



Discovering optimal flapping wing kinematics using active deep learning

Baptiste Corban^{1,†}, Michael Bauerheim¹ and Thierry Jardin¹

¹ISAE-SUPAERO, Université de Toulouse, 31055 Toulouse CEDEX 4, France

(Received 23 March 2023; revised 2 October 2023; accepted 3 October 2023)

This paper focuses on the discovery of optimal flapping wing kinematics using a deep learning surrogate model for unsteady aerodynamics and multi-objective optimisation. First, a surrogate model of the unsteady forces experienced by a 3-D flapping wing is built, based on deep neural networks. The model is trained on a dataset of randomly generated kinematics simulated using direct numerical simulation (DNS). Once trained, the neural networks can quickly predict the unsteady lift and torques experienced by the wing, using sparse information on the kinematics. This fast surrogate model allows multi-objective optimisation to be performed. The resulting Pareto front consists of new kinematics that may be very different from the kinematics of the initial dataset. A few arbitrarily chosen kinematics on the Pareto front are thus simulated using DNS and used to enhance the database. The new dataset is used to train again the networks, and this active deep learning/optimisation framework is performed until convergence, obtained after only two iterations. Overall, this method reduced the cost of optimisation by 83 %. Results reveal two distinct families of motions. Kinematics promoting high efficiency are characterised by large stroke amplitudes and relatively low angles of attack, as observed for fruit flies, honeybees or hawkmoths. For those, lift production is driven by quasi-steady effects and the formation of a stable leading edge vortex. Kinematics promoting high lift are characterised by small stroke amplitudes and high angles of attack, reminiscent of mosquitoes. Lift production is driven by the rapid generation of vorticity at the trailing edge.

Key words: machine learning, vortex dynamics, swimming/flying

1. Introduction

Flapping wings have attracted significant attention from both zoologists and aerospace engineers for the past three or four decades (Shyy *et al.* 2010). While the former seek,

† Email address for correspondence: baptiste.corban@gmail.com

in the broad sense, to understand the evolution of natural species, the latter draw their inspiration from nature to develop flapping wing robots.

In both cases, a detailed understanding of the physical mechanisms responsible for the fascinating flight performance of species like dragonflies and hummingbirds is required. In that regard, extensive experimental (e.g. Ellington 1984; Dickinson & Götz 1993; Dickinson, Lehmann & Sane 1999; Wang, Birch & Dickinson 2004; Lentink & Dickinson 2009) and numerical (e.g. Sun & Tang 2002; Wang *et al.* 2004; Wu & Sun 2004; Aono, Liang & Liu 2008; Bos *et al.* 2008; Jardin, Farcy & David 2012; Bhat *et al.* 2020) works have helped identify the prominent mechanisms at play, among which the leading edge vortex (LEV), wake capture, as well as rotational and added-mass effects are probably the most widespread phenomena. There exist other mechanisms pertaining to specific species, such as the clap-and-fling mechanism (Lighthill 1973) observed in butterflies. These mechanisms, which have been comparatively less studied, can however be considered as subsets of the more general mechanisms described above. Overall, these past studies have shown that the aerodynamics of flapping wings is characterised by the complex interaction between those key mechanisms, which renders the resulting physics highly unsteady and nonlinear, and emphasises its dependence on flapping wing kinematics.

A direct outcome is that purely physics-based reduced-order models aimed at predicting the aerodynamic loads experienced by a flapping wing, with relatively low cost, may fail under certain circumstances, for example, when wake capture effects (i.e. with strong unsteadiness and nonlinearity) are important. However, together with the significant increase in computational resources, recent models have relied on data-driven approaches where semi-empirical physics-based models are fitted to data from high-fidelity numerical simulations (e.g. Nakata, Liu & Bomphrey 2015; Cai *et al.* 2021; Van Veen *et al.* 2022). Such models have demonstrated reasonable accuracy over a relatively large range of flapping wing kinematics. This is of interest for the development of autonomous or semi-autonomous real-time flight controllers and can help improve the overall flight stability of a flapping wing robot. Moreover, it is suited to the optimisation of flapping wing kinematics, although it is not clear how the models developed in the works mentioned above perform in this context.

On the other side of the spectrum are purely data-driven models, which have benefited from the extensive research in machine learning over the past few years. In particular, neural networks have attracted the attention of the fluid community, since they are able to provide predictions at low cost but with high accuracy. For time-series predictions, the physics community first relied on the specific networks developed for the language processing applications, such as the long short-term memory (LSTM) networks introduced by Hochreiter & Schmidhuber (1997). For instance, LSTM networks have been used to propagate acoustic waves (Sorteberg *et al.* 2018) and predict the unsteady forces and wake behind a sphere at various Reynolds numbers (Gupta & Jaiman 2022). However, recurrent neural networks, such as LSTM networks, can be expensive to train and subject to vanishing gradient problems, due to long gradient chains. Moreover, predicting physics quantities usually needs only a short time history. As an alternative to recurrent networks, classical multi-layer perceptrons (MLPs) and convolutional neural networks (CNNs) have been adapted for unsteady predictions. To do so, a short history is used as input (autoregressive model), and the network learns the shift operator to predict the next time step. The complete time sequence is then computed by calling successively the shift operator while updating the input history. Alguacil *et al.* (2021, 2022) have used such techniques with a CNN to propagate acoustic waves in both space and time, showing high accuracy and generalisation capabilities compared with the LSTM methods

proposed by Sorteberg *et al.* (2018). Colombo, Bauerheim & Morlier (2023) developed a similar strategy based on a graph network used as a shift operator to estimate the unsteady aerodynamic forces and structural loading of a high-altitude long-endurance (HALE) concept aircraft subject to incoming flow perturbations. While such an approach has shown promising results, difficulties still exist, especially for long-time predictions. To tackle this issue while avoiding the complexity of recurrent networks, the learning of shift operators with short input history has been improved using long-term losses (LTLs), first introduced by Tompson *et al.* (2017) and further exploited by Ajuria-Illaramendi, Bauerheim & Cuenot (2022) and Colombo *et al.* (2023). The main idea is to collect during training new data by advancing in time the current network, and compare the predictions with the database. It works as a data augmentation technique, without backpropagation in time, where the network learns from its temporal mistakes. Recent works have shown a significant improvement of the quality of long-term predictions using LTL with a much lower training cost compared with recurrent networks. Ajuria-Illaramendi *et al.* (2022) also revealed that using LTL improves the robustness and generalisation capabilities of the temporal predictions made by neural networks for operating conditions far from those used during training. Here, we chose to use simpler MLPs with a time delay in the input, as they are less expensive and easier to implement and train than LSTM networks or RNNs. Some of the more complex solutions mentioned above could be useful if simple MLPs failed to provide sufficient precision, but were not needed in this study.

Note that this generalisation capability is usually a critical issue for neural networks, in particular when they are used for optimisation. Indeed, it would be unlikely that the training samples generated randomly are close to the optimal solutions. Therefore, the network is trained only on sub-optimal data: its use for optimisation necessarily suggests that the network has to generalise, and therefore no guarantee on the accuracy of these predictions can be made. Nevertheless, for low-dimensional spaces where dense sampling can be achieved, standard training strategies can be sufficient. For instance, Krügener *et al.* (2022) trained an MLP from Reynolds-averaged Navier–Stokes simulations to optimise the geometry of the combustion chamber of a rocket engine on a 10-dimensional space, showing an accurate Pareto front for this configuration. Similarly, Baqué *et al.* (2018) trained a geodesic convolutional neural network (GCNN) to minimise the drag of 3-D objects such as a sphere and a car. For more details on the various works employing machine learning for optimisation, one may refer to Li, Du & Martins (2022). However, to optimise flapping wings kinematics, the challenge associated with the generalisation of the neural network is critical, and will be tackled by active learning (AL), also referred to as optimal experimental design. It allows the training dataset to be built iteratively, by sampling new data based on the estimated optimal solutions. This approach is similar to Bayesian optimisation, where an acquisition function (also referred to as the infill sampling criterion) is used to select the new query points. Numerous works have used this technique using Gaussian processes (GPs), which was found effective when coupled to high-fidelity computational fluid dynamics (CFD) simulations, for which query points are expensive to compute (Roy *et al.* 2018; Campet *et al.* 2020). Active learning has been applied for optimisation in various scientific fields, from biology (Pandi *et al.* 2022) to materials science (Wang *et al.* 2022), including fluid mechanics to optimise turbine shapes (Wang *et al.* 2022). In the present work, active learning will be used by combining the MLP trainings with DNS to iteratively converge towards a Pareto front.

Consequently, the main objective of this paper is to train a deep neural network surrogate model on a dataset of high-fidelity numerical simulations through active learning towards the optimisation of three-dimensional flapping wing kinematics. The surrogate model

is used in place of numerical simulations, which are too computationally expensive, to provide inputs to a multi-objective evolutionary algorithm. In this particular context, a direct optimisation without active learning has recently been applied by Gehrke & Mulleners (2021) using experimental measurements as inputs. The authors performed a multi-objective optimisation of a hovering rectangular wing with parameters defining the pitch motion as variables. They showed that maximum efficiency was reached for nearly sinusoidal pitching motion with minimum angle of attack of approximately 30° while maximum lift was reached for trapezoidal motion with a minimum angle of attack of approximately 45° . In all cases, the flow dynamics and resulting lift were governed by the development of an LEV whose circulation was found to scale with the root-mean-square of the leading edge shear layer velocity. Furthermore, the maximum in LEV circulation was always reached once the leading edge had travelled 3.9 chords, i.e. before the end of the stroke. The evidence of non purely sinusoidal optimal pitching motions was also provided by Lee & Lua (2018) using a multi-fidelity approach, similar in spirit to that previously introduced by Zheng, Hedrick & Mittal (2013). The authors performed a two-stage optimisation where the first stage consisted of testing all possible combinations using a quasi-steady model and the second stage consisted of local optimisation using the simplex method and Navier–Stokes simulations. Again, optimality was closely correlated to the dynamics of the LEV. It is also informative to mention the recent work by Zheng *et al.* (2020) who performed the optimisation of three-dimensional hovering flapping wings using a self-adapted quasi-steady model. In this framework, a quasi-steady model was empirically fitted on data from Navier–Stokes simulations and used for the optimisation. The latter consisted of successive iterations where optimal solutions were computed using Navier–Stokes simulations and used to refine the empirical fit. The authors only considered purely sinusoidal motions, but investigated the role of elevation and pitching and flapping motions on aerodynamic performance. The results consistently pointed towards the role of LEV on optimal performance. Other data-driven models for flapping wings used GPs (Calado *et al.* 2023) or probabilistic recurrent state-space models (Bayiz & Cheng 2021), for example.

These optimisations are generally restricted to a subset of the parameter space, to reduce the associated time and costs. A common limitation in the optimisations mentioned above is the range of flapping amplitudes, which typically spans 60° to 180° , or the phase shift between flapping and pitching motions is constrained. Here, we use a deep neural network (NN) in conjunction with a genetic algorithm (GA) to identify optimal flapping wing kinematics over a large range of angles of attack and flapping amplitudes. We show that the NN-GA framework is able to identify optimal kinematics that lie well beyond (i.e. in the lift versus efficiency map) the dataset used for NN training. Furthermore, we identify distinct flow physics associated with high-efficiency motions on the one hand and high-lift motions on the other hand. High-efficiency motions are characterised by large flapping amplitudes and predominantly rely on leading edge vortex dynamics. Interestingly, optimal kinematics and underlying flow physics are similar to those observed in real species like fruit flies, bumblebees and hawkmoths, for example. Conversely, high-lift motions are characterised by low flapping amplitudes and predominantly rely on trailing edge vortex dynamics. This is reminiscent of mosquito wings kinematics, albeit at even lower flapping amplitudes, and shares similarities with the flow physics recently identified by Bomphrey *et al.* (2017) and Liu, Du & Sun (2020) on such species.

The remainder of the paper is structured as follows. In § 2, the wing and kinematics parameters are described, as well as the numerical set-up. Then in § 3, the active deep learning model is built and used for multi-objective optimisation. The Pareto front

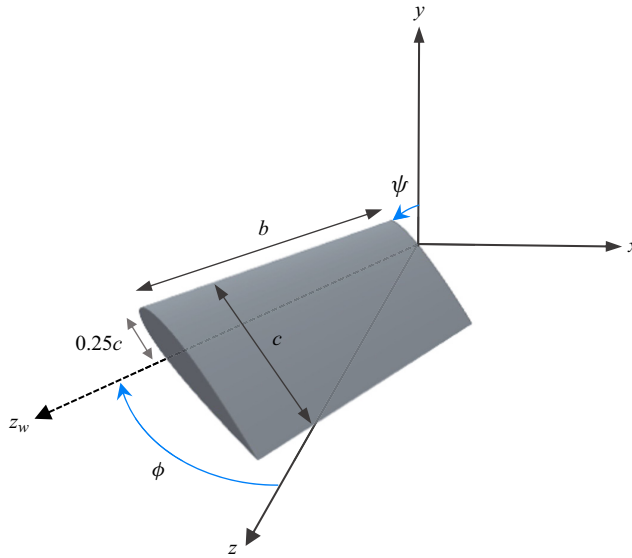


Figure 1. Definition of stroke and pitch angles, respectively ϕ and ψ , used to describe the wing motion.

and optimal kinematics obtained are then closely analysed in § 4 to highlight the key mechanisms at stake.

2. Wing model and numerical set-up

2.1. Wing model and kinematics

A rectangular wing model is commonly used to study flapping wing kinematics (Shyy *et al.* 2010; Diaz-Arriba *et al.* 2021; Gehrke & Mulleners 2021): even though insect wings have more complex shapes, a rectangular planform is a relevant simplified model allowing the accurate reproduction of the main physical phenomena at stake (namely the leading and trailing edge vortex dynamics). Thus, the wing model considered in this study is rectangular with a NACA0012 profile, where $c = 0.01$ m is the chord length and $b = 0.04$ m is the wingspan.

Two angles are used to describe the kinematics: the stroke, or revolution, angle ϕ and the pitch, or rotation, angle ψ . The elevation, or flap, angle is kept at zero as in actual insect motions; it has a low amplitude compared with the other two angles (Shyy *et al.* 2010). Figure 1 shows the angles' definition. The pitch axis z_w is attached to the wing, at quarter-chord, and stays in the xz plane throughout the motion.

The Reynolds number Re describes the ratio between inertial and viscous forces. It is defined as

$$Re = \frac{\rho \bar{U}_\phi c}{\mu}, \quad (2.1)$$

where $\bar{U}_\phi = 4bf\phi_0$ is the mean velocity at the tip of the wing during a stroke, induced only by the revolving motion of amplitude $2\phi_0$. In all cases throughout the paper, the Reynolds number is fixed to 1000, so that the flapping frequency f can be deduced as

$$f = \frac{\mu Re}{4\rho cb\phi_0}. \quad (2.2)$$

This work is restricted to periodic motions of frequency f , where kinematics (ψ, ϕ) are parametrised by a few Fourier coefficients (typically the f and $3f$ components), such as

$$\psi(t) = a_1 \cos(2\pi ft) + b_1 \sin(2\pi ft) + b_3 \sin(6\pi ft), \quad (2.3)$$

$$\phi(t) = \phi_0 \sin(2\pi ft - \delta), \quad (2.4)$$

where $\delta/2\pi f$ corresponds to the time delay between the revolution and rotation motions. The stroke motion amplitude is thus defined as $2\phi_0$. Note that such a parametrisation restricts the possible kinematic candidates for optimisation, yet the present definition allows for satisfying approximations of real insect flapping kinematics, further described in § 2.2. Additionally, the deep active learning framework proposed in § 3 can be extended to any other kinematic parametrisation if needed.

This study focuses on the hovering regime, where the motion objective is to produce a positive time-average lift, denoted \bar{L} . To achieve this lift performance, the system requires a rotation (respectively revolution) power $P_{rot} = T_{rot}\dot{\psi}$ (respectively $P_{rev} = T_{rev}\dot{\phi}$), T_{rot} (respectively T_{rev}) being the rotation (respectively revolution) torque applied to the wing root. As a consequence, the efficiency η can be defined as the ratio between the mean lift coefficient (\bar{C}_L) generated and the mean total power coefficient (\bar{C}_P), determined as

$$\bar{C}_L = \frac{\bar{L}}{\frac{1}{2}\rho bc\bar{U}^2}, \quad (2.5)$$

$$\bar{C}_P = \frac{P_{rot} + P_{rev}}{\frac{1}{2}\rho bc\bar{U}^3}, \quad (2.6)$$

$$\eta = \frac{\bar{C}_L}{\bar{C}_P}. \quad (2.7)$$

It is usual to use \bar{U}_ϕ as the normalisation velocity (Shyy *et al.* 2010; Gehrke & Mulleners 2021). Yet, the stroke amplitude is a degree of freedom in this study, and for kinematics with low amplitude, the stroke-induced velocity does not dominate the pitch-induced velocity. If the latter is not taken into account, a pure pitch motion would have an infinite lift coefficient, and the reference speed would not be adequate since $\bar{U}_\phi = 0$ in that case. Even though the lower bound of the study is $\phi_0 = 10^\circ$, for such kinematics, the relevant speed governing the unsteady aerodynamics is no longer \bar{U}_ϕ . As a consequence, another normalisation velocity \bar{U} is thus introduced, different from that used for the Reynolds number calculation: it is defined as the time-averaged leading-edge velocity at the tip of the wing, induced by both the stroke and pitch motions:

$$\bar{U} = \overline{\|\mathbf{u}_\phi + \mathbf{u}_\psi\|} = \frac{1}{T} \int_0^T \left[\left(\underbrace{b\dot{\phi}(t)}_{\mathbf{u}_\phi} + \frac{\xi}{2}\dot{\psi}(t) \cos(\psi(t)) \right)^2 + \left(\frac{\xi}{2}\dot{\psi}(t) \sin(\psi(t)) \right)^2 \right]^{1/2} dt. \quad (2.8)$$

Note that \mathbf{u}_ϕ is contained in the stroke plane, while \mathbf{u}_ψ is perpendicular to the wing chord, thus the normalisation velocity \bar{U} is not parallel to the stroke plane.

2.2. Kinematics generation

Real hovering kinematics from the hawkmoth, fruit fly and honeybee, described by Liu & Aono (2009), are used as a basis to generate the dataset used for neural network training.

Discovering optimal flapping wing kinematics

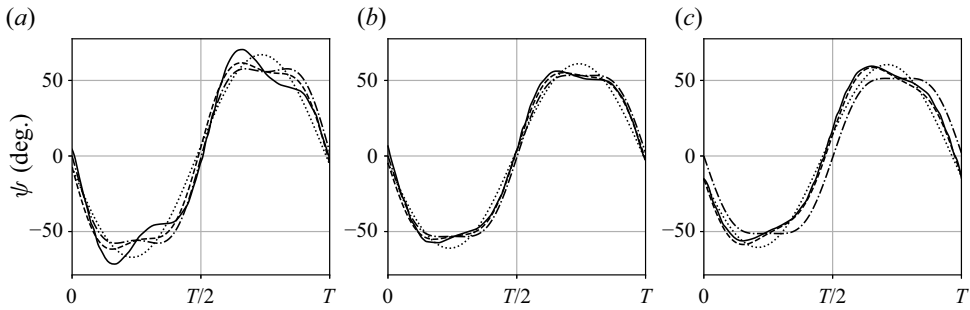


Figure 2. Reference pitch ψ -kinematics (solid line) for (a) the fruit fly, (b) the honeybee and (c) the hawkmoth, as well as their approximation using (a_1, b_1) coefficients (dotted line), (b_1, b_3) (dash-dotted line) and (a_1, b_1, b_3) coefficients (dashed line) (see (2.3)).

a_1	b_1	b_3	ϕ_0	δ
$[-0.5, 0.2]$	$[-1.8, -0.1]$	$[-0.3, 0]$	$[10, 60]^\circ$	$[0, 180]^\circ$

Table 1. Parameter bounds for the five parameters of the kinematics described by (2.3) and (2.4), used to generate randomly the 294 motions composing the initial dataset.

Figure 2 shows how their pitching motions are approximated using two or three Fourier coefficients. Using at least three Fourier coefficients is necessary to capture the reference pitching kinematics with reasonable accuracy, justifying the chosen pitching law in (2.3). Their stroke motion is close to sinusoidal, and their corresponding amplitude $2\phi_0$ is 120° for the hawkmoth, 90° for the fruit fly and 140° for the honeybee. The phase shift between revolving and pitching motions is 90° for all three cases.

To provide a large variety of data for the neural network training, 294 motions are initially created, of which 144 motions have a purely sinusoidal pitch profile (i.e. $a_1 = b_3 = 0$) and 150 motions are composed using the three Fourier coefficients. For each motion, a uniform law between the bounds shown in table 1 is used to randomly generate each parameter (a_1, b_1, b_3, ϕ_0 and δ). The bounds were chosen to contain the parameters of the three bio-inspired reference kinematics for a_1, b_1, b_3 and δ . For ϕ_0 , the higher bound is 60° . We believe this bound does not affect significantly the findings in this study. For amplitudes higher than 120° , the lift production mechanisms are similar to those described in § 4.2, but would need a mesh refined on a larger area to be simulated.

2.3. Computational set-up

DNS of the generated flapping motions is performed, and will later be used as the data for neural network training. The CFDs solver used to compute the flow around the flapping wing is StarCCM+. The three-dimensional Navier–Stokes equations are directly solved using a cell-centred finite volume method. Second-order numerical schemes are employed for spatial and temporal discretisations. The computational domain is a circular cylinder, $40c$ high and $30c$ wide, shown in figure 3. The wing moves parallel to the top and bottom ends of the far field. It is enclosed within a $2c$ -wide moving overset mesh. The displacement of the moving mesh is contained in a fixed control volume, in which the mesh is refined to resolve the wake with sufficient accuracy. The wing is modelled by a

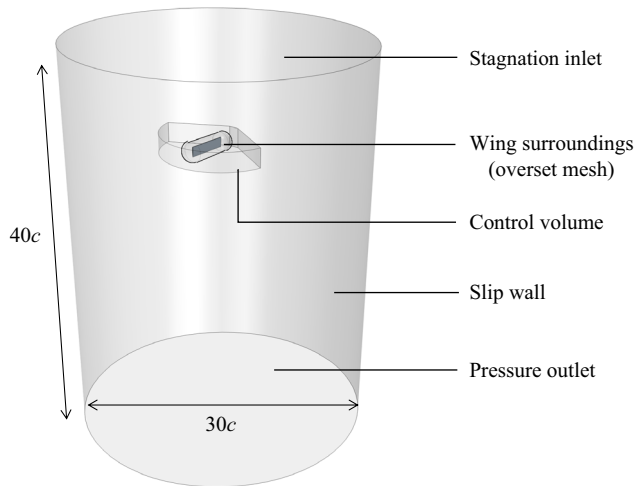


Figure 3. Computational fluid domain with details on the refinement zones and the boundary conditions employed.

non-slip wall, the top of the domain is a stagnation inlet boundary condition, the bottom end is a pressure outlet and the lateral boundary conditions are treated as a slip wall.

The spatial resolution chosen is $\Delta x = 0.02c$ on the wing surface and $\Delta x = 0.04c$ in the control volume, which yields a total number of cells of 1.9 million. The time resolution chosen is 250 time steps per period, where the period $T = 1/f$ corresponds to an upstroke followed by a downstroke motion. The numerical set-up is similar to that used by Diaz-Arriba *et al.* (2021), where it was shown that the results were converged with respect to both spatial and temporal resolutions. Four periods are simulated, and only the latest two are used in the dataset to avoid any transient effects in the data. Additional information on the decay of initial transients and convergence with respect to spatial and temporal resolutions is provided in the [Appendix](#).

For flow analysis on optimal cases predicted by neural networks (in § 4), a finer mesh will be used. The spatial discretisation for these simulations is $\Delta x = 0.005c$ on the wing surface and $\Delta x = 0.01c$ in the control volume, leading to a total number of 36 million cells. The time discretisation is identical to the coarser mesh, i.e. set to $\Delta t = T/250$.

3. Optimisation of flapping wing motions using neural networks

3.1. Unsteady aerodynamic model using deep learning

Lift, rotation torque and revolution torque are needed to compute η and \overline{C}_L , and assess a motion performance. Thus, three neural networks are trained to predict those three outputs. The neural networks are feed-forward neural networks (FNNs), coded and trained using Pytorch (Paszke *et al.* 2019). The FNN employed here is an MLP, composed of successive layers of neurons. The output layer size of each neural network is 1, meaning each neural network will be trained to predict a scalar output at an instant t .

3.1.1. Neural network input

Multiple architectures exist to predict temporal data, such as recurrent neural network (RNN), LSTM or echo-state network (ESN) among many others. While being effective for speech recognition and other tasks involving temporal sequences, their use in

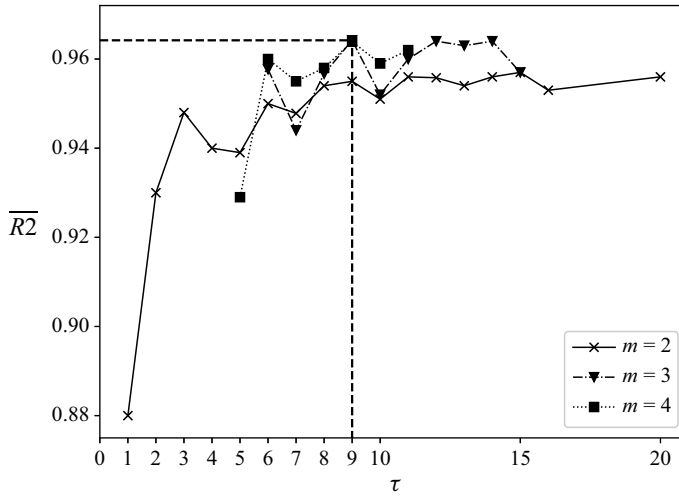


Figure 4. Optimal m and τ research, evaluated by the mean $R2$ score on the test set for lift prediction. The final choice $m = 3$ and $\tau = 9$ is indicated by dashed lines.

computational physics is still sparse, mostly because only a short time history is needed to predict the further time steps. A typical example is CFD, where time-marching is achieved using time integration with very few previous time samples. As a consequence, a simpler MLP is chosen here where the input contains a few samples at previous time steps. To make a prediction for an output at time t , each neural network takes as input a vector $\mathbf{X} = [\mathbf{x}(t), \mathbf{x}(t - \tau dt), \dots, \mathbf{x}(t - (m - 1)\tau dt)]$, with \mathbf{x} a state vector, m and τ two integers chosen in the following, and $dt = 1/250f$. Such a method has been shown sufficient to replicate CFD results from high-order numerical codes, as for instance by Alguacil *et al.* (2021, 2022) for acoustic waves propagation.

The state vector used is $\mathbf{x} = [\psi, \dot{\psi}, \ddot{\psi}, \phi]$ to provide information about the pitch and stroke kinematics to the neural network. Its components were chosen by testing different combinations. It appears that adding additional derivatives of the pitch angle improves the prediction performance, while for the revolving angle, only the first derivative is sufficient, as it follows a simpler sinusoidal law. Here, m and τ are determined by performing a partial gridsearch. Increasing m causes an increase in neural network size, as the input is larger and more neurons are needed to process it. It affects the training duration and prediction times once trained, thus m should be chosen as low as possible. Figure 4 shows the m and τ effect on the neural network performance, evaluated by the mean $R2$ score for lift prediction over a test set of 45 motions. For each motion in the test dataset, the $R2$ score is evaluated on both flapping periods (i.e. from $t = 0$ to $t = 2T$) and defined as

$$R2 = 1 - \frac{\sum_{t=0}^{2T} (y(t) - \hat{y}(t))^2}{\sum_{t=0}^{2T} (y(t) - \bar{y})^2}, \quad (3.1)$$

with \bar{y} the average value of the output y and \hat{y} the prediction from the neural network.

The chosen values are $m = 3$ and $\tau = 9$. As a consequence, for a prediction at the time step t , data are needed from $18 dt = 0.072T$ earlier. This is in line with the results of

Bayiz & Cheng (2021): they demonstrated that aerodynamic states depended on the wing motion in less than half a cycle, thanks to cross-correlation analysis of the latent states of their model and kinematic variables. Here, $0.072T$ is slightly lower than the delay they found for maximal correlation, often approximately $0.1-0.2T$. The input vector is therefore $X = [x(t), x(t - 9 dt), x(t - 18 dt)]$. It is sampled from the motion data and then fed to the trained networks that produce the three needed outputs.

As detailed in § 2, a dataset of 297 simulated motions is created, containing the 294 randomly generated kinematics as well as the three reference ones from the fruit fly, the hawkmoth and the honeybee. This dataset of motions is divided in three parts. First, 70 % of the motions are used to train the neural network by updating the weights and biases in each neuron. Then, 15 % for validation to check the neural network performance during training on a set of data on which the weights are not updated. This is useful to prevent the overfitting (i.e. the memorisation rather than learning) of the training data. The last 15 % are used to test the final network on unseen data and compare the trained models. The three reference kinematics (figure 2) are placed in the testing dataset. Each motion, having been assigned to a dataset, is then split in input vectors X of length $4m = 12$.

3.1.2. Prediction performance

The architecture chosen is identical for the three feed-forward neural networks: three hidden layers, with respectively 512, 1024 and 512 neurons, which correspond to a total of approximately 1 million tunable parameters per network. Each neuron activation function is a ReLU. The optimiser used to train the weights is Adam, with a learning rate of 10^{-4} . The loss is the mean square error (MSE). Each neural network is trained for 50 epochs, with a batch size of 512.

Figure 5 shows the neural network prediction on the three outputs for the three reference motions. The data from the CFD simulation are also displayed. The neural network prediction generally fits the numerical data accurately.

The best predicted output is the revolution torque with a mean R^2 score over the test dataset of $\overline{R^2} = 0.985$, then the lift with $\overline{R^2} = 0.983$ and third the rotation torque with $\overline{R^2} = 0.948$. Overall, the neural networks show good performance in predicting unsteady aerodynamics for the three reference cases and over a wide range of motions as the $\overline{R^2}$ scores show. These predictions of the unsteady aerodynamics take the order of 1 s to run on an 8-core CPU. This makes the neural networks suitable to be used as a surrogate model for an optimisation of the kinematics, as each evaluation of performance will be inexpensive compared with a CFD simulation (approximately 200 core-hours needed for each simulation in the dataset).

3.2. Multi-objective optimisation and active learning

3.2.1. NSGA II algorithm

The surrogate model is used to perform a multi-objective optimisation of the flapping motion. To do so, the unsteady aerodynamics predicted by the networks, using the kinematics as inputs, is coupled with a genetic algorithm for optimisation. The objectives are the stroke-average lift coefficient $\overline{C_L}$ and efficiency η defined in (2.5)–(2.7). The NSGA II algorithm, described by Deb *et al.* (2002), is implemented using the pymoo library (Blank & Deb 2020). Genetic algorithms are based on the survival of the fittest principle: like in natural evolution, the fittest individuals live through the generations. The stochastic nature of the genetic algorithm makes them suitable in evading local optima, which are

Discovering optimal flapping wing kinematics

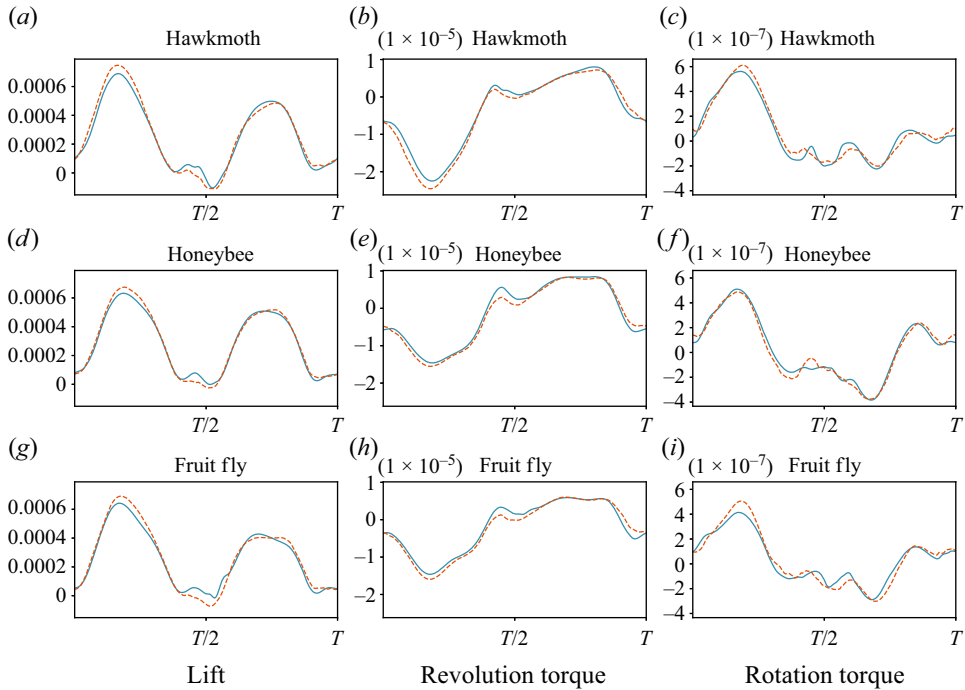


Figure 5. Neural network predictions on the reference motions: (a–c) hawkmoth, (d–f) honeybee and (g–i) fruit fly. Predictions are performed over a period $[0, T]$ for (a,d,g) the lift, (b,e,h) the revolution torque and (c,f,i) the rotation torque. The solid curve is the simulation results and the dashed line is the neural network prediction.

typically encountered in such non-convex optimisation problems. The NSGA II algorithm has proven to be efficient in predicting Pareto fronts in fluid mechanics, and has already been coupled to neural networks, as proposed by Krügener *et al.* (2022) to optimise the combustion chamber of a rocket engine. For each generation of the algorithm, a mating pool is formed by selecting individuals from the previous generation population. These individuals then go through mutation and cross-over to form offsprings. The offsprings and the previous population are gathered and the new population is created by non-dominated sorting, and finally by crowded distance sorting.

Each individual is composed of the five parameters $[a_1, b_1, b_3, \phi_0, \delta]$. The bounds of the search space are defined in table 1. The population is composed of 100 individuals. Fifty offspring are generated each generation. The algorithm usually converges after approximately 35 generations, using the convergence speed criterion based on the generational distance defined by Busch, Gehrke & Mulleners (2012), for a total of 1800 evaluations. This justifies the need for surrogate modelling, prior to the optimisation, since the 1800 evaluations cannot be achieved by CFD. In the present work, the cross-over method is the simulated binary cross-over, and the mutation method is the polynomial mutation (further details can be obtained from Blank & Deb 2020).

3.2.2. Active learning algorithm

Neural networks are known to lose accuracy when generalising outside the training dataset. Since the kinematics have been generated randomly, it is unlikely they correspond to

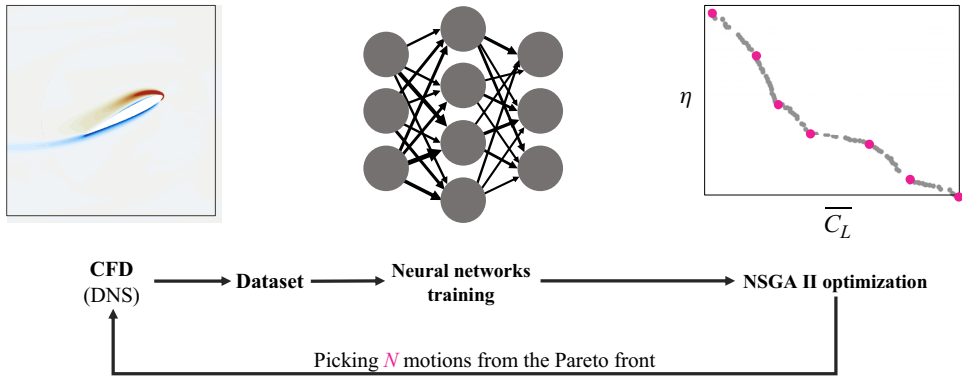


Figure 6. Complete framework used to discover optimal kinematics, involving the training of neural networks through an active learning approach to provide accurate predictions of the Pareto front by the NSGA II algorithm.

situations close to an optimum. As a consequence, the predicted Pareto front can be inaccurate because of the poor network generalisation capabilities. To evaluate this issue, CFD validations are performed on seven kinematics lying on the predicted Pareto front. The efficiency and lift coefficient obtained by CFD (∇) are compared with the network predictions (\circ) in figure 7 (step 1, top left). Results show that the global trend of the Pareto front is predicted correctly by the networks, but large local errors can be also observed, especially for cases with high lift coefficients (the differences between the CFD and network predictions are highlighted by the dotted lines in figure 7). In fact, no training data exist in this zone, the maximum lift coefficient of the training dataset is $\overline{C}_L \approx 0.9$, whereas the neural network predicts cases with coefficients up to $\overline{C}_L \approx 1.5$.

To improve the accuracy of neural networks, and thus of the multi-objective optimisation, an active learning approach is used. Lye *et al.* (2021) described an iterative process to improve partial differential equations optimisation using surrogate modelling. They tested it on an example of airfoil shape optimisation. One issue affecting the performance of deep learning surrogate models is that the training dataset is generated before the optimisation step, and the data points may be very different from the Pareto optimal points. The iterative surrogate model optimisation (ISMO) algorithm described by Lye *et al.* (2021) allows for an iterative construction of the dataset with data points from successive optimisation steps. This process was adapted here to the gradient-free multi-objective optimisation algorithm NSGA II, described in § 3.2.1.

In this study, an initial training dataset of 208 motions (70 % of the total dataset of 297 motions) was generated randomly to train the neural networks. The predictions from these first trained networks were detailed in § 3.1.2, and the validation of the Pareto front obtained with the NSGA II algorithm in figure 7 (step 1, top left). Now, $N = 7$ motions from the predicted Pareto front are arbitrarily selected, and numerical simulations are performed to check the accuracy of the surrogate model on kinematics unseen during the training. These new data are then added to the training dataset for another step of neural network training and multi-objective optimisation. This active learning approach is illustrated in figure 6.

Two iterations of the active learning algorithm were needed to obtain an accurate prediction of the Pareto front. The corresponding Pareto fronts are displayed in figure 7. The predicted Pareto front is compared with the DNS of seven selected individuals. Step 1 corresponds to the surrogate model trained on the initial dataset. As already mentioned,

Discovering optimal flapping wing kinematics

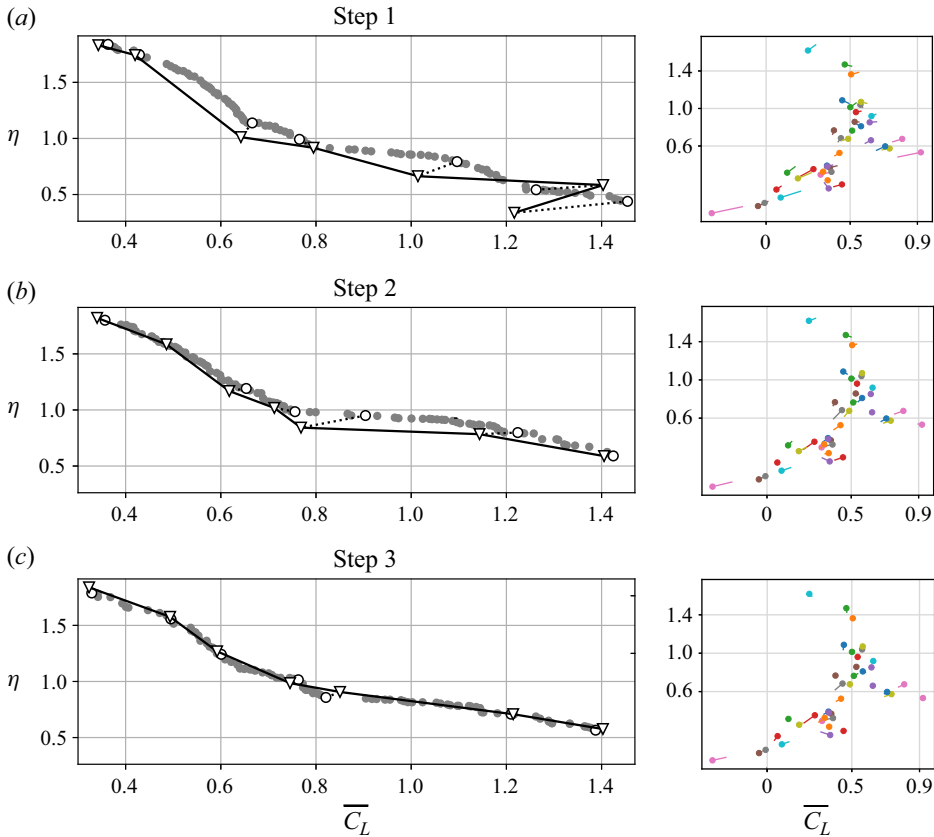


Figure 7. In the left column, Pareto fronts from the three steps of the active learning approach. The Pareto front predicted using the deep learning surrogate model (\circ) is compared with DNS (∇) on $N = 7$ individuals selected on the predicted Pareto front. The difference between network and DNS results is highlighted by the dotted lines. In the right column, the evolution of the prediction error of the model on the testing dataset in the (η , \overline{C}_L) plane. Dots (\bullet) are the DNS prediction and the lines point at the predicted position by the model.

it shows a good accuracy until $\overline{C}_L \approx 0.8$. For higher \overline{C}_L , the predicted Pareto front and DNS are further apart. Moreover, looking at the CFD simulations of the last two selected motions, which are linked to their initial predicted performance by a dotted line, the actual highest- \overline{C}_L motion is not actually the predicted one. The predicted Pareto front is not accurate in the high- \overline{C}_L area, leading to a wrong hierarchy between motions.

Figure 8 displays the DNS of the Pareto fronts and the initial dataset (+). The reference kinematics (hawkmoth, fruit fly and honeybee) used to create the dataset are shown (\star). They are located near the Pareto fronts in the high efficiency area. The initial dataset was randomly generated, as described in § 2.2, and has no samples with $\overline{C}_L > 0.9$. This explains why the neural network surrogate model is struggling with accuracy at Step 1. The model behaviour on its testing set in figure 7 (step 1, top right) already hinted at a lack of precision towards the high lift. On this plot, the dots (\bullet) are the DNS and a line links it to the predicted position. For $\overline{C}_L > 0.6$, errors are already clearly visible. Yet, it was able to predict high lift motions in an area where no data were available: even if errors are made, DNS shows that some optimal candidates have indeed a very high lift coefficient, up to $\overline{C}_L \approx 1.4$ when validated by DNS, while the network was predicting for this specific case

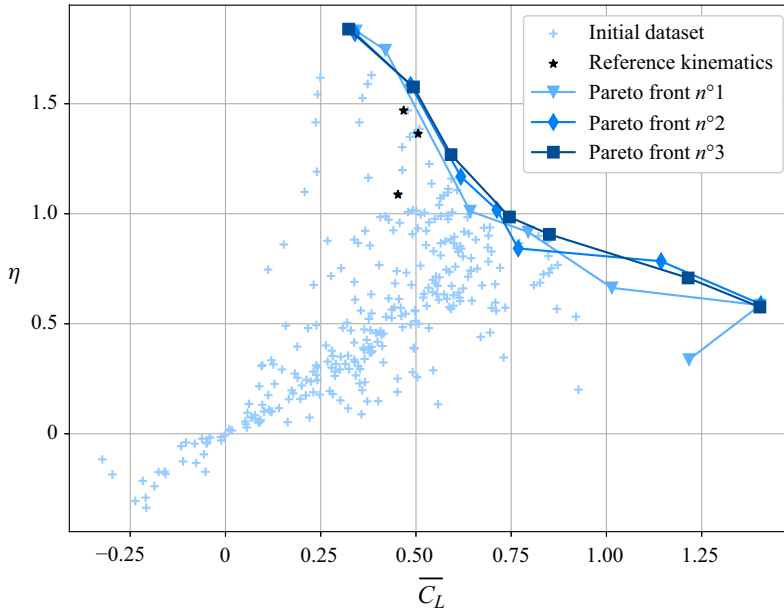


Figure 8. Comparison between the DNS of the seven selected motions from each Pareto front and the initial dataset (+). The reference motions used to generate the dataset are highlighted (★).

$\overline{C}_L \approx 1.2$. As a result, the highest lift obtained by DNS at Step 1 is showing an increase of 55 % in \overline{C}_L from the initial training dataset. At this point, it is thus remarkable that the deep learning model from Step 1 was able to discover new high lift kinematics, and predict their performance with a prediction error of 15 % on the lift coefficient. Similarly, on the other end of the Pareto front, the most efficient motions were improved by 12 %, yet more training samples lie in this zone, which results in less accuracy problems for the neural networks.

Following the active learning approach, these $N = 7$ new motions are added to the dataset for Step 2. The predicted Pareto front is now more accurate throughout the whole range (figure 7b). There is a small over-estimation of the efficiency and lift coefficients for most of the front. The best motions for each objective were not improved. Figure 8 shows that, except around $\overline{C}_L \simeq 0.75$, the motions in Step 2 dominate those from Step 1. Intermediate motions were thus improved, in addition to the accuracy of the model. A third step leads to the most accurate predicted Pareto front, shown in figure 7(c): the CFD simulations match the predictions throughout the whole Pareto front. There is also a local improvement compared to the previous steps: in the area $\overline{C}_L \in [0.55, 0.9]$, an increase in efficiency is observed compared with similar- \overline{C}_L motions from the first two steps, as visible in figure 8.

Overall, looking at the right column of figure 7, the accuracy over the whole testing set was improved by the two steps of active learning. The mean error (i.e. the mean size of the lines connecting DNS to neural net predictions on the plot) went from 0.081 for Step 1 to 0.056 for Step 3. The training dataset has been densified near the Pareto front by adding twice seven new motions, but it also benefited for the model performance over the whole range of motions.

The active deep learning process needed 314 CFD simulations of around 2 hours (running on 4 CPU nodes Intel Skylake 6126, 24 cores, 96Go RAM). As mentioned

Discovering optimal flapping wing kinematics

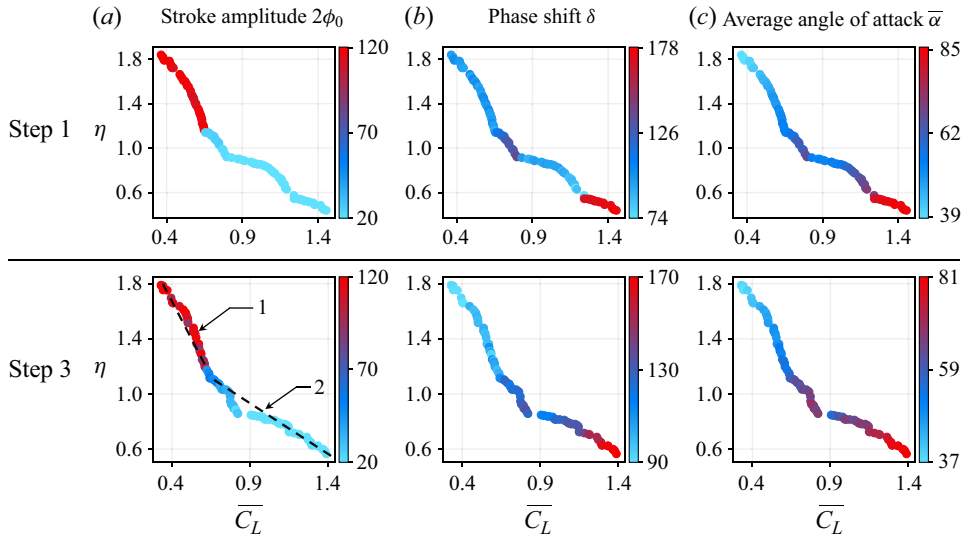


Figure 9. Colour-coded Pareto fronts highlighting the evolution of three parameters ϕ_0 , δ and $\bar{\alpha}$, for both the Step 1 and Step 3 Pareto fronts. Two regimes are identified (dashed lines) in the Step 3 Pareto front, one at high efficiency (1) and the other at high lift coefficient (2).

in § 3.2.1, the genetic algorithm converges after 1800 evaluations of the cost function. Using the surrogate model, each evaluation takes the order of a second. Thus, the total cost for the active learning process and optimisation is approximately 630 h. If we were to use CFD for the 1800 evaluations of the optimisation process, it would take approximately 3600 h. Our active deep learning model has thus allowed us to perform a two-objective optimisation for 17 % of the CFD cost.

Figure 9 allows to compare the evolution of three parameters and the differences of their distributions from Step 1 to Step 3: stroke amplitude $2\phi_0$ (a), the phase shift δ (b) and the stroke-average angle of attack $\bar{\alpha}$ (c). The latter is defined as the average angle during a stroke between the wing and the stroke plane xz . On the Step 1 Pareto front, there is a harsh change in stroke amplitude $2\phi_0$ around $\bar{C}_L = 0.5$ from 120° to 20° , whereas on the Step 3 front, there is a gradual decrease, passing by motions with an intermediate amplitude. For the phase shift, the decrease is monotonous when increasing the lift coefficient along the Step 3 front, while for Step 1 around $\bar{C}_L = 0.8$, there is a shift in trend. Similarly, the high values of phase shift around $\bar{C}_L = 1.2$ are attained more gradually in Step 3. The same remarks can be made for the average angle of attack. Overall, the evolution of those three quantities is smoother and monotonous on Step 3. The difference in trends in the Step 1 Pareto front is linked to its lack of precision, mixing the hierarchy between motions. The active learning process thus allowed a smoother transition between high-efficiency and high-lift motions to be found.

4. Analysis of the optimal kinematics

4.1. Pareto front analysis

To better understand what parameters drive these optimal kinematics, the evolutions of the stroke amplitude, phase shift and average angle of attack on the Step 3 Pareto front are observed in figure 9.

First, the stroke amplitude colouring in [figure 9\(a\)](#) highlights the difference between the motions at each end of the Pareto front. For high-efficiency motions, until $\overline{C}_L \simeq 0.6$ (regime 1), the stroke amplitude takes values between 90° and 120° . Then, there is a drop in amplitude to $\simeq 50^\circ$ and a gradual decrease reaching 20° from $\overline{C}_L = 0.9$ onwards (regime 2). This stroke amplitude difference between high-lift and high-efficiency motion hints at two distinct lift production mechanisms. The high-lift kinematics having a shorter stroke amplitude compared with the high-efficiency ones is in line with the results from Altshuler *et al.* (2005). They concluded that at constant wingtip velocity U_ϕ , decreasing stroke amplitude promoted lift while decreasing efficiency. The two regimes are highlighted by the dashed lines: (1) corresponds to high-efficiency kinematics and (2) corresponds to high-lift motions. They can be characterised by their slope in the (\overline{C}_L, η) plane. For regime 1 (i.e. from $\overline{C}_L = 0.32$ to 0.61) the efficiency decreases rapidly, with a slope $\Delta\eta/\Delta\overline{C}_L = -2.12$. This indicates that to gain a certain amount in lift coefficient, it costs roughly twice the amount in efficiency. For regime 2 (i.e. from $\overline{C}_L = 0.6$ to 1.4), the decline in efficiency is slower, with a typical slope $\Delta\eta/\Delta\overline{C}_L = -0.74$. A gain in lift coefficient is not as costly in efficiency. Motions from those two regimes will be discussed in § 4.2 for regime 1 and § 4.3 for regime 2. It is also to be noted that though the Reynolds number based on the revolution speed \overline{U}_ϕ is 1000 for all motions, the Reynolds number based on the force-normalisation velocity \overline{U} described in (2.8) varies between 400 (for high-efficiency cases) and 1300 (high-lift cases) along the Pareto front.

Second, the phase-shift colour-coded Pareto front ([figure 9b](#)) reveals a gradual evolution throughout the front. For efficient motions until $\overline{C}_L \simeq 0.65$, the phase shift is between 90° and 110° . That is, the wing angle passes through vertical slightly before stroke reversal. This kinematic property is commonly referred to as ‘advanced rotation’ (e.g. Sane & Dickinson 2001). Then for $\overline{C}_L \in [0.65, 1.2]$, δ varies between 110° and 135° . The highest lift motions ($\overline{C}_L > 1.2$) have higher values of phase shift, from 150° to 170° . Those high values of phase shift may seem counter-intuitive at first, and imply that the trailing edge of the wing is ahead of the leading edge during a large portion of the stroke. This relatively uncommon flapping regime will be further studied in § 4.3.

Third, the evolution of the stroke-average angle of attack $\overline{\alpha}$ on [figure 9\(c\)](#) shows an overall smooth increase from values of 37° for high- η motions to 81° for high-lift motions. This angle is influenced by the pitch kinematics and the phase-shift evolution. While values of 37° for high-efficiency motions are in line with previous results from Sane & Dickinson (2001), Gehrke & Mulleners (2021) and Diaz-Arriba *et al.* (2021), for example, values of 81° appear very large. This, again, suggests different mechanisms of force production between high-efficiency and high-lift motions. We note that the effective angle of attack of the wing is in fact much lower than $\overline{\alpha}$ due to the downwash induced during previous strokes (as a result of lift production).

4.2. High-efficiency kinematics

The most efficient flapping motion is studied here, as a typical example of kinematics of regime 1. Its parameters are $a_1 = 0.001$, $b_1 = -1.34$, $b_3 = -0.22$, $\phi_0 = 55^\circ$ and $\delta = 94^\circ$. This motion is characterised by a high stroke amplitude of 110° and an overall low angle of attack. The time evolution of the motion’s parameters are displayed in [figure 10](#), along with the lift and power coefficient evolution over a flapping period, obtained by DNS and the neural network model. The neural network error on the prediction is 2.8% for η and 1.8% for \overline{C}_L . It is interesting to note that this motion highly resembles the three reference

Discovering optimal flapping wing kinematics

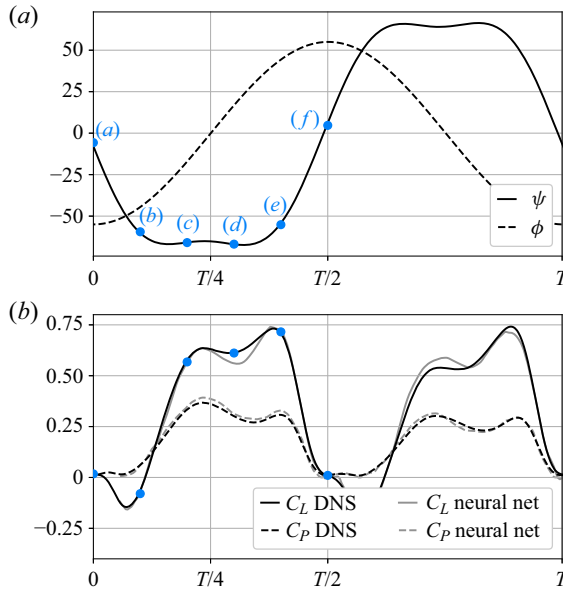


Figure 10. Time evolution of the (a) kinematics parameters, and (b) lift and power coefficients for the most efficient motion, obtained from DNS (black) and neural networks (grey). Key instants are located by blue dots.

kinematics observed in nature (figure 2). The pitch profile is trapezoidal: the pitch angle is almost constant at 65° during most of the stroke. This corresponds to an angle of attack of 35° , which was also observed in real hovering kinematics for small insects by Ellington (1984).

The lift coefficient evolution (figure 10) also has an almost trapezoidal shape during a single stroke ($t \in [0, T/2]$): a fast increase at the start of the stroke, then little variations in lift during the almost constant pitch angle phase, followed by a fast decrease at the end of the stroke. The lift seems mainly controlled by ϕ and ψ , which suggests that quasi-steady effects are dominant. Note that for this high-efficiency motion, the lift and the power coefficients evolution are in phase. During the part of the stroke when lift is produced, the power coefficient is half the value of the lift coefficient. This leads to the efficiency value $\eta = \overline{C_L}/\overline{C_P} = 1.79$.

To further analyse the dependency of lift to $\dot{\phi}$ and ψ , results are compared with predictions from the quasi-steady model by Lee *et al.* (2016), adapted to use our normalisation velocity \bar{U} defined in (2.8). The model includes effects from pure revolving motions, combined revolving and rotating motions, as well as from added mass. The total quasi-steady lift coefficient reads

$$C_L = C_{L,rev} + (C_{L,rot1} + C_{L,rot2}) \cos \alpha + C_{L,am} \cos \alpha, \quad (4.1)$$

where $C_{L,rev}$ is the quasi-steady lift force of a wing under pure revolving motion and is expressed as

$$C_{L,rev} = (f_{AR,tr})(f_{Ro,tr}) \frac{b^2 C_{L,tr}}{3\bar{U}^2} \dot{\phi}^2, \quad (4.2)$$

where $C_{L,tr}$ is the quasi-steady lift coefficient resulting from a translation motion and is equal to $1.966 - 3.94Re^{-0.429} \sin(2\alpha)$. Here, $f_{AR,tr}$ and $f_{Ro,tr}$ are correction factors associated with aspect ratio and Rossby number effects (see Lee *et al.* (2016) for further details).

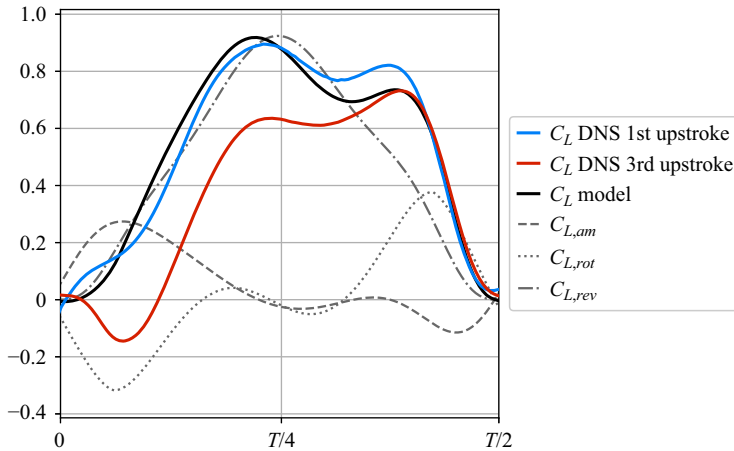


Figure 11. Quasi-steady lift model (black line) from Lee *et al.* (2016) and its different components (translational, rotational and added mass effects), compared with the DNS results during the the first (blue line) and third (red line) upstrokes for the most efficient motion.

Additionally, $C_{L,rot1}$ and $C_{L,rot2}$ are the quasi-steady lift forces of a wing undergoing combined rotating and revolving motion. They are derived as

$$C_{L,rot1} = (f_\alpha)(f_r)C_{rot} \frac{cb|\dot{\phi}|\dot{\alpha}}{\bar{U}^2} |\dot{\phi}|\dot{\alpha} \quad (4.3)$$

and

$$C_{L,rot2} = 0.778 \frac{c^2|\dot{\alpha}|\dot{\alpha}}{\bar{U}^2}, \quad (4.4)$$

where f_α and f_r are correction factors linked to the rotational axis position and the instantaneous angle of attack. Here, C_{rot} is equal to $0.842 - 0.507Re^{-0.1577}$.

Finally, the added mass coefficient is derived as

$$C_{L,am} = (f_a)(f_{AR,a}) \frac{\pi}{4\bar{U}^2} \left(cb\ddot{\phi} \sin \alpha + \frac{c^2}{2} \ddot{\alpha} \right), \quad (4.5)$$

with f_a and $f_{AR,a}$ correction factors empirically determined.

Figure 11 compares the lift predicted by the quasi-steady model with that obtained from the present simulations during the first and third flapping cycles. The lift coefficients obtained with the numerical simulation, during the first flapping cycle, and the quasi-steady model are found to match reasonably well. In particular, the lift generally increases up to $C_L \approx 0.9$ slightly before $T/4$ and then decreases before increasing again up to $C_L \approx 0.8$ around $3T/4$. The quasi-steady model suggests that the first increase is principally driven by quasi-steady effects from the revolving motion, and hence by kinematic parameters $\dot{\phi}$ and ψ . Added mass and rotational effects tend to shift the maximum lift to earlier times, when compared with that obtained from the sole contribution of the revolving motion (which occurs at $T/4$). However, the occurrence of the second peak in lift appears to be principally driven by rotational (pitch-up) effects. Overall, we note that quasi-steady effects from the revolving motion contribute positively to lift production throughout the whole stroke while rotational/added mass effects contribute negatively/positively during the first half of the stroke and positively/negatively during the

Discovering optimal flapping wing kinematics

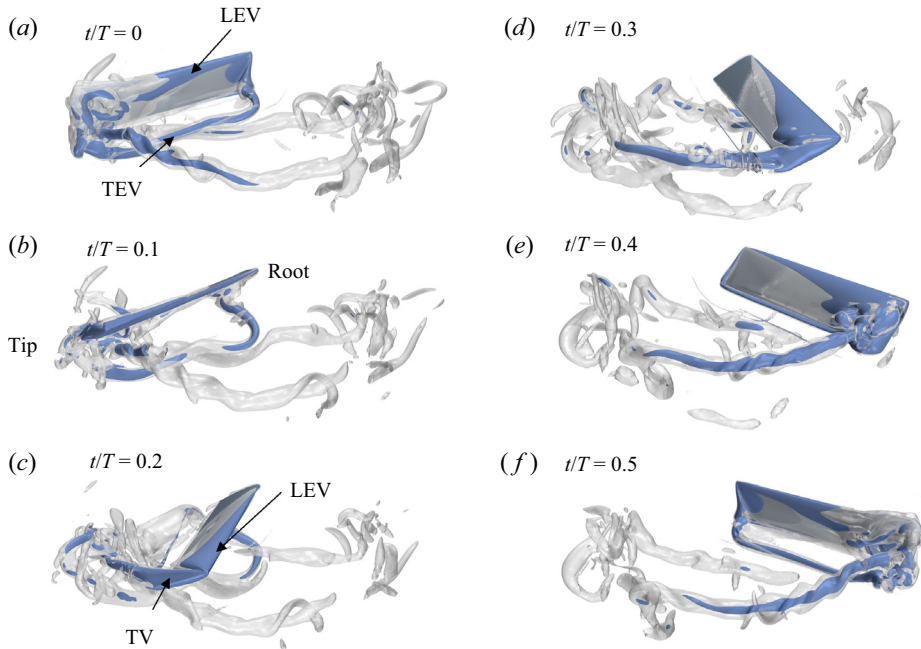


Figure 12. Non-dimensional Q -criterion isosurfaces obtained at different instants t/T , from 0 to 0.5, for the most efficient motion. The grey isosurfaces correspond to a value of 1 and the blue to a value of 10.

second half, respectively. Finally, by comparing the lift obtained during the first and third cycles, it is shown that wake effects (present during the third cycle but virtually absent during the first cycle) contribute to reducing lift. These effects can be classified into two contributions: one associated with the overall downwash induced by lift production during previous strokes and another one associated with local effects of wing–vortex interactions. The overall downwash tends to reduce the effective angle of attack of the wing and hence can be viewed as a general offset of the lift curve towards lower values. Wing–vortex interactions, however, may have very local effects, especially at stroke reversal where the lift is here found to reach negative values.

The occurrence of wing–vortex interactions can be investigated by looking at the flowfield in the vicinity of the wing. To this aim, isosurfaces of non-dimensional Q -criterion, defined as $Qc^2/\overline{U_\phi}^2$, are displayed in figure 12 at different instants throughout the motion for half a period (i.e. t/T from 0 to 0.5, corresponding to instants (a)–(f) in figure 10a). Two isosurfaces corresponding to values of 1 (grey) and 10 (blue) are shown to visualise the evolution of the vortex structures. The video corresponding to these snapshots is available in the supplementary movies available at <https://doi.org/10.1017/jfm.2023.832>. At the start of the stroke in panel (a), the wing is at its leftmost position and pitches clockwise. A trailing edge vortex (TEV) is visible in the vicinity of the trailing edge as a result of the rotation of the wing between strokes. The leading-edge vortex (LEV) generated during the previous stroke is also visible in the vicinity of the wing. In panel (b), the wing revolves from left to right. The LEV from the previous stroke sweeps on the lower surface, which contributes to negative lift by acting as a low-pressure region beneath the wing. In panels (c) to (e), a new leading-edge vortex is formed. It is cone-shaped and attached to the wing, as the velocity at the tip of the wing is higher than that near the

wing root. It reaches a quasi-steady state, which corresponds to the observations made on the lift in [figure 10](#). It is known that the leading-edge vortex is key to delay stall on low aspect ratio revolving wings and contributes to increasing the lift production throughout the stroke, by generating a low pressure area on the upper surface of the wing (Ellington *et al.* 1996; Dickinson *et al.* 1999; Eldredge & Jones 2019). A tip vortex (TV) is also visible. In panels (c) and (d), the TV and LEV appear as two distinct structures. In panel (e), the wing starts its counter-clockwise pitching motion to reverse for the next stroke. The wing angle passes through vertical before the wing reaches its rightmost position. This (slightly advanced) rotation mechanism increases the angle of attack and the leading-edge vortex, initially cone-shaped in panel (d), expands chord-wise and covers most of the wing while still being attached. In panel (e), the TV and LEV merge at the tip of the wing. Then, in panel (f), the leading-edge vortex merges with the 3-D structures of the tip vortex as the wing fully rotates for the next stroke. The decrease in revolution speed $\dot{\phi}$ of the wing weakens the spanwise pressure gradient stabilising the leading-edge vortex, as described by Medina & Jones (2016). The axial flow along the leading edge decreases as the pitch movement dominates in this reversal part of the flapping. The leading edge vortex loses its quasi-steady state, and this is linked to the fast and sudden drop in lift after the second peak.

4.3. High-lift kinematics

The flapping motion generating the highest lift is investigated in this section, as a typical example of motions in regime 2. The parameters are $a_1 = 0.186$, $b_1 = -0.586$, $b_3 = -0.3$, $\phi_0 = 10^\circ$ and $\delta = 167^\circ$. The corresponding ψ and ϕ time evolutions are displayed in [figure 13](#), along with the lift and power coefficient evolution over a flapping period, obtained by DNS and the neural network model. The neural network error on the prediction is 1.9% for η and 0.9% for $\overline{C_L}$. The motion is very different from the most efficient one and the three reference kinematics: it is a low revolution amplitude motion, with a complex pitching motion during the stroke. Due to the large phase shift between pitching and revolving motions, the pitch angle ψ changes sign very early in the stroke, i.e. soon after $t/T = 0.25$, meaning that the wing is oriented backwards during a large portion of the stroke. This is in contrast with previously studied motions (regime 1), where ψ changed sign around stroke reversal, i.e. at $t/T \approx 0.5$. Moreover, the pitching motion during a stroke can be divided in two phases: a main pitch-up motion ($t/T \in [0.2 - 0.4]$), and smaller pitching variations at the beginning and end of a stroke.

The lift coefficients produced in the upstroke (i.e. $t/T \in [0, T/2]$) and downstroke (i.e. $t/T \in [T/2, T]$) are different, due to the pitching motion not being fully symmetrical. The mean lift coefficient in the upstroke is 1.67 and is 1.19 in the downstroke. In both strokes, there are two peaks of lift production, the first one around half of the stroke and a second peak, smaller and flatter, towards the end of the stroke. During the upstroke, the first peak reaches $C_L = 7.5$, while the second one reaches $C_L = 0.96$. For the downstroke, the peaks reach maximum values of $C_L = 4.75$ and 2, respectively. The maximum lift value may change, but the overall lift evolution is similar between both strokes, thus a single stroke is analysed to understand the underlying flow characteristics leading to this performance.

Similarly to the previous subsection, the quasi-steady model from Lee *et al.* (2016) is used to estimate the main contributions to the lift peaks. [Figure 14](#) compares the lift predicted by the quasi-steady model with that obtained from the present simulations during the first and third flapping cycles. First, it is observed that the quasi-steady model predicts a lift peak slightly before $t = T/4$. This peak occurs at a similar instant to that obtained with

Discovering optimal flapping wing kinematics

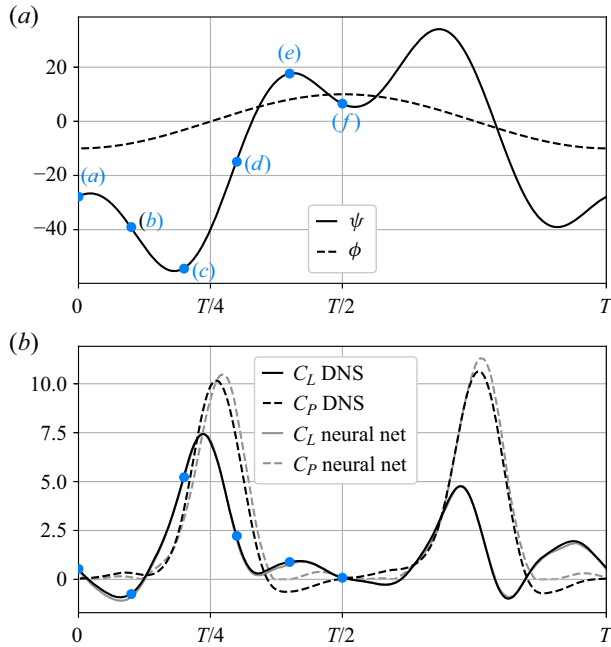


Figure 13. Time evolution of the (a) kinematics parameters, and (b) lift and power coefficients for highest-lift-generating motion, obtained from DNS (black) and neural networks (grey). Key instants are located by blue dots.

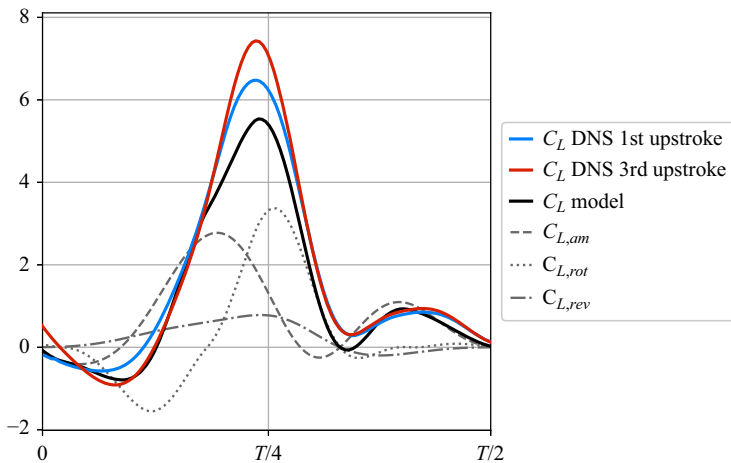


Figure 14. Quasi-steady lift model (black line) from Lee *et al.* (2016) and its different components (translational, rotational and added mass effects), compared with the DNS results during the the first (blue line) and third (red line) upstrokes for the most efficient motion.

numerical simulations but reaches lower levels. It appears to be dominated by both added mass effects and rotational terms. The difference in peak amplitude between simulations and the quasi-steady model suggests that a part of the lift is produced by unsteady, vortex-induced effects. This point will be further addressed later in this subsection. Second, the second small peak is well predicted by the quasi-steady model and seems to

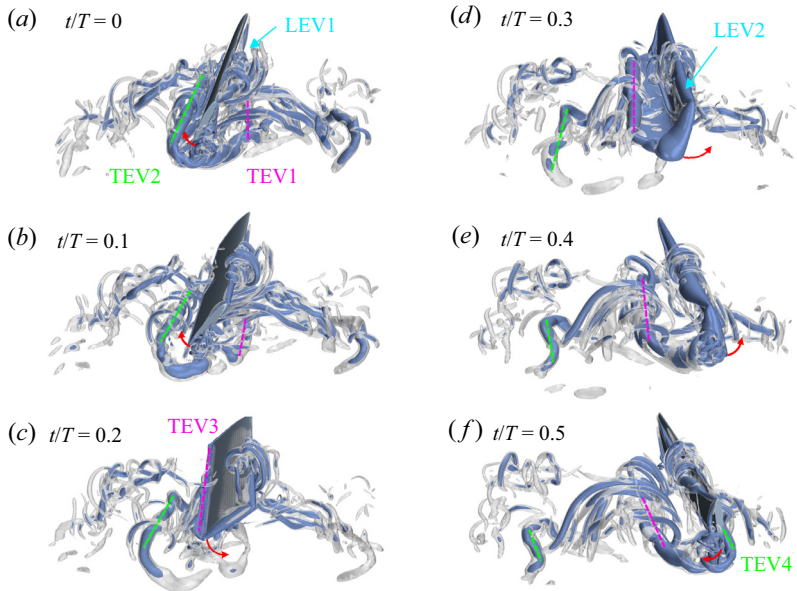


Figure 15. Q -criterion isosurfaces obtained at different instants t/T for the highest-lift-generating motion. The grey isosurfaces correspond to a value of 60, and the blue to a value of 200. Red arrows indicate the direction of the pitching motion of the wing, and dashed lines track the evolution of particular vortices.

be almost exclusively driven by added mass effects. Finally, it is shown that the amplitude of the lift peak is larger during the third flapping cycle than during the first flapping cycle. That is, wake effects appear to promote unsteady effects responsible for lift production near mid-stroke. However, the lift is slightly reduced before $t \approx T/6$ and very slightly increased after $t \approx T/3$. It is striking that the overall downwash induced by lift production during previous strokes does not have a detrimental influence on lift, conversely to what has been observed for the most efficient case analysed in the previous subsection. This is in line with the weak contribution of revolving motion-induced quasi-steady forces to the overall lift.

Figure 15 displays non-dimensional Q -criterion isosurfaces throughout one stroke, for values of $Qc^2/\bar{U}_\phi^2 = 60$ (grey) and 200 (blue). The video corresponding to these snapshots is available in the supplementary movies. Because of the low revolution amplitude and the complex 3-D vortical structures, spanwise vorticity obtained in the $r = 0.75b$ spanwise section is also displayed at various times in figure 16 to ease the visualisation. In the snapshot in panel (a), the wing is at its leftmost position and it initiates revolution from left to right. Three vortices are visible, generated in the previous strokes: one leading-edge vortex (LEV1) and two counter-rotating trailing edge vortices (TEV1 and TEV2). The TEV positions throughout the motion are highlighted using coloured dashed lines in figure 15. The line is pink if the vortex is created by the main pitch motion (panels c–e), and it is green if created by variations of smaller amplitude between strokes. In panels (a) and (b), the wing pitches clockwise and generates negative lift. LEV1 strongly interacts with the leading edge and promotes the generation of counter-rotating vorticity which appears to shed in panel (c). From panels (c)–(e), the wing performs a counter-clockwise pitching motion while revolving from left to right. The angle of attack increases from 25° to 107° . This pitch-up motion creates the main lift peak. The combination of the rotation and revolution implies a high trailing edge speed where the flow separates and rolls up

Discovering optimal flapping wing kinematics

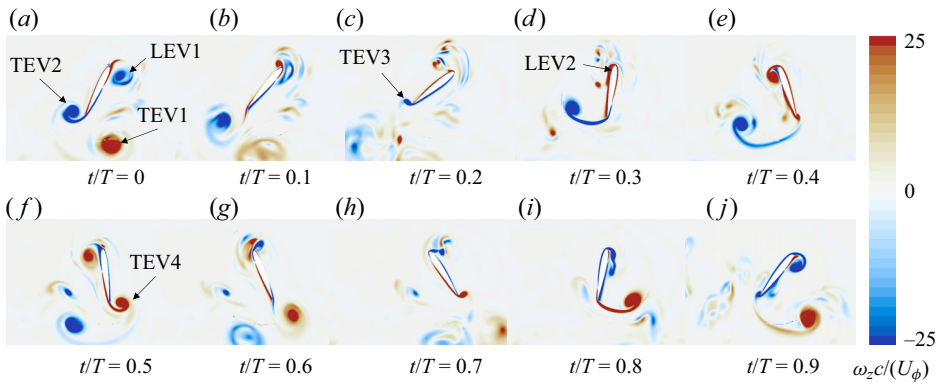


Figure 16. Spanwise vorticity contours obtained in the $r = 0.75b$ spanwise cross-section at different instants t/T for the highest-lift-generating motion.

into a trailing edge vortex TEV3. The fast generation of this intense TEV3 is clearly visible from panels (c) and (d). A leading-edge vortex LEV2 is also generated during this phase, as highlighted in panel (d). The fast formation of both leading and trailing edge vortices from panels (c)–(e) generates a large time variation in the first moment of vorticity which, following the impulse theory (Wu 1981), is directly related to force production. Interestingly, this mechanism resembles that recently observed for mosquitoes (Liu *et al.* 2020), which flap their wings with a relatively small amplitude, i.e. typically 45° . In panels (c) and (e), the leading edge vortex develops close to the wing but remains small due to the limited revolving amplitude. Lift generation is here very different to that observed in the high-efficiency case where the LEV on its own, by acting as a low-pressure region on the upper surface of the wing, is the main contributor to lift generation. In other words, the ‘delayed stall’ mechanism is not relevant for this low stroke amplitude motion because the leading-edge vortex does not have enough time to form (Wang *et al.* 2004; Zhu & Sun 2017; Liu *et al.* 2020). In panel (e), LEV2 is wrapped by vorticity braids around the mid-span that result from the development of 3-D spanwise instabilities. The latter may be triggered by the relative proximity of 3-D instabilities that have already developed in LEV1 and TEV2.

Between panels (e) and (f), the revolving speed decreases to zero and the wing pitches clockwise. TEV3 and LEV2 can be viewed as a vortex dipole that generates a fluid jet oriented towards the trailing edge. Together with the trailing edge speed induced by the pitching motion, this jet leads to the formation of TEV4 as the flow separates past the trailing edge. This mechanism was also observed for mosquitoes (Bomphrey *et al.* 2017). Again, the generation of TEV4 contributes to the time rate of change of the first moment of vorticity, and hence aerodynamic force. This mechanism, which is favoured by the interaction with a TEV3 and LEV2 induced jet, is found to be correlated with the occurrence of the second small lift peak in figure 14, although it was suggested that the latter is driven by added mass effects. At the end of the stroke, LEV2 seems to burst into smaller scale structures with vorticity braids developing along its full span.

Then the next stroke begins, involving similar phenomena. We note from panel (g), and similar to panel (b), that the contra-rotating vortex dipole formed by TEV2 and TEV4 rapidly advects downstream while LEV2 interacts with the leading edge of the wing.

5. Conclusion

This paper reports on the discovery of optimal, hovering flapping wing kinematics using deep neural networks and multi-objective optimisation.

First, a large database is produced using direct numerical simulation (DNS) of randomly generated kinematics. Then, the database is used to train deep neural networks (NNs) that predict the instantaneous lift and revolution and rotation torques experienced by the wing. Using sparse information on the kinematics, these NNs have very low runtime and can thus be used as inputs to a multi-objective optimisation that identifies the Pareto front of optimal solutions (i.e. optimal kinematics) in the lift versus efficiency space. DNS of a few arbitrarily selected optimal kinematics is then conducted to augment the initial database and perform another NN training. This (iterative) active learning strategy allows convergence towards an accurate Pareto front that is found to include solutions that lie far away (in the lift versus efficiency space) from the initial dataset. Specifically, the best lift reached is improved by 52 % and the best efficiency by 12 %.

The main advantage of this method is the reduced computational cost. The multi-optimisation was performed for 17 % of what it would have costed using only CFD. The active learning algorithm is independent of the optimisation problem or the type of surrogate model. It also is an interesting method for surrogate modelling to use in case of a pre-existing dataset not sampled enough near the optimal points. The iterative process allows for a low-cost improvement of the model, instead of random sampling of the search space.

Interestingly, the Pareto front obtained by the active learning approach reveals two distinct regimes, associated with high-efficiency and high-lift kinematics, respectively. The multi-objective optimisation allowed the generation of the whole range of motions transitioning between the two regimes, enabling the study of which parameter drives the lift production or efficiency.

On the one hand, the most efficient motion from the Pareto front is resembling the reference bio-inspired kinematics (i.e. hawkmoth, fruit fly and honeybee), characterised by a high stroke amplitude (110°), a low average angle of attack (37°), and an almost 90° phase shift between revolving and pitching motions. The flow is weakly unsteady, with the formation of an attached leading-edge vortex that reaches a quasi-steady state.

On the other hand, the highest lift motion has a low stroke amplitude (20°), with a fast pitch-up motion during the stroke combined with a large phase shift (i.e. much larger than 90°) between revolving and pitching motions. The lift profile is characterised by a first large lift peak at midstroke, followed by a second smaller peak. The first peak results from the fast pitch-up motion of the wing and is dominated by strong added mass effects and the fast generation of a strong trailing edge vortex, although quasi-steady effects from both revolving and rotating motions also have important contributions. The second peak, which occurs at the end of the stroke, is due to a reversed pitching motion. Because of the large phase shift between revolving and pitching motions, the leading edge of the wing is oriented backward and hence the reversed pitching motion is again a pitch-up motion and generates lift principally through added mass effects. Here, the interaction with the wake promotes the generation of a new counter-rotating trailing edge vortex. Both trailing edge vortices form a counter-rotating dipole that advects downstream, which can be viewed as the footprint of lift production. Conversely to high-efficiency cases, the kinematics and flow physics are reminiscent of those recently observed for mosquitoes, albeit at smaller flapping amplitudes.

Discovering optimal flapping wing kinematics

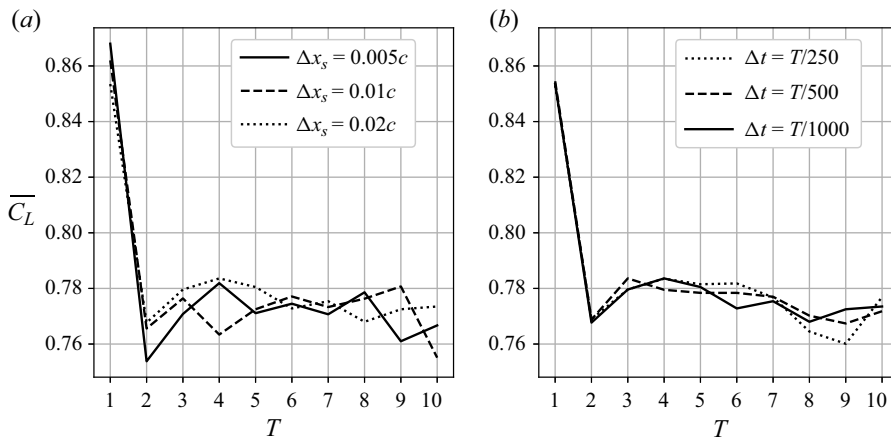


Figure 17. Cycle-averaged lift coefficient for ten flapping cycles, using three (a) spatial and (b) temporal discretisations.

Finally, the wake capture effects are also compared between the two studied motions. This phenomenon appears detrimental for lift production in the high-efficiency case, and conversely, is beneficial for the high-lift case.

Supplementary movies. Supplementary movies are available at <https://doi.org/10.1017/jfm.2023.832>.

Funding. This research received no specific grant from any funding agency, commercial or not-for-profit sectors.

Declaration of interests. The authors report no conflict of interest.

Author ORCIDs.

 Baptiste Corban <https://orcid.org/0009-0008-3412-6660>;

 Thierry Jardin <https://orcid.org/0000-0001-9704-2984>.

Appendix

Figure 17 displays the cycle-averaged lift coefficients obtained for spatial discretisations $\Delta x = 0.02c$, $0.01c$ and $0.005c$ in panel (a) and temporal discretisations $\Delta t = T/250$, $T/500$ and $T/1000$ in panel (b). Here, Δt is set to $T/1000$ in panel (a) and Δx is set to $0.02c$ in panel (b). The flapping motion considered is similar to that used in a number of previous experimental and numerical studies by Jardin, David & Farcy (2009), Jardin *et al.* (2012) and Diaz-Arriba *et al.* (2021). The flapping amplitude is set to 120° . Revolving and pitching motions are defined by fifth-order polynomials, as detailed by Diaz-Arriba *et al.* (2021). The minimum angle of attack is set to 45° and the Reynolds number based on the mean revolving speed at the wing tip is approximately 1000.

The figure shows that initial transients have sufficiently decayed after two flapping cycles, in line with observations from Liu *et al.* (2020) and Bhat *et al.* (2020), for example. Furthermore, it is found that results with lowest spatial and temporal resolutions approximate within reasonable accuracy those obtained with highest spatial and temporal resolutions. From the second period, the cycle-averaged lift coefficients do not deviate from that obtained at the tenth period using the highest spatial and temporal resolutions by more than 2.5%. Hence, the lowest resolutions are used in the present work to generate the

initial database on which the neural networks are trained. This leads to a computational cost that is approximately 75 times less than that with the highest resolutions.

Finally, it is worth mentioning that the numerical solver has been previously validated on various cases at similar Reynolds numbers, including axisymmetric bluff bodies (Bury & Jardin 2012), revolving wings (Jardin & David 2017) and perching airfoils (Jardin & Doué 2019).

REFERENCES

- AJURIA-ILLARAMENDI, E., BAUERHEIM, M. & CUENOT, B. 2022 Performance and accuracy assessments of an incompressible fluid solver coupled with a deep convolutional neural network. *Data-Centric Engng* **3**, e2.
- ALGUACIL, A., BAUERHEIM, M., JACOB, M.C. & MOREAU, S. 2021 Predicting the propagation of acoustic waves using deep convolutional neural networks. *J. Sound Vib.* **512**, 116285.
- ALGUACIL, A., BAUERHEIM, M., JACOB, M.C. & MOREAU, S. 2022 Deep learning surrogate for the temporal propagation and scattering of acoustic waves. *AIAA J.* **60** (10), 5890–5906.
- ALTSHULER, D.L., DICKSON, W.B., VANCE, J.T., ROBERTS, S. & DICKINSON, M.H. 2005 Short-amplitude high-frequency wing strokes determine the aerodynamics of honeybee flight. *Proc. Natl Acad. Sci.* **102** (50), 18213–18218.
- AONO, H., LIANG, F. & LIU, H. 2008 Near-and far-field aerodynamics in insect hovering flight: an integrated computational study. *J. Expl Biol.* **211** (2), 239–257.
- BAQUÉ, P., REMELLI, E., FLEURET, F. & FUA, P. 2018 Geodesic convolutional shape optimization. In *Proceedings of the 35th International Conference on Machine Learning*, vol. 80, pp. 472–481. PMLR.
- BAYIZ, Y.E. & CHENG, B. 2021 State-space aerodynamic model reveals high force control authority and predictability in flapping flight. *J. R. Soc. Interface* **18** (181), 20210222.
- BHAT, S.S., ZHAO, J., SHERIDAN, J., HOURIGAN, K. & THOMPSON, M.C. 2020 Effects of flapping-motion profiles on insect-wing aerodynamics. *J. Fluid Mech.* **884**, A8.
- BLANK, J. & DEB, K. 2020 pymoo: multi-objective optimization in Python. *IEEE Access* **8**, 89497–89509.
- BOMPHELY, R., NAKATA, T., PHILLIPS, N., *et al.* 2017 Smart wing rotation and trailing-edge vortices enable high frequency mosquito flight. *Nature* **544**, 92–95.
- BOS, F.M., LENTINK, D., VAN OUDHEUSDEN, B.W. & BIJL, H. 2008 Influence of wing kinematics on aerodynamic performance in hovering insect flight. *J. Fluid Mech.* **594**, 341–368.
- BURY, Y. & JARDIN, T. 2012 Transitions to chaos in the wake of an axisymmetric bluff body. *Phys. Lett. A* **376** (45), 3219–3222.
- BUSCH, C., GEHRKE, A. & MULLENNERS, K. 2012 On the parametrisation of motion kinematics for experimental aerodynamic optimisation. *Exp. Fluids* **63**, 10.
- CAI, X., KOLOMENSKIY, D., NAKATA, T. & LIU, H. 2021 A CFD data-driven aerodynamic model for fast and precise prediction of flapping aerodynamics in various flight velocities. *J. Fluid Mech.* **915**, A114.
- CALADO, A., POLETTI, R., KOLOSZAR, L. & MENDEZ, M.A. 2023 A robust data-driven model for flapping aerodynamics under different hovering kinematics. *Phys. Fluids* **35** (4), 047122.
- CAMPET, R., ROY, P.T., CUENOT, B., RIBER, E. & JOUHAUD, J.-C. 2020 Design optimization of an heat exchanger using Gaussian process. *Intl J. Heat Mass Transfer* **150**, 119264.
- COLOMBO, M., BAUERHEIM, M. & MORLIER, J. 2023 Prediction of gust aeroelastic dynamics of HALE using graph neural networks. *AIAA SCITECH 2023 Forum*, 2570.
- DEB, K., PRATAP, A., AGARWAL, S. & MEYARIVAN, T. 2002 A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **6** (2), 182–197.
- DIAZ-ARRIBA, D., JARDIN, T., GOURDAIN, N., PONS, F. & DAVID, L. 2021 Numerical investigation of three-dimensional asymmetric hovering flapping flight. *Phys. Fluids* **33** (11), 111907.
- DICKINSON, M.H. & GÖTZ, K.G. 1993 Unsteady aerodynamic performance of model wings at low Reynolds numbers. *J. Expl Biol.* **174** (1), 45–64.
- DICKINSON, M.H., LEHMANN, F.O. & SANE, S.P. 1999 Wing rotation and the aerodynamic basis of insect flight. *Science* **284** (5422), 1954–1960.
- ELDREDGE, J.D. & JONES, A.R. 2019 Leading-edge vortices: mechanics and modeling. *Annu. Rev. Fluid Mech.* **51**, 75–104.
- ELLINGTON, C.P. 1984 The aerodynamics of hovering insect flight. *Phil. Trans. R. Soc. Lond.* **305** (1122), 1–181.
- ELLINGTON, C.P., VAN DEN BERG, C., WILLMOTT, A.P. & THOMAS, A.L. 1996 Leading-edge vortices in insect flight. *Nature* **384** (6610), 626–630.

- GEHRKE, A. & MULLENERS, K. 2021 Phenomenology and scaling of optimal flapping wing kinematics. *Bioinspir. Biomim.* **16** (2), 026016.
- GUPTA, R. & JAIMAN, R. 2022 Three-dimensional deep learning-based reduced order model for unsteady flow dynamics with variable Reynolds number. *Phys. Fluids* **34**, 033612.
- HOCHREITER, S. & SCHMIDHUBER, J. 1997 Long short-term memory. *Neural Comput.* **9** (8), 1735–1780.
- JARDIN, T. & DAVID, L. 2017 Root cutout effects on the aerodynamics of a low-aspect-ratio revolving wing. *AIAA J.* **55** (8), 2717–2726.
- JARDIN, T., DAVID, L. & FARCY, A. 2009 Characterization of vortical structures and loads based on time-resolved PIV for asymmetric hovering flapping flight. *Exp. Fluids* **46**, 847–857.
- JARDIN, T. & DOUÉ, N. 2019 Influence of pitch rate on freely translating perching airfoils. *J. Fluid Mech.* **873**, 49–71.
- JARDIN, T., FARCY, A. & DAVID, L. 2012 Three-dimensional effects in hovering flapping flight. *J. Fluid Mech.* **702**, 102–125.
- KRÜGENER, M., ZAPATA USANDIVARAS, J., BAUERHEIM, M. & URBANO, A. 2022 Coaxial-injector surrogate modeling based on Reynolds-averaged Navier–Stokes simulations using deep learning. *J. Propul. Power* **38**, 1–16.
- LEE, Y.J. & LUA, K.B. 2018 Optimization of simple and complex pitching motions for flapping wings in hover. *AIAA J.* **56** (6), 2466–2470.
- LEE, Y.J., LUA, K.B., LIM, T.T. & YEO, K.S. 2016 A quasi-steady aerodynamic model for flapping flight with improved adaptability. *Bioinspir. Biomim.* **11** (3), 036005.
- LENTINK, D. & DICKINSON, M.H. 2009 Rotational accelerations stabilize leading edge vortices on revolving fly wings. *J. Expl Biol.* **212** (16), 2705–2719.
- LI, J., DU, X. & MARTINS, J.R.R.A. 2022 Machine learning in aerodynamic shape optimization. *Prog. Aerosp. Sci.* **134**, 100849.
- LIGHTHILL, M. 1973 On the Weis-Fogh mechanism of lift generation. *J. Fluid Mech.* **60** (1), 1–17.
- LIU, H. & AONO, H. 2009 Size effects on insect hovering aerodynamics: an integrated computational study. *Bioinspir. Biomim.* **4** (1), 015002.
- LIU, L., DU, G. & SUN, M. 2020 Aerodynamic-force production mechanisms in hovering mosquitoes. *J. Fluid Mech.* **898**, A19.
- LYE, K., MISHRA, S., RAY, D. & CHANDRASHEKAR, P. 2021 Iterative surrogate model optimization (ISMO): an active learning algorithm for PDE constrained optimization with deep neural networks. *Comput. Meth. Appl. Mech. Engng* **374**, 113575.
- MEDINA, A. & JONES, A.R. 2016 Leading-edge vortex burst on a low-aspect-ratio rotating flat plate. *Phys. Rev. Fluids* **1** (4), 044501.
- NAKATA, T., LIU, H. & BOMPHELY, R.J. 2015 A CFD-informed quasi-steady model of flapping-wing aerodynamics. *J. Fluid Mech.* **783**, 323–343.
- PANDI, A., *et al.* 2022 A versatile active learning workflow for optimization of genetic and metabolic networks. *Nat. Commun.* **13**, 3876.
- PASZKE, A., *et al.* 2019 PyTorch: an imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pp. 8024–8035. Curran Associates, Inc.
- ROY, P.T., SEGUI, L.M., JOUHAUD, J.-C. & GICQUEL, L. 2018 Resampling strategies to improve surrogate model-based uncertainty quantification: application to LES of LS89. *Intl J. Numer. Meth. Fluids* **87** (12), 607–627.
- SANE, S.P. & DICKINSON, M.H. 2001 The control of flight force by a flapping wing: lift and drag production. *J. Expl Biol.* **204** (15), 2607–2626.
- SHYY, W., AONO, H., CHIMAKURTHI, S.K., TRIZILA, P., KANG, C.-K., CESNIK, C.E.S. & LIU, H. 2010 Recent progress in flapping wing aerodynamics and aeroelasticity. *Prog. Aerosp. Sci.* **46** (7), 284–327.
- SORTEBERG, W.E., GARASTO, S., POUPLIN, A.S., CANTWELL, C.D. & BHARATH, A.A. 2018 Approximating the solution to wave propagation using deep neural networks. In *32nd Conference on Neural Information Processing Systems - NeurIPS*, arXiv:1812.01609. Springer.
- SUN, M. & TANG, J. 2002 Unsteady aerodynamic force generation by a model fruit fly wing in flapping motion. *J. Expl Biol.* **205** (1), 55–70.
- TOMPSON, J., SCHLACHTER, K., SPRECHMANN, P. & PERLIN, K. 2017 Accelerating Eulerian fluid simulation with convolutional networks. In *Proceedings of the 34th International Conference on Machine Learning*, in Proceedings of Machine Learning Research, vol. 70, pp. 3424–3433.
- VAN VEEN, W.G., VAN LEEUWEN, J.L., VAN OUDHEUSDEN, B.W. & MUIJRES, F.T. 2022 The unsteady aerodynamics of insect wings with rotational stroke accelerations, a systematic numerical study. *J. Fluid Mech.* **936**, A3.

- WANG, Z.J., BIRCH, J.M. & DICKINSON, M.H. 2004 Unsteady forces and flows in low Reynolds number hovering flight: two-dimensional computations versus robotic wing experiments. *J. Expl Biol.* **207**, 449–460.
- WANG, A., LIANG, H., MCDANNALD, A., TAKEUCHI, I. & KUSNE, A.G. 2022 Benchmarking active learning strategies for materials optimization and discovery. *Oxford Open Mater. Sci.* **2** (1), itac006.
- WANG, P., WANG, Y., LIU, T. & ZHANG, D. 2022 Active learning-based multi-objective optimization for aerodynamic performance of a supercritical carbon dioxide turbine. *Struct. Multidiscipl. Optim.* **65**, 267.
- WU, J.C. 1981 Theory for aerodynamic force and moment in viscous flows. *AIAA J.* **19** (4), 432–441.
- WU, J.H. & SUN, M. 2004 Unsteady aerodynamic forces of a flapping wing. *J. Expl Biol.* **207** (7), 1137–1150.
- ZHENG, L., HEDRICK, T.L. & MITTAL, R. 2013 A multi-fidelity modelling approach for evaluation and optimization of wing stroke aerodynamics in flapping flight. *J. Fluid Mech.* **721**, 118–154.
- ZHENG, H., XIE, F., JI, T. & ZHENG, Y. 2020 Kinematic parameter optimization of a flapping ellipsoid wing based on the data-informed self-adaptive quasi-steady model. *Phys. Fluids* **32** (4), 041904.
- ZHU, H.J. & SUN, M. 2017 Unsteady aerodynamic force mechanisms of a hoverfly hovering with a short stroke-amplitude. *Phys. Fluids* **29**, 081901.