to the measurement plane. By invoking Taylor's hypothesis [7], we allow the assumption that the flow remains relatively unchanged as it passes through the measurement plane. This hypothesis implies that there is no significant time dependence of a spatial velocity distribution over the timescale required for observation. Zaman and Hussain [8] showed that the hypothesis works well for an isolated coherent structure if a constant convection velocity, equal to the structure centre velocity, is used in the case of shear flows.

The wake composite image is generated by offsetting each consecutive PIV image with a calculated instantaneous convection velocity and then overlapping the images, while keeping the mid region of each instantaneous PIV image. The instantaneous convection velocity, which determines the offset if each PIV image, is calculated based on a cross-correlation algorithm that examine the match of the velocity vector fields (u, v) or the fluctuating velocity vector fields (u', v') of two consecutive PIV maps. The cross-correlation coefficient, which determines the match of two consecutive PIV maps, is calculated as follows:

$$C_p(X, Y, T) = \sum_{i=1,j=1}^{I,J} \frac{\left[p(x_i, y_i, t) - \overline{p(t)}\right]\left[p(x_i + X, y_i + Y, t + T) - \overline{p(t + T)}\right]}{IJ\sigma_p(t)\sigma_p(t + T)} \quad (1)$$

If $C = 1$, the two PIV maps are identical; i.e., perfectly matched. If the two images are different, then $C < 1$. I, J is the size of the overlapping area and σ_p is the standard deviation of the flow property p. Here, p is to be replaced with u and v for computing the cross-correlation coefficients with respect to the velocity vector maps (C_u, C_v), or with u' and v' for computing the cross-correlation coefficients with respect to the velocity fluctuations vector maps $(C_{u'}, C_{v'})$. The spatial shift (X, Y) of any instantaneous PIV map is computed based on overlapping that achieves the maximum correlation coefficient; i.e., $\max[C_u, C_v]$ or $\max[C_{u'}, C_{v'}]$. Please note that for the cross-correlation algorithm to work properly, the sampling rate of the PIV velocity vector maps must be at least 1.7 orders-of-magnitude higher than the wake flow field base frequency, thus allowing sufficient temporal resolution to track the wake flow structures as they propagate from one velocity map to another. For example, if the wake is measured behind a bird in flapping flight with a flapping

frequency of 10 Hz, the PIV velocity vector maps must be sampled at a rate of 500 Hz or more. i.e., the sampling rate of the PIV raw images has to be 1 kHz or higher. If the cross-correlation failed during the *get*WAKE wake reconstruction process (e.g., due to poor PIV data), the spatial shift of each instantaneous PIV map is determined from the freestream advection velocity U_∞ and the time difference Δt; i.e., $X = U_\infty \Delta t$. Please note to not remove to many vector rows/columns from the PIV map (using "Horizontal cut" or "Vertical cut"), otherwise the cross-correlation may fail due to insufficient overlapping area.

In the example wake below the bird flew from right to left; therefore, the downstream distance is measured as positive chord lengths. What appears as downstream essentially happened earlier, while what appears as upstream happened later.

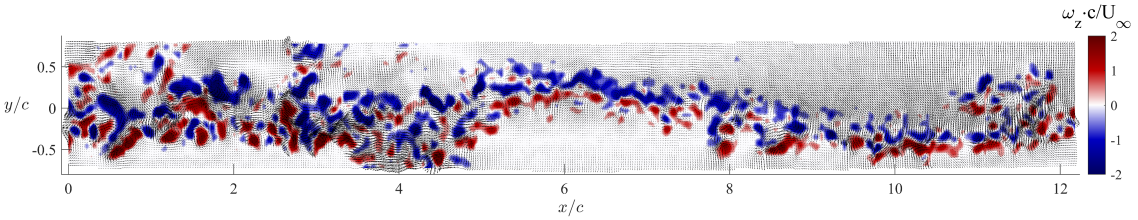


Figure 22: Example of a wake reconstructed by the *get*WAKE from PIV wake measurements taken behind a freely flying Starling during a complete wingbeat. The contour is the normalized spanwise vorticity in the wake

5 Forces Estimation

5.1 Drag coefficient C_d

The variation of the drag force can be estimated from the PIV wake data based on Ben-Gida et al. [1]. The analysis leads to the following equation for the two-dimensional drag coefficient computed in the wake:

$$C_d = \underbrace{\frac{2}{cU_\infty}\int_0^h u\left(1-\frac{u}{U_\infty}\right)\mathrm{d}y}_{C_{d_0}\text{ - Steady part}} - \underbrace{\frac{2}{cU_\infty}\frac{\partial}{\partial t}\int_0^h\int_0^l \frac{u}{U_\infty}\mathrm{d}x\mathrm{d}y}_{C_{d_1}\text{ - Unsteady part}} \tag{2}$$

The steady drag coefficient component C_{d_0} is referred to as the velocity deficit drag, whereas the unsteady drag coefficient component C_{d_1} is added due to the unsteady flow motion. While the steady drag term can be obtained from the near wake velocity field, the unsteady drag term requires information regarding the entire control surface surrounding the source over time. We assume most of the unsteady disturbances generated by the unsteady motion are obtained from the velocity field at the near wake where both unsteady contribution and viscous effects have not dissipated yet. Therefore, we approximate the full surface integral of the unsteady term to include only the velocity field obtained from the PIV experiments in the near wake behind the birds/bodies. Here, U_∞ is the mean undisturbed streamwise velocity, and h and l are the vertical and horizontal extent of the computed velocity field

in the wake, respectively. For the steady drag coefficient, we average the various profiles along the streamwise extent of each PIV map, thus yielding a single C_{d_0} value to represent each PIV map. Fig. 20 depicts an example of the drag coefficient variation in the wake measured behind a freely flying Starling.

5.2 Cumulative circulatory lift coefficient $\Delta C_{l_{circ}}$

The variation of the cumulative circulatory lift coefficient can be estimated from the PIV wake data based on Stalnov et al. [2], Ben-Gida et al. [9] and Nafi et al. [3]. Any unsteady motion of a lifting surface is accompanied by shedding of vortices into the wake. Assuming potential flow, one can utilize the unsteady thin airfoil theory [10, 11], which assumes a planar wake evolution, to compute the time-dependent lift of a two-dimensional lifting surface. Theodorsen [10] derived an expression for total time-dependent lift which is composed of non-circulatory and circulatory terms whereas Sears and von-Kármán [11] unsteady lift derivation contains quasi-steady term (produced by instantaneous bound circulation), added mass term (generated by inertia of the fluid moving along with the lifting surface) and wake-induced lift term (produced by the wake vorticity). For simple harmonic motion, both methods provide similar expressions. The non-circulatory term from Theodorsen's [10] formulation is essentially the added mass term whereas the circulatory term is equivalent to sum of quasi-steady lift term and wake-induced lift term.

In the near wake of flapping birds, we assume that the wake has not deformed yet and interactions between the vortices shed into the wake are not significant [2], thus allowing the use of the planar wake assumption and consequently the unsteady thin airfoil theory. Although general trends may be well described with the use of such theory, it is the small scales in the wake that are actually of interest. When analyzing these vortical structures in the near wake of flying birds, one can deduce a more suitable viscous-based method for the unsteady lift estimation. The estimation of increment in the time-dependent lift throughout the wake is evaluated here from the PIV velocity fields by utilizing Wu's viscous flow approach [12], which was later expressed by Panda and Zaman [4]. Assuming two-dimensional, incompressible flow and neglecting the added mass term, the time-dependent circulatory lift force can be expressed as [4]:

$$L_{circ}(t) = \underbrace{\rho \frac{\mathrm{d}}{\mathrm{d}t} \iint_A x\omega_z(t)\mathrm{d}x\mathrm{d}y}_{\text{x-moment of the vorticity field}} + \underbrace{\rho U_\infty \int_0^t \int_0^h \nu \left(\frac{\partial^2 u(t)}{\partial x^2} + \frac{\partial^2 u(t)}{\partial y^2} \right) \mathrm{d}y\mathrm{d}t}_{\text{Diffusion contribution}} \quad (3)$$

where x and y are the Cartesian coordinates defining the wake flow plane, with x axis in the freestream direction. In the above equation, the first integral from left is the first x-moment of the vorticity field (A) in the wake, with $\omega_z(t)$ as the instantaneous spanwise vorticity field ($= \partial v/\partial x - \partial u/\partial y$) that is evaluated directly from the PIV data using a least squares differentiation scheme. The second integral from left is the contribution from the viscous term (diffusion), with ν as the kinematic viscosity and h denoting the vertical extent of the computed velocity field in the wake. This term may be negligible in the far wake, yet, here we take it into account for near-wake effects. ρ is the fluid density. Please note that in our notation in Eq. (3), a

minus sign is **not** placed ahead of the first integral from left. This is because in our analysis we define positive vorticity values in the wake region as counter-clockwise vortices, whereas Panda and Zaman [4] defined them as clockwise vortices.

Applying Taylor's hypothesis (as introduced above for the wake reconstruction), $\mathrm{d}x = U_\infty \mathrm{d}t$, one can transform the spatial derivative in the left-side integral of Eq. (3) into a temporal one. Moreover, by interchanging the left-side integral in Eq. (3) with the time derivative (using Leibniz integral rule), one can re-write Eq. (3) as follows:

$$L_{circ}(t) = \rho U_\infty \int_0^t \left[\int_0^h u\omega_z(t)\mathrm{d}y + \int_0^h \nu \left(\frac{\partial^2 u(t)}{\partial x^2} + \frac{\partial^2 u(t)}{\partial y^2} \right) \mathrm{d}y \right] \mathrm{d}t \tag{4}$$

Therefore, the change in the lift in time $\delta\tau$ can be expressed accordingly:

$$\delta L_{circ} = \rho U_\infty \delta\Gamma \tag{5}$$

where the corresponding change in the circulation $\delta\Gamma$ is given by:

$$\delta\Gamma = \int_0^h u\omega_z(t)\mathrm{d}y + \int_0^h \nu \left(\frac{\partial^2 u(t)}{\partial x^2} + \frac{\partial^2 u(t)}{\partial y^2} \right) \mathrm{d}y \tag{6}$$

Since at the beginning of the unsteady motion the lift is unknown, we shall refer to the estimated lift component as an increment in the circulatory lift [2, 9] that is generated from the beginning of the motion. Based on Eq. (4), the cumulative circulatory lift at time t, $\Delta L_{circ}(t)$, is computed accordingly:

$$\Delta L_{circ}(t) = \rho U_\infty \int_0^t \zeta(t)\mathrm{d}t = \rho U_\infty \Gamma(t) \tag{7}$$

with the vorticity flux term $\zeta(t)$ being expressed as follows:

$$\zeta(t) = \int_0^h u_c\omega_z(t)\mathrm{d}y + \int_0^h \nu \left(\frac{\partial^2 u(t)}{\partial x^2} + \frac{\partial^2 u(t)}{\partial y^2} \right) \mathrm{d}y \tag{8}$$

Here, u_c is the convection velocity at which the characteristics of the wake collectively travel downstream. The cumulative circulatory lift coefficient at time t is expressed as:

$$\Delta C_{l_{circ}}(t) = \frac{2}{cU_\infty} \int_0^t \zeta(t)\mathrm{d}t = \frac{2\Gamma(t)}{cU_\infty} \tag{9}$$

As discussed in Section 3.8, the *get*WAKE suggests four methods for computing the cumulative circulatory lift coefficient in the wake. In the *default* option entitled **Panda & Zaman (1994), based on PIV individual maps (threshold applied)**, the vorticity flux $\zeta(t)$ defined by Eq. (8) is estimated **for each individual PIV map obtained in the near wake** behind the source as function of time; after applying a threshold on the vorticity contours. For each PIV map (or time t), the convection velocity is defined as the mean streamwise velocity component along the x-direction, $u_c \approx \langle u \rangle_x(y)$, and the spanwise vorticity field is estimated as the local mean spanwise vorticity along the x-direction, $\omega_z(x, y, t) \approx \langle \omega_z \rangle_x(y, t)$. Moreover, for each PIV map, the second order derivatives of u are computed using a least squares differentiation scheme and then approximated

with their local mean value along the x-direction, $\partial^2 u(x,y,t)/\partial x^2 \approx \langle \partial^2 u/\partial x^2 \rangle_x(y,t)$ and $\partial^2 u(x,y,t)/\partial y^2 \approx \langle \partial^2 u/\partial y^2 \rangle_x(y,t)$. All the above vectors, representing each instantaneous velocity map, are then integrated over the y-direction to yield the vorticity flux $\zeta(t)$, as presented in Eq. (8), and the cumulative circulatory lift coefficient in the wake (see Eq. (9)).

In the option entitled **Panda & Zaman (1994), based on generated wake (threshold applied)**, the vorticity flux $\zeta(t)$ defined by Eq. (8) is estimated from the **complete wake image reconstructed by the user** behind the source; after applying a threshold on the vorticity contours. For the complete reconstructed wake image, the convection velocity is approximated as the freestream velocity, $u_c \approx U_\infty$. The spanwise vorticity field array is taken directly from the complete reconstructed wake (after the threshold selected by the user is applied), $\omega_z(x,y)$, where x-direction and time t are interchangeable by applying $\mathrm{d}t = U_\infty \mathrm{d}x$. The second order derivatives of u are computed using a least squares differentiation scheme directly from the complete reconstructed wake, $\partial^2 u(x,y)/\partial x^2$ and $\partial^2 u(x,y)/\partial y^2$, where x-direction and time t are interchangeable by applying $\mathrm{d}t = U_\infty \mathrm{d}x$. All the above arrays, representing the full reconstructed wake map, are then integrated over the y-direction (at each x location) to yield the vorticity flux $\zeta(t)$, as presented in Eq. (8), and the cumulative circulatory lift coefficient in the wake (see Eq. (9)).

The option entitled **Panda & Zaman (1994), based on generated wake** is identical to the former one and based on the complete wake image reconstructed by the user, but without applying the threshold on the processed vorticity field.

In the option entitled **Cumulative sum** the cumulative circulatory lift coefficient is estimated from the **individual PIV maps** accordingly:

$$\Delta C_{l_{circ}}(t) = \frac{2}{cU_\infty} \int_0^t \gamma(t)\mathrm{d}t = \frac{2\Gamma(t)}{cU_\infty} \tag{10}$$

where $\gamma(t)$ is the instantaneous circulation value calculated for each individual PIV map, as follows:

$$\gamma(t) = \int_0^l \int_0^h \omega_z(x,y,t)\mathrm{d}x\mathrm{d}y \tag{11}$$

h and l are the vertical and horizontal extent of the PIV velocity field, respectively.

References

[1] Hadar Ben-Gida, Adam Jonathon Kirchhefer, Zachary J. Taylor, Wayne Bezner-Kerr, Christopher G. Guglielmo, Gregory A. Kopp, and Roi Gurka. Estimation of unsteady aerodynamics in the wake of a freely flying european starling (*Sturnus vulgaris*). *Plos One*, 8(11):e80086, 2013.

[2] Oksana Stalnov, Hadar Ben-Gida, Adam J. Kirchhefer, Christopher G. Guglielmo, Gregory A. Kopp, Alexander Liberzon, and Roi Gurka. On the estimation of time dependent Lift of a european Starling (*Sturnus vulgaris*) during flapping flight. *Plos One*, 10(9), 2015.

[3] Asif Nafi, Hadar Ben-Gida, Christopher G. Guglielmo, and Roi Gurka. Aerodynamic forces acting on birds during flight: A comparative study of a shorebird, songbird and a strigiform. *Experimental Thermal and Fluid Science*, 113:110018, 2020.

[4] J. Panda and K. B. M. Q. Zaman. Experimental investigation of the flow field of an oscillating airfoil and estimation of lift from wake surveys. *Journal of Fluid Mechanics*, 265:65–95, 1994.

[5] Roi Gurka, Krishnamoorthy Krishnan, Hadar Ben-Gida, Adam J. Kirchhefer, Gregory A. Kopp, and Christopher G. Guglielmo. Flow pattern similarities in the near wake of three bird species suggest a common role for unsteady aerodynamic effects in lift generation. *Interface Focus*, 7(20160090), 2017.

[6] Jonathan Lawley, Hadar Ben-Gida, Krishnan Krishnamoorthy, Erin E. Hackett, Gregory A. Kopp, Gareth Morgan, Christopher G. Guglielmo, and Roi Gurka. Flow Features of the Near Wake of the Australian Boobook Owl (Ninox boobook) During Flapping Flight Suggest an Aerodynamic Mechanism of Sound Suppression for Stealthy Flight. *Integrative Organismal Biology*, 1(1), 2019.

[7] G. I. Taylor. The spectrum of turbulence. *Proc. R. Soc. Lond. A*, 164:476–490, 1938.

[8] K. B. M. Q. Zaman and Hussain A. K. M. F. Taylor hypothesis and large-scale coherent structures. *Journal of Fluid Mechanics*, 112:379–396, 1981.

[9] Hadar Ben-Gida, Oksana Stalnov, Christopher G. Guglielmo, Gregory A. Kopp, and Roi Gurka. Unsteady aerodynamics loads during flapping flight of birds ; case study : Starling and Sandpiper. In 12^{th} *International Conference on Heat Transfer, Fluid Mechanics and Thermodynamics*, pages 121–127, 2016.

[10] T. Theodorsen. General theory of aerodynamic instability and the mechanism of flutter. *NACA TR-496*, pages 291–311, 1935.

[11] W. R. Sears and T. von Kármán. Airfoil theory for non-uniform motion. *Aeronaut. Sci.*, 5(10), 1938.

[12] J. C. Wu. Theory for aerodynamic force and moment in viscous flows. *AIAA Journal*, 19(4):432–441, 1981.