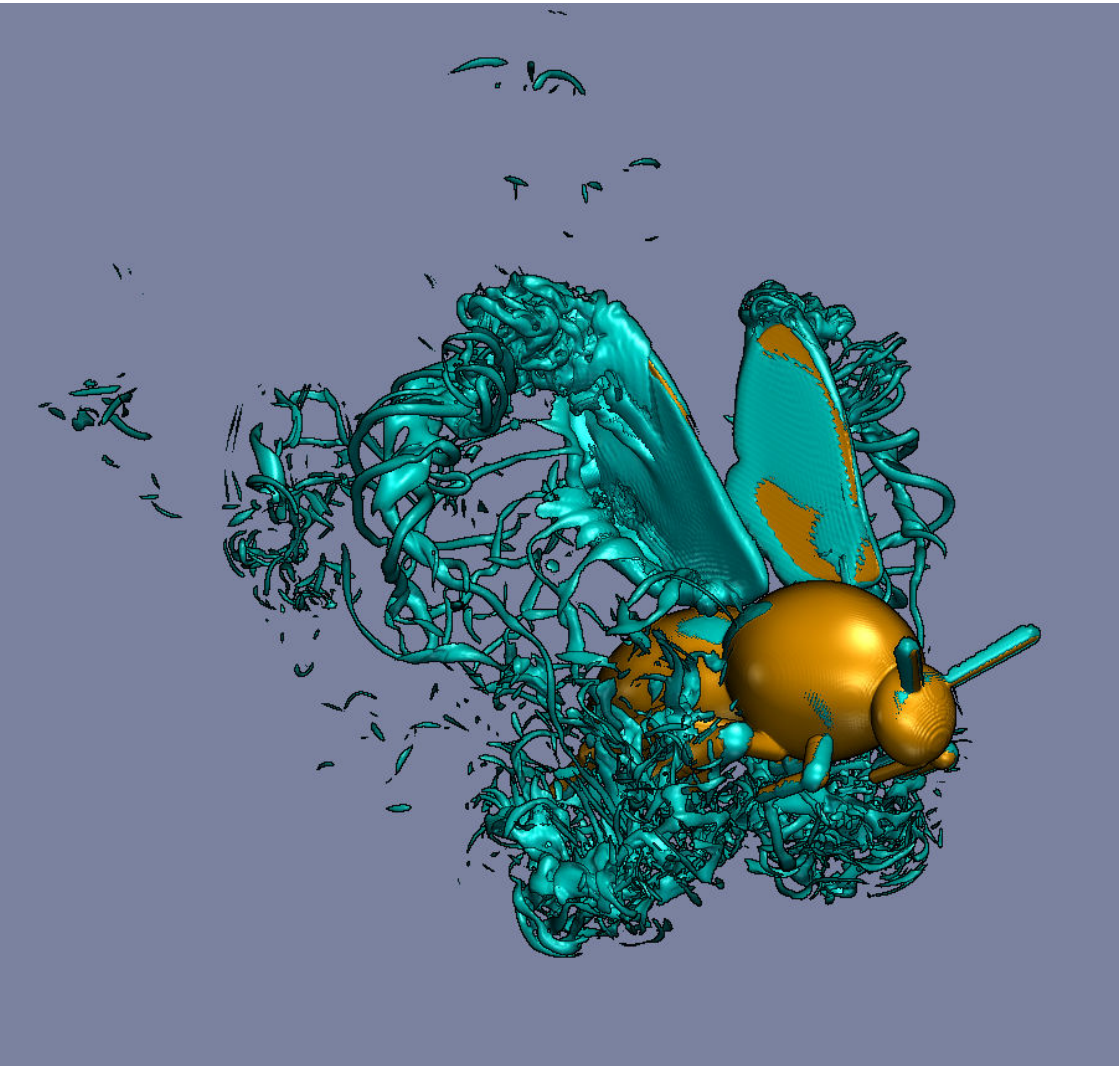
Numerical Modeling of Fluid-Structure Interaction in Bio-Inspired Propulsion

Thomas Engels



Sculpting a lion is actually quite easy. You start with a solid block and simply remove everything that does not look like lion.

Dietmar Engels

Dedicated to my mum, Margareta Engels, who is doubtlessly the greatest mum ever to walk the planet.

UNIVERSITÉ D'AIX-MARSEILLE

TECHNISCHE UNIVERSITÄT BERLIN

ECOLE DOCTORALE 353 SCIENCES POUR L'INGÉNIEUR:
MÉCANIQUE, PHYSIQUE, MICRO ET NANOÉLECTRONIQUE

UFR SCIENCES

LABORATOIRE DE MÉCANIQUE, MODÉLISATION & PROCÉDÉS
PROPRES UMR7340

Thèse présentée pour obtenir le grade universitaire de docteur

Discipline : Mécanique et Physique des Fluides

Thomas ENGELS

Numerical Modeling of Fluid–Structure Interaction in Bio–Inspired Propulsion

Soutenue le 10/12/2015 devant le jury :

Hao LIU	Chiba University, Japan	Rapporteur
Angelo IOLLO	Université de Bordeaux 1	Rapporteur
Fritz-Olaf LEHMANN	Universität Rostock	Examinateur
Marie FARGE	École Normale Supérieure Paris	Examinatrice
Dmitry KOLOMENSKIY	Chiba University, Japan,	Invité
Kai SCHNEIDER	Aix-Marseille Université	Directeur de thèse
Jörn SESTERHENN	Technische Universität Berlin	Directeur de thèse

Technische Universität Berlin

Aix*Marseille université

Numerical Modeling of Fluid–Structure Interaction in Bio–Inspired Propulsion

Thomas Engels, M.Sc.

Der Fakultät Verkehrs- und Maschinensysteme der Technischen Universität Berlin zur Erlangung des akademischen Grades Doktor der Ingenieurwissenschaften (Dr.-Ing.) genehmigte Dissertation

Tag der mündlichen Aussprache: 10.12.2015

Promotionsausschuss:

Hao LIU	Chiba University, Japan	Gutachter
Angelo IOLLO	Université de Bordeaux 1	Gutachter
Fritz-Olaf LEHMANN	Universität Rostock	Prüfer
Marie FARGE	École Normale Supérieure Paris	Prüfer
Dmitry KOLOMENSKIY	Chiba University, Japan,	Gast
Kai SCHNEIDER	Aix-Marseille Université	Betreuer
Jörn SESTERHENN	Technische Universität Berlin	Betreuer

Contents

Acknowledgements

First of all, I want to express my upmost gratitude to my two supervisors, Kai Schneider in Marseille and Jörn Sesterhenn in Berlin, for their constant support over the four years of this PhD program. I am very grateful for the freedom of spirit they allow and the degree of independence that comes with it, and for trusting me to succeed this way. I want to express the same gratitude to Dmitry Kolomenskiy, who was my informal supervisor during the past five years, starting during my masters. He is also the main reason I became interested in the field of biomechanics and insect flight, so I would like to thank him for the great inspiration.

Malcolm Roberts was very helpful teaching me good practices of open-source programming and general code development, and I benefited a lot from his presence in Marseille.

My work would have been much harder without the excellent computer experts in both laboratories, Michel Pognant and Dominique Fougère in Marseille and Martin Franke and Lars Oergel in Berlin, and I would like to thank them for their kin support.

Moving twice a year between different countries was a difficult task, but it would have been twice as difficult without the help of the director of M2P2, Patrick Bontoux, who helped me each year finding an apartment, rapidly signing papers and even lending me his car.

I want to thank Fritz-Olaf Lehmann and Masateru Maeda for the fruitful discussions on insect flight, and especially Fritz-Olaf Lehmann for the help and advice preparing our joint research proposals.

I am grateful for countless meetings with Romain Nguyen van Yen on the penalization method, which unfortunately didn't result in a new active penalization method, but was fruitful nonetheless, and Marie Farge for her support, especially regarding wavelets, adaptivity and the research proposals.

It was very nice to have discussions with Julius Reiss, Jens Brouwer, Juan-Jose Pena-Fernandez, Matthias Lemke, Lewin Stein, Stefano Alois, Sergio Bengoechea Lozano, Robert Wilke and Sonja Hoßbach in the laboratory in

Berlin, and Amin Ghaffary, Stéphane Viazzo, Matthieu Leroy, Gustavo Lopez, Braulio Bernales and Marianna Pepona in Marseille.

I would also thank Margareta Engels for the effort of no longer beginning a phone call with “why didn’t you call?”, and my good friends Juliana Börner, Janina Motsch, Jens Rippel, Anna Sobkowiak, Janine Schwarz, Jana Mieseler, Claudia Höfler, Julia Radke, Georgia Druhm, Christian Gatzki, Annika Amling, Malcolm Roberts, Anne Cloarec, Vanessa Gontard, Matthieu Leroy, Romain Nguyen van yen, Dmitry Kolomenskiy, Margatera, Dietmar, Michael and Alexander Engels and all my other friends for providing me the desperately needed moral support during my thesis, and especially my love Fleur Grelet, whom I never would have met without a french-german lifestyle.

Abstract

English abstract

Flying and swimming animals have developed efficient ways to produce the fluid flow that generates the desired forces for their locomotion. These bio-inspired problems couple fluid dynamics and solid mechanics with complex geometries and kinematics. The present thesis is placed in this interdisciplinary context and uses numerical simulations to study these fluid–structure interaction problems with applications in insect flight and swimming fish. Based on existing work on rigid moving obstacles, using an efficient Fourier discretization, a numerical method has been developed, which allows the simulation of flexible, deforming obstacles as well, and provides enhanced versatility and accuracy in the case of rigid obstacles. The method relies on the volume penalization method and the fluid discretization is still based on a Fourier discretization. The code, designed to run on massively parallel supercomputers, is entirely open source and freely available on the internet. We first apply this method to insects with rigid wings, where the body and other details, such as the legs and antennae, can be included. After presenting detailed validation tests, we proceed to studying a bumblebee model in fully developed turbulent flow. Our simulations show that turbulent perturbations affect flapping insects in a different way than human-designed fixed-wing aircrafts. While in the latter, upstream perturbations can cause transitions in the boundary layer, the former do not present systematical changes in aerodynamic forces. We conclude that insects rather face control problems in a turbulent environment than a deterioration in force production. In the next step, we design a solid model, based on a one–dimensional beam equation, and simulate coupled fluid–solid systems. Applications deal, in a two-dimensional setup, with insect flight, but also with simplified three–dimensional models for swimming fish. In these 'swimmers', consisting of a flexible plate with one rigid direction, we study the influence of the shape on the hydrodynamic efficiency. A contracting shape, as found in some amphibians, is found to swim faster and require less power than an expanding shape, which is more similar

to most caudal fins observed in fish. We present evidence that this finding can be explained by a favorable interaction with the tip-vortices in the case of the contracting shape.

Résumé français

Les animaux volants et flottants ont développé des façons efficaces de produire l'écoulement de fluide qui génère les forces désirées pour leur locomotion. Ces problèmes "bio-inspirés" couplent la dynamique des fluides avec la mécanique des solides, y compris des géométries et cinématiques complexes. Cette thèse est placée dans ce contexte interdisciplinaire et utilise des simulations numériques pour étudier ces problèmes d'interaction fluides-structure, et les applique au vol des insectes et à la nage des poissons. Basée sur les travaux existants sur les obstacles mobiles rigides, utilisant une discrétisation de Fourier efficace, une méthode numérique a été développée, permettant également la simulation des obstacles déformables et fournissant une polyvalence et précision accrues dans le cas des obstacles rigides. L'algorithme se repose sur la méthode de pénalisation volumique et la discrétisation fluide est toujours basée sur une discrétisation de Fourier. Le code, conçu pour fonctionner sur des supercalculateurs massivement parallèles, est entièrement open source et disponible librement sur internet. Nous appliquons cette méthode d'abord aux insectes avec des ailes rigides, où le corps et d'autres détails, tels que les pattes et les antennes, peuvent être inclus. Après la présentation de tests de validation détaillée, nous procédons à l'étude d'un modèle de bourdon dans un écoulement turbulent pleinement développé. Nos simulations montrent que les perturbations turbulentes affectent les insectes volants d'une manière différente de celle des avions aux ailes fixées et conçues par l'humain. Dans le cas de ces derniers, des perturbations en amont peuvent déclencher des transitions dans la couche limite, tandis que les premiers ne présentent pas de changements systématiques dans les forces aérodynamiques. Nous concluons que les insectes se trouvent plutôt confrontés à des problèmes de contrôle dans un environnement turbulent qu'à une détérioration de la production de force. Lors de l'étape suivante, nous concevons un modèle solide, basé sur une équation de barre monodimensionnelle, et nous passons à la simulation des systèmes couplés fluide–structure. Les applications concernent d'abord des configurations en deux dimensions spatiales, spécifiques au vol de l'insecte, mais aussi des modèles tridimensionnels représentant la nage des poissons. Avec ces 'nageurs', constitués d'une plaque flexible avec une direction rigide, nous étudions l'influence de la forme sur l'efficacité hydrodynamique. Nous concluons qu'une forme contractée, c'est-à-dire une silhouette élargie à la tête et qui s'affine à la queue, que l'on trouve dans certains amphibiens, est plus efficace et permet une nage plus rapide qu'une forme exponentielle, qui est pourtant plus similaire à la plupart des nageoires caudales

observées chez les poissons. Nous présentons des preuves que cet effet peut être expliqué par une interaction favorable des vortex de bord avec la plaque dans le cas de la forme contractée.

Zusammenfassung auf Deutsch

Fliegende und Schwimmende Tiere haben im Laufe ihrer Evolution effiziente Wege gefunden, die Strömung zu erzeugen die die gewünschten Kräfte für ihre Fortbewegung produziert. Diese biologisch inspirierten Probleme koppeln Fluiddynamik mit Festkörpermechanik und komplexer Geometrie und Kinematik. Die vorliegende Arbeit ist in diesem interdisziplinären Kontext platziert und verwendet numerische Simulationen, um solche Fluid–Struktur Wechselwirkungsprobleme zu studieren, und sie auf den Insektenflug und schwimmende Fische anzuwenden. Basierend auf vorhergehenden Arbeiten zu starren, sich bewegenden Hindernissen, basierend auf einer effizienten Fourier Diskretisierung, wurde eine numerische Methode entwickelt, die die Simulation von flexiblen, verformbaren Hindernissen ermöglicht, und im Fall starrer Körper eine verbesserte Vielseitigkeit und Genauigkeit bietet. Das Verfahren beruht auf der "Volume Penalization Method"[1] und die Diskretisierung des Fluides ist immer noch auf einer Fourier Diskretisierung basiert. Der Code, ausgelegt für massiv parallele Supercomputer, ist komplett quelloffen und im Internet frei verfügbar. Zunächst wenden wir diese Methode auf Insekten mit starren Flügeln an, wobei der Körper und andere Details, wie die Beine und Fühler, mit einbezogen werden können. Anschließend an detaillierte Validierungstests gehen wir zum Studium einer Modellhummel in voll entwickelter turbulenter Strömung über. Unsere Simulationen zeigen, dass turbulente Störungen flatternde Insekten in einer fundamental anderen Weise als menschengemachte Starrflügler beeinflussen können. Während in letzteren stromaufwärts eingebrachte Störungen Transitionen in der Grenzschicht verursachen können, zeigen erstere keine signifikanten und systematischen Änderungen in ihren aerodynamischen Kräften. Wir schließen daraus, dass Insekten in turbulenten Strömungen eher mit erhöhten Kontrollanforderungen konfrontiert werden als mit einer Beeinträchtigung in der Produktion ihrer Auftriebskräfte. Im nächsten Schritt entwickeln wir ein mechanisches Modell, basierend auf einer eindimensionalen Balkengleichung für große Verformungen, und Simulieren gekoppelte Fluid-Struktur Systeme. Anwendungen behandeln, in zweidimensionalen Konfigurationen, Modelle für den Insektenflug, aber auch dreidimensionale Modelle von schwimmenden Fischen. Bei diesen "Schwimmern", bestehend aus einer flexiblen Platte mit einer starren Richtung, untersuchen wir den Einfluss der Form auf der hydrodynamischen Effizienz. Wir finden dass eine kontrahierende Form, wie sie in einigen Amphibien gefunden

[1] Etwa: Methode der Volumen-basierten Strafterme

wird, schneller schwimmt und weniger Energie benötigt als eine expandierende Form, die mehr Ähnlichkeit mit den in den meisten Fischen beobachteten Schwanzflossen hat. Wir präsentieren Hinweise, dass dieser Umstand durch eine günstige Wechselwirkung der Kantenwirbel mit der Struktur im Falle der kontrahierenden Form erklärt werden kann.

Version abrégée en français

Cette thèse traite la modélisation numérique des problèmes d'interaction fluide-structure, ainsi que leur simulation sur les supercalculateurs à grande échelle. Les applications traitent des problèmes inspirés par la nature, et plus particulièrement par les insectes volants.

Ces animaux ont développé des mécanismes de vol efficaces qui sont très différents de ceux utilisés par les avions conçus par les humains. Les insectes battent leurs ailes tandis que les avions utilisent des ailes fixes.

Les ailes sont généralement flexibles, ce qui signifie qu'elles peuvent se déformer lors de leurs mouvements. Par ailleurs la flexibilité est une caractéristique commune dans la nature. Les ailes d'insectes peuvent se plier et se tordre, les poissons sont flexibles et les plantes peuvent se plier avec le vent.

Par conséquent, la modélisation de ces problèmes d'interaction fluide-structure est essentielle pour mieux comprendre la nature.

Cette thèse est organisé en trois parties. La première partie traite de la méthode numérique de base, commune aux deux parties suivantes. La deuxième partie applique cette méthode à des problèmes avec des objets rigides, notamment des insectes avec des ailes rigides. La troisième partie étend la méthode à des obstacles flexibles, et nous l'appliquons à des problèmes pertinents pour les poissons qui nagent.

Partie I : Introduction et méthode numérique de base

Pour la simulation numérique des problèmes d'interaction fluide-structure, différentes méthodes existent.

Les méthodes classiques utilisent des maillages conformes qui sont alignés avec l'obstacle. Le maillage peut être structuré ou non structuré. Ces méthodes ont en général une bonne précision sur la surface. Cependant, ils nécessitent

la génération du maillage de discrétisation, ce qui est une procédure compliquée. Les maillages non-structurés souffrent en outre de leur structure de données inefficaces. D'ailleurs, le maillage doit être généré à chaque pas de temps si l'obstacle se déplace ou se déforme.

Plus récemment, d'autres méthodes ont été développées. Parmi eux, la méthode de frontière immergée[2] et la méthode de pénalisation volumique[3]. Ces méthodes ne nécessitent pas de maillages conformes. Elles sont plutôt résolues sur des maillages réguliers cartésiens qui ne changent pas dans le temps. Afin de prendre en compte les conditions aux limites, des termes supplémentaires sont inclus dans l'équation de Navier–Stokes.

Cette idée a d'abord été proposée par Peskin pour simuler l'écoulement dans le cœur humain. Son idée utilise des termes de forçage sur l'interface entre le fluide et le solide. Plus tard, Angot a introduit la méthode de pénalisation volumique. Dans cette méthode, le volume entier du corps est forcé, en ajoutant le terme $-(\chi/C_\eta)\underline{u}$ aux équations de Navier–Stokes, où $\chi = 1$ à l'intérieur de l'obstacle et $\chi = 0$ à l'extérieur. La méthode est inspirée par l'intuition physique qu'un milieu poreux avec une porosité C_η peut imiter une paroi solide imperméable, si la porosité est suffisamment faible.

Dans cette thèse nous utilisons exclusivement la méthode de pénalisation volumique. Nous allons montrer qu'elle peut fournier des résultats précis même pour des problèmes assez complexes.

La méthode de pénalisation est une technique pour inclure les conditions aux limites dans l'équation originale, ainsi découplant la discrétisation spatiale et la géométrie du problème. Cependant, une discrétisation est toujours nécessaire. Dans notre cas, les équations de Navier–Stokes pénalisées sont résolues en utilisant une méthode spectrale basée sur la transformée de Fourier. Les dérivées spatiales d'une quantité u sont calculées dans l'espace de Fourier, en multipliant ces coefficients de Fourier, $\widehat{u}$, par ik, où k est le nombre d'onde. L'opérateur de Laplace devient une multiplication par $-|k|^2$, ce qui est donc un opérateur diagonal. Par conséquent la résolution numérique d'une équation de Poisson, comme elle apparaît souvent dans les écoulements incompressibles, est une division simple. Ceci est un grand avantage par exemple sur les méthodes de différences finies, où l'opérateur de Laplace est une matrice creuse mais non-diagonale.

Pour basculer entre l'espace physique et celui de Fourier, la transformation rapide de Fourier est utilisée. Cet algorithme a une complexité de $N \log N$. Il est nécessaire de basculer entre les deux espaces car le terme non-linéaire

[2] Traduit littérairement de l'anglais : "Immersed boundary method"
[3] Traduit littérairement de l'anglais : "Volume penalization method"

contient des produits en espace physique, ce qui se transforme dans des produits de convolution dans l'espace de Fourier. Ces derniers sont très coûteux à évaluer, avec une complexité de N^2. En calculant le terme non-linéaire dans l'espace physique nous évitons ces produits de convolution.

Même si la transformée de Fourier rapide (FFT) est plusieurs ordres de grandeur plus efficace que la transformée lente, cela reste une opération coûteuse. Par contre, d'un point de vue algorithmique, la méthode spectrale permet de concentrer la plupart des efforts de calcul dans un seul endroit, en effet la FFT et son inverse. Ce fait simplifie l'optimisation du code numérique. On peut d'ailleurs bénéficier des bibliothèques numériques existantes qui sont très efficaces et consacrées uniquement à la transformée de Fourier. Par conséquent, le calcul numérique peut se faire facilement dans les grands centres de calcul, utilisant les plus grands ordinateurs au niveau mondial. Les résultats présentés dans cette thèse on été obtenus avec de tels machines en utilisant plusieurs milliers de cœurs de calcul.

–

Puisque la transformée de Fourier est périodique par définition et ne tiens pas compte des conditions aux limites, nous devons utiliser la méthode de pénalisation pour simuler l'écoulement autour des obstacles complexes, comme les insectes que l'on analyse dans cette étude. Néanmoins, la méthode de pénalisation volumique introduit un nouveau paramètre, C_η, pouvant être interprété comme une porosité artificielle. Il est intuitivement clair que pour imiter un obstacle solide, C_η doit être "petit". Par contre, l'opérateur de pénalisation, $-(\chi/C_\eta)\underline{u}$, devient plus raide si C_η est plus petit. Cela a des conséquences importantes, car on est obligé de traiter la discrétisation temporelle d'une manière explicite. Par conséquent, le pas de temps Δt doit être inférieur à C_η, d'où le désir de choisir C_η "petit, mais aussi grand que possible". Nous examinons en détail comment choisir C_η pour obtenir des résultats optimaux avec une erreur minimale, et un coût de calcul aussi faible que possible.

Puisque la transformation de Fourier est périodique, nous discutons des techniques d'absorption de sillage. Sans traitement spécifique, les tourbillons générés par un obstacle, dans notre cas un insecte, ne quittent pas le domaine de calcul. Pour compléter le répertoire de méthodes numériques que l'on possède, on développe ainsi une technique d'éponge. Il s'agit d'un terme de pénalisation pour la vorticité $\underline{\omega} = \nabla \times \underline{u}$, plus précisément $-(\chi_{sp}/C_{sp})\underline{\omega}$. Pour l'ajouter à l'équation de Navier–Stokes pénalisée, il faut appliquer l'opérateur de Biot-Savart, ce qui est l'inverse du rotationnel. Par conséquent, le terme devient $-\nabla \times \frac{\left(\frac{\chi_{sp}}{C_{sp}}(\underline{\omega}-\underline{\omega}_0)\right)}{\nabla^2}$. Il faut donc résoudre trois problèmes de Poisson, mais ceci est facilement possible dans l'espace de Fourier, comme nous l'avons

expliqué précédemment. Le terme d'éponge a des avantages dans le cas d'un écoulement extérieur, car il est de divergence nulle et il n'a pas d'impact sur l'écoulement moyen. Il ne faut alors pas savoir en avance à quelle vitesse l'écoulement moyen se déplace, ni de modifier l'équation de Poisson pour la pression.

Les forces intégrales exercées sur l'obstacle peuvent dans notre cas être calculées par intégration sur le volume du corps, $\underline{F} = \int \frac{\chi}{C_\eta} \underline{u} dV$.

La méthode de pénalisation a été proposée d'abord pour des obstacles immobiles. Dans ce cas, la fonction de masque, χ, a été choisie d'une manière discontinue, $\chi = 1$ dans l'obstacle et $\chi = 0$ sinon. Cette fonction de masque n'est pas très appropriée si l'obstacle se déplace ou se déforme, puisqu'elle ne peut être déplacée par un point de grille entier. Ce mouvement saccadé produit des grandes oscillations dérangeant les forces hydrodynamiques. Par conséquent, on introduit une fonction de masque glissé, avec une couche fine de glissement. Dans la limite $\Delta x \to 0$, elle converge vers une fonction caractéristique. Par contre, la couche de glissement permet de la déplacer par des incréments plus petits que Δx, et permet ainsi la simulation des obstacles déformables. En pratique nous introduisons la fonction de distance signée, δ, qui donne pour chaque endroit $\underline{x}$ l'éloignement de l'interface fluide-solide. Cette fonction est à la base de notre algorithme. La fonction de masque est ensuite calculée à partir de la fonction de distance, $\chi(\delta)$.

Tous les aspects mentionnés ci-dessus sont implémentés dans le logiciel `FluSI`, notre code open-source pour la simulation des insectes avec la méthode de pénalisation combinée avec une discrétisation spectrale.

Partie II : Insectes avec des ailes rigides

Dans la prochaine partie nous appliquons notre méthode décrite précédemment à la simulation des insectes avec des ailes rigides.

Nous commençons par décrire le modèle mathématique qui représente un insecte. Ce modèle suppose que les ailes sont des plaques fines et rigides sans profil aérodynamique. Pour une grande gamme d'insectes cette hypothèse est valable. Par contre, dans certains cas, des grandes déformations des ailes ont été observées expérimentalement et il faut donc être conscient de cette limitation du modèle. Dans le cas d'un papillon, par exemple, les déformations peuvent être non-négligeables. Cependant, le problème complet, c'est-à-dire un insecte avec des ailes déformables, est nettement plus difficile à déduire. Il existe très peu de données sur la flexibilité des ailes, et leur structure très

complexe, composée de membranes et barres flexibles, est un champ de recherche entier et peu exploré. Les ailes flexibles sont alors hors de portée de cette thèse.

Nous supposons qu'un insecte est composé d'un corps et de deux ailes. Même si beaucoup d'insectes on effectivement quatre ailes, les ailes se comportent souvent comme une aile effective. Par contre, notre approche permet également d'ajouter plus d'ailes indépendantes, si nécessaire, mais nous allons laisser de côté cette possibilité. Le corps de l'insecte a trois degrés de liberté de rotation et trois de translation. Les deux ailes peuvent tourner autour de l'épaule par trois angles. Par conséquent, un insecte avec des ailes rigides possède 12 degrés de liberté.

Nous décrivons d'abord les transformations linéaires pour changer entre les différents repères, ce qui est effectué par une séquence de matrices orthogonales de rotation. Par exemple, pour passer du repère global (ou de laboratoire) à celui fixé sur le corps, nous utilisons

$$\begin{aligned} \underline{x}^{(b)} &= M_{\text{body}}(\psi,\beta,\gamma)\left(\underline{x}^{(g)} - \underline{x}^{(g)}_{\text{cntr}}\right) \\ M_{\text{body}} &= R_x(\psi)\, R_y(\beta)\, R_z(\gamma), \end{aligned}$$

où $\underline{x}^{(g)}$ est la position dans le repère global et $\underline{x}^{(b)}$ celle dans le repère fixé au corps. Les matrices de rotation $R_x(\psi)$, $R_y(\beta)$ et $R_z(\gamma)$ sont des matrices orthogonales de roulis, tangage et lacet. Des transformations similaires sont utilisées pour passer au repère des ailes.

La cinématique des ailes, c'est-à-dire les trois angles de rotation en fonction du temps, est prescrite par une série de Fourier ou d'Hermite. De cette façon, un mouvement arbitraire de l'aile peut être simulé, et ce mouvement est lu à partir d'un fichier. L'évolution temporelle de ces angles est fournie par des expériences publiées précédemment.

La forme de l'aile est décrite dans son repère en utilisant les coordonnées polaires. Ceci est possible car l'aile est supposée être plate (il s'agit d'un objet à deux dimensions). Cette fonction est aussi périodique et donc à nouveau une série de Fourier est utilisée pour stocker la forme. Comme pour la cinématique, la forme est prélevée des articles publiés.

La puissance aérodynamique est la puissance nécessaire pour déplacer l'aile contre la résistance du fluide, tandis que la puissance inertielle est la puissance requise pour déplacer l'aile dans le vide. Nous discutons la façon de calculer les deux composantes de la puissance. On note d'ailleurs qu'il est très difficile de mesurer la puissance expérimentalement, beaucoup de résultats expérimentaux sont en effet basés sur des modèles simplifiés. Cependant,

notre méthode permet d'obtenir des valeurs précises pour la puissance aérodynamique, au moins dans les limites du modèle, notamment les ailes rigides.

Dans le calcul numérique des insectes, deux situations principales apparaissent souvent : le vol fixe et le vol libre. Dans le premier cas, l'insecte est attaché d'une manière fixe ; il ne peut donc pas bouger. Ceci est utile pour étudier la production de force et la consommation d'énergie en négligeant le problème du contrôle de vol. Dans le cas du vol libre, la position du corps et de l'orientation est calculée à partir des forces aérodynamiques. Il a été démontré que le vol des insectes est instable en soi, c'est-à-dire que l'animal doit contrôler sa position et son orientation d'une manière continue. Sans contrôle actif, l'insecte commence à tourner est s'écrase finalement. Dans un premier temps, on considère le vol fixe. Par contre, les équations pour le vol libre sont également décrites et prêtes à être utiliser.

Après avoir introduit le modèle, nous présentons des cas de test de validation. Ceci est une étape importante pour démontrer que nos résultats sont valables, par contre, il a été dur de trouver des cas de test suffisamment bien décrits pour les reproduire. Nous commençons avec une seule aile pour comparer avec l'expérience de Suzuki et ses collègues. L'aile est un simple rectangle et elle suit un mouvement facile. D'ailleurs, elle est d'épaisseur finie. Nos résultats collent très bien à ceux obtenus par Suzuki et al.

Ensuite, pour conclure la section de validation scientifique, nous procédons à un cas test plus complexe. On considère alors le modèle d'une drosophile, une petite mouche (longueur des ailes ≈ 2.5 mm). Son nombre de Reynolds, basé sur la vitesse moyenne des ailes et l'envergure moyenne, est de $Re = 136$. Ce cas test a été choisi car Maeda et al. ont fourni une description suffisamment précise pour permettre la reproduction de leurs résultats. Dans ce cas, on inclut le corps ainsi que la deuxième aile. De nouveau, nos résultats coïncident avec les résultats obtenus par l'autre équipe, avec une différence de moins de 2%.

Après avoir décrit et validé notre modèle d'insecte, nous allons faire de la recherche originale. L'animal choisi est un bourdon, un insecte nettement plus grand qu'une drosophile (longueur d'aile $\approx$ 13 mm). Par conséquent, son nombre de Reynolds est plus élevé, $Re \approx 2000$. Dans ce cas, la mécanique des fluides est plus compliqué, et le champ de vorticité produit par les ailes présente des vortex de toute taille. Ainsi le sillage peut être considéré comme turbulent. Le bourdon a été choisi car d'autres équipes avait publié des donnés sur la cinématique de ses ailes, ce dont nous avions besoin pour construire notre modèle. Par ailleurs, le bourdon est connu pour voler même dans des conditions difficiles, par exemple en cas de jours très venteux.

L'idée de base de cette partie de la thèse est d'étudier pour la première fois le

vol d'insecte dans un écoulement turbulent. Malgré le fait que tous les insectes volent dans un environnement plein d'obstacles qui génèrent un écoulement turbulent et complexe, presque tous les travaux existants ont considéré l'animal dans un écoulement non-perturbé et laminaire. Spécifiquement, il n'y avait aucune simulation numérique de cette configuration. D'ailleurs, le code décrit et développé précédemment est bien adapté à une telle configuration : puisque la méthode spectrale résout les équations sur un maillage équidistant, le fait d'avoir une distribution homogène des échelles en espace n'augmente pas le coût de calcul. Au contraire, inclure de la turbulence est presque gratuit de ce point de vue.

Puisqu'il s'agit des premières simulations de ce type, il nous faut d'abord un point d'orientation quelle type de turbulence peut être importante pour l'insecte, ainsi que quelle intensité. Dans la nature, l'écoulement généré, par exemple, par un arbre, est à priori non-isotrope et inhomogène. Par contre, dans une certaine distance, la turbulence a tendance à s'homogénéiser. Par conséquent, le modèle de turbulence que l'on considère est celui de la turbulence isotrope et homogène. Pour choisir son intensité, on considère le sillage généré par le bourdon dans un écoulement laminaire. Dans le voisinage proche de l'insecte, on observe des intensités relatives de turbulence de plus de 60%, où on définit l'intensité relative comme $Tu = \left|u'_{RMS}\right| / |u_\infty|$. Dans une distance de cinq fois la longueur d'aile en aval, on trouve toujours une intensité de plus de 16%. Ces intensités de turbulence sont grandes par rapport à celles que l'on considère dans l'ingénierie, où 0.8-2% est appelé "turbulent".

Avec le modèle de bourdon, qui est conçu pour être en équilibre en vol laminaire, et le modèle de turbulence, on peut ensuite passer à l'étude de la combinaison des deux. On conçoit une soufflerie numérique avec un flux d'air turbulent en amont. L'insecte reste fixé dans le repère du laboratoire, ce qui nous permet d'étudier l'impact de la turbulence sur la production des forces uniquement. La question que l'on se pose est de savoir si les perturbations turbulentes peuvent déstabiliser le tourbillon de bord d'attaque. Si tel est le cas, nous pouvons nous attendre à une réduction significative de la production de la force de portance.

Vu que la turbulence est un phénomène intrinsèquement aléatoire, il faut répéter la simulation avec une autre réalisation de turbulence avec les même propriétés statistiques (intensité, nombre de Reynolds, etc.). Ensuite, calculer la moyenne d'ensemble sur les quantités d'intérêt, par exemple la force de portance, permet de tirer des conclusions statistiques.

On varie l'intensité de la turbulence entre 16% et 99%. Dans le dernier cas, les fluctuations de vitesse sont aussi importantes que l'écoulement moyen,

ce qui représente alors un écoulement fortement turbulent. Cependant, les moyennes statistiques des forces, moments et puissance aérodynamique ne changent pas avec l'intensité de turbulence. Évidemment, leur variances augmentent, mais la valeur moyenne reste constante. Ce comportement diffère de celui des profils aérodynamiques. Dans leur cas, des petits perturbations en amont peuvent déclencher des effets transitoires dans la couche limite - par conséquent, les forces moyennes ne sont plus les mêmes que dans le cas laminaire.

Le fait que les forces ne sont pas statistiquement changées dans le cas du bourdon peut être expliqué par le fait que le tourbillon de bord d'attaque n'est pas déstabilisé par l'afflux turbulent. En effet, la visualisation du champ de vorticité, moyennée statistiquement, montre clairement l'existence de ce vortex, même dans les conditions les plus turbulentes considérées ici.

Des études expérimentales ont démontré que des abeilles s'écrasent dans un écoulement suffisamment turbulent. Combiné avec nos résultats, on peut conclure que voler dans une zone turbulente pose un problème de contrôle pour l'animal, plutôt que des difficultés de générer suffisamment de portance pour rester en air. Nous pouvons également faire le lien avec ces expériences en intégrant le moment aérodynamique sur le temps de réaction, et nous confirmons avec nos donnés que le bourdon ne peux pas contrôler son orientation dans la turbulence avec une intensité de 99%.

Partie III : Obstacles flexibles

Jusqu'à présent, les ailes d'insectes ont été considérées comme des plaques rigides. Or, on sait que ceci n'est qu'une approximation ; en réalité, les ailes sont tellement fines qu'elles se déforment durant leurs mouvements. On a déjà discuté que du point de vue mécanique, les ailes sont des structures énormément complexes, et pour les modéliser d'une façon rigoureuse, il faut mesurer leurs propriétés élastiques. Par conséquent, un tel modèle est au-delà de la portée de cette thèse.

Cependant, nous allons faire les premiers pas dans cette direction, et dans la troisième partie de cette thèse, nous passons à des obstacles qui peuvent se déformer sous l'action du fluide. Le but final de ces efforts est de modéliser les insectes complets avec des ailes flexibles en écoulement turbulent.

Nous nous limitons donc à des modèles plus simples dans un premier temps, et nous considérons une plaque qui est flexible dans une direction et rigide dans les autres. Ainsi, elle est décrite par une équation à une dimension ; effectivement, il s'agit d'une barre. D'abord, nous dérivons cette équation.

On suppose que le matériel soit linéaire, c'est-à-dire il est décrit par une loi constitutive linéaire. Par contre, nous admettons des grandes déformations, le modèle est donc géométriquement non-linéaire. Ceci est motivé par le fait que dans les problèmes que l'ont veut considérer, appliqués aux problèmes en biologie, les déformations sans habituellement grandes. Contrairement à cela, les problèmes qui se posent dans l'ingénierie traitent souvent des petites déformations ce qui entraîne une linéarisation. Après avoir dérivé le modèle, nous développons et validons un code numérique pour le résoudre. Ce code est basé sur une discrétisation en différences finies, et le schéma temporel est implicite. Ce type de discrétisation temporel est nécessaire car les équations contiennent un terme du type $\partial_{xxxx} u$. Par conséquent, le équations sont très raides, et un schéma explicite devrait satisfaire une condition du type $\Delta t < C\Delta x^4$. On conclut la description du modèle en comparant avec des résultats fournis dans la littérature, et on confirme sa validité.

Ensuite, nous souhaitons coupler cette équation avec les équations de Navier–Stokes pour simuler des problèmes d'interaction fluide-structure. Cela nécessite de générer la fonction de masque, $\chi(\underline{x}, t)$, ainsi que le champ de vitesse à l'intérieur du solide, $\underline{u}_s$. Pour la fonction de masque, on décrit comment on peut calculer la fonction de distance pour chaque point $\underline{x}$ du maillage eulérien. On propose un algorithme qui est généralisé mais à la fois efficace. La vitesse $\underline{u}$ est interpolé avec la projection de $\underline{x}$ sur la ligne au milieu de la plaque. Avec les deux composantes, χ et $\underline{u}_s$, un peut inclure le solide dans le fluide. Il ne reste alors que le couplage dans l'autre sens.

La structure flexible tient compte du fluide par les forces sur la surface, notamment le tenseur de tensions newtoniens. Il faut alors utiliser une technique d'interpolation pour transférer cette quantité, qui est une quantité eulérienne, sur la barre qui est une structure lagrangienne. On discute quels types d'interpolation peuvent être utilisés pour effectuer ce transfert, et quels problèmes peuvent apparaître.

On a jusqu'à présent décrit les termes de couplage fluide-solide et solide-fluide. Par contre, une partie importante de la solution numérique pour les problèmes d'interaction fluide-structure est le couplage temporel. Les deux champs physiques ont des caractéristiques très différentes. Les équations du modèle solide sont raides et nécessitent une discrétisation temporelle implicite. Le grand nombre d'inconnus pour le fluide exige cependant une discrétisation temporelle explicite. Nous décrivons comment ces différentes discrétisations peuvent être combinées, et les problèmes pour la stabilité qui peuvent se produire. Il est en général plus simple de considérer un solide nettement plus dense que le fluide. Si le rapport de densité s'approche de un, des instabilités se produisent. On peut les stabiliser en performant des itérations.

Un autre aspect important est la conservation de l'énergie ; on souhaite que l'énergie transmise de la structure au fluide soit la même que dans l'autre sens. Autrement dit, on voudrait que la couplage n'introduise pas d'énergie artificielle. On montre que ceci n'est pas le cas dans notre algorithme en considérant un exemple bidimensionnel. Le chapitre est clos avec des tests de validation, qui montre que le code complet donne des résultats valables pour des problèmes de 2D et 3D.

Ensuite, nous appliquons le nouveau code à des problèmes à deux dimensions, avant de passer aux problèmes tridimensionnels.

Le premier scénario traite une configuration archétype des problèmes interaction fluide-structure, celle d'une barre immergé dans un fluide parallèle à la barre. Le comportement physique, c'est-à-dire sa stabilité, est étudié. Le problème est décrit par trois paramètres : le nombre de Reynolds, l'élasticité et la densité relative du solide. En fonction de ces paramètres, trois types de comportement sont possibles. La barre peut être stable, c'est-à-dire elle revient dans sa configuration initiale si elle est perturbée, ou montrer une réponse dynamique. Le cas dynamique est divisé dans un comportement périodique et chaotique. Ce type de problème est aussi lié aux insectes, même si d'une façon plus abstraite. Vu que les ailes sont flexibles, elles peuvent avoir le même comportement, par exemple en vol plané. Il est important que l'aile soit dans le régime stable pour éviter la fatigue du matériau.

Le deuxième problème 2D qu'on étudie est plus directement inspiré par les insectes. On considère deux barres, qui sont censées représenter deux sections d'ailes, par exemple d'une libellule. Ces animaux ont quatre ailes indépendantes. Nos barres suivent un mouvement de soulèvement imposé à leur bord d'attaque. Suite à leur flexibilité, l'angle d'attaque effectif est non-zéro, même si on n'impose pas de rotation active. Par conséquent, les deux barres produisent une force de propulsion. On s'intéresse particulièrement au décalage entre les deux mouvements d'ailes.

Pour conclure, nous considérons les problèmes en trois dimensions spatiales dans le dernier chapitre. La configuration est inspirée par les poissons, qui ont tendance à générer une force propulsive avec leurs nageoires caudales. Des travaux expérimentaux ont montré comment les effets de taille finie peuvent influencer la vitesse de natation, ici nous étudions numériquement l'influence de la forme de la nageoire sur sa performance. Nous trouvons qu'une forme convergente, comme on le voit dans certains nageurs amphibies, surpasse une forme divergente, qui est observée chez les poissons. Nous expliquons ces résultats par la dynamique des fluides.

Part I

Introduction and basic numerical method

Chapter 1

Introduction

The capabilities of animals for flying, swimming and running have fascinated researchers for a long time, and they are of high interest for a large, interdisciplinary community of biologists, engineers, physicists and mathematicians. Many animals have developed, through natural selection, efficient ways of transportation, which are mostly quite different from current human-designed vehicles. For example, animals have legs while cars have wheels, fish swim using their fins and submarines use rotors, and insects flap their wings, unlike fixed-wing or rotary flight in aircraft. In many cases, these 'unusual' solutions may be more efficient, which fosters the interest of understanding and imitating them. Though walking and running are fascinating fields [63], we will focus on swimming and flying, and more precisely on their numerical modeling, for the rest of this thesis.

To give an example, engineers try to understand and copy the aerodynamic strategies employed by insects, in order to build bio-inspired micro air vehicles (MAV), allowing the remote observation of hazardous environments inaccessible to ground vehicles [100]. But also besides such a technological interest, the biological success of this class of invertebrates requires to know their aerodynamic capabilities, which is relevant for example for agriculture or for medicine, as several tropical diseases are transmitted by insects [74]. It is thus important to be able to predict their possible spreading and migration.

Flexibility is an important feature of animal locomotion. While rotor blades are mainly designed to be stiff or almost rigid, fish fins or insect wings can undergo severe deformation without being damaged [32, 91, 152]. Moreover, animals actively exploit the properties of the coupled fluid–solid system to maximize their efficiency [66, 134, 189]. This implies that to study the aero– or hydrodynamics of animal locomotion, the fluid–structure interaction problem can mostly not be neglected, thus coupling the fields of fluid and

solid mechanics [15].

The traditional way to investigate the fluid mechanics of animal locomotion is through observation and experiments. Specifically designed experiments isolate certain factors and reduce the overall complexity of the problem, compared to the actual animal. Insects, for example, have developed sophisticated sensory systems and neural feedback loops that constantly measure impinging inflow and flight conditions, and adapt the wing beat accordingly. This active response to aerial perturbations is thus highly complex and currently poorly understood.

However, the rapid progress in the field of numerical fluid dynamics during the last decades has opened an alternative to the experimental approach: the numerical simulation. This is the method employed in this thesis. Simulations are based on solving the Navier–Stokes equation which govern the fluid [7] using numerical approximations on high performance supercomputers [53]. Though still challenging, numerical simulations of the fluid dynamics of animal locomotion offers unique opportunities. This is not limited to the availability of the entire flow field, possibly down to the smallest vortex, in a quality unobtainable experimentally. Moreover, simulations allow to easily vary the parameters of a system beyond natural limits, which helps identifying the influence of these parameters. The fluid viscosity is such a parameter; it is difficult to vary experimentally, for example when studying insect flight in air. Furthermore, simulations allow turning off steering mechanisms and studying the flight without the active control system. In combination with experiments, this allows extracting the animal's active response to aerial perturbations, which is of major interest likewise for biology and engineering.

The present thesis is placed in this interdisciplinary field of nature-inspired fluid dynamics, and the focus lies on numerical simulations thereof. This naturally involves advancing numerical tools and algorithms, notably the two–dimensional numerical method for fixed obstacles presented in [144], which has later been extended to three-dimensional as well [86]. This method, which is the starting point of the thesis, allows studying moving, rigid obstacles, but could not be applied to deforming ones. Therefore, a new approach, based on the signed distance function, has been developed. This algorithm also proved beneficial for rigid obstacles, due to a significant reduction in computing time while increasing both versatility and precision. It is therefore used exclusively in the present thesis. Being the common ground for both rigid and flexible obstacles, the basic numerical method is presented first, in chapter 2.

Due to their differences in physics and applications, the remaining part of this work is divided into two major parts, dealing with rigid and flexible obstacles, respectively. Part II thus deals with rigid obstacles exclusively, and the focus of the applications lies on flapping insects. Like previous studies on flapping

insect flight, we thus assume that the wings operate as rigid, non–deformable, flat plates [29, 97, 158, 174, 182, 188]. The employed numerical approach allows studying arbitrarily complex geometries, thus our virtual insects can include wings, body, head and even legs and antennae. The model is described in detail in chapter 3, including detailed validation tests proving its correctness. We then proceed and exploit another advantage of our method by studying a bumblebee in forward flight through fully developed turbulence (chapter 4). These simulations are the first of their kind and show that basic aerodynamic mechanisms, like the leading edge vortex, are not vitally harmed by turbulent perturbations. We conclude that turbulence faces insects with an increased demand for control, and not a systematically deteriorated force production. This is different from airfoil-based flight at low-Reynolds numbers, as the transition of the boundary layer from laminar to turbulent can drastically alter the force production [117].

The long-term goal of our research is to study insects with flexible wings in turbulent flow. Our basic numerical method allows simulating these fluid-structure interaction problems as well, and we begin part III in chapter 5 with specific ingredients unused in the previously considered rigid models. Fluid-structure interaction problems naturally require a suitable solid model. As the solid mechanics of real insect wings are highly complex [22, 23], a model fully describing them is beyond the scope of this thesis. Instead, we rely on a one–dimensional model, which we use to study two– and three–dimensional problems. In the latter case, our solids are assumed rigid in one direction. The coupled fluid–solid system is studied in chapter 6 in a two–dimensional setting. We consider first the canonical problem of the fluttering instability, from which we proceed to study the thrust generation by flexible heaving foils. The latter configuration is inspired by a 'mechanical insect' designed to study the coupling between the flapping frequency and the structural resonant frequency [134].

In chapter 7, we present three–dimensional simulations of fluid–structure interaction problems. The application chosen is the thrust production in swimming fish. Therefore, simplified 'swimmers' are devised, consisting of a flexible plate with a driven pitching motion. This setup has also been used experimentally [135]. Many contributions in this field investigate the role of structural resonances, and how animals exploit these to maximize their efficiency [28, 135, 186]. Our contribution instead focuses on the influence of the swimmers shape, since actual fish obviously differ from the rectangular plates. Our surprising finding is that a contracting shape, as observed in some amphibians, swims faster and consumes less power than an expanding shape, which is similar to the caudal fins in many fish. We explain this finding by a favorable impact on the tip vortices which are due to finite-size effects.

Finally, conclusions on the developed method and the obtained results are

given, and some perspectives for future work are discussed.

Chapter 2

Basic numerical method

The content of this chapter is adopted from [44], [46] and [45].

The numerical simulation of bio-inspired fluid dynamics has become an important cornerstone in understanding aerial and aquatic bio-locomotion. While experiments are most reliable in terms of realism, it is difficult to precisely control the input parameters, and some quantities, like the flight forces, are subject to appreciable uncertainty, if available at all. Experiments with living animals account for the whole complexity, that is, the generation of hydrodynamic forces, measurements by the animal's sensory system and corresponding kinematic response, in addition to the difficulty to encourage animals to fly or swim under laboratory conditions, preferably in a repetitive fashion. Insects, for example, often refuse to fly in a smokey environment, which renders laser measurements of the flow difficult. Numerical experiments on virtual insects or fish allow studying non-controlled locomotion with prescribed motion patterns, which in combination with experiments allows drawing conclusions on the role of active steering and stabilization mechanisms.

Fluid–structure interaction (FSI) problems involve, as the name suggests, the interaction of fluid with a solid body. The problems can be divided into *rigid* and *flexible* obstacles, and whether the deformation is *actively* computed from the fluid forces, i.e., the coupled problem is solved, or *passively* imposed, e.g., measured externally and then prescribed in the simulation.

For the numerical solution of these problems, an abundant variety of methods and strategies exist, see for example the books [15, 16]. A broad classification can be made based on the grid used for discretization, depending on whether or not it is adapted to the problem's geometry. In the former, classical category, the computational grid is fitted to the boundary. It is used, for example, to simulate insects with rigid flapping wings [96, 97]. For flexible

obstacles with active FSI, Gomes et al. [67] used a structured, partitioned mesh that matches the interface of a flexible appendix in the wake of a cylinder, and studied the flow-induced vibrations both numerically and experimentally. A general difficulty of this family of methods is the generation of suitable meshed, a question which has been addressed for example by Wick [177] for structured, moving meshes or by Bathe et al. [8] based on unstructured grids. In fact, the grid generation alone is a challenging problem [162] in its own. However, these methods have been used successfully for a wide range of active FSI problems, e.g., [31, 165, 166].

The difficulty of grid generation fueled the development of methods that do not have this requirement, most prominently the immersed boundary method, originally proposed by Peskin to compute the flow in the human heart [130]. Nowadays, a large variety of variants of the immersed boundary method have been developed, see e.g., the reviews [112, 131]. A popular variant is the direct forcing method, which essentially tries to impose the surface forcing at one time step to reach exactly the desired boundary condition at the next instant [167].

In this work, we employ the volume penalization method to take the boundary condition into account. The idea of modeling solid obstacles as permeable media with a small permeability has first been proposed by Arquis and Caltagirone [5]. The forcing term acts on the entire volume of the solid and not only on its surface, as it is the case in the immersed boundary methods, and corresponds to the Darcy drag. The method's distinctive feature is the existence of a rigorous convergence proof by Angot et al. [4], which has later been refined [18]. It has been extended to model not only Dirichlet conditions in moving, rigid obstacles [87], but also homogeneous Neumann conditions, which is relevant for studying the turbulent mixing of a passive scalar [80]. As animals forage following odor traces, this technique can potentially be attractive for insect flight simulations as well. Variants of the penalization method have been developed for compressible flow [14, 98, 148] and magneto-hydrodynamics [114, 115]. An interesting recent development has been proposed by Introini et al. [10, 75], the core idea being a modified projection scheme in a finite-differences context. Then, the Neumann boundary condition proposed by Kadoch et al. [80] appears for the pressure. Another variant, specifically adopted to impulsively started flow is the iterative penalization method proposed by Hejlesen et al. [70].

In the context of biolocomotion, the penalization method has been used to study the C–start of a fish at rest by Gazzolla et al. [61], which showed, using a genetic algorithm, that the observed, natural starting behavior is the best solution for this type of problem. A main advantage of the penalization method is its simplicity in implementation and computation, which is a crucial perquisite for the usage of genetic algorithms. It is also straightforward to

study several animals, for example the collective behavior of fish schools [60].

The remainder of this chapter is organized as follows. After a short discussion about the assumption of incompressible fluid, we introduce the penalized version of the incompressible Navier–Stokes equations in section 2.1. The volume penalization method is discussed in some detail and extended to arbitrarily moving and deforming obstacles in section 2.3, and we briefly revisit the governing equations in two spatial dimensions in section 2.2. Ways to compute hydrodynamic forces and moments are discussed in section 2.4, as well as imposing outflow boundary conditions in section 2.5. The discretization used in this work is summarized in sections 2.7-2.9. As this works deals with rigid and flexible obstacles, we present in this chapter only their common ground, and discuss specific details, for example the numerical fluid–solid coupling, in the respective chapters.

2.1 Model equations and penalization method

This work deals with fluid–structure interaction problems in biological contexts, namely flying insects and swimming fish. In those problems, the fluid can typically be approximated as incompressible. Batchelor [7, pp. 167] states five conditions required for this assumption to be valid, of which the following two apply to the present work:

$$\frac{U^2}{c^2} \ll 1 \quad \frac{f^2 L^2}{c^2} \ll 1,$$

where c is the speed of sound and U, f, L are relevant scales for velocity, frequency and length, respectively. The first condition is the well known statement of small Mach numbers $Ma = U/c$. For example, we find for the Mach number based on the wingtip velocity of a fruitfly (FF) and a bumblebee (BB), both flying in air, to be $Ma_{FF} = 0.0075$ and $Ma_{BB} = 0.024$, respectively. The Mach numbers are thus appreciably smaller than $Ma = 0.3$, which is the limit often used in engineering. The second condition represents the unsteadiness of the flow, and all phenomena studied here are essentially unsteady. Again considering a fruitfly and a bumblebee for illustration, we find the Strouhal number $St = fL/c$ to be $St_{FF} = 0.016$ and $St_{BB} = 0.0057$. For these insects both conditions are thus met, and this applies to virtually all flying insects, justifying the incompressibility assumption. The same is true for swimming in water. The fluid is thus governed by the incompressible Navier–Stokes equa-

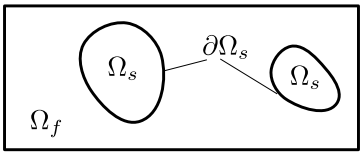


Figure 2.1.1 – Domain definitions. Solid obstacles Ω_s immersed in a fluid domain Ω_f.

tions

$$\partial_t \underline{u} + \underline{\omega} \times \underline{u} = -\nabla\Pi + \nu\nabla^2\underline{u} + \underline{F}_p \tag{2.1.1a}$$

$$\nabla \cdot \underline{u} = 0 \tag{2.1.1b}$$

$$\underline{u}|_{\partial\Omega_s(s,t)}(s,t) = \underline{u}_s(s,t) \tag{2.1.1c}$$

$$\underline{u}(\underline{x}, t=0) = \underline{u}_0(\underline{x}) \tag{2.1.1d}$$

where $\underline{u}$ is the fluid velocity, $\underline{\omega} = \nabla \times \underline{u}$ is the vorticity, ν is the kinematic viscosity, the fluid density ϱ_f is normalized to unity and $\underline{F}_p$ is an external force, usually a pressure gradient. These equations are formulated in dimensionless units, using the scales L, T, $U = L/T$ and M for length, time, velocity and mass, respectively. As the fluid density is always normalized to unity, the mass scale is given by $M = \varrho_f L^3$. The nonlinear term in eqn. (2.1.1a) is written in the rotational version, thus one is left with the gradient of the total pressure $\Pi = p + \frac{1}{2}\underline{u} \cdot \underline{u}$ instead of the static pressure p [132]. This formulation is chosen because of its favorable properties when discretized with spectral methods, namely conservation of momentum and energy [132, pp. 210]. A sketch of the problem is given in figure 2.1.1. The no–slip boundary condition, eqn. (2.1.1c), is satisfied on the fluid–solid interface $\partial\Omega_s$. The boundary condition is inhomogeneous since moving and deforming obstacles are considered.

Equations (2.1.1a-2.1.1d) can, in general, not be solved analytically. It is thus common to employ numerical approximations, however, eqns. (2.1.1a-2.1.1d) pose two major challenges. The most severe difficulty is the fact that the pressure p (or, equivalently, Π) is not a thermodynamic variable satisfying an equation of state, but rather a Lagrangian multiplier that ensures the divergence-free condition (2.1.1b). It adapts instantaneously and not with a finite speed of sound [17]. In addition, the boundary condition (2.1.1c) must be satisfied on a possibly complicated, moving and deforming fluid–solid interface. Traditional numerical approaches rely on boundary-fitted grids to cope with the latter and fractional step methods [82], requiring to solve a Poisson problem for the pressure, to cope with the former difficulty.

Since the generation of boundary-fitted grids for complicated geometries is

a challenging problem on its own [162], which possibly has to be repeated at every time step, alternatives have been developed. The principal idea is to extend the computational domain to the interior of obstacles as well, i.e., $\Omega = \Omega_f \bigcup \Omega_s$ in figure 2.1.1. The boundary is then taken into account by adding supplementary terms to equation (2.1.1a). A variety of methods in this spirit has been developed, starting from Peskin's immersed boundary method [130, 131]. The approach chosen here is the volume penalization method, which is physically motivated by replacing the solid obstacle by a permeable medium with small permeability C_η. The penalized problem reads

$$\partial_t \underline{u} + \underline{\omega} \times \underline{u} = -\nabla \Pi + \nu \nabla^2 \underline{u} + F_p - \frac{\chi}{C_\eta} (\underline{u} - \underline{u}_s) \tag{2.1.2a}$$

$$\nabla \cdot \underline{u} = 0 \tag{2.1.2b}$$

$$\underline{u}(\underline{x}, t = 0) = \underline{u}_0(\underline{x}) \tag{2.1.2c}$$

Its distinctive feature is the existence of a rigorous convergence proof [4, 18] which shows that the solution of the penalized Navier–Stokes equations (2.1.2a-2.1.2c) tends for $C_\eta \to 0$ indeed towards the exact solution of the Navier–Stokes equation imposing no-slip boundary conditions (2.1.1a-2.1.1d), with a convergence rate of $\mathcal{O}\left(\sqrt{C_\eta}\right)$. The parameter C_η should thus be chosen to a small enough value. The mask function χ is defined as

$$\chi(\underline{x}, t) = \begin{cases} 0 & \text{if } \underline{x} \in \Omega_f \\ 1 & \text{if } \underline{x} \in \Omega_s \end{cases}. \tag{2.1.3}$$

In anticipation of the application to moving boundaries, we note that we will replace the discontinuous χ-function by a smoothed one, with a thin smoothing layer centered around the interface. The mask function encodes all geometric information of the problem, and eqns. (2.1.2a-2.1.2c) do not include boundary conditions; at the outer boundary $\partial\Omega$, periodicity is assumed.

Based on the physical interpretation of C_η as permeability, it is intuitively clear that C_η must be set small enough for the penalization method to yield reasonable results which is reflected by the convergence rate of $\mathcal{O}\left(\sqrt{C_\eta}\right)$. However, the choice of C_η is subject to constraints, as the penalized equations are discretized and solved numerically. The modeling error of order $\mathcal{O}\left(\sqrt{C_\eta}\right)$ should be of the same order as the discretization error [122]. It is first noted that in eqn. (2.1.2a), C_η has the dimension of a time. It is instructive to put the nonlinear, viscous and pressure terms aside for a moment. One is then left with $\partial_t u = -u/C_\eta$ inside the solid, with the obvious solution $u = u_0 \exp\left(-t/C_\eta\right)$. Thus, C_η can be directly identified as the relaxation time. Interfering with the time step Δt, usually implying $\Delta t / C_\eta < \text{const}$, where the constant depends on the time marching scheme, this simple fact has important

consequences for the numerical solution. It indicates that a good choice for C_η is not only "small enough", but also "as large as possible". Further insight in the properties of the penalization method can be obtained considering a laminar channel flow (Poisseuille flow) in its penalized version, yielding

$$\begin{aligned} -\partial_x p + \nu \partial_{yy} u &= 0 \qquad \text{in } \Omega_f \\ -\partial_x p + \nu \partial_{yy} u - \frac{u}{C_\eta} &= 0 \qquad \text{in } \Omega_s. \end{aligned}$$

A sketch of this setup is given in 2.1.2 (left). Assuming $\partial_x p = P = \text{const}$, we look for a solution with continuous velocity and derivatives at the fluid–solid interface. In the solid domain, the solution is

$$u(y) = P\,\delta_\eta \frac{\left(a\left(e^{y/\delta_\eta} + e^{-y/\delta_\eta}\right) + \delta\left(e^{a/\delta_\eta} - e^{-a/\delta_\eta}\right)\right)}{e^{a/\delta_\eta} - e^{-a/\delta_\eta}}$$

where $\delta_\eta = \sqrt{\nu C_\eta}$ has been introduced. The velocity on the wall is $u(y = a) \approx 2\delta$. The solution of the problem is illustrated in figure 2.1.2 (right). It shows the solution for two values of C_η. A boundary layer of thickness δ_η forms inside the solid region Ω_s, which has also been pointed out in [18], and a small slip velocity is found right on the channel walls, which has been demonstrated in greater detail in [122]. At this stage, it should be noted that the solution of the Poisseuille flow is in fact the solution of a steady heat equation problem. Thus, a more physical interpretation can be given in this case. The viscous term, active in both Ω_s and Ω_f, imposes a viscous flux over the boundary $\partial\Omega_s$, which is rapidly consumed by the penalization term, which acts as a sink. The competition between viscous transport and the penalization damping determines the boundary layer thickness. This finding also illustrates that penalization works better for higher Reynolds numbers, and that it will require very small values of C_η in the Stokes regime. It should however be pointed out that in realistic flow configurations, the thickness of the penalization boundary layer is not solely determined by the viscosity, but also depends on the pressure and the employed projection scheme.

When increasing the resolution, the number of points per thickness $K = \delta_\eta / \Delta x$ should be kept constant, which implies slaving $C_\eta \propto (\Delta x)^2$, consistent with [122], where the penalized Laplace and Stokes operators were analyzed analytically. With the scaling for C_η, one still has to choose the constant K. In fact, for any value of K, the method will converge with the same convergence order, but the error offset can be tuned [46]. As this tuning is discretization specific, it will be discussed later.

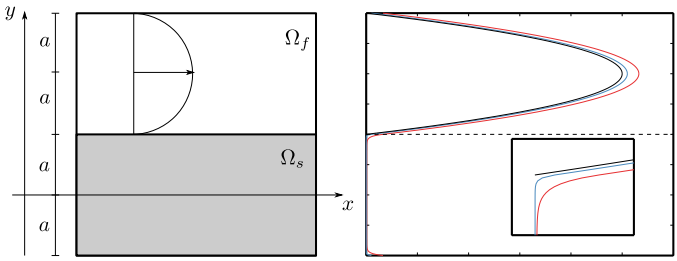


Figure 2.1.2 – Penalized Poisseuille flow. Domain definitions (left) and exact solution (right) for a large (red) and a small (blue) value of C_η. The inset shows a zoom to illustrate the penalization boundary layer. Units are arbitrary.

2.2 Model equations in two dimensions

Simulations in a two-dimensional space are orders of magnitude faster to compute, and still they can give valuable insight in some phenomena. In the 2D case, the curl of $\underline{u}$, the vorticity, has only one non-vanishing component, it is thus a scalar. Taking the curl of eqn (2.1.2a) yields the vorticity transport equation

$$\partial_t \omega + (\underline{u} \cdot \nabla)\,\omega \quad = \quad \nu \nabla^2 \omega - \nabla \times \left(\frac{\chi}{C_\eta} (\underline{u} - \underline{u}_s) \right). \tag{2.2.1}$$

The velocity can be determined as $\underline{u} = \nabla^\perp \psi$, introducing the streamfunction ψ, which satisfies $\nabla^2 \psi = \omega$, and $\nabla^\perp = \left(-\partial_y, \partial_x\right)^T$ as the orthogonal gradient. The velocity $\underline{u}$ is defined up to an irrotational contribution that can be forced independently, i.e., $\underline{u} = \nabla^\perp \psi + \underline{U}$. Since the only periodic potential flow is a constant velocity, $\underline{U} = \underline{u}_\infty$ can be interpreted as the mean velocity. It should be noted that the pressure gradient $\nabla \Pi$ drops out of eqn. (2.1.2a) as its curl is identically zero.

2.3 Penalization method for moving and flexible obstacles

The volume penalization method as discussed so far assumes a discontinuous mask function in the form of eqn. (2.1.3). When applying the method to a non-grid aligned body, for example a circular cylinder, the mask function exhibits the "staircase" effect, which geometrically approximates the boundary to first order in Δx. Results for stationary cylinder using the discontinuous

mask function are acceptable [144, 146], but spurious oscillations in the case of moving ones are reported [87]. The reason is that the discontinuous mask can be translated only by integer multiples of the grid spacing, and this jerky motion causes large oscillations in the hydrodynamic forces. Kolomenskiy and Schneider [87] proposed an algorithm to shift the mask function in Fourier space instead of physical space. To see how this technique basically works, and why we cannot use it in the present work, consider a pure translation motion. Basically, the approach consists in solving a transport equation for the mask function,

$$\partial_t \chi + \underline{u}_s \cdot \nabla \chi = 0\,, \tag{2.3.1}$$

using the same discretization as for the Navier–Stokes equation. The gradient operator being diagonal in Fourier space, this equation can be solved exactly. Thus, the obstacle at time t^n is described by

$$\chi\left(\underline{x}, t^n\right) = \mathcal{F}^{-1}\left(e^{-i\underline{k}\cdot\underline{u}_s t}\mathcal{F}\left(\chi\left(x, t^0\right)\right)\right). \tag{2.3.2}$$

Using equation (2.3.2) the obstacle can be displaced by less than one grid point. Always using the initial mask, $\chi\left(x, t^0\right)$, avoids error accumulation. For general u_s, including arbitrary rotations, the motion can be decomposed in a sequence of one dimensional transformations [87]. Hence, arbitrary motion can be simulated with this approach. In order to avoid Gibbs oscillations, the discontinuous χ-function can be smoothed by solving a few steps of a diffusion equation prior to the simulation.

In the present work, we employ a different approach, for two reasons. First, as visible in eqn. (2.3.2), displacing the mask in Fourier space involves (additional) Fourier transforms, which are computationally expensive. It can also only be applied on equidistant grids, as required by the Fourier transform. Second, and more important, the Fourier approach cannot be used for flexible obstacles, since the diagonality of the transport operator is lost.

The idea in this work is to directly assume the mask function to be smoothed over a thin smoothing layer, which has also been used in [87], but for a different reason. To this end, we introduce the signed distance function $\delta(\underline{x}, t)$ [127], which in the case of a cylinder of radius R_0 centered around $\underline{x}_0(t)$ simply reads

$$\delta\left(\underline{x}, t\right) = \left\|\underline{x} - \underline{x}_0\left(t\right)\right\|_2 - R_0.$$

Then, the mask function can be computed from the signed distance, for example using

$$\chi(\delta) = \begin{cases} 1 & \delta \leq -h \\ \frac{1}{2}\left(1 + \cos\left(\pi\frac{\delta+h}{2h}\right)\right) & -h < \delta < +h \\ 0 & \delta > +h \end{cases} \tag{2.3.3}$$

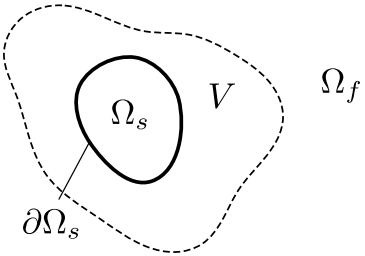


Figure 2.4.1 – Computation of hydrodynamic force.

where the semi-thickness of the smoothing layer, h, is used. It is typically defined relative to the grid size, $h = C_{\text{smth}}\Delta x$, thus eqn. (2.3.3) converges to a Heaviside step function as $\Delta x \to 0$. Nonetheless, it can be translated by less than one grid point, and then be resampled on the Eulerian fluid grid. Eqn. (2.3.3) is not an unique choice, in fact, it is rather arbitrary. In [44, 46] we used

$$\chi(\delta) = \frac{1}{2}\left(\text{erf}\left(-\frac{\delta}{h}\right) + \text{erf}\left(\frac{\delta}{h}\right)\right) \tag{2.3.4}$$

with $h = C_{\text{smth}}\Delta x \ \max(\partial\chi/\partial\delta) = \frac{C_{smth}\Delta x}{\sqrt{\pi}}(\exp(-4) - 1)$ instead. This choice is motivated by the fact that the Fourier coefficients of (2.3.4) decay faster, representing thus a better compromise of support width in physical and Fourier space. However, the solution of eqns. (2.1.2a-2.1.2c) is in C^1 at best, which is the reason why eqn. (2.3.3) turns out to be sufficient.

2.4 Hydrodynamic forces and moments

When simulating fluid–structure interaction problems, one is usually interested in the hydrodynamic forces (e.g., drag and lift) and moments (i.e., yaw, pitch, roll). It is noted that the penalization term in eqn. (2.1.2a) has the dimension of a force, still, the intuition of simply integrating it over the obstacle, as done in [4], is correct only if the obstacle is at rest or in steady motion. In the case of arbitrarily moving obstacles, additional terms occur, as first pointed out in [167]. The purpose of this section is to give a clear derivation of the expressions for the force and moment. The force $\underline{F}$ on an obstacle is defined as the surface integral of the stress tensor over the obstacle's surface:

$$\underline{F} = \oint_{\partial\Omega_s} \sigma \cdot \underline{n}\, \mathrm{d}\gamma. \tag{2.4.1}$$

$$\underline{\underline{\sigma}} = -\nabla p + \frac{\nu}{2}\left(\nabla u + (\nabla u)^T\right) \tag{2.4.2}$$

It can, in principle, be computed using eqn. (2.4.1). However, if one is interested in the integral forces only, as opposed to the local distribution, volume integration can be employed, which is computationally more efficient than surface integration. We consider a control volume V, that may or may not coincide with an obstacle, as illustrated in figure 2.4.1. This way, we include both the strategy of integrating only over Ω_s employed by Uhlmann [167] and the idea of integrating a volume surrounding, but excluding the obstacle, as proposed by Bergmann and Iollo [11]. It is useful to rewrite eqn. (2.1.2a) with the stress tensor $\underline{\underline{\sigma}}$ and the non-linear term in convection form, yielding

$$\partial_t \underline{u} + \underline{u} \cdot \nabla \underline{u} = \nabla \cdot \underline{\underline{\sigma}} - \frac{\chi}{C_\eta} (\underline{u} - \underline{u}_s) \tag{2.4.3}$$

Integrating (2.4.3) over V yields

$$\int_V (\partial_t \underline{u} + \underline{u} \cdot \nabla \underline{u}) \, \mathrm{d}V = \int_V \nabla \cdot \underline{\underline{\sigma}} \, \mathrm{d}V - \int_V \frac{\chi}{C_\eta} (\underline{u} - \underline{u}_s) \, \mathrm{d}V,$$

and applying Gauss' theorem gives the expression for the force acting on the control volume V

$$\underline{F} = \oint_{\partial V} \sigma \cdot \underline{n} \, \mathrm{d}\gamma = \int_V \frac{\chi}{C_\eta} (\underline{u} - \underline{u}_s) \, \mathrm{d}V + \int_V (\partial_t \underline{u} + \underline{u} \cdot \nabla \underline{u}) \, \mathrm{d}V. \tag{2.4.4}$$

If we now consider $V = \Omega_s$, then $\underline{F}$ is directly the force acting on the obstacle. The second term on the left hand side is labeled unsteady correction. Rewriting it with the material derivative yields the same form as in [167]:

$$\underline{F} = \int_{\Omega_s} \frac{\chi}{C_\eta} (\underline{u} - \underline{u}_s) \, \mathrm{d}V + \int_{\Omega_s} \frac{\mathrm{D}u}{\mathrm{D}t} \mathrm{d}V$$

The velocity inside the obstacle is equal to $\underline{u}_s$ in a very good approximation, and thus the final expression for the force is

$$\underline{F} = \int_{\Omega_s} \frac{\chi}{C_\eta} (\underline{u} - \underline{u}_s) \, \mathrm{d}V + \frac{\mathrm{d}}{\mathrm{d}t} \int_{\Omega_s} \underline{u}_s \mathrm{d}V. \tag{2.4.5}$$

Eqn (2.4.5) is the one used in the present work. The unsteady correction can, for rigid bodies, be computed as $V_s \dot{\underline{u}}_s$, as described in [167], where V_s is the obstacle's volume and $\dot{\underline{u}}_s$ its acceleration. For flexible obstacles, the integral is evaluated numerically. If the obstacle is at rest, eqn (2.4.4) reduces to $\underline{F} = \int_V \frac{\chi}{\eta} \underline{u} \mathrm{d}V$, which is the classical formula given by Angot et al. [4]. For completeness, we also consider V to only surround the obstacle, thus excluding Ω_s [11]. The surface integral in eqn (2.4.4) consists then of two parts, and $\chi \equiv 0$:

$$\oint_{\partial \Omega_s} \sigma \cdot \underline{n} \, \mathrm{d}\gamma = \oint_{\partial V} \sigma \cdot \underline{n} \, \mathrm{d}\gamma - \int_V (\partial_t \underline{u} + \underline{u} \cdot \nabla \underline{u}) \, \mathrm{d}V$$

where the remaining surface integral on the left hand side is evaluated on a rectangle coinciding with the grid.

The situation is similar for the torque acting on the body. It can be computed using

$$\underline{m} = \oint_{\partial V} \underline{r} \times (\sigma \cdot \underline{n}) \, \mathrm{d}\gamma = \int_V \underline{r} \times (\partial_t \underline{u} + \underline{u} \cdot \nabla \underline{u}) \, \mathrm{d}V + \int_V \underline{r} \times \frac{\chi}{C_\eta} (\underline{u} - \underline{u}_s) \, \mathrm{d}V \tag{2.4.6}$$

where $\underline{r} = \underline{x} - \underline{x}_c$ is the position vector relative to the point of reference. Again the unsteady correction appears, which reduces to

$$\int_V \underline{r} \times (\partial_t \underline{u} + \underline{u} \cdot \nabla \underline{u}) \, \mathrm{d}V = J_c \ddot{\Theta}_c$$

in the case of rigid objects. Therein, J_c is the geometrical moment of inertia and Θ is the angle of rotation.

2.5 Wake removal techniques

The penalized Navier–Stokes eqns (2.1.2a-2.1.2c) do, by principle, not require boundary conditions. It is however possible to supplement them with in- and outflow boundary conditions, and model only the immersed obstacle with penalization. It may seem surprising, however, that suitable open boundary conditions for incompressible flows are unknown to this day. For example, the open boundary conditions mini-symposium in 1991 is reported to have been "an exercise in frustration" [27, 141]. The general difficulty is that domain truncation is essentially nonphysical, and that applies even to compressible flows, where characteristics can be used to model outflow conditions [149]. An intuitive example where posing an outflow condition is difficult is the case of a traveling vortex pair. As soon as the first vortex disappears, i.e., it crosses the outflow, the second one looses the velocity induced by its partner. Outflow conditions thus require modeling to some extend the structures outside the computational domain. In this spirit, Gautier et al, for example, simulated the flow past a cylinder at $Re = 40$ with an immersed boundary method, and imposed the potential flow solution as far-field inflow condition [59].

Here, we follow an alternative approach, and employ a periodic discretization of eqns. (2.1.2a-2.1.2c). In such a periodic setting, the wake re-enters the domain, which is an undesired artifact. To overcome it, a supplementary sponge (sp) penalization term can be added to the vorticity-velocity eqn. (2.2.1), to gradually damps the vorticity [44, 46]. The equation then reads

$$\partial_t \omega + (\underline{u} \cdot \nabla) \, \omega = \nu \nabla^2 \omega - \nabla \times \left(\frac{\chi}{C_\eta} (\underline{u} - \underline{u}_s) \right) - \frac{\chi_{sp}}{C_{sp}} (\omega - \omega_0) \, . \tag{2.5.1}$$

The desired vorticity profile ω_0 is usually zero. The sponge penalization parameter C_{sp} is usually set to a much larger value than C_η, typically $C_{sp} = 10^{-1}$. The larger value and its longer relaxation time ensure that if a traveling vortex pair enters the sponge region, the leading one is not dissipated too fast, because it otherwise could leave the partner orphaned in the domain. In primitive variables and three spatial dimensions, we find for the same term

$$\partial_t \underline{u} + \underline{\omega} \times \underline{u} = -\nabla \Pi + \nu \nabla^2 \underline{u} - \frac{\chi}{C_\eta} (\underline{u} - \underline{u}_s) - \nabla \times \frac{\left(\frac{\chi_{sp}}{C_{sp}} (\underline{\omega} - \underline{\omega}_0) \right)}{\nabla^2}, \quad (2.5.2)$$

since $\nabla \cdot \underline{u} = 0$, $\underline{u}$ takes the form $\underline{u} = \nabla \times \underline{\psi}$ and the vorticity is $\underline{\omega} = \nabla \times \underline{u} = \nabla \times \nabla \times \underline{\psi} = -\nabla^2 \underline{\psi}$, thus for each component ψ_i of the streamfunction $\underline{\psi}$, we solve a Poisson equation $\nabla^2 \psi_i = \omega_i$, and afterwards, the curl of the streamfunction yields the velocity. By construction the sponge term is divergence-free, which is important since it otherwise would contribute to the pressure, which in turn would be modified even in regions far away from the sponge. Moreover, it leaves the mean flow untouched. This technique is adapted to spectral discretizations, since the solution of 3 Poisson problems would be prohibitively expensive in a finite-difference setting.

Dirichlet conditions on the velocity can be imposed directly with the volume penalization, which applies for example to channel walls. For simulations in a uniform, unbounded free-stream, we use both techniques; in a small layer at the domain borders, the Dirichlet condition $\underline{u} = \underline{u}_\infty$ is imposed with the same precision as the actual obstacle, and a preceding, thicker vorticity-sponge layer ensures that the upstream influence is minimized [46]. The sponge technique is similar to the "fringe regions" proposed by Schlatter et al [143]. However, they used a velocity sponge, which lacks the favorable properties of the vorticity sponge described here.

2.6 Mean flow equations

The mean flow, that is the zeroth Fourier mode, has a special role in our framework. In the vorticity streamfunction formulation, see section 2.2, the mean flow has to explicitly added, since the inversion of the Poisson eqn $\nabla^2 \psi = \omega$ is only defined up to a constant. The mean flow $\underline{u}_\infty$ can be forced independently, for example to a constant value.

In primitive variables, eqn. (2.1.2a), we can derive an ordinary differential equation for $\underline{u}_\infty$ by spatially averaging equation (2.1.2a) (i.e., we integrate

over the torus $\mathbb{T}^3$),

$$\int_{\mathbb{T}^3} \partial_t \underline{u}\, \mathrm{d}\underline{x} + \underbrace{\int_{\mathbb{T}^3} \underline{\omega} \times \underline{u}\, \mathrm{d}\underline{x}}_{=0} = \underline{F}_p - \underbrace{\int_{\mathbb{T}^3} \nabla q\, \mathrm{d}\underline{x}}_{=0} + \nu \underbrace{\int_{\mathbb{T}^3} \nabla^2 \underline{u}\, \mathrm{d}\underline{x}}_{=0} - \int_{\mathbb{T}^3} \frac{\chi}{C_\eta} (\underline{u} - \underline{u}_s)\, \mathrm{d}\underline{x} \tag{2.6.1}$$

$$\frac{\mathrm{d}\underline{u}_\infty}{\mathrm{d}t} = \underline{F}_p - \int_V \frac{\chi}{C_\eta} (u - u_s)\, \mathrm{d}\underline{x}. \tag{2.6.2}$$

in the absence of an external pressure gradient $\underline{F}_p$, the mean flow acceleration is thus determined by the drag force. If we set $\underline{u}_\infty$ to a fixed value, we thus imply an external pressure gradient that always compensates the drag force [46].

When looking for a steady cruising speed, achieved for example by a swimmer model, it is useful to modify eqn. (2.6.2), to take into account a fixed fluid mass. The equation then becomes

$$\frac{\mathrm{d}\underline{u}_\infty}{\mathrm{d}t} = \frac{\left\langle -\frac{\chi}{C_\eta} (u - u_s) \right\rangle}{m_{\text{fluid}}} \tag{2.6.3}$$

where $\langle\rangle$ denotes spatial average. The acceleration of the mean flow then does no longer depend on the domain size, and a smaller value of m_{fluid} can be set to reach the steady state faster [47].

2.7 Discretization in space

The model equations described so far can be discretized with any numerical scheme. Ghaffari et al., for example, used compact finite differences and studied two–dimensional models of swimming fish [65], while Boiron et al. employed a finite volume discretization and studied supersonic flows interacting with bluff bodies [14]. Even Chebyshev collocation method have been used by Pasquetti et al. [128], which involve comparatively high computational cost when solving Poisson's equation.

In the present work, we will use a Fourier pseudospectral discretization. In two spatial dimensions, this approach has been used quite extensively, for example to compute flows past bundles of cylinders [144, 146], or the decaying turbulence in circular containers [145]. The general idea is to represent variables as truncated Fourier series, thus in three dimensions we have for any

quantity q (velocity, pressure, vorticity)

$$q(\underline{x},t) = \sum_{k_x=0}^{N_x-1} \sum_{k_y=0}^{N_y-1} \sum_{k_z=0}^{N_z-1} \widehat{q}(\underline{k},t) \exp(\imath \underline{k} \cdot \underline{x}) \qquad \underline{x} \in [0,2\pi)^3$$

where $\underline{k} = (k_x, k_y, k_z)^T$ is the wavevector, $\imath = \sqrt{-1}$ and $\widehat{q}$ are the discrete complex Fourier coefficients of q. As the variable q is, in the present application, always real-valued, its Fourier coefficients are Hermitian symmetric. Therefore, we store only half as many Fourier coefficients $\widehat{q}$ as we have grid points q, still, as complex numbers consist of two real numbers, $\widehat{q}$ and q require the same amount of memory. The Fourier coefficients can be computed with the fast Fourier transform (FFT). This algorithm is generally accredited to Cooley and Tukey [26] and, reducing the order of complexity from N^2 to $N \log N$, it is one of the most widely used numerical recipe throughout virtually all fields of science. Its widespread use is also reflected in the existence of highly tuned numerical libraries; here, we use P3DFFT in the 3D case [129] and FFTW in the 2D case [55]. In the case of Fourier discretizations, most (in our code, 80%) of the computational effort is spend on FFTs, and thus these codes are easier to optimize than others.

Once the Fourier coefficients are known, computing derivatives is greatly simplified. The gradient of a scalar can for instance be obtained by multiplying with the wavevector and the complex unit, $\widehat{\nabla q} = \imath \underline{k} \widehat{q}$. The Laplace operator becomes a simple multiplication by $-|\underline{k}|^2$, it is thus diagonal in Fourier space. When using, e.g., finite differences, the dominant part of computational efforts is, in incompressible flows, spent on solving Poisson's equation in every time step [78]. This is a strong motivation to employ a Fourier discretization, as inverting a diagonal operator becomes a simple division. Inserting the truncated Fourier series into the model equations and requiring that the residual vanishes with respect to all test functions (which are identical with the trial functions $\exp(\imath \underline{k} \cdot \underline{x})$) yields a Galerkin projection and results in an evolution equation for the Fourier coefficients of the velocity. The nonlinear and penalization terms contain products, which become convolutions in Fourier space. To facilitate computation, the products are calculated in physical space. This last introduces aliasing errors which are virtually eliminated by the 2/3 rule [17], meaning that only 2/3 of the Fourier coefficients are retained. Such a mixture of spectral and physical computations is generally labeled "pseudo-spectral" and is, when de-aliased, equivalent to a Fourier-Galerkin scheme.

In the turbulence community, Fourier spectral methods are the predominant tool, starting with the pioneering work of Orzag and Patterson (1972) [123] with a resolution of 32^3 points, until the simulations as big as 4096^3 by Ishihara et al. in 2003 [76, 77] or 8192^3 by Yeung et al. [187].

2.8 Discretization in time

The spatially discretized equations describing two and three dimensional flows can be advanced in time in either physical or Fourier space, and the choice is made by counting the necessary Fourier transforms. Time stepping Fourier space is found to be more efficient.

The discretization in time depends on the viscosity; for low Reynolds number flows, the viscosity should be treated in an implicit fashion to avoid the time step restriction $\Delta t < A\Delta x^2/\nu$ (e.g., $A = 0.5$ for the Runge–Kutta 4 scheme). Two-dimensional flows are somewhat limited in terms of Reynolds numbers, since as soon as Re is high enough, three-dimensional effects occur.

2.8.1 Two-dimensional flows

For two-dimensional flows, we discretize the vorticity-velocity eqn. (2.2.1). Denoting the Fourier transformation with $\mathcal{F}$, i.e., $\widehat{\omega} = \mathcal{F}(\omega)$, we can write the semi-discrete version of equation (2.2.1):

$$\partial_t \widehat{\omega} = -\mathcal{F}\left(\underline{u} \cdot \mathcal{F}^{-1}\left(\iota \underline{k} \widehat{\omega}\right)\right) - \nu \left|\underline{k}\right|^2 \widehat{\omega} - \iota \underline{k} \times \mathcal{F}\left(\frac{\chi}{C_\eta}\left(\underline{u} - \underline{u}_s\right)\right) \tag{2.8.1}$$

$$\widehat{\underline{u}} = \frac{\iota \underline{k}^\perp}{\left|\underline{k}\right|^2}\widehat{\omega} + \underline{u}_\infty, \quad \left|\underline{k}\right| \neq 0 \tag{2.8.2}$$

When discretizing eqn. (2.8.1) in time, we can again enjoy the diagonality of the Laplace-operator to integrate the diffusive term exactly. This method is known as integrating factor technique [17, 132]. To simplify the notation, we rewrite equation (2.8.1) as

$$\partial_t \widehat{\omega} + \nu \left|\underline{k}\right|^2 \widehat{\omega} = \widehat{f}\left(\widehat{\omega}\right) \tag{2.8.3}$$

we can easily find the solution of the homogeneous problem. The solution of the complete equation (2.8.3) then reads

$$\widehat{\omega}\left(\underline{k}, t^{n+1}\right) = \widehat{\omega}\left(\underline{k}, t^n\right) \exp\left(-\Delta t \nu \left|\underline{k}\right|^2\right) \ldots + \int_{t^n}^{t^{n+1}} \exp\left(-\tau \nu \left|\underline{k}\right|^2\right) \widehat{f}\left(\omega\left(\underline{x}, t^{n+1} - \tau\right)\right) \mathrm{d}\tau. \tag{2.8.4}$$

The Duhamel–integral on the right hand side is discretized with an explicit Adams–Bashforth scheme of second order accuracy. The exact treatment of the diffusive term avoids introducing an additional stability condition on the

time step Δt. On the other hand, the penalty term introduces additional numerical stiffness and thus imposes a restriction on the time step. Using this time marching scheme, the stability condition for the time step is

$$\Delta t \leq \min\left(C_\eta, \mathrm{CFL}\frac{\Delta x^\alpha}{u_{\max}}\right), \quad 0 < \mathrm{CFL} < 1 \tag{2.8.5}$$

where $\alpha = 4/3$ [147]. Further details about this time marching scheme and the spatial discretization can be found in [87, 144].

2.8.2 Three-dimensional flows

For three-dimensional flows, we employ two different schemes, depending on the Reynolds number. Low-Reynolds numbers are treated with an Adams-Bashforth scheme, which is similar to the one used in two-dimensional flows and treats the viscous term in an exact fashion. Higher Reynolds number simulations, where the CFL condition limits the time step, are treated with the classical fourth order Runge-Kutta scheme, treating all terms explicitly.

As discussed previously, the pressure in eqn. (2.5.2) is not a state variable, but rather a Lagrangian multiplier. We can derive the defining equation by taking the divergence of eqn. (2.5.2), i.e., dot-multiplying by ∇:

$$\begin{aligned}\nabla\cdot(\partial_t\underline{u}+\underline{\omega}\times\underline{u}) &= \nabla\cdot\left(-\nabla\Pi+\nu\nabla^2\underline{u}-\frac{\chi}{C_\eta}(\underline{u}-\underline{u}_s)\right.\\ &\qquad\left.-\nabla\times\frac{\left(\frac{\chi_{sp}}{C_{sp}}(\omega-\omega_0)\right)}{\nabla^2}\right) \qquad (2.8.6)\\ \nabla^2\Pi &= \nabla\cdot\left(-\underline{\omega}\times\underline{u}-\frac{\chi}{C_\eta}(\underline{u}-\underline{u}_s)\right) \qquad (2.8.7)\end{aligned}$$

Thus, we apply a projection to the right hand side of the (penalized) Navier–Stokes equations, which is different from the classical splitting projection methods [68], where the velocity field obtained after a preliminary time step is projected. Denoting the nonlinear term with $\mathcal{N}$, the penalization term with $\mathcal{P}$ and the sponge term with $\mathcal{S}$, the right hand side reads

$$\widehat{RHS} = \underline{\widehat{\mathcal{N}}} + \underline{\widehat{\mathcal{P}}} - \nu\,|\underline{k}|^2\,\widehat{u} + \underline{\widehat{\mathcal{S}}} + \imath\underline{k}\left[\frac{\imath\underline{k}}{|k|^2}\cdot\left(\underline{\widehat{\mathcal{N}}}+\underline{\widehat{\mathcal{P}}}\right)\right]. \tag{2.8.8}$$

The right hand side (2.8.8) is thus divergence free, but it is still important to remark that one should also start with an divergence-free initial condition. It is further worthwhile to point out that the penalization term is diagonal in

physical space and full in Fourier space. However using an integrating factor in physical space, to overcome the associated stiffness (cf section 2.1), is not possible, since the χ-function is not continuous.

2.9 Convergence properties

We now consider a simple academic flow, the Couette flow between two coaxial cylinders, to see how the numerical method works in practice. Let the cylinders have radii $R_1 = 0.5$ and $R_2 = 1.0$. The inner one rotates with an angular velocity Ω. The exact solution for the steady state is an azimuthal velocity field [113]:

$$u_\Theta(r) = -\Omega \frac{R_1^2}{R_2^2 - R_1^2} r + \Omega \frac{R_1^2 R_2^2}{R_2^2 - R_1^2} \frac{1}{r}. \tag{2.9.1}$$

Our numerical solution is obtained using a $[0, 2.5] \times [0, 2.5]$ domain and a viscosity of $\nu = 0.1$. A discontinuous mask function is used. The initial condition is taken to equal the exact solution, and the computation is stopped when the time derivatives of both L_2 and L_∞ norms are smaller than 10^{-4}, which corresponds to a final time of around $T = 1.0$. This test case being computationally cheap, we can afford to test five resolutions, 32^2, 64^2, 128^2, 256^2 and 512^2 with 30 values of C_η each, distributed logarithmically equidistant in $C_\eta = \left[10^{-6}, 10^{-1}\right]$. The time step is $\Delta t = \min\left(10^{-4}, C_\eta\right)$ to keep the time discretization error negligible, and the Adams-Bashforth scheme with integrating factor is used.

Figure 2.9.1 (left) illustrates the relative error in L_2 norm as a function of C_η and the spatial resolution. It exhibits the coupling between the spatial resolution and the penalization parameter. First, one observes the $\sqrt{C_\eta}$ slope predicted from theory [4] and the saturation of the error for small C_η. The behavior is consistent with the loss of the $\mathcal{C}^1$ regularity of the exact solution of the penalized problem in the limit of small C_η. For each resolution, one can identify an optimal value of C_η, that minimizes the error, consistent with [122]. In view of the connection between C_η and both the spatial and temporal discretization, it is clear that the strategy of choosing some fixed, small value of C_η is sure to give sub-optimal results. Instead, one should keep the number of points per thickness of the penalization boundary layer constant, $K = \sqrt{\nu C_\eta}/\Delta x$, as discussed in section 2.1. Figure 2.9.1 right illustrates the spatial convergence for different values of K. All curves exhibit first- to second order convergence, but the choice of K modifies the error offset. The smallest error is obtained for $K = 0.128$, and in all cases, the boundary layer is thus under-resolved ($K < 1$).

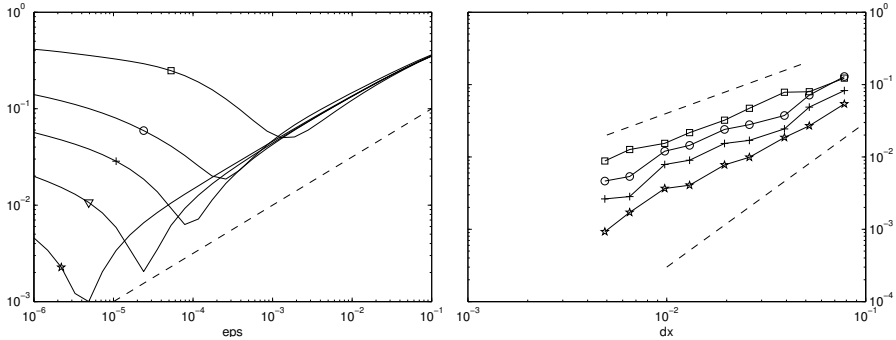


Figure 2.9.1 – Relative L_2 error $\left|\left|\underline{u}^{ex} - \underline{u}_\varepsilon^N\right|\right| / \left|\left|\underline{u}^{ex}\right|\right|$ for the Taylor-Couette flow. Left: convergence with respect to C_η. Resolutions are 32^2 (□), 64^2 (◯), 128^2 (+), 256^2 (∇) and 512^2 ($\star$). The dashed line marks the $\sqrt{C_\eta}$ slope. Right: convergence with respect to Δx, for $K = \sqrt{\nu C_\eta}/\Delta x = \text{const}$, symbols indicate $K = 0.405$ (□), $K = 0.064$ (◯), $K = 0.091$ (+) and $K = 0.128$ ($\star$). Note an intermediate value K yields best results. Dashed lines indicate first (top) and second (bottom) order convergence.

At this stage we should point out that the choice of C_η does not exclusively depend of the viscosity. If so, it would be possible to give more sophisticated estimates for its determination by considering the penalized heat equation. For small values of C_η, it is in addition the projection to the incompressible manifold that regularizes the loss of $\mathcal{C}^1$ regularity, since the divergent part of the penalization term is canceled by the contribution to the pressure, eqn. (2.8.7). It is, therefore, likely to be impossible to give a universal way to choose C_η. However, we always found K_{opt} between 0.1 and 0.4 for a large variety of validation cases, not all of which are reported in this manuscript, and consider $K = 0.1, \ldots, 0.4$ as a recommendation for usage in practice.

2.10 Concluding remarks

The proposed numerical method relies on the volume penalization and a Fourier pseudospectral discretization. We outlined the essential properties of this combination of methods, especially the parameter of the volume penalization method, C_η, has been discussed in detail. We provided evidence that one in fact chooses $K = \sqrt{\nu C_\eta}/\Delta x = \text{const}$ instead. Care must be taken however, since C_η is also a time scale. Thus, when it comes to turbulent flows near the walls, C_η should roughly not exceed the shortest turbulent times

scales by too much, even though the viscous sub-layer will likely prevent the smallest eddies from reaching the wall.

The time step of the proposed time marching methods must not exceed C_η, which, with $K = \text{const}$, implies $\Delta t = \left[K^2/\nu\right] \Delta x^2$. The viscosity thus determines the relative influence of the penalization, since for ν small, the time step is determined by the CFL condition, and we conclude therefore that the method works better for lower viscosities ν.

All numerical codes used in this work are written in FORTRAN95 and are freely available on the internet. The proposed framework is intended to be applicable also for higher Reynolds number flow, for which small spatial and temporal vortical structures appear. To resolve these scales, high resolution and therefore the usage of high-performance computers is required. To this end, our code is designed to compute on massively parallel machines with distributed memory architectures. The parallel implementation is based on the MPI protocol. The FFTs are computed using the P3DFFT library [129], which provides a parallel data decomposition framework, and FFTW [55], which is used for the one–dimensional Fourier transforms inside P3DFFT. The flow variables are stored on the three–dimensional Cartesian computational grid. Each MPI process only holds a portion of the total data, and the parallel decomposition is performed on at most two indices, i.e., a pencil decomposition is used. The x-direction is not split among processes, the code can thus run on $N_y N_z$ processes at most. This limitation is however not of practical importance, since $N_y N_z$ usually exceeds the number of available CPUs. Some details about the three-dimensional code FLUSI[1] are given in Appendix A

[1]The FluSI (acronym for FLUid Structure Interaction) code is freely available for non-commercial use under https://github.com/pseudospectators/FLUSI

Part II

Fluid-structure interaction with rigid obstacles

Chapter 3

Virtual insects: modeling and validation

The content of this chapter is adapted from [45].

3.1 An introduction to insect flight

Insect fly by flapping their wings, a mechanism entirely different from human aircraft. Their wings are basically flat without a notable aerodynamic profile (since the wings deform while flapping, this is only strictly true on the ground) and operate typically at high angle of attack. If a flat plat in steady flow is operating at this angle of attack, it is not even close to producing enough lift to balance the weight, which led to the widespread myth "insects cannot fly, according to physics" [39], although it is clear that it is not the insect doing something wrong.

The fact that they do fly implies that they use unusual and potent aerodynamic strategies for force generation. The first mechanism was discovered by Weis-Fogh in 1973 [176] and is termed 'clap–fling–sweep'. It is mainly encountered in small insects, for example in *Encarsia formosa*, the chalcid wasp (< 1 mm body length) [85]. The insect brings its wings closely together ('clap'), then opens a gap in V-shape ('fling') and then sweeps the wing apart. During the fling, strong vortices are generated, providing enhanced force production. The kinematic pattern is illustrated in figure 3.1.1. The clap–fling–sweep mechanism was an important result, but it could explain only some aspects of insect flight. Especially for larger specimen, the existence of a leading edge vortex (LEV) was hypothesized, and in 1996 Ellington et al. [39] presented

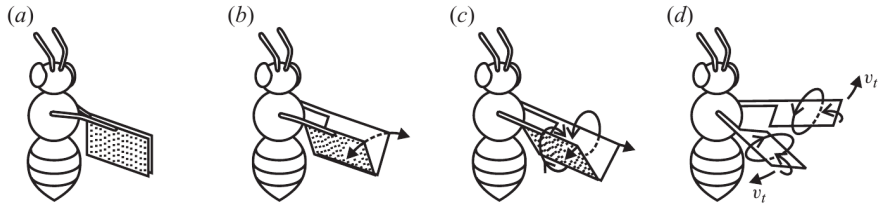


Figure 3.1.1 – Illustration of the clap (a) fling (b,c) sweep (d) mechanism, figure adapted from Weis-Fogh 1973 [176].

the first conclusive evidence for this hypothesis. The LEV is typically attached to the wing, and at each half-stroke, a new vortex is generated. Spanwise flow along the wing has been suggested as stabilization mechanism, see e.g., [93]. Figure 3.1.2 shows schematic illustration of the vortical system, consisting of the LEV and a tip vortex that constitutes the wake. Dickinson et al. [29] used a flapping wing robot in mineral oil, operating at the Reynolds number of a fruit fly, and found three distinct mechanisms, delayed stall, rotational circulation and wake capture. Delayed stall is associated with the formation of the LEV, and rotational circulation refers to the fact that between half strokes, the insect necessarily rotates its wing, which in turn contributes to the total circulation and thus force production. The relative timing of this rotation could, according to [29], be used to modulate the force production by either rotating before, during or after stroke reversal. The stroke cycle-averaged lift coefficient can thus be modified by a factor of approximately 1.7 [29]. However, the significance of this mechanism for lift enhancement in real animals is not clear because recent 3D reconstruction of wing motion in freely flying 1.0 mg fruit flies suggests kinematic patterns at which rotational circulation is minimal at each half stroke reversal [151].

Finally, while flapping back and forth the wing could interact with its own wake from the previous half stroke, which is referred to as wake capturing. This mechanism also depends on the stroke trajectory, as shown in [92]. Wake capturing is particularly relevant for insects with four independent wings, like dragonflies, since the small distance between fore- and hindwing necessarily invokes wing–wake interactions, which in turn also offers opportunities for control [106] .

The key to insect flight is their wing kinematics, which varies tremendously between species [36], airspeed [33] and even between individual strokes of the same animal [154]. Earlier studies tethered the insect in a wind tunnel and recorded the wing motion with high-speed cameras, until evidence was presented [154] that even though insects do flap when being tethered

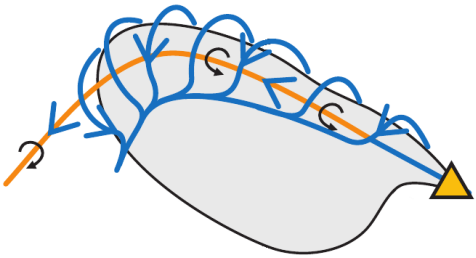


Figure 3.1.2 – Sketch of the leading edge and tip vortices, figure adopted from Lentink and Dickinson [93]

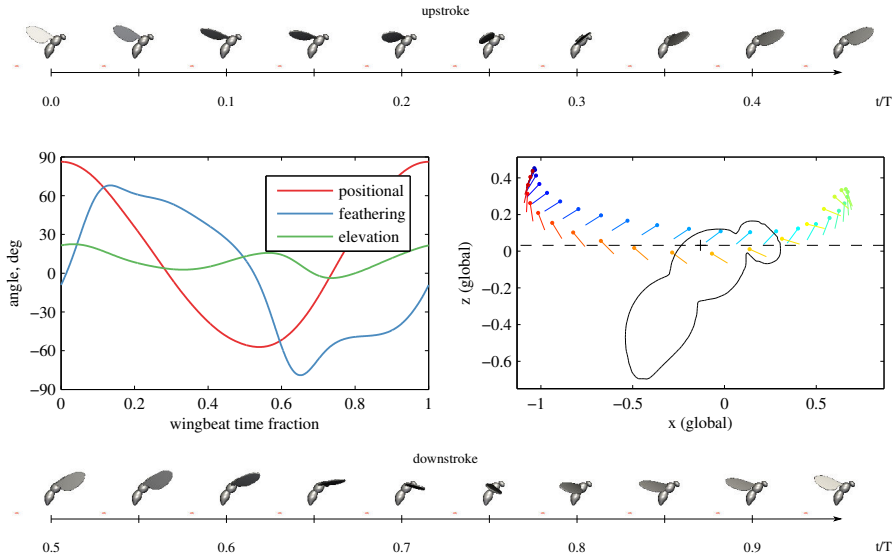


Figure 3.1.3 – Fruit fly in hovering with the kinematics data according to [57]. Top and bottom row show the wingbeat during up- and downstroke. Centered figures illustrate the wing angles (left) and the wing trajectory (right). Dashed line is the stroke plane.

in an axial flow, this flapping is strikingly different from what they do in voluntary free flight [57]. Nowadays, insects are thus filmed in free flight for kinematic reconstruction [56, 190]. The flapping frequency decreases with increasing size, from 400 Hz in the tiny calcid wasp *Encarsia formosa* [176], 218 Hz in fruit fly *Drosophila melangogaster* [57], 152 Hz in bumblebees *Bombus trerrestris* [33] to 26 Hz for a hawkmoth [96, 97]. The mosquito *Aedes aegypti* is exceptional in that it flaps at 800 Hz [74].

In hovering flight for many insects, the stroke plane is almost horizontal, i.e., the wings flap in a plane approximately parallel to the ground. Exceptions are for example the hoverflies *Episyrphus balteatus*, with an inclination angle of more then 30° with respect to horizontal [36], and dragonflies [20]. The hovering flight of a fruit fly is illustrated in figure 3.1.3, showing the typical features regarding body posture, horizontal stroke plane and high angles of attack. The kinematics are derived from [57], and the precise definition of positional, feathering and elevation angle will be given in section 3.2. In forward flight, the stroke plane is effectively tilted with respect to horizontal, as is the body. The body posture also depends on the forward flight speed, as experiments on honeybees suggest. The bees have been reported to hold their abdomen higher with increasing airspeed, and the authors of the study suggest that this is a mechanism for drag reduction [99, 161].

Flapping flight allows for great maneuverability. Fruit flies, for example, regularly perform maneuvers termed 'saccades', which are changes in the flight direction by 90° or more, over a timescale of a few wing beats (40 ms) only [56]. Surprisingly, these maneuvers are associated with only subtle differences in the kinematics. Even more impressive maneuvers have been observed in fruit flies when they fear a predator, both during an escape take off [190] and in free flight [118]. Attempts to synthesize stereotypical steering patterns, e.g., to induce a yaw rotation, have for instance been proposed in [30]. However, the apparent price to pay for this great maneuverability in flapping flight seems to be intrinsic instabilities that require constant control [138, 139, 183].

The Reynolds number of an insect, of great relevance for the fluid dynamics, is commonly defined as

$$Re = \frac{U_{\text{tip}} R}{\nu} = \frac{2\Phi f R \cdot c_m}{\nu},$$

where U_{tip} is the mean wingtip velocity and equal to $2\Phi f R$, and c_m is the mean chord length defined as A/R, where A is the wing's surface. This Reynolds number is convenient since it involves many important parameters of the insect (unlike, e.g., $Re_f = f^2 R/\nu$). For a fruitfly, $Re \approx 130$ [103], while a bumblebee *bombus terrestris* operates at about 2000 [43]. Despite their remarkably high wingbeat frequency, mosquitoes like *Aedes aegypti* operate at

low Reynolds numbers, typically 75, owing to their reduced stoke amplitude [74]. Hawkmoths are larger specimen and their Reynolds number is typically 6300 [96], which is of the same order of magnitude as hummingbirds [175].

3.2 Virtual insects

In the following, we describe the virtual insect framework used in this work in detail. We remind the modularization of the framework which means that the previously described fluid module can take any obstacle into account through the volume penalization method, i.e., through the mask function χ and the solid velocity field $\underline{u}_s$. The insect module is thus concerned with creating these latter functions on the Eulerian fluid grid by using the signed distance function. Our virtual insect consists of a body and two wings, all of which are performing solid body rotations around three axis. Therefore, we will make excessive use of the rotation matrices

$$R_x(\xi) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\xi & \sin\xi \\ 0 & -\sin\xi & \cos\xi \end{pmatrix} \qquad R_y(\xi) = \begin{pmatrix} \cos\xi & 0 & -\sin\xi \\ 0 & 1 & 0 \\ \sin\xi & 0 & \cos\xi \end{pmatrix}$$

$$R_z(\xi) = \begin{pmatrix} \cos\xi & \sin\xi & 0 \\ -\sin\xi & \cos\xi & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

to define the different reference frames

$$\underline{x}^{(g)} \text{ global} \quad \underline{x}^{(b)} \text{ body} \quad \underline{x}^{(w)} \text{ wing} \quad \underline{x}^{(s)} \text{ stroke plane} \quad ,$$

in which the geometry is defined. As described above, the mask function is constructed in each evaluation of the right hand side as a function of the signed distance function, $\chi(\underline{x}) = \chi(\delta(\underline{x}))$, according to eqn. (2.3.3).

3.2.1 Body system

The insect's body is responsible for a major part of the total drag force. Earlier numerical contributions, like the one by Ramamurti and Sandberg [133], neglected the body to simplify the problem. For body fitted grids, the presence of the body introduces a significant complication. One way to deal with this complexity is the use of overset grids, where a mesh for each solid part (body, wings) is defined. These grids overlap and require interpolation and synchronization, which is complicated on distributed memory architectures. This method is employed, for example, by the group of Hao Liu [96]. The penalization method used in this work allows using arbitrarily complex bodies, including legs and antennae.

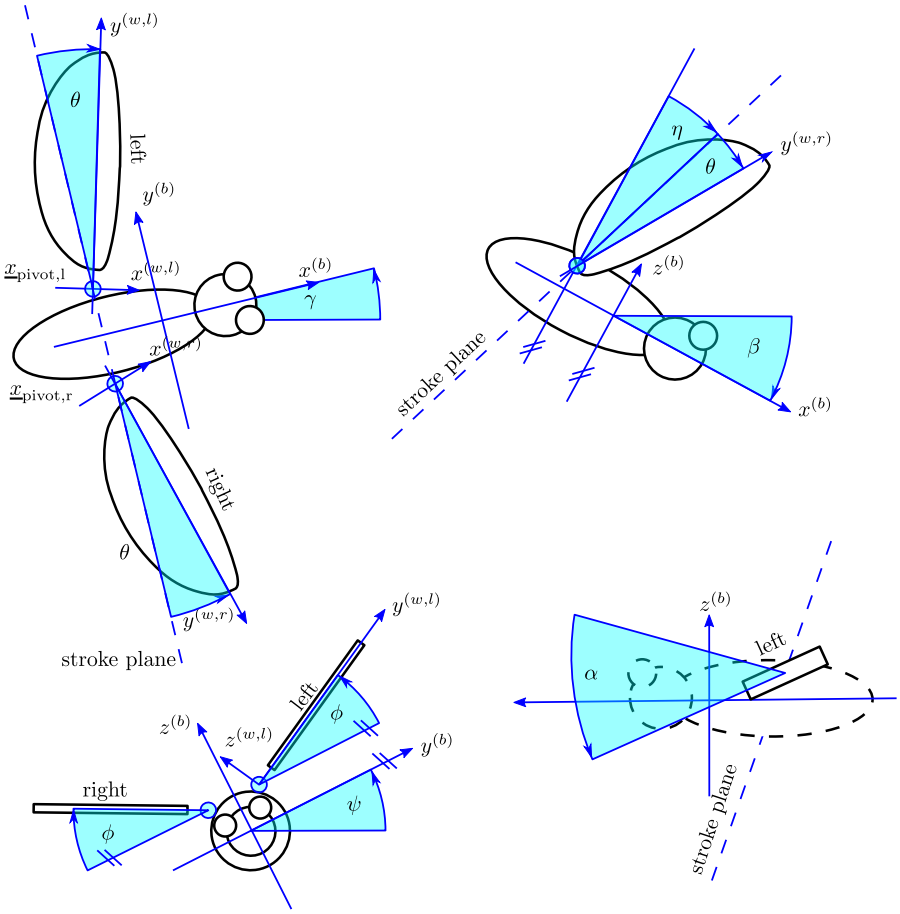


Figure 3.2.1 – A simple model insect ("Jerry"). Definitions of the body angles γ (yaw), β (pitch), ψ (roll) and the anatomical stroke plane angle η, definitions of the wing coordinate systems and the wing angles θ (deviation), ϕ (position) and α (feathering).All angles are shown with positive sign.

The insect's body is described by logical center $\underline{x}^{(g)}_{\text{cntr}}$, the translational velocity $\underline{u}^{(g)}_{\text{cntr}}$ and the body angles β (pitch), γ (yaw) and ψ (roll), see figure 3.2.1. The center point $\underline{x}^{(g)}_{\text{cntr}}$ does not necessarily coincide with the center of gravity, it is rather an arbitrary point of reference. A point $\underline{x}^{(g)}$ in the global coordinate system can be transformed to the body system using the following linear transformation

$$\begin{aligned} \underline{x}^{(b)} &= M_{\text{body}}(\psi,\beta,\gamma)\left(\underline{x}^{(g)} - \underline{x}^{(g)}_{\text{cntr}}\right) \\ M_{\text{body}} &= R_x(\psi)\,R_y(\beta)\,R_z(\gamma)\,. \end{aligned} \tag{3.2.1}$$

Since rotation matrices do not commute, it is important to note that the body is first yawed, then pitched and finally rolled, which is conventional in flight mechanics. The geometry of the body is defined in the body reference frame, for example composed by ellipsoids and spheres like the one illustrated in figure 3.2.1. The angular velocity of the body in the global system is

$$\begin{aligned} \underline{\Omega}^{(g)}_b &= R_z^{-1}(\gamma)\left[\begin{pmatrix}0\\0\\\dot{\gamma}\end{pmatrix} + R_y^{-1}(\beta)\left[\begin{pmatrix}0\\\dot{\beta}\\0\end{pmatrix} + R_x^{-1}(\psi)\begin{pmatrix}\dot{\psi}\\0\\0\end{pmatrix}\right]\right] \\ \underline{\Omega}^{(b)}_b &= M_{\text{body}}\,\Omega^{(g)}_b \end{aligned}$$

which defines the velocity field inside the insect resulting from the body motion,

$$\underline{u}^{(g)}_b = \underline{u}^{(g)}_{\text{cntr}} + M^{-1}_{\text{body}}\left(\underline{\Omega}^{(b)}_b \times \underline{x}^{(b)}\right). \tag{3.2.2}$$

Equation (3.2.2) is valid also in the wings, since the flapping motion is relative to the body.

3.2.2 Body shape

The body shape is described in the body reference frame described previously. For instance, for the body depicted in figure 3.2.1, which is composed of an ellipsoidal shaped thorax and spheres for the head and eyes, the signed distance function is the intersection of the distance functions for the thorax, head and eyes,

$$\delta_{\text{body}} = \max\left(\delta_{\text{thorax}}, \delta_{\text{head}}, \delta_{\text{eyes}}\right). \tag{3.2.3}$$

The max operator of the signed distances in eqn. (3.2.3) represents the intersection operator [127]. The signed distances for the components read

$$\begin{aligned}\delta_{\text{thorax}}\left(\underline{x}^{(b)}\right) &= \sqrt{\left(y^{(b)}\right)^2+\left(z^{(b)}\right)^2}-\sqrt{b^2\left(1-\left(x^{(b)}/a\right)^2\right)}\\ \delta_{\text{head}}\left(\underline{x}^{(b)}\right) &= \left|\underline{x}^{(b)}-\underline{x}^{(b)}_{0,\text{head}}\right|\\ \delta_{\text{eyes}}\left(\underline{x}^{(b)}\right) &= \left|\underline{x}^{(b)}-\underline{x}^{(b)}_{0,\text{eyes}}\right|,\end{aligned}$$

where a, b define the axes of the thorax ellipsoid and $\underline{x}^{(b)}_{0,\text{head,eyes}}$ are the centers of the spheres. A second example is the more realistic body shape in figure 3.5.1, which is obtained by sweeping the radius

$$R(s)=\begin{cases}\max\left(0,-1.299s^2+0.94s+0.03\right) & s<0.6333 & \text{abdomen}\\ \max\left(0,-2.16s^2+3.47s-1.21\right) & 0.6333<s<1 & \text{thorax}\\ \max\left(0,-12.68s^2+27.49s-14.73\right) & 1<s<1.2 & \text{head}\end{cases}$$

around an arc-segment of radius $r_{\text{bc}}=0.95$ centered at $x^{(b)}_{\text{bc}}=-0.245$, $z^{(b)}_{\text{bc}}=-0.93$.

3.2.3 Wing system

We consider only insects with two wings, one on each side, which applies for example to true flies (*Dipterae*), but also to a variety of insects that have four wings, but move them together so they act like one wing [13]. The extension to four-winged insects that can move all wings independently, like Dragonflies [168], is straightforward. The wings are rotating about the pivot points $\underline{x}^{(b)}_{\text{pivot,r}}$ and $\underline{x}^{(b)}_{\text{pivot,l}}$. These pivots do not necessarily lie on the body surface; we rather allow a gap between wings and body. This gap avoids problems with non-solenoidal velocity fields at the wing base. It is conventional to introduce a stroke plane, which is a plane tilted with respect to the body by an angle η. The coordinate in the stroke plane reads

$$\underline{x}^{(s)}=M_{\text{stroke}}\left(\underline{x}^{(b)}-\underline{x}^{(b)}_{\text{pivot}}\right).$$

Since three angles fully describe the solid body rotation, the redundant stroke plane is introduced for convenience only. It is mainly motivated by the visual impression of the wings moving in such a stroke plane, even though in reality there is an out-of-plane motion as well. Here, we use an anatomical stroke angle, that is, the angle η is defined relative to the body. Some insects tilt their stroke plane when changing from hovering to forward flight, others change

the body pitch angle β. Within the stroke plane, the wing motion is described by the angles α (feathering angle or angle of attack), ϕ (positional or flapping angle), θ (deviation or out-of-stroke angle). Applying two rotation matrices transforms from the body to the wing coordinate system:

$$\underline{x}^{(w)} = M_{\text{wing}}\underline{x}^{(s)} = M_{\text{wing}}M_{\text{stroke}}\left(\underline{x}^{(b)} - \underline{x}^{(b)}_{\text{pivot}}\right).$$

When flapping symmetrically, i.e., both wings following the same motion protocol, the stroke and wing rotation matrices for the left and right wing are given by

$$\begin{aligned} M_{\text{stroke,l}} &= R_y(\eta) & M_{\text{stroke,r}} &= R_x(\pi)\,R_y(\eta) \\ M_{\text{wing,l}} &= R_y(\alpha)\,R_z(-\theta)\,R_x(\phi) & M_{\text{wing,r}} &= R_y(-\alpha)\,R_z(-\theta)\,R_x(-\phi) \end{aligned}$$

due to the rotation $R_x(\pi)$ the sign of θ for the right wing does not have to be inverted. The angular velocities of the wings are given by

$$\begin{aligned} \underline{\Omega}^{(b)}_w &= M^{-1}_{\text{stroke}}\left[R^{-1}_x(\phi)\left[\begin{pmatrix}\dot{\phi}\\0\\0\end{pmatrix} + R^{-1}_z(-\theta)\left[\begin{pmatrix}0\\0\\-\dot{\theta}\end{pmatrix} + R^{-1}_y(\alpha)\begin{pmatrix}0\\\dot{\alpha}\\0\end{pmatrix}\right]\right]\right] \\ \underline{\Omega}^{(w)}_w &= M_{\text{wing}}\,\underline{\Omega}^{(b)}_w \end{aligned}$$

which is used to compute the velocity field resulting from the wing motion,

$$\begin{aligned} \underline{u}^{(w)}_w &= \underline{\Omega}^{(w)}_w \times \underline{x}^{(w)} \\ \underline{u}^{(g)}_w &= M^{-1}_{\text{body}}\,M^{-1}_{\text{wing}}\,M^{-1}_{\text{stroke}}\,\underline{u}^{(w)}_w. \end{aligned} \tag{3.2.4}$$

The total velocity field inside the wings is given as the superposition of the body and wing rotation:

$$\underline{u}^{(g)}_s\left(\underline{x} \in \{\underline{x}_w\}\right) = \underline{u}^{(g)}_w + \underline{u}^{(g)}_b$$

The actual kinematics, i.e., the angles $\alpha(t)$, $\phi(t)$ and $\theta(t)$ are parametrized by either Fourier or Hermite interpolation coefficients.

3.2.4 Wing shape

In the previous section we defined the wing reference frame $\underline{x}^{(w)}$, in which we now describe the wing's signed distance function δ. In general, we define a set of several signed distance functions, each of which describes one surface of the wing. The signed distance function of the entire wing is then given by their intersection. For some model insects, we consider simple wings for

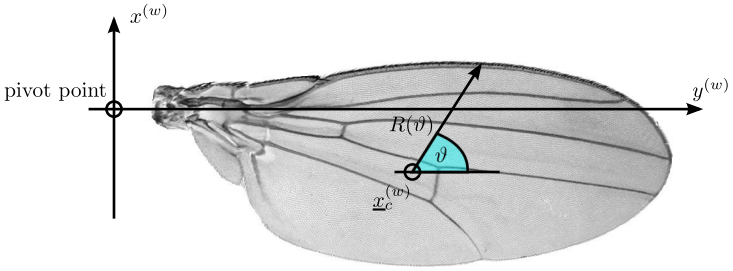


Figure 3.2.2 – Realistic wing shapes are described in polar coordinates $R(\vartheta)$.

which straightforward analytical expressions are available. For a rectangular wing, for instance the one illustrated in figure 3.4.1 (c), we find for the signed distance function

$$\delta\left(\underline{x}^{(w)}\right) = \max\left(x^{(w)} - b; x^{(w)} - (B-b); y^{(w)} - 1; a - y^{(w)}; |z| - h/2\right). \tag{3.2.5}$$

For realistic insect wings however, we parametrize the wing shape in polar coordinates. As illustrated in figure 3.2.2, the shape in the wing plane is described by the center point $\underline{x}_c^{(w)}$, which is arbitrary as long as the function $R(\vartheta)$ is unique for all ϑ. To sample the wing on a computational grid, we need a function $R(\vartheta)$ that can be evaluated for all ϑ. As $R(\vartheta)$ is naturally 2π-periodic, a truncated Fourier series can be used:

$$R(\vartheta) = \frac{a_0}{2} + \sum_{i=1}^{N} a_i \cos(2\pi i\vartheta) + \sum_{i=1}^{N} b_i \sin(2\pi i\vartheta) \tag{3.2.6}$$

In practice, eqn (3.2.6) has to be evaluated for all grid points in the vicinity of the wing, which requires $\mathcal{O}\left(N_x N_y N_z\right)$ evaluations with a small constant. The computational cost can however be significant, which is why eqn. (3.2.6) is evaluated for 25 000 values of ϑ once during initialization. Afterwards, linear interpolation is used for its lower computational cost. The signed distance for such a wing then reads

$$\delta\left(\underline{x}^{(w)}\right) = \max\left(\left|z^{(w)}\right| - h/2; r(\vartheta) - R(\vartheta)\right).$$

If a wing cannot be described by one radius, because $R(\vartheta)$ is not unique, several radii and center points can be used [13].

3.2.5 Power requirement

Actuating the wings requires power expenditures that are very difficult to measure directly. For larger animals, like hummingbirds, tiny masks can be placed over the nares in order to measure oxygen consumption [52]. For insects, this technique cannot be directly applied, instead, one could measure the oxygen concentration in a completely sealed wind tunnel. Most results about power consumption reported in the literature are instead based on models [34, 37, 38].

In numerical simulations, the power can be obtained directly, since the aerodynamic torque moment $\underline{m}$ with respect to the wing pivot point is available from eqn. (2.4.6). The power P_{aero} required to move one wing is found to be

$$P_{\text{aero}} = -\underline{m} \cdot (\underline{\Omega}_w - \underline{\Omega}_b) \tag{3.2.7}$$

which is equivalent to the definition

$$P_{\text{aero}} = \int_{\partial\Omega_s} \underline{u} \mathrm{d}\underline{F}$$

given in [103]. The power can also be computed directly from the penalization term,

$$P_{\text{aero}} = \int_{\Omega} \underline{u}_s \frac{\chi}{C_\eta} (\underline{u} - \underline{u}_s)\,, \tag{3.2.8}$$

if the body is at rest. If it moves, the body velocity field has to be substracted from eqn. (3.2.8), since the body motion does not contribute to the power. The above definitions include all contributions from the fluid, which are usually treated separately in analytical models, e.g., according to [34]

$$P_{\text{aero}} = P_{\text{par}} + P_{\text{ind}} + P_{\text{pro}} + P_{\text{acc}}$$

where P_{par} is the parasitic power, P_{ind} the induced power, P_{pro} the profile power and P_{acc} the power to accelerate the virtual mass. In addition to the aerodynamic power, the inertial power has to be expended, i.e., the power required to move the wing in vacuum. As the flapping motion is periodic, its stroke averaged value is zero. The inertial power P_{inert} is positive is the wing is accelerated (power consumed) and negative if it is decelerated. The definition is

$$P_{\text{inert}} = \underline{\Omega}_w^{(w)} \cdot \left(\underline{\underline{J}}^{(w)} \dot{\underline{\Omega}}_w^{(w)} + \underline{\Omega}_w^{(w)} \times \underline{\underline{J}}^{(w)} \underline{\Omega}_w^{(w)} \right)$$

with the wing tensor of inertia $\underline{\underline{J}}^{(w)}$ [12]. As the wings are typically very thin, the inertia tensor reduces to

$$\underline{\underline{J}}^{(w)} = \begin{pmatrix} J_{xx} & J_{xy} & 0 \\ J_{xy} & J_{yy} & 0 \\ 0 & 0 & J_{zz} \end{pmatrix}.$$

The sum of inertial and aerodynamic power can be negative during deceleration phases, which would mean that the insect can store energy in its muscles. It is unknown to what extend this can be realized, and it is thus often conservatively assumed that energy storage is not possible, in which case the total power P_{total} is given by

$$P_{\text{total}} = \max\left(P_{\text{inert}} + P_{\text{aero}}, 0\right).$$

3.2.6 Governing equations in free flight

Until now we have considered the insect to be fixed, i.e., tethered in the computational domain. In free flight, additional equations have to be solved together with the fluid, namely Newton's law. The body translation is then governed by

$$M\,\underline{\dot{u}}_{\text{cntr}}^{(g)} = \underline{F}^{(g)}$$

where $\underline{F}^{(g)}$ contains the aerodynamic and gravitational forces and M is the mass of the insect. For simplicity, $\underline{x}_{\text{centr}}$ and $\underline{u}_{\text{centr}}$ correspond to the center of gravity in the case of free flight. To handle the rotational degrees of freedom, we employ a quaternion based formulation, similar to the one proposed in [102], which avoids the 'Gimbal lock' problem. The governing equation for the angular velocity $\underline{\Omega}^{(b)}$ in the body reference frame reads

$$\underline{\underline{J}}^{(b)}\underline{\dot{\Omega}}^{(b)} + \begin{pmatrix} 0 & -\Omega_z^{(b)} & \Omega_y^{(b)} \\ \Omega_z^{(b)} & 0 & -\Omega_x^{(b)} \\ -\Omega_y^{(b)} & \Omega_x^{(b)} & 0 \end{pmatrix} \underline{\underline{J}}^{(b)}\underline{\Omega}^{(b)} = \underline{m}^{(b)},$$

where $\underline{\underline{J}}^{(b)}$ is the moment of inertia around the body axes $\left(x^{(b)}, y^{(b)}, z^{(b)}\right)$ and $\underline{m}$ is the vector of torque moments as defined in eqn. (2.4.6). The skew-symmetric term stems from the change into a moving reference frame. We introduce the normalized quaternion ε with components ε_i, $i = 0, \ldots, 3$, $\sum \varepsilon_i^2 = 1$. The governing equations for the quaternion state are

$$\dot{\underline{\varepsilon}} = \frac{1}{2}\underline{\underline{S}}^T\,\underline{\Omega}^{(b)}$$

with the matrix

$$S = \begin{pmatrix} -\varepsilon_1 & \varepsilon_0 & \varepsilon_3 & -\varepsilon_2 \\ -\varepsilon_2 & -\varepsilon_3 & \varepsilon_0 & \varepsilon_1 \\ -\varepsilon_3 & \varepsilon_2 & -\varepsilon_1 & \varepsilon_0 \end{pmatrix}.$$

Assuming $\underline{\underline{J}}^{(b)}$ to be constant the body axes to be the principal axes of inertia (i.e., $\underline{\underline{J}}^{(b)}$ is diagonal), the following first order system of 13 differential

equations is obtained

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix} x^{(g)}_{\mathrm{cntr}} \\ y^{(g)}_{\mathrm{cntr}} \\ z^{(g)}_{\mathrm{cntr}} \\ u^{(g)}_{\mathrm{cntr,x}} \\ u^{(g)}_{\mathrm{cntr,y}} \\ u^{(g)}_{\mathrm{cntr,z}} \\ \varepsilon_0 \\ \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \Omega^{(b)}_x \\ \Omega^{(b)}_y \\ \Omega^{(b)}_z \end{pmatrix} = \begin{pmatrix} u^{(g)}_{\mathrm{cntr,x}} \\ u^{(g)}_{\mathrm{cntr,y}} \\ u^{(g)}_{\mathrm{cntr,z}} \\ F^{(g)}_x/M \\ F^{(g)}_y/M \\ \left(F^{(g)}_z/M - g\right) \\ \left(-\varepsilon_1\Omega^{(b)}_x - \varepsilon_2\Omega^{(b)}_y - \varepsilon_3\Omega^{(b)}_z\right)/2 \\ \left(\varepsilon_0\Omega^{(b)}_x - \varepsilon_3\Omega^{(b)}_y + \varepsilon_2\Omega^{(b)}_z\right)/2 \\ \left(\varepsilon_3\Omega^{(b)}_x + \varepsilon_0\Omega^{(b)}_y - \varepsilon_1\Omega^{(b)}_z\right)/2 \\ \left(-\varepsilon_2\Omega^{(b)}_x + \varepsilon_1\Omega^{(b)}_y + \varepsilon_0\Omega^{(b)}_z\right)/2 \\ \left(\left(J^{(b)}_y - J^{(b)}_z\right)\Omega^{(b)}_y\Omega^{(b)}_z + m^{(b)}_x\right)/J^{(b)}_x \\ \left(\left(J^{(b)}_z - J^{(b)}_x\right)\Omega^{(b)}_z\Omega^{(b)}_x + m^{(b)}_y\right)/J^{(b)}_y \\ \left(\left(J^{(b)}_x - J^{(b)}_y\right)\Omega^{(b)}_x\Omega^{(b)}_y + m^{(b)}_z\right)/J^{(b)}_z \end{pmatrix} \tag{3.2.9}$$

which is solved with the same time discretization as the fluid. The rotation matrix M_{body} is then computed from the quaternion ε_i

$$M_{body} = \begin{pmatrix} \varepsilon_0^2+\varepsilon_1^2-\varepsilon_2^2-\varepsilon_3^2 & 2\left(\varepsilon_1\varepsilon_2-\varepsilon_3\varepsilon_0\right) & 2\left(\varepsilon_1\varepsilon_3+\varepsilon_2\varepsilon_0\right) \\ 2\left(\varepsilon_1\varepsilon_2+\varepsilon_3\varepsilon_0\right) & \varepsilon_0^2-\varepsilon_1^2+\varepsilon_2^2-\varepsilon_3^2 & 2\left(\varepsilon_2\varepsilon_3-\varepsilon_1\varepsilon_0\right) \\ 2\left(\varepsilon_1\varepsilon_3-\varepsilon_2\varepsilon_0\right) & 2\left(\varepsilon_2\varepsilon_3+\varepsilon_1\varepsilon_0\right) & \varepsilon_0^2-\varepsilon_1^2-\varepsilon_2^2+\varepsilon_3^2 \end{pmatrix}, \tag{3.2.10}$$

which replaces the definition in eqn. (3.2.1) in the free-flight case. The initial values at time $t = 0$ for ε_i can conveniently be computed from a set of yaw, pitch and roll angles,

$$\begin{pmatrix} \varepsilon_0 \\ \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \end{pmatrix} = \begin{pmatrix} \cos(\psi/2)\cos(\beta/2)\cos(\gamma/2) + \sin(\psi/2)\sin(\beta/2)\sin(\gamma/2) \\ \sin(\psi/2)\cos(\beta/2)\cos(\gamma/2) - \cos(\psi/2)\sin(\beta/2)\sin(\gamma/2) \\ \cos(\psi/2)\sin(\beta/2)\cos(\gamma/2) + \sin(\psi/2)\cos(\beta/2)\sin(\gamma/2) \\ \cos(\psi/2)\cos(\beta/2)\sin(\gamma/2) - \sin(\psi/2)\sin(\beta/2)\cos(\gamma/2) \end{pmatrix}.$$

In the actual implementation, we multiply the right hand side of equation (3.2.9) with a six component vector with zeros or ones, to deactivate some degrees of freedom.

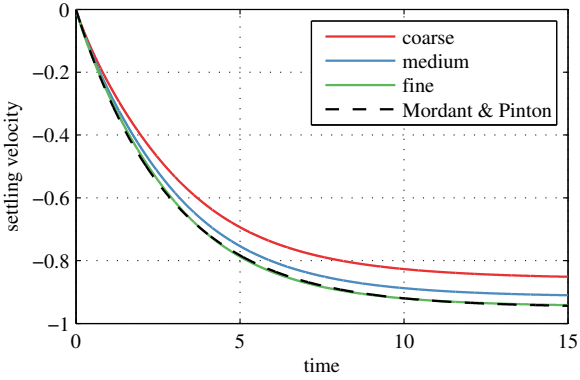


Figure 3.3.1 – Settling velocity of a falling sphere, experimental results by Mordant & Pinton [116] and present results for coarse, medium and fine grids.

3.3 Validation case of a falling sphere

The first test case to validate the flow solver is the sedimentation of a sphere, which in our terminology is an insect without wings and with a spherical body. We consider case 1 proposed by Mordant and Pinton [116], who studied the sedimentation problem experimentally in a water tank. The sphere of unit diameter and mass $M = 1.3404$ is falling in fluid of viscosity $\nu = 0.0228$ and unit density. The dimensionless gravity is $g = 0.8036$ and the terminal settling velocity obtained from the experiments is $U = 0.9488$. We perform a grid convergence test using the domain size $8 \times 8 \times 16$, and parameters $N_x \times N_y \times N_z \times C_\eta$ of a coarse ($96 \times 96 \times 192 \times 10^{-3}$), medium ($192 \times 192 \times 384 \times 2.5 \cdot 10^{-4}$) and fine ($384 \times 384 \times 768 \times 6.25 \cdot 10^{-5}$) grid. The number of points per penalization boundary layer is $K = 0.0573$. The results of the convergence study are illustrated in figure 3.3.1 and the settling velocity for the finest resolution differs from the experimental findings by less than 1%. The finest resolution required 30 GB of memory and 34 400 CPU hours on 1024 cores to perform 266 667 time steps. The computational cost is relatively high, since small values of C_η are required as the Reynolds number is small.

3.4 Validation case of a rectangular flapping wing

We consider the setup proposed in [158], Appendix B2, by Suzuki and co-workers. It considers only one rectangular wing with a finite thickness, neglecting thus the body and the second wing. The fact that the thickness is

finite and the geometry rather simple, compared to actual insects, motivates the choice of this setup as a first test case. The wing kinematics are given by

$$\begin{aligned} \phi &= \phi_m \cos(2\pi t) \\ \alpha &= \frac{\alpha_m}{\tanh c_\alpha} \tanh(c_\alpha \sin(2\pi t)) \\ \theta &= 0 \end{aligned}$$

where $\phi_m = 80^\circ$, $\alpha_m = 45^\circ$, $c_\alpha = 3.3$; the motion is symmetric for the up- and downstroke and depicted in figure 3.4.1 (a-b). The rectangular wing and the wing coordinate system are illustrated in figure 3.4.1 (c). We normalize the distance from pivot to tip to unity, which yields $a = 1.6667$, $b = 0.0667$, $B = 0.4167$ and a wing thickness of $h = 0.04171$. The Reynolds number is set to $\mathrm{Re} = U_{\mathrm{tip}} B/\nu = 100$ with $U_{\mathrm{tip}} = 2\pi\phi_m$, yielding the kinematic viscosity $\nu = 0.0366$. In the present simulations, we discretize the domain of size $3 \times 3 \times 3$ using $512 \times 512 \times 512$ points and a penalization parameter of $C_\eta = 1.25 \cdot 10^{-4}$ ($K = 0.365$). The reference computation is performed in a domain of size $4.16 \times 4.16 \times 4.16$, using a fine grid near the wing and a coarse one in the far-field. Based on the resolution of the fine grid, $\Delta x = B/50$, the equidistant resolution would be 500^3. In our simulations, the body reference point is located at $\underline{x}^{(g)}_{\mathrm{cntr}} = (1.5, 1.5, 1.7)$ and coincides with the wing pivot point, thus $\underline{x}^{(b)}_{\mathrm{pivot}} = 0$. The orientation of the body coordinate system is given by $\psi = 0^\circ$, $\beta = -45^\circ$, $\gamma = 45^\circ$ and $\eta = -45^\circ$, where the 45° yaw angle was added to keep the wing as far away from the vorticity sponges as possible. A total of four strokes has been computed, starting with a quiescent initial condition, $\underline{u}(\underline{x}, t = 0) = 0$. The outer boundary conditions on the domain are homogeneous Dirichlet conditions in z-direction and a vorticity sponge, extending over 32 grid points with $C_{\mathrm{sp}} = 10^{-1}$, in the remaining ones. The simulation required 35 GB of memory and 5785 CPU hours on 1024 cores. A total of 27 701 time steps was performed.

The resulting time series of the vertical force is illustrated in figure 3.4.1 (d). It takes the first two wingbeats to develop a periodic state, since the motion is impulsively started, and then the following strokes are almost identical. The present solution agrees well with the reference solution, and the relative R.M.S difference is $\|F - F_{\mathrm{ref}}\|_2 / \|F_{\mathrm{ref}}\|_2 \approx 4\%$ over the last two periods.

3.5 Hovering flight of a fruit fly model

A major difficulty in validating numerical codes for insect flight is the necessity of sufficiently precise experimental data, including all details of the apparatus, like the wing shape, axis of rotation, position of pivot points and time

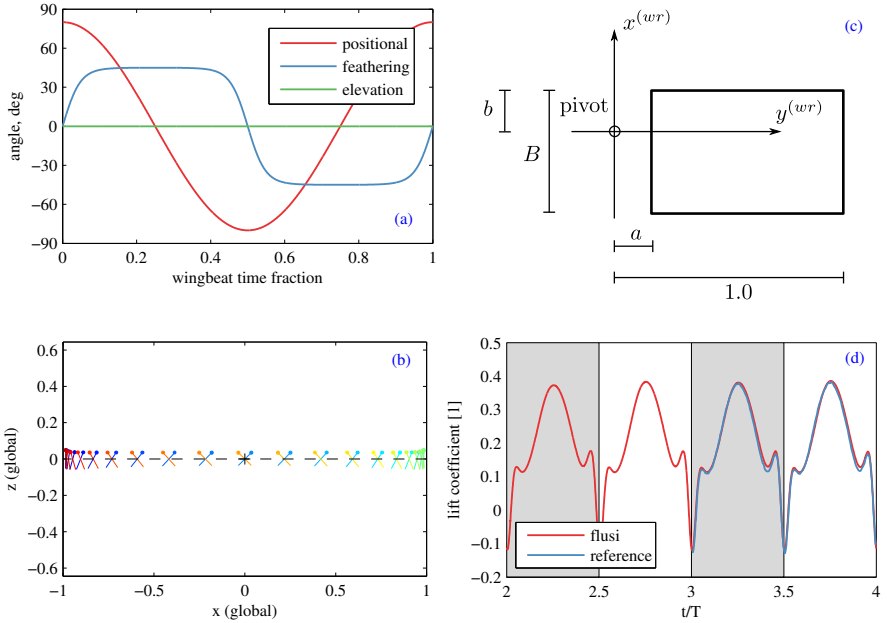


Figure 3.4.1 – Flapping rectangular wing. Top left: kinematics used in the test case, as given by Suzuki et al. [158]. Top right: Geometry of the flapping rectangular wing. Contrary to [158], we normalize the distance pivot-wing tip to unity. Bottom left: visualization of the wing kinematics by a chord section (without body, color represents time) Bottom right: time evolution of the vertical force acting on the wing for the computed four strokes, with the reference solution presented in [158], which was only shown for the fourth stroke.

series of the rotation angles and stroke plane. Some numerical contributions tried to compare with the experiments by Dickinson et al. [29], for example [158, 179]. The agreement is at best qualitative and of the right order of magnitude, which can be attributed to uncertainties in the input parameters.

For a more precise validation, we instead rely on a numerical solution as reference, circumventing the mentioned difficulties. The model insect is a fruit fly, *Drosophila melanogaster*, of which numerical simulations have been presented by Maeda and Liu [103]. Their simulations are based on the chimera approach, i.e., a body-fitted grid for the wings ($65 \times 65 \times 11$ each) and the body ($61 \times 61 \times 9$), as well as a background grid ($161 \times 141 \times 127$) have been used to solve the incompressible Navier–Stokes equations, approximated in the artificial compressibility formulation, using a finite volume discretization.

The fruit fly considered is defined in figure 3.5.1. The wing length from pivot

point to wing tip, $R = 2.47\,\mathrm{mm}$[1], the fluid density $\varrho_f = 1.225\,\mathrm{Kg/m^3}$ and the wingbeat frequency $f = 218\,\mathrm{Hz}$ are used for normalization. The body, depicted in figure 3.5.1 (a,c), has an elliptical cross-section with center points following an arched centerline of radius $r_{\mathrm{bc}} = 0.94644$ centered at $x_{\mathrm{bc}}^{(b)} = -0.244769$, $z_{\mathrm{bc}}^{(b)} = -0.9301256$. The wing pivot points are located at

$$\underline{x}_{\mathrm{pivot,rl}}^{(b)} = (-0.12,\ \pm 0.1445,\ 0.08)\,.$$

The insect is tethered at $\underline{x}_c^{(g)} = (1.6,\ 1.6,\ 1.9)$ in a computational domain of size $3.2 \times 3.2 \times 3.2$, discretized with $640 \times 640 \times 640$ Fourier modes and a penalization parameter of $C_\eta = 1.15 \cdot 10^{-4}$ ($K = 0.23$). Its body orientation is given by $\psi = 0$, $\beta = -45°$, $\gamma = 45°$ and $\eta = -45°$, where we added the yaw rotation to minimize the influence of periodicity. The fruit fly hovers, thus the body position and orientation does not change in time. A total of 4 wing beats has been computed, and the initial condition is fluid at rest, $\underline{u}\,(\underline{x}, t = 0) = \underline{0}$. We apply a vorticity sponge in the x and y direction (32 grid points thick with $C_{\mathrm{sp}} = 10^{-1}$) and impose no-slip boundary conditions in the z-direction, i.e., we impose a floor.

The wing shape is illustrated in figure 3.5.1 (d). It has a mean chord $c_m = A/R = 0.33$, which yields with the kinematic viscosity of air, $\nu = 1.5 \cdot 10^{-5}\,\mathrm{m^2/s}$ the Reynolds number

$$\mathrm{Re} = \frac{U_{\mathrm{tip}} c_m}{\nu} = \frac{2\Phi R f c_m}{\nu} = 136$$

where $\Phi = 2.44\,\mathrm{rad}$ is the stroke amplitude of the positional angle. The time evolution of the wing kinematics is illustrated in figure 3.5.1 (b). It is inspired by but not identical to the measurements by Fry et al. [57], and thus also slightly from figure 3.1.3. The up- and downstroke are not symmetric, and the wing trajectory, figure 3.5.1 (a), shows the characteristic $\cup$-shape.

The results for the lift force, normalized with the weight, and the aerodynamic power in W/kg body mass are depicted in figure 3.5.2. The normalization is chosen because it yields more intuitive values. As the mass of the model insect was not given in [103], we assumed the fourth stroke of the reference data to balance the weight, yielding $m = 1.02\,\mathrm{mg}$. Small circles at mid-stroke indicate stroke-averaged values. For the stroke-averaged vertical force, we find 10.03%, 6.19%, 4.03% and 3.26% relative difference to the reference data for the four strokes computed, which can be explained by the impulsively started motion at the beginning of the first stroke. The time evolution, e.g., the occurrence of peaks, are similar in both data. The agreement is even better

[1] The value $R = 2.39\,\mathrm{mm}$ in [103] is the distance root-tip and not pivot-tip, according to the authors.

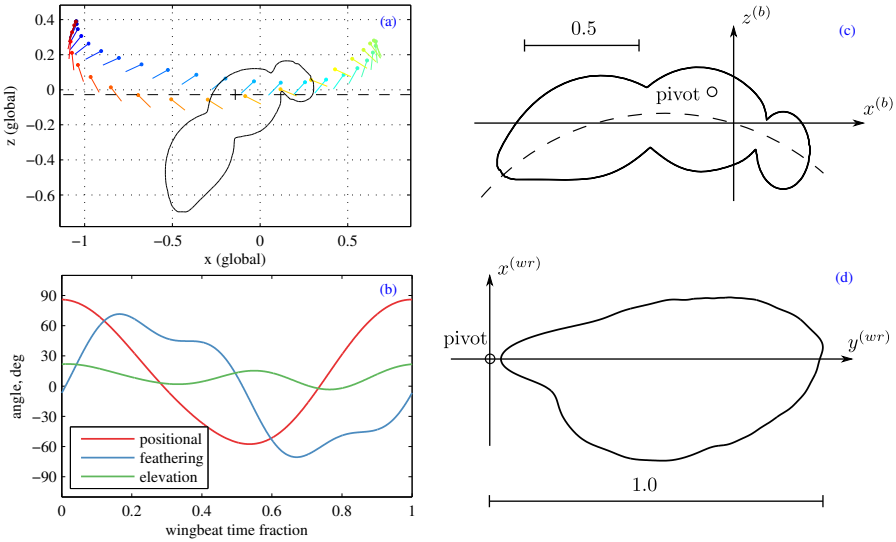


Figure 3.5.1 – Hovering fruit fly, for comparison with Maeda and co-workers. Left: overview of the setup in perspective view. Sponges are applied in the x and y-direction, while in the z-direction a floor is imposed. Top right: To-scale drawing of the insect's body in the plane $y^{(b)} = 0$. The body is rotationally symmetric with a circular center line. Bottom right: To-scale drawing of the wing shape, together with its pivot point.

for the aerodynamic power, with relative differences of 0.57%, 0.06%, −0.95% and −1.00% for the stroke-averaged values. Both power and lift peak during the translation phase of the up- and downstroke, and reach a minimum value at the reversals. This is remarkably different from the patterns suggested in [29], in which peaks occur near the reversals, and may be linked to the non-zero deviation angle θ.

The flow field generated by the fruit fly model is visualized in figure 3.5.3 by iso-surfaces of the Q-criterion, for two time instants during the downstroke. For incompressible flows, the Q-criterion can be computed as $Q = \nabla^2 p/2$, see [94, p. 23]. The first snapshot at $t/T = 0.2$ is during the early downstroke, when the maximum angle of attack (cf. figure 3.5.1 (b)) is reached, i.e., it is at the beginning of the translation phase. The leading edge vortex on both wings is already formed, and the lift production is close to its peak (cf. figure 3.5.2). A vortex ring can be observed at the left wing, connecting the wing root at tip. It is formed when the wing begins to change direction at the stroke reversal and therefor termed starting vortex. The tip vortex, however, is continuously formed throughout the stroke, as can be seen in the

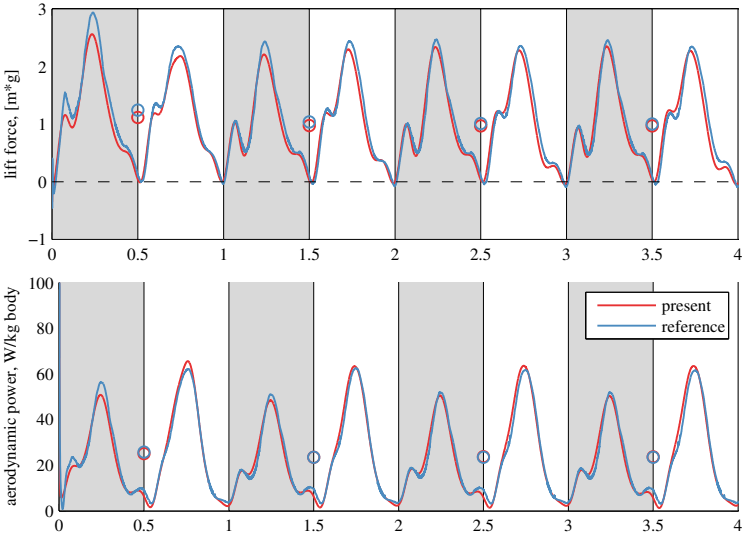


Figure 3.5.2 – Hovering fruit fly, comparison with the results obtained by Maeda and co-workers. Top: wing kinematics over time, note that all strokes have the same kinematics. Middle: total vertical force, stroke average indicated by circles. Bottom: aerodynamic power. Gray shaded areas denote upstrokes. The mean values during the last stroke differ by 3.26% and 1.00% for the vertical force and the power, respectively. The overall agreement is good.

second snapshot at $t/T = 0.4$, which is shortly before stroke reversal. It is transported downwards by the fluid flow, since the vertical force production requires to accelerate fluid momentum in that direction. The first snapshot shows the tip vortex shed during the previous upstroke, which is at that time a coherent tube. In the second snapshot, it is already largely dissipated by the fluid viscosity and no longer connected. The leading edge vortex can also be observed at $t/T = 0.4$, illustrating that it indeed remains stably attached throughout the stroke.

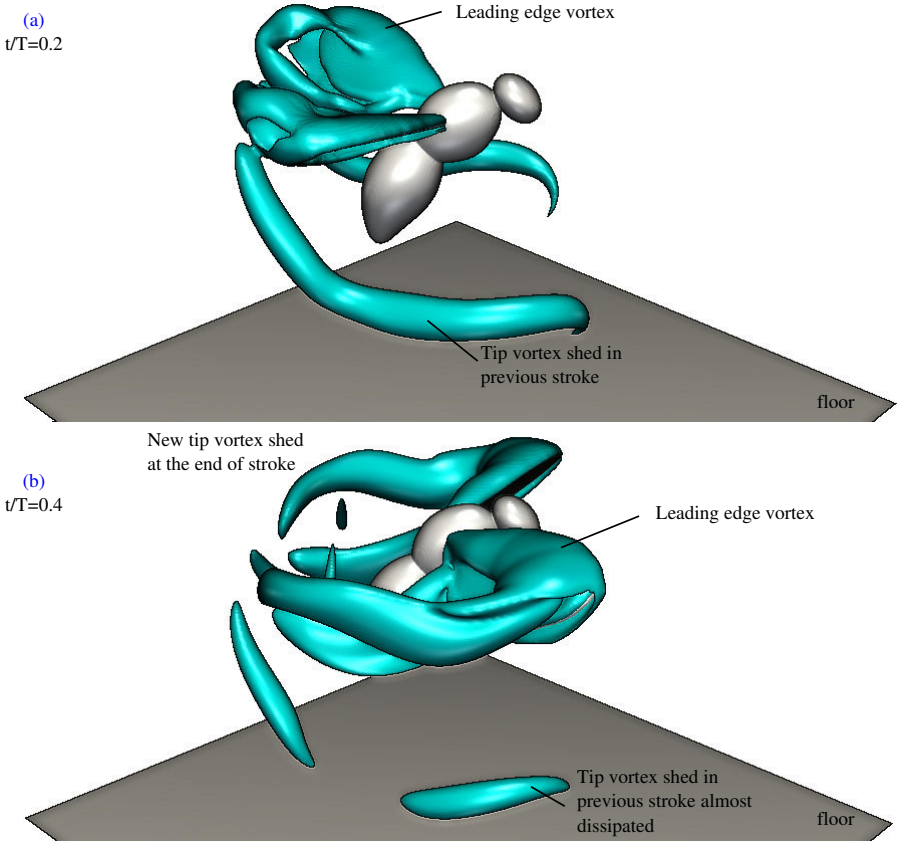


Figure 3.5.3 – Snapshot of the vortical structures generated by a hovering fruit fly model, visualized by iso-surfaces of the Q-criterion ($Q = 100$), at two different times of the downstroke (a) at the beginning of the downstroke $t/T = 0.2$ and (b) shortly before the ventral stroke reversal $t/T = 0.4$. The leading edge vortex is formed at the beginning of the downstroke, and at $t/T = 0.2$ is already visible, and remains stably attached until the ventral stroke reversal. The tip vortex shed in the previous upstroke is visible in (a) and almost dissipated in (b).

Chapter 4

Simulation of a model bumblebee in turbulent flow

The content of this chapter is adopted from [43].

4.1 Introduction

Insect flight currently receives considerable attention from both biologists and engineers. Its fundamental aerodynamics were first explored assuming that insects move in artificially quiescent air. However, natural environment is usually turbulent and the influence of atmospheric turbulence including canopy flows on aerodynamic forces and flow structures in flapping flight are vast research areas [58, 181]. Although numerical simulations allow better control of flow parameters, only experimental contributions exist so far [24, 124, 137, 170].

Field studies show variations of insect behavior with changing weather conditions, suggesting a pronounced dependency of aerodynamic force generation from atmospheric turbulence [159]. The interaction between insects and turbulence is complex because it includes sensing, wing motion, body kinematics, and mechanical responses of the flight apparatus. The behavior of wild orchid bees flying freely in an turbulent air jet has been studied by Combes and Dudley [24]. They found that turbulent flow conditions have a destabilizing effect on the body, most severe about the animal's roll axis. In response to this flow, orchid bees try to compensate the induced moments by an extension of their hindlegs, increasing the roll moment of inertia. Interaction of bumblebees with wake turbulence has also been considered by Ravi et al.

[137]. Their experiments were performed in a von Kármán-type wake behind horizontal and vertical cylinders that produced large vortical flow structures. Under these conditions, bees display large rolling motions, pronounced lateral accelerations, and a reduction in their upstream flight speed. Vance et al. [170] performed a comparative study on the sensitivity of bees and stalk-eye flies to localized wind gusts. The studied flight sequences were divided into perturbation and recovery periods, each of them consisting of a passive phase with symmetric wing kinematics and an active phase with asymmetric kinematics. The authors found that bees and stalk-eye flies respond differently to aerial perturbations, either causing roll instabilities in bees or significant yaw rotations in stalk-eye flies. Ortega-Jimenez et al. [124] analyzed feeding flights of Hawkmoths in vortex streets past vertical cylinders. Depending on distance of the animal from the cylinder and cylinder size, they observed destabilizing effects on yaw and roll and a reduction in the animal's maximum flight speed. Kinematic responses to large helical coherent structures were also found in Hawkmoths flying in a vortex chamber [125]. A study on the energetic significance of kinematic changes in hummingbird feeding flights, further demonstrated a substantial increase in metabolic rate during flight in turbulent flows, compared to flight in undisturbed laminar inflow [126, 136].

It is challenging to conduct experiments with living, freely flying animals in turbulent flows because of their complex, sensor-dependent changes in wing kinematics and wing-wake interaction. Besides vision-guided responses, animals also change body posture and flight heading according to feedback from their antennae that detects wind speed and odors. It has been shown that vortex streets produce odor plumes that provide cues for navigation, which are not available in laminar flows [119]. To isolate specific effects of turbulence on aerodynamic mechanisms and power expenditures in flight, numerical simulations are useful tools with which parameters of flows and wake dynamics may be studied beyond natural limits including entire instantaneous flow fields.

In this chapter, we focus on the impact of turbulence on specific aerodynamic features used for force production in forward flight. The model insect is considered to be tethered and the wing motion is prescribed regardless of the flow conditions. As described in the previous chapter, insect flight relies on a leading edge vortex for force production, and a portion of the power expenditures is spent on the tip vortices [88]. The question we address is whether these features are systematically altered by turbulent perturbations, e.g, the leading edge vortex is destabilized and detaches from the wing. This destabilization would have tremendous consequences on the force production. The choice of the model insect, the bumblebee *bombus terrestris,* is motivated by the fact that they are all-weather foragers, and thus encounter a particularly

large variety of natural flow conditions [137]. Bumblebees also are medium sized (wing length 13.2 mm), thus the associated Reynolds numbers are sufficiently large. Furthermore, this species has been considered in the literature, both in experimental [33, 34] and numerical work [180, 183], thus various data are available.

The remainder of this chapter is organized as follows. In section 4.2, we present the model bumblebee, i.e., the wing and body geometry and other parameters. We then discuss the wingbeat kinematics suggested in the literature in section 4.2.2, and derive the kinematics that lead to force and torque balanced flight under laminar flow conditions. The laminar case also serves as reference and orientation prior to studying the model in turbulence, and we thus discuss some characteristics of the wake turbulence generated by the insect itself in section 4.2.3. In section 4.3, we study the model with turbulent inflow conditions. To this end, we discuss homogeneous isotropic turbulence as canonical model in section 4.3.1, and discuss how it is rescaled and imposed as inflow condition in sections 4.3.2 and 4.3.3 respectively. The results of the study are presented thereafter.

4.2 Bumblebee forward flight in laminar flow

4.2.1 Bumblebee model

The parameters of the bumblebee model are derived from [33], case BB01. The animal's typical body mass, M, is approximately 175 mg and wing length, R, amounts to 13.2 mm. We assume the insect to be composed of linked rigid bodies and model the insect's body shape by sweeping an elliptical section of variable size along a curvilinear centerline. Compared to other insects, bumblebees have relatively thick legs that potentially create non-negligible aerodynamic effects. We thus include all legs, the proboscis and the antennae in our model as circular cylindrical sections joined by spheres, and further assume bilateral symmetry of the insect. Figure 4.2.1A-B shows side and top views of the modeled animal with all linear dimensions normalized to wing length. We numerically compute the inertia tensor, $\underline{\underline{I}}$, of the bumblebee neglecting wings and assuming uniform body density of $\varrho_b = M/V = 362\,\mathrm{kg/m^3}$, where $V = 0.48\,\mathrm{cm^3}$, from equation

$$\begin{aligned}\underline{\underline{I}} &= \varrho_b \int_v \left((\underline{r}\cdot\underline{r})\,\underline{\underline{E}} - \underline{r}\otimes\underline{r} \right) dV \\ &= \begin{pmatrix} 0.183 & 0 & 0.1307 \\ 0 & 0.400 & 0 \\ 0.1307 & 0 & 0.339 \end{pmatrix}.\end{aligned}$$

Diagonalizing the inertia tensor yields the principal moments of inertia $I_{11} = 0.1092$, $I_{22} = 0.3998$ and $I_{33} = 0.4136$, all in units of $10^{-8}\,\mathrm{kgm}^2$. The data show that moment of inertia about the roll axis (x-axis) is approximately 4-times smaller than about the yaw and pitch axes, which is in agreement with [137] and [24].

Wing contour was digitized from [1] and the area scales to $48.37\,\mathrm{mm}^2$ of a single wing and 3.66 mm mean wing chord, c (figure 4.2.1E). The wingbeat kinematics are shown in figure 4.2.1C-D; details about their modeling are given in section 4.2.2. The flight speed is set at the intermediate value of 2.5m/s and the wingbeat frequency f is 152 Hz according to values previously measured in freely flying bumblebees [33, 183]. The inclination, η, of the stroke plane against the longitudinal body axis is 37.5°, and body pitch angle, β, is 24.5° with respect to horizontal (Fig. 4.2.1 C). Modeled wings flap at mean wing tip velocity of $U = 2\Phi Rf = 8.75\,\mathrm{m/s}$ and a Reynolds number of 2042 based on U, c and the kinematic viscosity of air at 300 K, $\nu = 1.57 \cdot 10^{-5}\,\mathrm{m}^2/\mathrm{s}$. The fluid density is $\varrho_f = 1.177\,\mathrm{kg/m}^3$.

4.2.2 Wingbeat kinematics and aerodynamic forces and power

The present model is based on the work of Dudley and Ellington [33, 34]. This contribution is relatively old and lacks the precision of modern high-resolution high speed cameras. It is however the only published data on actual forward flight kinematics. The time evolution of the angles and the stroke trajectory are illustrated in figure 4.2.2 (top). Especially the feathering angle is subject to uncertainty, since only 8 points during the cycle are given. Figure 4.2.2 (top) therefore shows the result of a curve fitting process.

To evaluate the kinematics, we carry out a computation of the bumblebee tethered at $\underline{x}_{\mathrm{cntr}} = (2,2,2)$ in a $6 \times 4 \times 4$ domain, with the mean flow imposed to $u_\infty = 1.246$, which corresponds to 2.5 m/s. Vorticity sponges are used at the in- and outlet of the domain to remove the wake. The penalization parameter is $C_\eta = 2.5 \cdot 10^{-4}$ ($K = 0.074$), and the spatial resolution $1152 \times 768 \times 768$. A reference computation at even higher resolution ($1536 \times 1024 \times 1024$) yields no significant difference and thus confirms that the lower resolution is sufficient. We compute four strokes, and the cycle-averaged aerodynamic force, moment and power are given in table 4.1 for the last cycle. We find that the lift force is 1.14 the weight force, i.e., excess lift is produced, as well as a thrust force of 0.373. The model would thus accelerate. The roll and yaw moments are zero, owing to the bilateral symmetry, and we find a significant pitch moment. The latter finding however is not a major concern, since the point of reference, $\underline{x}_{\mathrm{cntr}}$, is set arbitrarily, and the actual center of gravity is unknown. A nonzero mean pitch moment could thus in principle be balanced by the weight. The deviation from unit lift is

already notable, and the thrust force of more than one third the weight is unacceptable. The worst feature is however the power requirement of 165 W/kg body mass, which is significantly larger than those values reported by Dudley and Ellington [34] (56 W/kg body mass), Berman et al. [12] (53 W/kg body mass) and Ellington [38] (88 W/kg body mass). Muscles cannot produce that much power, especially not the instantaneous aerodynamic power, which reaches peaks well above the cycle-average value.

Sun and co-workers likewise reported that they were unable to achieve force and torque balanced flight using the original set of kinematics, although the power was not presented [180, 183]. In these contributions, the authors suggest the use of a simplified kinematics set, consisting of a sinusoidal positional angle, constant deviation and piecewise constant feathering angle with a sinusoidal transition during the time interval τ. The time evolution of the angles during the cycle are given by

$$\phi = \overline{\phi} + \Phi \cos(2\pi t) \tag{4.2.1}$$

$$\theta = \theta_0 \tag{4.2.2}$$

$$\alpha = \begin{cases} \alpha_{\text{down}} - A\left(t - \frac{\tau}{2} - \frac{\tau}{2\pi}\sin\left(2\pi\frac{t-T_1}{\tau}\right)\right) & 0 < t < \frac{\tau}{2} \\ \alpha_{\text{down}} & \frac{\tau}{2} < t < T_1 \\ \alpha_{\text{down}} + A\left(t - T_1 - \frac{\tau}{2\pi}\sin\left(2\pi\frac{t}{\tau}\right)\right) & T_1 < t < T_2 \\ \alpha_{\text{up}} & T_2 < t < T_3 \\ \alpha_{\text{up}} - A\left(t - T_3 - \frac{\tau}{2\pi}\sin\left(2\pi\frac{t-T_3}{\tau}\right)\right) & T_3 < t < 1 \end{cases} \tag{4.2.3}$$

$$T_1 = \frac{1}{2} - \frac{\tau}{2}; \quad T_2 = T_1 + \tau; \quad T_3 = 1 - \frac{\tau}{2}; \quad A = \frac{\alpha_{up} - \alpha_{down}}{\tau}.$$

These equations contain six adjustable parameters, α_{down}, α_{up}, τ, $\overline{\phi}$, Φ and θ_0, but the choices presented in [180, 183] did not lead to balanced flight using our model. We thus proceed and instead derive the parameters from Dudley's data, yielding $\overline{\phi} = 24°$, $\Phi = 62.5°$, $\theta_0 = 12.55°$, $\alpha_{\text{down}} = 50°$, $\alpha_{\text{up}} = 0°$ and $\tau = 0.22$, which however still did not yield acceptable results for the forces and the power. Since the feathering angle α is the most uncertain in the original data, we varied it until balanced flight was achieved, similar to [180, 183], yielding $\alpha_{\text{down}} = 70°$ and $\alpha_{\text{up}} = -40°$. This set of kinematics is illustrated in figure 4.2.2 (center), and we term it "simplified & balanced". The aerodynamic measures (table 4.1) show that this set of kinematics is a good choice. Especially the aerodynamic power has dropped by a factor of two with respect to the kinematics presented by Dudley, and the forces and moments are a good approximation for balanced flight.

However, when comparing the simplified and original kinematics in the stroke trajectories shown in figure 4.2.2, the constant deviation angle is a striking difference, since the figure-of-eight shape is lost. As a last step, we thus keep

Kinematics set	$F_h\ [Mg]$	$F_v\ [Mg]$	P_{aero}	M_{roll}	M_{pitch}	M_{yaw}
Dudley [33]	-0.37	1.14	165.00	0	-0.307	0
Simplified & Balanced	-0.08	1.02	84.05	0	0.01	0
Simplified figure-of-eight	-0.14	1.07	96.30	0	0.03	0

Table 4.1 – Stroke averaged aerodynamic measures for the different kinematics sets. Forces are normalized by the weight force Mg, moments by MgR, power in W/kg body mass.

ϕ and α according to eqn. (4.2.1,4.2.3), respectively, with the parameters found previously for balanced flight. The deviation angle θ, eqn. (4.2.2), is replaced by the data reported by Dudley. This set of kinematics is shown in figure 4.2.2 (bottom) and termed "simplified figure-of-eight". The cycle-averaged flight data is reported in table 4.1. The thrust force is observed to be almost twice as large as in the previous case, and the excess lift increases as well. The aerodynamic power is about 14% higher. Based on this data, the simplified & balanced kinematics (figure 4.2.2 center) are preferred, and we employ them for the remainder of this chapter.

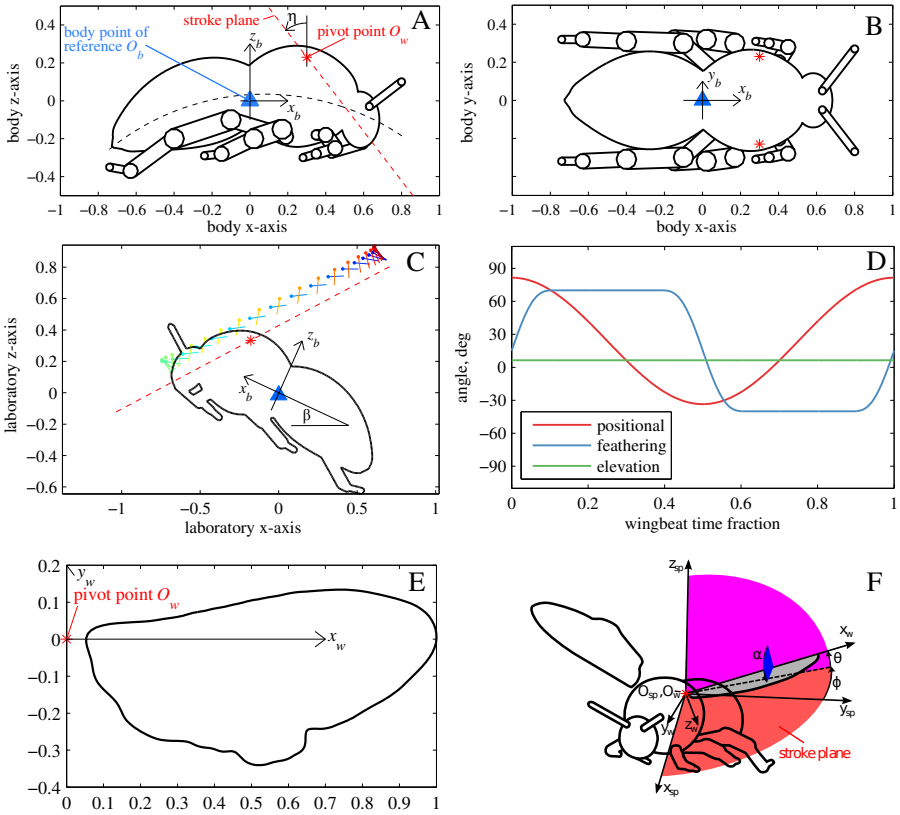


Figure 4.2.1 – Bumblebee model. (A-B) side and top view of the bumblebee body with the definitions of the body point of reference and the wing pivot points. The body is obtained by sweeping a rotationally symmetric body around an arc (black dashed line). The wings rotate around the pivot points (red stars) within the stroke plane, which is inclined by $\eta = 37.5°$ with respect to the body normal. The entire body is inclined by $\beta = 24.5°$ with respect to the horizontal (C). The two wings follow the symmetric motion protocol illustrated in D. The wing geometry is illustrated in E, where the distance pivot point - wing tip is normalized to unity. Wing angles are defined in F.

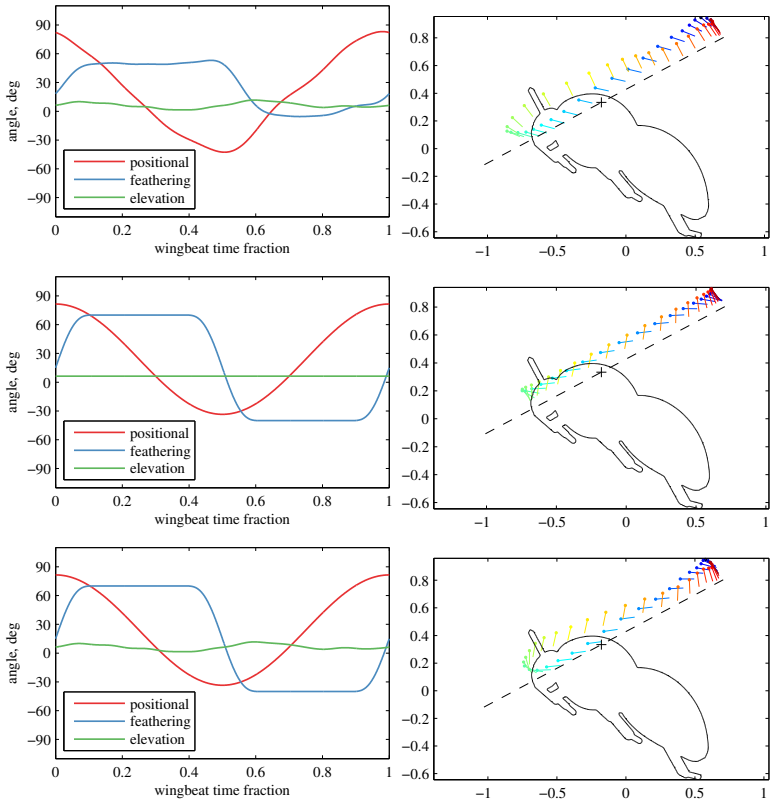


Figure 4.2.2 – Bumblebee kinematics. Top row: data extracted from experimental work by Dudley and Ellington [33]. Mid row: Simplified and balanced kinematics used here. Bottom: Simplified ϕ and α, but θ from the experimental data, yielding the figure-of-eight motion type.

4.2.3 Wake turbulence generated by the bumblebee

Since the properties of turbulence encountered by bumblebees are essentially unknown, we consider the turbulence generated by the insect itself as an orientation. We visualize the instantaneous flow field at $t/T = 0.3$, i.e., during the downstroke, in figure 4.2.3 by the $\|\underline{\omega}\| = 100$ isosurface of vorticity magnitude. Besides the typical vortex features of flapping insect flight, namely the leading edge and tip vortices, small scaled structures are visible in the wake.

To further analyze the flow properties, we compute the turbulence intensity Tu, which we define as the RMS velocity normalized by the free-stream velocity u_∞, $Tu = \sqrt{2\overline{e'_{\text{kin}}}/3}/u_\infty$. In the latter equation, e'_{kin} is the turbulent kinetic energy defined as $\underline{u}' \cdot \underline{u}'/2$, where $\underline{u}'$ is the instantaneous turbulent velocity, which is defined as $\underline{u}' = \underline{u} - \overline{\underline{u}}$. The overline denotes time average. The computation of $\overline{e'_{\text{kin}}}$ can be tedious, as during the computation, the time-averaged velocity field is not available. It is computed on the fly during the simulation according to

$$\overline{\underline{u}^{n+1}} = \left(\underline{u}^n \Delta t^n + t^n \overline{\underline{u}^n}\right) / t^{n+1},$$

where $\overline{\underline{u}^n}$ is the value of $\overline{\underline{u}}$ after n time steps at time t^n. The final value of $\overline{\underline{u}}$ is of course available only after the simulation. To compute the time averaged turbulent kinetic energy, it is possible to use the identity

$$\overline{e'_{\text{kin}}} = \left(\overline{\underline{u}^2} - \overline{\underline{u}}^2\right)/2$$

to facilitate the computation, since both contributions can be computed on the fly. The resulting Tu is visualized in figure 4.2.3 by means of three semi-transparent isosurfaces. In the vicinity of the wing and directly behind the abdomen, relative turbulence intensity exceeds $Tu = 0.6$. In a distance $4R$ downstream, Tu dropped to 0.4, and the 0.2 isosurface is bounded by the outflow condition, indicating that the isosurface extends to more than $4R$, which is the limit of the computational domain.

The analysis of the model bumblebee thus shows that it generates peak turbulence intensities of up to 0.4 at distances relevant for a following insect. The flow condition may thus also be directly relevant to bumblebees flying closely to each other, for instance at the nest entrance, where wing–wake interaction likely occurs.

The wake turbulence intensity generated by the bumblebee is thus orders of magnitudes larger than in typical engineering applications. Mueller et al. [117], for instance, studied the influence of inlet turbulence on low Reynolds number airfoils, where "low" refers to values of about $2 \cdot 10^5$, experimentally in a wind tunnel with increased disturbance level. The turbulence intensities

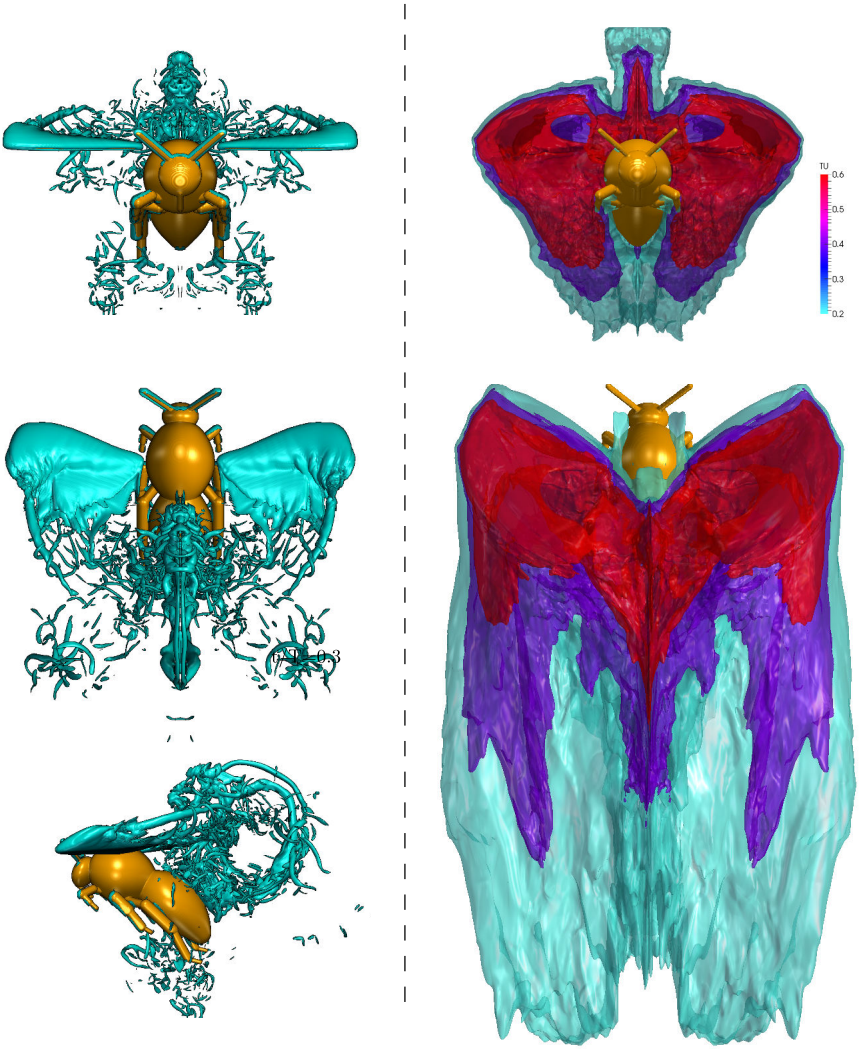


Figure 4.2.3 – Model bumblebee in forward flight under laminar inflow conditions. Left column: instantaneous flow field at $t/T = 0.3$, visualized by the $\|\underline{\omega}\| = 100$ isosurface. Right: isosurfaces of the turbulence intensity generated by the insect and normalized by free-stream velocity, $Tu = \sqrt{2\overline{u'^2}/3}/u_\infty$.

in that wind tunnel were below 1%, yet the flow field around the airfoil was extremely sensitive to the perturbations.

4.3 Bumblebee forward flight in turbulent flow

The simulations of the bumblebee model in laminar inflow presented above are used to validate our model, and the wake turbulence generated by the flapping wings gives an estimate of relevant turbulence intensities. In the following, we study bumblebee flight in turbulent air.

4.3.1 Model turbulence: Homogeneous isotropic turbulence

Insects successfully fly in turbulent environments [124, 137, 170]. However, the kind of aerial perturbation they actually encounter is not well known and depends on a large set of parameters, like the weather, landscape, flight altitude, etc. Atmospheric turbulence and in particular the small scale vortical structures are vast research areas [58, 181].

We model the inflow perturbations by homogeneous isotropic turbulence (HIT). This is a reasonable assumption for the small turbulent scales relevant to insects, and HIT is a well established "numerical experiment" [140]. As the name suggests, all three spatial directions are periodic, and thus eqns (2.1.1a-2.1.1b) are solved without boundary conditions. In a triply periodic setup, Fourier spectral methods are particularly beneficial because of their favorable numerical properties.

In general, such a turbulence simulation is performed by starting from a random, solenoidal initial condition with prescribed spectrum, which is evolved according to the incompressible Navier–Stokes eqn. The energy dissipation by the friction term is, approximately or exactly, compensated by a forcing term, that feeds energy into the system. This forcing usually acts on the largest few wavenumbers. Scales smaller than the forcing scale will then develop according to the Navier–Stokes equations, and the flow field converges towards a statistically steady state, which is independent of the initial condition.

For convenience, we consider only domains of size $(2\pi)^3$. Our numerical code is de-aliased using the 2/3-rule, thus the largest wavenumber retained by the simulation is $k_{\max} = 2\,(N/2-1)\,/3$. For all isotropic turbulence simulations we use the Runge–Kutta 4 time stepper with explicit treatment of the friction term, since the time step is limited by the CFL condition.

Homogeneous isotropic turbulence can be characterized by a variety of inte-

gral quantities that are summarized in the following:

$$E\left(|\underline{k}|\right) = \frac{1}{2} \sum_{k-\frac{1}{2} \leq |\underline{k}| \leq k+\frac{1}{2}} \left|\widehat{\underline{u}}'\left(\underline{k}\right)\right|^2 \qquad \text{Radial energy spectrum}$$

$$\epsilon = 2\nu \int_0^{k_{\max}} |\underline{k}|^2 \left|\widehat{\underline{u}}'\left(\underline{k}\right)\right|^2 \mathrm{d}k = \nu \left\langle \underline{\omega} \cdot \underline{\omega} \right\rangle \qquad \text{Mean rate of energy dissipation}$$

$$e' = \frac{1}{2}\left\langle \underline{u}' \cdot \underline{u}' \right\rangle = \frac{3}{2}U^2 = \int_0^{k_{\max}} E\left(|\underline{k}|\right) \mathrm{d}k \qquad \text{Mean turbulent kinetic energy}$$

$$\Lambda = \frac{\pi}{2U^2} \int_0^{k_{\max}} k^{-1} E\left(|\underline{k}|\right) \mathrm{d}k \qquad \text{Integral (length)scale}$$

$$\lambda = \left(\frac{15\nu U^2}{\epsilon}\right)^{1/2} \qquad \text{Taylor Micro(length)scale}$$

$$\ell_\eta = \left(\frac{\nu^3}{\epsilon}\right)^{1/4} \qquad \text{Kolomogorov length scale}$$

$$\tau_\eta = \left(\frac{\nu}{\epsilon}\right)^{1/2} \qquad \text{Kolomogorov time scale}$$

$$u_\eta = \left(\nu\epsilon\right)^{1/4} \qquad \text{Kolomogorov velocity scale}$$

$$T_0 = \frac{\Lambda}{U} \qquad \text{Eddy turnover time}$$

$$U = \sqrt{\frac{1}{3}\left\langle \underline{u}' \cdot \underline{u}' \right\rangle} \qquad \text{RMS velocity}$$

$$R_\lambda = \frac{\lambda U}{\nu} \qquad \text{Turb. Reynolds number}$$

The time average is denoted by the overbar and primes denote turbulent fluctuation quantities, e.g., $\underline{u}' = \underline{u} - \overline{\underline{u}}$, where $\overline{\underline{u}} = 0$ in homogeneous isotropic turbulence. Spatial average is denoted by $\langle\rangle$. The radial energy spectrum is the energy of a shell in the three-dimensional wavenumber space. A typical spectrum consists of the energy-carrying range at low wavenumbers, the inertial range which follows the well-known $-5/3$ power law, and the dissipative range at the highest wavenumbers. The turbulent kinetic energy e' is dissipated by the dissipation rate ϵ, both are spatial averages.

Turbulence is a multiscale phenomenon and a variety of important scales can be defined. The integral scale Λ is a representative length for large, energy-carrying structures, and the Taylor mirco scale λ is a length scale in the inertial range of the spectrum. Both can also be computed from autocorrelations of the velocity. In the dissipative range, the Kolmogorov scales ℓ_η, u_η and τ_η are relevant. By definition, the Reynolds number based on these scales, $Re_\eta = u_\eta \ell_\eta / \nu$ is unity.

In many simulations, scientists have aimed for the highest possible Reynolds number on a given grid, in which case the product $k_{\max}\ell_\eta$ is ≈ 1, meaning that the Kolmogorov length scale is the smallest resolved spatial structure. In any case, independent of ℓ_η, it should further be assured that the Taylor macroscale is smaller than the box size, $\Lambda < 2\pi$, because otherwise the effect of periodicity spoils the results.

As stated previously, simulating isotropic turbulence requires a suitable forcing term that feeds energy into the system. Otherwise, all energy would eventually be dissipated. A variety of different forcing schemes have been reported in the literature. We consider two alternatives here, the method by Machiels [101] and the method by Jimenez et. al [79].

The first forcing method we use here is the one used by Machiels [101]. The forcing term acts on the modes with $|\underline{k}| \leq K_f$, where $K_f = 2.5$. The method aims at forcing a given dissipation rate ϵ_f, which is convenient since it directly implies a given ℓ_η. The forcing is thus easy to adjust for a given resolution. We give the forcing term without derivation:

$$\underline{\widehat{f}}\left(\underline{k},t\right) = \begin{cases} \epsilon_f \underline{\widehat{u}}\left(\underline{k},t\right) / 2\int_0^{K_f} E\left(|\underline{k}|,t\right) & |\underline{k}| \leq K_f \\ 0 & \text{otherwise} \end{cases} \tag{4.3.1}$$

We will see, though, that the effective dissipation rate is not constant but oscillates with mean value $\epsilon = \epsilon_f$.

Another forcing method was originally proposed by Jiménez [79], and is also employed by Kaneda and co-workers in the current state of the art simulations. The forcing aims at keeping the turbulent kinetic energy, e', constant over time. It thus feeds the dissipated energy back in the system. The viscosity is then adapted to the numerical resolution. We start the derivation of the forcing term by noting the Navier–Stokes equation with the non-linear term written in divergence-form

$$\partial_t \underline{u} + \nabla \cdot (\underline{u} \otimes \underline{u}) \quad = \quad -\nabla p + \nu \nabla^2 \underline{u} + \underline{f}$$

Dot-multiplying this equation by $\underline{u}$ yields the energy equation, which is then averaged spatially. The non-linear term and the pressure gradient term then vanish due to periodicity, and one is left with

$$\partial_t \int_{\mathbb{T}} \frac{\underline{u}^2}{2} \mathrm{d}V \quad = \quad -2\nu Z + \int_{\mathbb{T}} \underline{f} \cdot \underline{u} \mathrm{d}V, \tag{4.3.2}$$

with the enstrophy $Z = \int_{\mathbb{T}} \frac{1}{2}\underline{\omega} \cdot \underline{\omega} \mathrm{d}V$. The last integral in eqn (4.3.2) should be evaluated in Fourier space, as the forcing term is conveniently defined there. Applying Parseval's theorem yields

$$\frac{\mathrm{d}}{\mathrm{d}t}\langle E \rangle = \epsilon + \sum_k \underline{\widehat{f}}_k \cdot \underline{\widehat{u}}_k^* \tag{4.3.3}$$

the ansatz for the forcing is

$$\widehat{\underline{f}}\left(\underline{k}\right) = \begin{cases} -c\widehat{\underline{u}}\left(\underline{k}\right) & |\underline{k}| \leq K_f \\ 0 & \text{otherwise} \end{cases}$$

where $c = c\left(t\right)$ is a $|\underline{k}|$-independent constant yet to be determined. For the kinetic energy to be constant, we demand the left hand side of eqn (4.3.3) to be zero, we are then left with

$$c = \frac{\epsilon}{\sum_0^{K_f} |\widehat{\underline{u}}|^2 \, dk} \tag{4.3.4}$$

By noting that the denominator is the energy in the forced modes, we show the similarity between the two forcing methods. They simply differ in that Machiels uses a pre-defined ϵ_f and Jiménez chooses the current dissipation rate ϵ.

To generate the initial condition, we follow Rogallo's work [140]. The aim is to generate a divergence-free, random velocity field imposing a given energy spectrum $E\left(|\underline{k}|\right)$. This can be achieved in Fourier space using

$$\widehat{\underline{u}}\left(\underline{k}, t=0\right) = \begin{pmatrix} \alpha \, |\underline{k}| \, k_y + \beta k_x k_z \\ -\alpha \, |\underline{k}| \, k_x + \beta k_y k_z \\ -\beta k_h^2 \end{pmatrix} / \, |\underline{k}| \, k_h$$

where $k_h = \sqrt{k_x^2 + k_y^2}$ and

$$\alpha = 2\sqrt{\frac{E\left(|\underline{k}|\right)}{4\pi \, |\underline{k}|^2}} \exp\left(\imath\Theta_1\right) \cos\left(\Phi\right); \qquad \beta = 2\sqrt{\frac{E\left(|\underline{k}|\right)}{4\pi \, |\underline{k}|^2}} \exp\left(\imath\Theta_2\right) \sin\left(\Phi\right).$$

The variables $\Theta_{1,2}$ and Φ are random numbers between 0 and 2π. The prescribed spectrum is in our case $E\left(|\underline{k}|\right) = \left(|\underline{k}|^4 / k_p^5\right) \exp\left(-2\left(|\underline{k}| / k_p\right)^2\right)$, where k_p is the wavenumber with the highest energy in the field; it is usually equal to two. The same spectrum is used, for instance, in [77]. The resulting flow field is then rescaled to have the desired enegery.

We compare both forcing methods using a 256^3 simulation with an initial turbulent kinetic energy $e' = 0.5$. The viscosity is set to $\nu = 2 \cdot 10^{-3}$. For Machiels forcing, we set $\epsilon_f = 0.0936$. The time evolution of energy, dissipation and R_λ are shown in figure 4.3.1. We first note that the kinetic energy e' is not perfectly constant using Jimenez forcing approach; it rather increases slightly with the simulation time. We should, however, note that we compute 50 eddy turnover times ($T_0 = 2.1$), which is longer than many simulations in

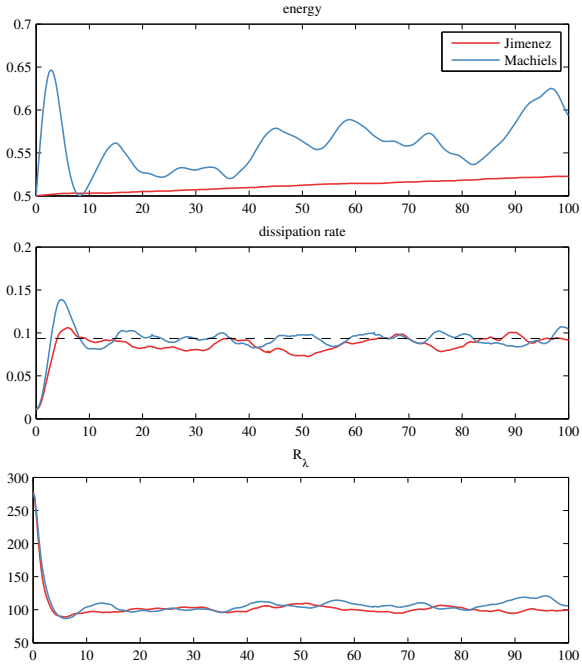


Figure 4.3.1 – Homogeneous isotropic turbulence. Time evolution of the turbulent kinetic energy e', the dissipation rate ϵ and the Reynolds number based on the Taylor-microscale R_λ, for both types of forcing discussed in the text. The dashed lines marks the dissipation rate set for Machiels forcing, $\epsilon_f = 0.0936$. Bottom right is the energy spectrum at $t = 100$.

turbulence research are performed. The reason is that we can save more uncorrelated velocity fields when running the simulation longer. Using Machiels forcing, fluctuations of roughly 20% the initial energy can be observed. It is rather surprising that the dissipation rate fluctuates in the same manner for both forcing types, and in Machiels case its mean value is indeed ϵ_f. The turbulent Reynolds number R_λ likewise is very similar in both forcing types and oscillates around $R_\lambda \approx 100$. Figure 4.3.2 (left) visualizes an instantaneous flow field by means of the $\|\underline{\omega}\| = 18$ isosurface of absolute vorticity. The flow exhibits the typical "vortex tubes" that have first been reported in [172] and then later in [79]. The energy spectrae are shown in figure 4.3.2 (right) for both forcing types. They are virtually identical, and exhibit only a small inertial range, since the Reynolds number is not high enough for a long inertial range to exist. In both cases, the simulation is slightly overresolved ($k_{\max}\ell_\eta \approx 2$).

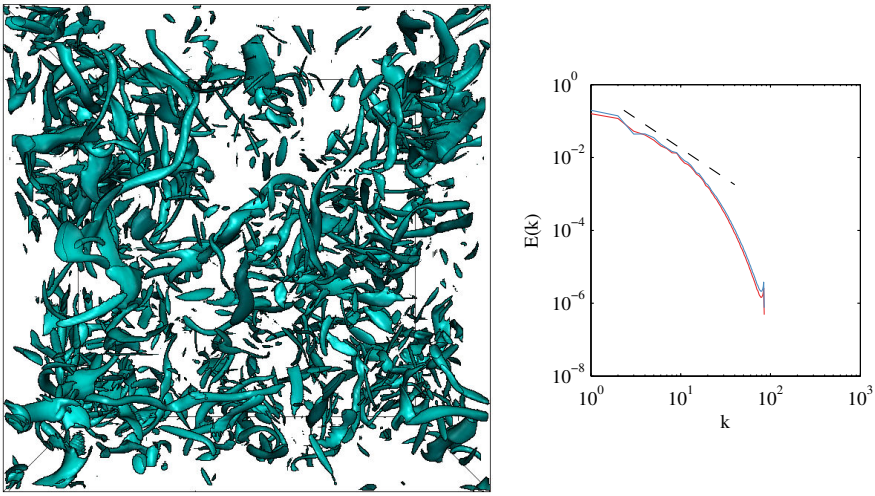


Figure 4.3.2 – Homogeneous isotropic turbulence. Left: visualization of the flow field by the $\|\underline{\omega}\| = 18$ isosurface of absolute vorticity. In the simulation R_λ is equal to 100, and the typical "vortex tubes" [79] are visible. Forcing method is negative viscosity (Jiménez). Right: energy spectrum for both forcing types, dahed line is $k^{-5/3}$.

4.3.2 Rescaling to insect dimensions

The isotropic turbulence data was computed conventionally in a domain of size $(2\pi)^3$, and before using it as inflow turbulence, it must be rescaled. For the sake of clarity, we henceforth state quantities from the HIT simulations with a tilde overset. Reynolds similarity requires keeping $Re = L_y \underline{u}'/\nu$ constant in both simulation, and thus the scaling relation is simply

$$\underline{u}' = \frac{2\pi\nu}{L_y\widetilde{\nu}}\,\widetilde{\underline{u}}'. \tag{4.3.5}$$

Note that eqn. (4.3.5) is defined only up to a free parameter, the lateral size of the computational domain in the insect flight simulation, L_y, which is arbitrary as long as the insect fits in the domain. Varying L_y is equivalent to changing the animal's size relative to the length scales of the turbulence field. Thus the parameter L_y needs to be large enough to reduce the effect of periodicity in the lateral direction and small enough to produce turbulent length scales similar to natural perturbations. We thus used an intermediate value for the lateral size of $4R$.

4.3.3 Numerical wind tunnel with turbulent inflow

The precomputed isotropic turbulence fields can now be used as turbulent inflow condition for a "numerical wind tunnel". Figure 4.3.3 illustrates the devised setup. A turbulent velocity field is rescaled according to eqn. (4.3.5), and upsampled by zero padding in Fourier space to the target resolution, since the latter is higher than the turbulence resolution. In the inlet section of the actual setup, the turbulent velocity field is superimposed on a laminar free stream velocity in axial direction, $\underline{u}_\infty = (u_\infty, 0, 0)$. The inflow velocity imposed through penalization thus reads

$$\underline{u}_s\left(x,y,z,t\right) = \underline{u}_\infty + \underline{u}'\left(x - u_\infty t, y, z\right).$$

The turbulence field is periodic, and thus we treat the x-coordinate in a periodic fashion, therefore we feed in the same field several times. The repetition time is $T_{\text{rep}} = L_y/u_\infty$. The upsampled turbulence field is interpolated linearily in the x-direction. At the outlet, we gradually damp the vorticity using a sponge. The basic idea of generating inflow perturbations by pre-computed HIT has also been reported in [104] in the context of "classical" simulations, i.e., without volume penalization and with strictly imposed Dirichlet inflow conditions. The strategy has also been extended to use wall-bounded pipe turbulence instead of HIT as turbulent inflow condition for jets, e.g., in [2].

We now place the bumblebee model in the turbulence-enhanced numerical wind tunnel.

4.3.4 Results

We studied the model insect in turbulent inflow considering four different turbulence intensities, with turbulent Reynolds number $R_\lambda = \lambda u'/\nu$ ranging from 90 to 228. In the latter equation u' is RMS velocity, λ the Taylor micro scale, which is a length scale of turbulent eddies in the inertial range and ν the kinematic viscosity of air. Properties of the inflow data are summarized in table 4.3. For all tested intensities, the Kolmogorov length scale ℓ_η of dissipating eddies is significantly smaller than the wing length and the length scale of energy-carrying structures, the integral scale Λ, is similar to the wing length. The Taylor micro scale λ varies between 0.25 and 0.1, respectively. Since turbulence is erratic, we performed a number of simulation runs N_R increasing from 4 to 27 for increasing R_λ in order to obtain statistically reliable mean values and variances. Fig. 4.3.5 presents the slab-averaged turbulence intensity

$$\langle Tu \rangle = \int_{y_0-1.3R}^{y_0+1.3R} \int_{z_0-1.3R}^{z_0+1.3R} Tu(x,y,z)\mathrm{d}y\mathrm{d}z / \left(2.6R\right)^2$$

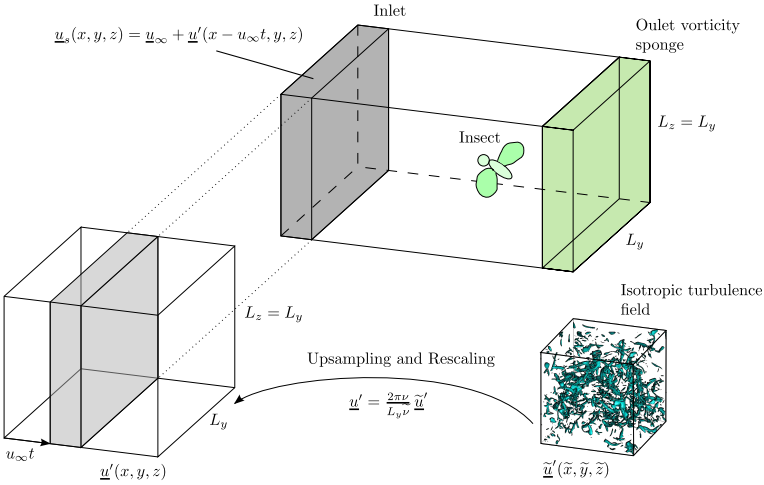


Figure 4.3.3 – Numerical wind tunnel with turbulent inlet. The precomputed isotropic turbulence field (bottom right) is upsampled in Fourier space to the resolution of the flight simulation, and rescaled according to eqn (4.3.5). In the inlet section of the wind tunnel, is it superimposed to the laminar mean flow.

as a function of the downstream distance. The black line corresponds to the laminar inflow condition. The subdomain used for averaging is centered around the insect ($y_0 = 2R$, $z_0 = 2R$). The data for laminar inflow (black line) show that the bumblebee model generates relative peak intensities of 0.25 at the wings and mean intensities of approximately 0.16 at five wing lengths downstream distance.

Fig. 4.3.4E-F illustrates the flow around the insect under turbulent inflow conditions for $Tu = 0.33$ and 0.99 relative intensity and shows that weak turbulence is associated with relatively coarse flow structures in the inflow. By contrast, flow patterns near the wings are similar in size and intensity to the structures present in the inflow at strong turbulence.

The considered range of turbulent Reynolds numbers covers the flow regime that a bee typically encounters in its natural habitats. It has been reported, for example, that bumblebees fly at wind speeds of 8 m/s [178]. At this speed, natural habitats with cylindrical trees of about 10 cm in diameter yield turbulent Reynolds numbers in the range considered in the simulation. Fig. 4.3.4B-D shows that lift, thrust and power of single simulation runs at turbulent conditions differ from the measures obtained for the laminar case. However, the generic features of the data, i.e., the location of peaks and valleys are similar under all tested flow conditions. Wingbeat-averaged and ensemble-averaged data including statistics are shown in table 4.2. These mean values

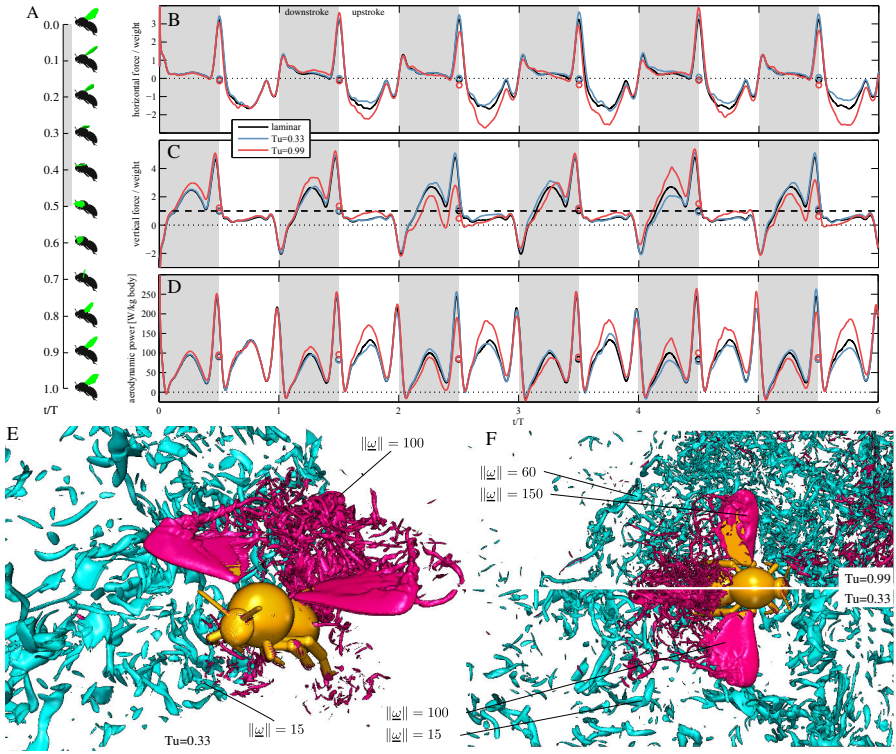


Figure 4.3.4 – Bumblebee in turbulent flow. (A) Visualization of the prescribed wingbeat, where T is period time. (B-D) Time evolution of horizontal (B) and vertical (C) force, and aerodynamic power (D) under laminar, moderately turbulent ($Tu = 0.33$) and highly turbulent ($Tu = 0.99$) conditions. Circular markers represent cycle-averaged values. (E-F) Visualization of flow structures by means of isosurfaces of the absolute vorticity $\|\underline{\omega}\|$, normalized by the wingbeat frequency f. (E) Perspective view for a realization with $Tu = 0.33$. The purple and blue isosurfaces visualize stronger and weaker vortices, respectively, and weaker vortices are shown only for $3.7R \leq y \leq 4R$ for visibility. (F) Top view, with the upper half showing flow at elevated turbulent ($Tu = 0.99$), and the lower half at moderate turbulent ($Tu = 0.33$) intensity. Weaker vortices, i.e., smaller values of $\|\underline{\omega}\|$, are shown only for $0 \leq z \leq 0.3R$ for visibility.

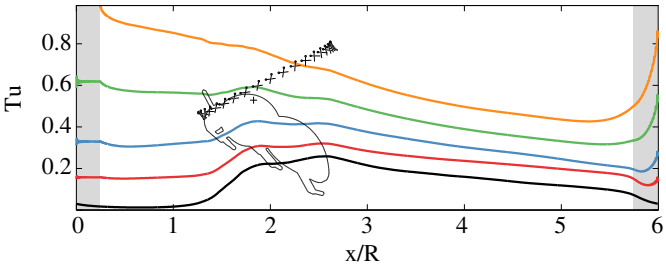


Figure 4.3.5 – Slab averaged turbulence intensity as a function of the axial coordinate, with the insect drawn to scale for orientation. Black: laminar case. Red, blue, green and orange lines correspond to Tu equal to 0.17, 0.33, 0.63 *and* 0.99, *respectively. The gray shaded areas mark regions where the in- and outflow is imposed.*

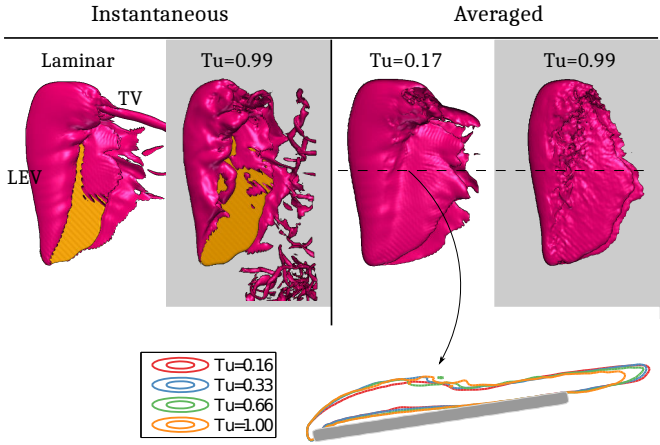


Figure 4.3.6 – Top: Isosurface of normalized absolute vorticity, $\|\underline{\omega}\| = 100$, *in the vicinity of the right wing at* $t/T = 0.3$. *Snapshots of instantaneous vorticity distribution during laminar and turbulent inflow is shown on the left. Phase- and ensemble averaged vorticity from* 16 *and* 108 *wing beats and at* $Tu = 0.17$ *and* $Tu = 0.99$, *respectively, is shown on the right. Bottom: averaged* $\|\underline{\omega}\| = 50$ *isolines at mid-span for all values of Tu. The leading edge vortex persists on average even under strongest inflow perturbations.*

Tu	Forward force F_h		Vertical force F_v		Aerodynamic power P_{aero}	
	$\overline{x} \pm \delta_{95}$	σ	$\overline{x} \pm \delta_{95}$	σ	$\overline{x} \pm \delta_{95}$	σ
0	-0.08	0	1.02	0	84.05	0
0.17	-0.10 ± 0.04	0.08	1.04 ± 0.09	0.18	83.72 ± 1.77	3.61
0.33	-0.06 ± 0.09	0.18	1.10 ± 0.10	0.21	85.02 ± 2.03	4.14
0.63	$+0.02 \pm 0.10$	0.29	1.04 ± 0.13	0.40	83.32 ± 3.13	9.57
0.99	-0.10 ± 0.07	0.37	1.01 ± 0.10	0.54	85.44 ± 1.98	10.47

Tu	Moment M_x (roll)		Moment M_y (pitch)		Moment M_z (yaw)	
	$\overline{x} \pm \delta_{95}$	σ	$\overline{x} \pm \delta_{95}$	σ	$\overline{x} \pm \delta_{95}$	σ
0	0.00	0	0.01	0	0.00	0
0.17	-0.01 ± 0.01	0.03	$+0.00 \pm 0.02$	0.03	-0.01 ± 0.02	0.03
0.33	-0.01 ± 0.04	0.08	-0.01 ± 0.03	0.06	$+0.04 \pm 0.02$	0.05
0.63	-0.02 ± 0.04	0.12	$+0.02 \pm 0.04$	0.12	$+0.07 \pm 0.04$	0.13
0.99	$+0.01 \pm 0.04$	0.19	-0.04 ± 0.03	0.13	-0.03 ± 0.04	0.21

Table 4.2 – Aerodynamic forces, power and moments obtained in the numerical experiments. Forces are normalized by the weight mg, moments by mgR, power is given in W/kg body mass.

demonstrate only tiny differences between turbulent and laminar flow conditions and even at the strongest turbulent perturbation, the bumblebee model generates aerodynamic forces close to those derived during unperturbed, laminar inflow, at virtually the same energetic cost. This aerodynamic robustness of insect wings is in striking contrast to the properties of streamlined airfoils that are highly sensitive to the laminar-turbulent transition. Fig. 4.3.6 shows the vortical structure at the wing at 0.3 stroke cycle, represented by the $\|\underline{\omega}\| = 100$ isosurface of normalized vorticity $\underline{\omega} = (\nabla \times \underline{u})\,/\,f$. The laminar case is a snapshot of the flow field, while turbulent data are also phase-averaged over N_w independent strokes for each value of Tu (see table 4.3). Although turbulence alters shape and size of the wing's tip vortex, the leading edge vortex remains visible in phase-averaged flow fields even at maximum inflow turbulence intensity.

Previous studies highlighted that turbulent flows may destabilize body pos-

R_λ	Tu	ℓ_η	λ	Λ	N_R	N_w
90.5	0.17	0.013	0.246	0.772	4	16
130.1	0.33	0.008	0.179	0.782	4	16
177.7	0.63	0.005	0.129	0.759	9	36
227.9	0.99	0.004	0.105	0.759	27	108

Table 4.3 – Parameters of inflow turbulence used in numerical experiments. The Kolmogorov length scale ℓ_η, the Taylor micro λ and the integral scale Λ are normalized by the wing length R. For each value of Tu, a number of N_R realizations has been performed, yielding in total N_w statistically independent wingbeats.

ture of an insect [24]. Roll, in particular, is prone to instability because moment of inertia about the roll axis is approximately four times smaller than about the yaw and pitch axes. Our results in table 4.2 show that mean aerodynamic moments about yaw, pitch, and roll axes do not change with increasing turbulence. However, we observed characteristic changes in moment fluctuation. Assuming that during perturbation the insect begins to rotate from rest at time t_0, we may approximate the final angular roll velocity from,

$$\Omega_{roll}(t_0+\tau) = \frac{1}{I_{roll}} \int_{t_0}^{t_0+\tau} M_{roll}(t)\,\mathrm{d}t, \tag{4.3.6}$$

with M_{roll} the roll moment, I_{roll} the roll moment of inertia with respect to the body x-axis, and τ the response delay (see below) [56].

The maximum turbulence-induced roll velocity that a freely flying bumblebee encounters depends on the reaction time of the animal in response to changes in body posture. Many insects compensate for posture perturbations by asymmetrically changing their bilateral wing stroke amplitude. Since bumblebees have no hind wing-condensed, gyroscopic halteres, the response delay mainly depends on the speed with which visual information from the retina of the compound eyes is converted into neural commands for motor control. Mean visual transduction in insects typically lasts 50 ms [69], which converts into 10 stroke cycles at 200 Hz stroke frequency in fruit flies [138] and 7.6 cycles at 152 Hz in the honeybee [170]. Previous studies on freely flying honeybees, however, reported slightly shorter response delays of approximately 20 ms or 4.5 stroke cycles, suggesting the use of ocellar pathways for body stability reflexes in this species [170].

To predict the maximum delay that allows a bumblebee to recover from turbulence-induced roll, response delay in equation (4.3.6) was 2, 3, and 4 stroke periods (Fig. 4.3.7). Fig. 4.3.7 shows how the RMS final roll ve-

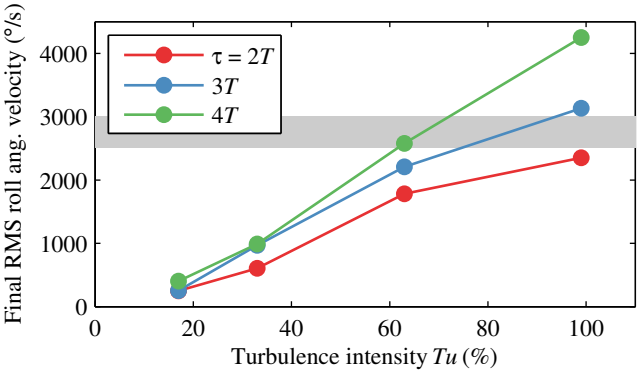


Figure 4.3.7 – RMS value of the final roll angular velocity Ω_{roll} versus the inflow turbulence intensity, calculated over all flow realizations. Different colors correspond to different response delay times τ. The gray shaded area represents the limit of sensor saturation [138, 170], see text for details.

locity increases under these conditions with increasing turbulence intensity. Behavioral measurements previously showed that the maximum body angular velocity from which a bee may recover in free flight amounts to approximately 3000 °/s (gray area in Fig. 4.3.7)[170]. Fig. 4.3.7 thus predicts that bumblebees recover form roll-induced turbulence intensities of up to 60% assuming response delays between two and four stroke periods. By contrast, posture recovery at turbulence intensities of 99% requires reduced reaction times of not more than two cycle periods. The latter finding implies that bumblebees cannot achieve stable flight at 99% turbulence intensity, which is in agreement with experimental observations of animals crashing in elevated turbulent flows when flying freely [24].

4.3.5 Conclusions and outlook

Our high-resolution numerical experiments of a bumblebee model in perturbed forward flight highlighted several unexpected results with respect to alterations in aerodynamic forces, flight stability, and aerodynamic power expenditures. The simulations imply that even strongest background turbulence does not vitally harm structure and efficacy of the lift-enhancing leading edge vortex and thus averaged forces and moments are almost identical compared to laminar flow conditions. Turbulent inflow conditions are thus of little significance for the overall flight performance of an animal in tethered flight. However, these fluctuations cause temporal transient instabilities. Thus, in a freely flying insect in which the body may rotate, absolute angular velocities

about yaw, pitch, and roll axes might reach elevated values, which in turn would require decreasing visuo-motor response delays for body stabilization with increasing turbulence. Owing to its small moment of inertia, roll is especially prone to turbulence-induced fluctuations. An important consequence of body roll is the deflection of the wingbeat-averaged resultant aerodynamic force from the vertical direction.

Thus, at large roll angles, the animal must increase the magnitude of this force so that its vertical component can support the weight of the insect. Incidentally, it has been reported that hummingbirds increase the wingbeat frequency and amplitude [136]. The finding that an increase in the turbulence intensity, superimposed to laminar flow, has no significant effect on power expenditures for tethered flight is most surprising and significant with respect to flight endurance and migration of insects. Since it has been suggested that flight of insects is limited by power rather than force production [40], any biological and physical mechanisms that help an insect to limit its wing and body drag-dependent power expenditures is of great value and may increase the animal's biological fitness.

Part III

Fluid-structure interaction with flexible obstacles

Chapter 5

Extension of the numerical method for flexible obstacles

The content of this chapter is adopted from [44], [46] and [47].

After considering insects with rigid wings, we now turn our attention to fluid–structure interaction problems that involve deformable solids. The collective long-time goal of these efforts is the simulation of insects with flexible wings. The solid mechanics of insect wings, however, is a vast research field on its own, with little data available. For instance, in one of the few available studies, Combes et al. [22, 23] presented flexibility measurements, together with a simplified finite-elements model that takes into account the wing venation by anisotropic (but continuous) local rigidity. Taking realistic, flexible wings into account is thus beyond the scope of this work, and we will resort to simplified solid models instead.

This chapter is dedicated to development and validation of the numerical method, applications will be presented in chapters 6-7. This chapter is organized as follows. In section 5.1, we derive the non-linear beam equation we use to model chord- and spanwise flexibility, and discuss its numerical solution as well as a validation case. We then proceed to the core problem in FSI simulations, the coupling of fluid and solid in section 5.2. The coupled approach is then validated in two and three spatial dimensions in section 5.3.

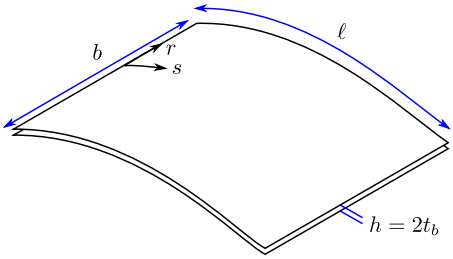


Figure 5.1.1 – Geometrical definitions of a rectangular plate, which is flexible in the s-direction and rigid in the r-direction.

5.1 Solid model: non-linear beam equation

Solving fluid–structure interaction problems requires a suitable mathematical description of the flexible object under consideration. In this work we focus on solids that are flexible in one direction only, considering the remaining directions to be perfectly rigid. This simplification of the solid mechanics part, which is actually a three-dimensional continuum as is the fluid, allows for example to separate the effects of chord- and spanwise flexibility. Insect wings are, for example, much more flexible in the chord- than in the spanwise direction [22]. Thus, we focus on a well-controlled solid model with only few parameters and, owing to its one-dimensionality, negligible computational efforts.

In the following, we will derive the governing equations for a one-dimensional beam composed of linear-elastic material, similar to [108]. We assume negligible structural damping and consider the beam to be inextensible but with finite flexural stiffness. Large deflections are admissible, thus the model is geometrically non-linear. Similar models have been used in [110, 111], assuming in addition negligible inertia.

As for the fluid, the governing solid model equation will be formulated in dimensionless units, using the scales L, T, $U = L/T$ and M for length, time, velocity and mass, respectively. As the fluid density is always normalized to unity, the mass scale is given by $M = \varrho_f L^3$. When using the solid model together with the fluid, we always normalize the beam length ℓ to unity, i.e., $L = \ell$. The basic geometrical definitions of the plate are illustrated in figure 5.1.1.

To derive the beam equation, we consider an infinitesimal beam element as illustrated in figure 5.1.2. The internal stresses are represented by the normal and tangential forces ($\underline{N}$ and $\underline{T}$, respectively) and the bending moment $\underline{M}$. The external forces are the gravity $\underline{G}$ in negative $y^{(g)}$ direction and the pres-

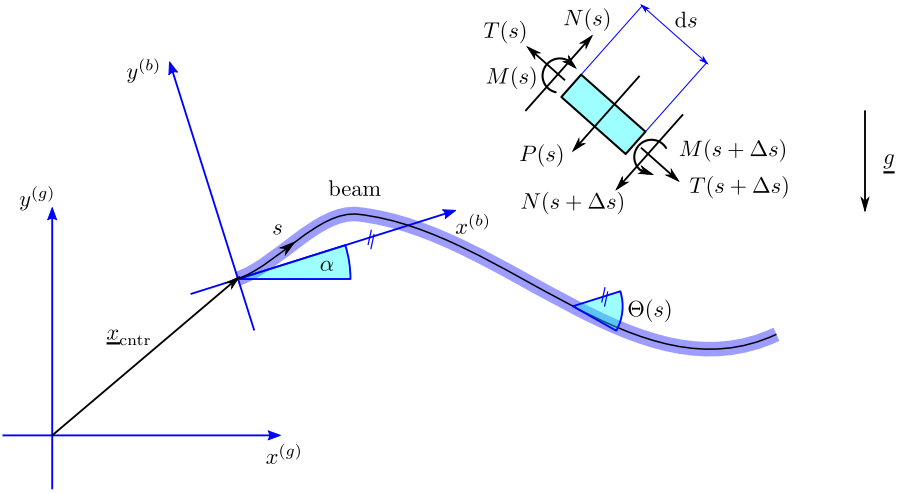


Figure 5.1.2 – Derivation of the beam equation. The slender, inextensible beam is clamped in the relative reference frame $x^{(b)}$, $y^{(b)}$, which is translated by $\underline{x}_{\text{cntr}}$ and rotated by α with respect to the global reference frame. The governing equations are derived considering the beam element (inset) with the forces and moments acting on it. Gravity is assumed in negative $y^{(g)}$ direction.

sure force $\underline{P}$. The latter is the interface to the fluid model, since the pressure jump across the beam, $[p]^{\pm}$, is dictated by the flow conditions. The shear stress on the surface, likewise resulting from the fluid, is denoted by $\underline{\tau}$. It can often be neglected. The forces and moments are described in the complex plane,

$$\underline{N} = N\,\imath\,e^{\imath\Theta}e^{\imath\alpha} \qquad \underline{T} = T\,e^{\imath\Theta}e^{\imath\alpha} \qquad \underline{P} = -\,[p]^{\pm}\,\imath\,\mathrm{d}s\,e^{\imath\Theta}\,e^{\imath\alpha}$$

$$\underline{G} = -\,g\,\mu\,\mathrm{d}s\,\imath \qquad \underline{\tau} = [\tau]^{\pm}\,\mathrm{d}s\,e^{\imath\Theta}e^{\imath\alpha}$$

where $\imath^2 = -1$. The angles $\Theta(s,t)$ and $\alpha(t)$ are the local deflection angle and the orientation of the relative system, respectively, see figure 5.1.2. All forces and moments are described in the global reference frame, but the superscript (g) has been dropped for ease of notation. The parameter μ is the dimensionless mass per unit length of the beam,

$$\mu = \frac{\varrho_s A}{M/L} = \frac{\varrho_s A}{\varrho_f L^2},$$

where A is the cross section and ϱ_s the solid density in $\left[\mathrm{kg/m^3}\right]$. We assume μ to be constant along the beam. Denoting the position of the beam element

with $\underline{z}$, Newtons second law reads

$$\begin{aligned} \mu\, \mathrm{d}s\, \ddot{\underline{z}} &= \sum \underline{F} \\ \mu\, \ddot{\underline{z}} &= \partial_s \left(\underline{T} + \underline{N}\right) - \iota\, \mu g - \iota\, [p]^{\pm}\, e^{\iota\Theta} e^{\iota\alpha} + [\tau]^{\pm}\, e^{\iota\Theta} e^{\iota\alpha} \end{aligned} \tag{5.1.1}$$

The normal force $\underline{N}$ can be eliminated using the torque balance for the beam section. The inertial moment of this element scales like $\mathrm{d}s^2$ and can therefore be neglected, yielding

$$N + \partial_s M = 0$$

for the torque balance. The moment can be linked to the curvature using the Euler-Bernoulli assumption for slender beams,

$$M = \eta \partial_s \Theta,$$

where η is the dimensionless bending rigidity

$$\eta = \frac{EI}{ML^3/T^2} = \frac{EI}{\varrho_f L^3 U^2}$$

with EI in $\left[\mathrm{Nm}^2\right]$. Note for a one-dimensional equation, Youngs modulus has to be corrected by the Poisson ratio, $\widetilde{E} = E/(1-n^2)$ [90]. Combing the preceding equations with eqn. (5.1.1) yields

$$\begin{aligned} \ddot{\underline{z}} \mu e^{-\iota\alpha} &= e^{\iota\Theta} \Big(\partial_s T + \iota T \partial_s \Theta - \iota\eta \partial_{sss}\Theta + \eta \partial_{ss}\Theta \partial_s \Theta - \iota\, [p]^{\pm} + [\tau]^{\pm}\Big) \\ &\quad - \iota g \mu e^{-\iota\alpha} \end{aligned} \tag{5.1.2}$$

The beam is considered an inextensible structure. Therefore the beam element can only rotate, and the derivative of the coordinate in the relative system is directly related to the local deflection angle:

$$\partial_s \underline{z}^{(b)} = e^{\iota\Theta}. \tag{5.1.3}$$

The position of the beam element in the global reference frame is $\underline{z} = \underline{z}_{\mathrm{cntr}} + e^{\iota\alpha}\underline{z}^{(b)}$, thus we find for the velocity and acceleration

$$\dot{\underline{z}} = \dot{\underline{z}}_{\mathrm{cntr}} + e^{\iota\alpha}\left(\dot{\underline{z}}^{(b)} + \iota\dot{\alpha}\underline{z}^{(b)}\right) \tag{5.1.4}$$

$$\ddot{\underline{z}} = \ddot{\underline{z}}_{\mathrm{cntr}} + e^{\iota\alpha}\left(\iota\left(\underline{z}^{(b)}\ddot{\alpha} + 2\dot{\alpha}\dot{\underline{z}}^{(b)}\right) + \ddot{\underline{z}}^{(b)} - \underline{z}^{(b)}\dot{\alpha}^2\right) \tag{5.1.5}$$

In order to combine these equations with the inextensibility condition (5.1.3) we need to derive $\dot{z}$ with respect to s,

$$\partial_s \dot{\underline{z}} = e^{\iota\alpha}\left(\iota\left(\partial_s \underline{z}^{(b)}\ddot{\alpha} + 2\dot{\alpha}\partial_s \dot{\underline{z}}^{(b)}\right) + \partial_s \ddot{\underline{z}}^{(b)} - \partial_s \underline{z}^{(b)}\dot{\alpha}^2\right),$$

and the inextensibility condition (5.1.3) with respect to time,

$$\partial_s \underline{z}^{(b)} = e^{\imath\Theta} \qquad \partial_s \dot{\underline{z}}^{(b)} = \imath\dot{\Theta} e^{\imath\Theta} \qquad \partial_s \ddot{\underline{z}}^{(b)} = e^{\imath\Theta}\left(\imath\ddot{\Theta} - \dot{\Theta}^2\right).$$

We thus obtain the total acceleration

$$\partial_s \ddot{\underline{z}} \;=\; e^{\imath\alpha} e^{\imath\Theta}\left(\imath\left[\ddot{\alpha} + \ddot{\Theta}\right] - \left[\dot{\alpha} + \dot{\Theta}\right]^2\right).$$

Combined with eqn. (5.1.2) and after separation of real and imaginary part, we eventually obtain the beam equation:

$$\begin{aligned}
\frac{\partial^2 T}{\partial s^2} - T\left(\frac{\partial \Theta}{\partial s}\right)^2 &= -[p]^{\pm}\frac{\partial \Theta}{\partial s} - 2\eta\frac{\partial \Theta}{\partial s}\frac{\partial^3 \Theta}{\partial s^3} - \eta\left(\frac{\partial^2 \Theta}{\partial s^2}\right)^2 \cdots \\
&\quad -\mu\left(\dot{\Theta} + \dot{\alpha}\right)^2 - \frac{\partial [\tau]^{\pm}}{\partial s} \qquad (5.1.6)\\
\mu\ddot{\Theta} + \mu\ddot{\alpha} + \frac{\partial [p]^{\pm}}{\partial s} &= -\eta\frac{\partial^4 \Theta}{\partial s^4} + \left(T + \eta\left(\frac{\partial \Theta}{\partial s}\right)^2\right)\frac{\partial^2 \Theta}{\partial s^2} \cdots \\
&\quad +2\frac{\partial T}{\partial s}\frac{\partial \Theta}{\partial s} + [\tau]^{\pm}\frac{\partial \Theta}{\partial s}, \qquad (5.1.7)
\end{aligned}$$

which is a partial differential equation of second order in time and fourth order in space. Similar to the incompressible Navier–Stokes equations, we are thus left with one evolution equation and one constraint without time derivative. Parametrizing the beam with the local deflection angle Θ is numerically beneficial, since any set of (discrete) angles $\{\Theta\}_i$ results in the same length. From Θ and $\dot{\Theta}$, the position and velocity vectors can be obtained using

$$\begin{aligned}
\underline{x}_c(s) &= \underline{x}_{\text{cntr}} + \int_0^s \begin{pmatrix} \cos\alpha\cos\Theta - \sin\alpha\sin\Theta \\ \cos\alpha\sin\Theta + \sin\alpha\cos\Theta \end{pmatrix} \mathrm{d}s \qquad (5.1.8)\\
\underline{u}_s(s) &= \dot{\underline{x}}_{\text{cntr}} + \int_0^s \begin{pmatrix} \cos\alpha\left(-\dot{\Theta}\sin\Theta - \dot{\alpha}\sin\Theta\right) \\ \cos\alpha\left(-\dot{\Theta}\sin\Theta + \dot{\alpha}\cos\Theta\right) \end{pmatrix} \mathrm{d}s \cdots \qquad (5.1.9)\\
&\quad + \int_0^s \begin{pmatrix} -\sin\alpha\left(-\dot{\Theta}\sin\Theta + \dot{\alpha}\cos\Theta\right) \\ +\sin\alpha\left(-\dot{\Theta}\sin\Theta - \dot{\alpha}\sin\Theta\right) \end{pmatrix} \mathrm{d}s.
\end{aligned}$$

We consider the beam with clamped-free boundary conditions, i.e., at the leading edge, the position and thus the acceleration is prescribed and a zero-slope condition is set in the relative reference frame. Pitching motion can be imposed by rotating the relative system. The boundary conditions at the

leading edge are thus:

$$\left.\begin{array}{rcl} \Theta = 0 & & \\ \frac{\partial T}{\partial s} + \eta \frac{\partial^2 \Theta}{\partial s^2} \frac{\partial \Theta}{\partial s} & = & \mu \left(\ddot{x}_{\text{cntr}} \cos\alpha + \ddot{y}_{\text{cntr}} \sin\alpha \cdots \right. \\ & & \left. + g \sin\alpha \right) - [\tau]^{\pm} \\ T \frac{\partial \Theta}{\partial s} - \eta \frac{\partial^3 \Theta}{\partial s^3} & = & \mu \left(\ddot{y}_{\text{cntr}} \cos\alpha - \ddot{x}_{\text{cntr}} \sin\alpha \cdots \right. \\ & & \left. + g \cos\alpha + f_{\text{ext}} \right) + [p]^{\pm} \end{array}\right\} \text{at } s = 0, \quad (5.1.10)$$

where f_{ext} is an external force defined later. At the trailing edge, a free end is assumed, thus

$$\left.\begin{array}{rcl} T & = & 0 \\ \frac{\partial \Theta}{\partial s} & = & 0 \\ \frac{\partial^2 \Theta}{\partial s^2} & = & 0 \end{array}\right\} \text{at } s = 1. \quad (5.1.11)$$

The above derivation of the solid model assumes constant coefficients η and μ along the beam. The corresponding set of equations, without derivation, for the case of non-constant coefficients, i.e., $\eta = \eta(s)$ and $\mu = \mu(s)$, can be found in appendix B.

5.1.1 Numerical solution

The set of equations (5.1.6-5.1.7, 5.1.10-5.1.11, 5.1.8-5.1.9) are solved numerically using second order finite difference approximations of the differential operators. Special care has to be devoted to time integration. First, the equations are stiff and require an implicit time marching scheme. Second, the eigenvalues of the discrete (linearized) operator lie on the imaginary axis, which happens to be the stability limit of the second order Crank-Nicolson scheme which was used in previous works [44]. Hence we employ a second order backward differentiation scheme with variable time steps [9]:

$$w^{n+1} = \frac{(1+\xi)^2}{1+2\xi} w^n - \frac{\xi^2}{1+2\xi} w^{n-1} + \frac{1+\xi}{1+2\xi} \Delta t^n f\left(w^{n+1}\right) \quad (5.1.12)$$

where $\xi = \Delta t^n / \Delta t^{n-1}$. The vector $w = \begin{pmatrix} \Theta & \dot{\Theta} \end{pmatrix}$ has been introduced to rewrite equation (5.1.7) as a first order system and $f\left(w^{n+1}\right)$ is the corresponding right-hand side. All terms are treated implicitly and the inextensibility constraint (5.1.6) is fulfilled at the new time level. The resulting nonlinear system is solved using Newton-Raphson iterations, with a relative error below 10^{-10} as stopping criterion. The Jacobian is computed analytically and hard-coded in the solver. Typically, three iterations are performed until the stopping criterion is reached. Details on the discretization in space and time, as well as the Jacobian can be found in Appendix B. In the fluid–structure interaction case, the time step Δt is determined by the fluid.

	x-displacement $\left[\times 10^{-3}\right]$	y-displacement $\left[\times 10^{-3}\right]$
Reference [165]	$-14.305 \pm 14.305\,[1.0995]$	$-63.607 \pm 65.160\,[1.0995]$
Present	$-13.306 \pm 13.356\,[1.1031]$	$-63.734 \pm 65.263\,[1.1031]$
Relative difference	$6.98\% \pm 6.6\%\,[0.33\%]$	$0.2\% \pm 0.16\%\,[0.33\%]$

Table 5.1 – Validation of the solid model, results for a beam oscillating in vacuum in a gravity field. Data given as mean ± amplitude[frequency].

5.1.2 Quantitative validation of the solid model

The solid model described above is relatively simple, compared to the three-dimensional Navier–Lamé equations. We test its accuracy using a benchmark proposal by Turek & Hron [165], case CSM3, where an elastic structure of length $\ell = 0.35$ and thickness $h_b = 0.02$ is considered. The dimensionless numbers characterizing our model are found

$$\mu = \frac{\varrho_s h_b \ell}{\varrho_f \ell^2} = 0.0571 \qquad \eta = \frac{E \ell h_b^3 / 12(1-n^2)}{\varrho_f \ell^3 U^2} = 0.0259$$

where we arbitrarily set the width of the beam to ℓ and U to 1 m/s. The beam is considered in vacuum and performs undamped oscillations in a gravity field with $g = 0.7$. In the limit of small deflections, linearized theory predicts the first eigenfrequency to be $f_1 = 3.516\sqrt{\eta/\mu}/2\pi = 1.077$, see, e.g., [48].

Reference data is given as mean±amplitude, where mean $= \frac{1}{2}(\max + \min)$ and amplitude $= \frac{1}{2}(\max - \min)$. Present and reference data is assembled in table 5.1. The comparison is favorable, and the difference in the x-displacement can be explained by the inextensibility assumption, whereas [165] considered a slightly compressible material. For the vertical displacement and the oscillation frequency, the relative difference is below 0.33%. We note that though the vertical displacement is 37% of the beam length, the oscillation frequency agrees to within 1% with the prediction by linear theory.

5.2 Coupling fluid and solid

5.2.1 Construction of the mask function for flexible objects

To include the flexible object described above in the fluid solver, the smoothed χ function and the solid velocity field $\underline{u}_s$ are required on the Eulerian fluid grid. As described in section 2.3, we rely on the signed distance function for

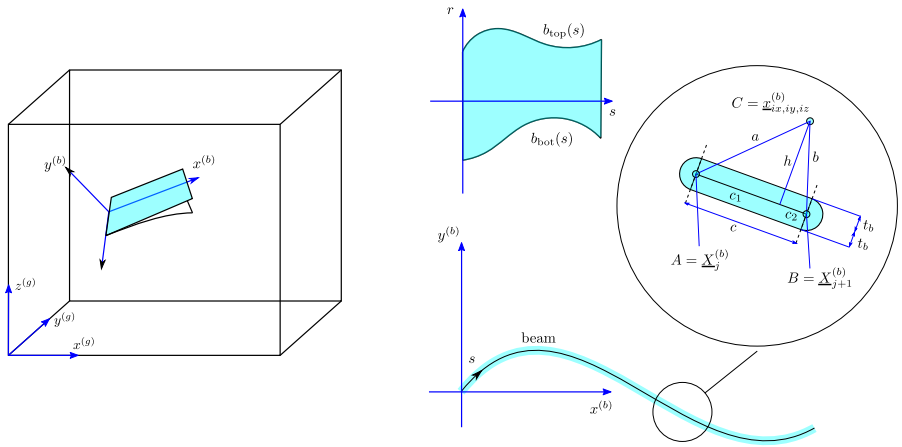


Figure 5.2.1 – On the construction of the mask function for flexible obstacles. Left: perspective view in three-dimensional space. Right: two-dimensional illustration in the relative system. The beam is composed of a centerline (black) and the thin solid layer (cyan). The centerline is given as a set of points $\{\underline{X}^{(b)}\}_j$, the inset shows one segment and the geometrical definitions used in algorithm 5.1

the former task. Figure 5.2.1 (left) illustrates the general setup. The spanwise flexible plate is defined in the relative system $\underline{x}^{(b)}$, which is defined as described in section 3.2.1 for insects. We allow for all six degrees of freedom. The beam is given as a set of coordinates $\{\underline{X}^{(b)}\}_j$ and we approximate the centerline by a linear Bézier curve. The spacing between two points on the beam, Δs, is of about the same size as the fluid grid spacing Δx, which is why the piecewise linear approximation is justified.

Algorithm 5.1 describes how the distance to the beam centerline, δ, is computed for each point on the Eulerian fluid grid. It is easier to first compute the distance to the centerline, which is greater or equal to zero, and then to substract the beam thickness t_b, than to directly compute the signed distance itself. Algorithm 5.1 cycles over all Eulerian fluid nodes. Points outside the bounding box are skipped to save computing time. For each remaining Eulerian node, we check the distances to all N_s solid nodes. If the distance is small enough, we detect whether the point in question lies on a segment, as illustrated in the inset in figure 5.2.1, or in the hinges between segments. The solid velocity is interpolated linearly on the segments. Non-rectangular shapes can be taken into account by introducing the functions $b_{\mathrm{bot}}(s)$ and $b_{\mathrm{top}}(s)$ in the same spirit as described in section 3.2.4. The proposed algorithm is local, i.e., each MPI process executes it independently with no intra-processor communication involved, and the total computational effort

is below 5% of the total cost of a simulation.

5.2.2 Force interpolation

The solid deformation is dictated by the balance of inertia, internal and external forces. The latter contain the fluid forces, through which the fluid interacts with the solid. In the opposite way, the resulting deformation alters the fluid flow. For the solid deformation, we assume viscous fluid tension to be negligible in front of the pressure. The total aerodynamic forces and moments, however, are always computed using the volume integration discussed in section 2.4, and thus include viscous contributions.

Figure 5.2.2 illustrates the beam in a three-dimensional setup; it is thus a plate with one rigid direction. The plate is composed of two surfaces, on which the scalar-valued pressure is interpolated. The resulting force is normal to the surface and points to the suction side, where the lower pressure is encountered.

In the one-dimensional model eqn (5.1.6-5.1.7), the pressure is a line-load, i.e., it is a force per unit length. The two-dimensional surface pressure distribution, as shown in figure 5.2.2, is thus integrated over the r-direction,

$$[p]^{\pm}(s) = \int_{r=b_{\text{bot}}(s)}^{r=b_{\text{top}}(s)} \left(p_{\text{top}}(s,r) - p_{\text{bottom}}(s,r)\right) \mathrm{d}r$$

Since the computational domain is usually split among CPU, the interpolation requires a layer of ghost nodes around the sub-domains to be synchronized between processes prior to interpolation.

For the actual interpolation, we use a technique borrowed from the immersed-boundary community, namely the regularized delta functions D. Their usage has been proposed by Peskin already [130]. Since then, different functions have been devised, among which we use the one proposed by Yang et al. [185],

$$D(\xi) = \begin{cases} \frac{3}{8} + \frac{\pi}{32} - \frac{\xi^2}{4} & |\xi| \leq 0.5 \\ \frac{1}{4} + \frac{1-|\xi|}{8}\sqrt{-2 + 8|\xi| - 4\xi^2} \cdots & \\ \quad -\frac{1}{8}\arcsin\left(\sqrt{2}(|\xi| - 1)\right) & 0.5 \leq |\xi| \leq 1.5 \\ \frac{17}{16} - \frac{\pi}{64} - \frac{3|\xi|}{4} + \frac{\xi^2}{8} \cdots & \\ \quad + \frac{|\xi|-2}{16}\sqrt{16|\xi| - 4\xi^2 - 14} \cdots & \\ \quad + \frac{1}{16}\arcsin\left(\sqrt{2}(|\xi| - 2)\right) & 1.5 \leq |\xi| \leq 2.5 \\ 0 & 2.5 \leq |\xi|, \end{cases}$$

Algorithm 5.1 Given the beam nodes $\{\underline{X}^{(b)}\}_j$ and the orientation of the relative system, construct the signed distance function $\delta(\underline{x})$ and the solid velocity field $\underline{u}_s(\underline{x})$. For the sake of brevity we denote $\sqrt{\lceil\cdot\rceil} = \sqrt{\max(\cdot, 0)}$.

Set $d_{\max} = \sqrt{\Delta s^2 + (t_b + h_{\text{smth}})^2}$ ▷ largest possible distance of a fluid node to beam centerline
for $ix = 0, nx-1; iy = 0, ny-1; iz = 0, nz-1$ **do**
  Get coordinates of point $\underline{x}_{ix,iy,iz}$
  Transform to body system $\underline{x}^{(b)}_{ix,iy,iz} = M_{\text{body}}\left(\underline{x}_{ix,iy,iz} - \underline{x}_{\text{cntr}}\right)$
  Set $\delta_{ix,iy,iz} = \infty$, $\underline{u}_{s,ix,iy,iz} = 0$ ▷ Initialization
  if $\underline{x}^{(b)}_{ix,iy,iz}$ is in bounding box **then** ▷ Saves computing time
    for $is = 0, ns-2$ **do** ▷ Loop over beam segments
      $a \leftarrow \overline{AC}$
      $b \leftarrow \overline{BC}$
      if $a \le d_{\max}$ or $b \le d_{\max}$ **then** ▷ One node of segment is close enough
        $b \leftarrow \overline{AB}$
        $h \leftarrow \sqrt{\left\lceil a^2 - \frac{a^2+c^2-b^2}{2c}\right\rceil}$
        $c_1 \leftarrow \sqrt{\lceil a^2 - h^2\rceil}$
        $c_2 \leftarrow \sqrt{\lceil b^2 - h^2\rceil}$
        if $a \le d_{\max}$ and $b \le d_{\max}$ and $c_1 \le c$ and $c_2 \le c$ **then** ▷ On segment
          if $h \le \delta_{ix,iy,iz}$ **then**
            $\delta_{ix,iy,iz} \leftarrow h$
            $s_1 \leftarrow j\Delta s$
            $s_2 \leftarrow (j+1)\Delta s$
            $s \leftarrow s_1 + c_1$
            $\underline{u}_{s,ix,iy,iz} \leftarrow \underline{u}^{(b)}_j + \frac{s-s_1}{s_2-s_1}\left(\underline{u}^{(b)}_{j+1} - \underline{u}^{(b)}_j\right)$ ▷ Linear Interpolation
          end if
        else ▷ Not on segment, but possibly hinge
          if a<b **then** ▷ Left hinge is closer
            if $a \le \delta_{ix,iy,iz}$ **then**
              $s \leftarrow j\Delta s$
              $\delta_{ix,iy,iz} \leftarrow a$
              $\underline{u}_{s,ix,iy,iz} \leftarrow \underline{u}^{(b)}_j$
            end if
          else if b<a **then** ▷ Right hinge is closer
            if $b \le \delta_{ix,iy,iz}$ **then**
              $s \leftarrow (j+1)\Delta s$
              $\delta_{ix,iy,iz} \leftarrow b$
              $\underline{u}_{s,ix,iy,iz} \leftarrow \underline{u}^{(b)}_{j+1}$
            end if
          end if
        end if
      end if
    end for
    $\delta_{ix,iy,iz} \leftarrow \delta_{ix,iy,iz} - t_b$ ▷ Shift zero-isosurface from centerline to outer contour:
    $\delta_{ix,iy,iz} \leftarrow \max(\delta_{ix,iy,iz}, z^{(b)} - b_{\text{top}}(s), b_{\text{bot}}(s) - z^{(b)})$ ▷ Take finite width into account:
  end if
end for

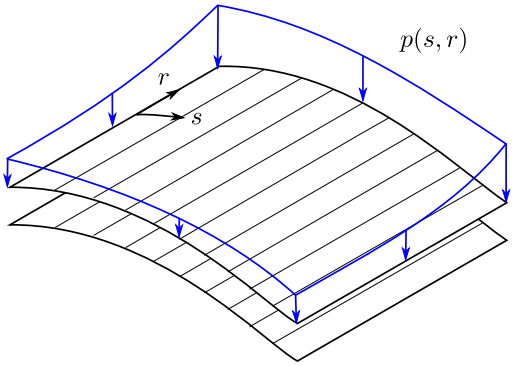


Figure 5.2.2 – Surfaces forces on the beam in a 3D configuration, where it becomes a plate with one rigid direction (r). The pressure is interpolated on the top- and bottom surfaces, and the resulting force is normal to the surface.

where $\xi = x/\Delta x$. This function is designed to reduce spurious oscillations in the interpolated values, which is desirable for fluid–structure interaction. The three-dimensional function is constructed as tensor product of three functions of one variable,

$$D\left(\underline{x}\right) = D\left(\frac{x}{\Delta x}\right) D\left(\frac{y}{\Delta y}\right) D\left(\frac{z}{\Delta z}\right),$$

and the interpolated pressure on a point $\underline{X}$ on the interface then reads

$$p\left(\underline{X}\right) = \sum p_{ix,iy,iz} D\left(\underline{x}_{ix,iy,iz} - \underline{X}\right). \tag{5.2.1}$$

5.2.3 Time stepping of the coupled system

We previously described the necessary modules for the simulation of fluid structure interaction. On the one hand we have the fluid, which takes into account the time-varying solid geometry though the volume penalization method. On the other hand, we derived the solid model which governs the structural mechanics part. The coupling between both modules is achieved through the construction of the mask function χ and solid velocity field $\underline{u}_s$, and the interpolation of the pressure on the solid's surface.

In the following, we describe how the coupled fluid–solid system is advanced in time. We start by noting that the time discretizations for the fluid and solid part are fundamentally different. The flow field is, in general, three-dimensional and discretized with an explicit time marching scheme. An implicit scheme would require the solution of a large non-linear system, which

is computationally expensive. The Fourier discretization would furthermore entail dense matrices instead of sparse ones, since the Fourier transformation is non-local. The solid model equation on the other hand is numerically very stiff, owing to the fourth order derivative in eqn (5.1.7), which restricts the time step of explicit schemes to $\Delta t = C\Delta s^4$ where C is a constant depending on the rigidity η and the time marching scheme. An implicit scheme is therefore required.

This mélange of time discretizations has been employed by other researchers [51, 153], and this class of methods is usually termed *partionned methods*. Their advantage lies in the high degree of modularization, which allows great flexibility. However, the time discretization of the coupled system is subject to instabilities, even if, for a given Δt, both individual solvers are stable.

The source for these numerical problems lies in the temporal treatment of the interface condition, see e.g., the reviews [51, 153]. Most contributions consider Dirichlet-Neumann conditons, for the displacement and the stresses, respectively [51, 89, 142, 153], but other conditions are possible, namely Robin-Neumann [6]. In the Dirichlet-Neumann case, the interface conditions read

$$\begin{aligned} \frac{\partial \underline{X}(t)}{\partial t} &= \underline{u}\left(\underline{X}(t), t\right) \quad \text{on } \partial\Omega_s \\ \underline{\underline{\sigma}}^f\left(X(t), t\right) &= \underline{\underline{\sigma}}^s\left(\underline{X}(t), t\right) \quad \text{on } \partial\Omega_s \end{aligned}$$

where we denote the Lagrangian position of the interface with $\underline{X}$. Both, continuity of velocity and stresses explicitly depend on time, and they should ideally be satisfied at the old and new time levels. In *monolithic* solvers, the coupled system is evolved as a whole and the interface conditions are indeed satisfied precisely at both time levels, see e.g., [62, 73]. These approaches are unconditionally stable and energy conserving, but do not allow modularization of the fluid and solid solvers [153].

The stability of partionned methods, as we employ here, is reportedly [19, 51, 89, 153, 155, 169] strongly dependent on the artificial added mass effect, which depends on the density ratio ϱ_s/ϱ_f. Smaller density ratios require iterative stabilization. In the following, we describe the two partionned coupling schemes used in the present work, the semi-implicit staggered scheme and an iterative scheme.

5.2.3.1 Semi-implicit staggered scheme

Partionned coupling schemes are commonly divided into strong (or implicit) and weak (or explicit) coupling, depending on how the interface conditions are treated. In a weak coupling scheme, the equations are advanced in a

staggered manner, i.e., advancing first the fluid and then the solid. These schemes are popular in the aeroelasticity community, see e.g., [50], because of their stability for the large density ratios typically involved in this field. The staggered time advancing however implies a phase lag in both fields, which is equivalent to the explicit treatment of either or both interface conditions [51]. The simplest scheme is the sequential staggered scheme, where both solvers are advanced in parallel and exchange the coupling information only a the beginning of a time step. We found this scheme to be inaccurate and unstable for our problems, and instead employ a semi-implicit staggered scheme.

Algorithm 5.2 Semi-implicit staggered scheme for the time advancement of coupled fluid–structure problems.

Require: Fluid velocity field $\underline{u}^n$ and solid state $\mathcal{S}^n$ at time t^n

1: Construct mask function χ^n and solid velocity field $\underline{u}_s^n$ from solid state $\mathcal{S}^n$ (see Algorithm 5.1)
2: Compute source terms for the fluid $\underline{f}^n = \underline{f}(\underline{u}^n, \chi^n, \underline{u}_s^n)$
3: Advance fluid to new time level $\underline{u}^n \to \underline{u}^{n+1}$ using the AB2 scheme
4: Compute static pressure $p^{n+1} = p(\underline{u}^{n+1}, \chi^n, \underline{u}_s^n)$
5: Interpolate pressure jump $[p]^{\pm}(t^{n+1}) = \mathcal{I}(p^{n+1})$
6: Advance solid state $\mathcal{S}^n \to \mathcal{S}^{n+1}$ using $[p]^{\pm}(t^{n+1})$ and the BDF2 scheme
7: **return** New fluid state $\underline{u}^{+1}$ and new solid state $\mathcal{S}^{n+1}$

The present formulation of this scheme is presented in algorithm 5.2. The fluid is discretized using the Adams-Bashforth scheme with integrating factor, as described in 2.8, and the solid using the fully implicit BDF scheme in eqn. (5.1.12). The latter evaluates the RHS only at the new time level, therefore, the interpolated pressure jump at the old time level t^n is not required in algorithm 5.2. The computation of the pressure at the new time level requires an additional evaluation of the right hand side of the penalized Navier–Stokes equations. The weak coupling character is visible in line 4 of the algorithm, where the source terms are computed with the solid model from the previous time step. This is equivalent to treating the dynamic coupling condition (i.e., the continuity of stresses) in an implicit way, while the displacement condition is treated explicitly. A similar scheme has for instance been presented in [105]. Various improvements of this method have been proposed, aiming at stabilization for light structures [153]. The effect of small density ratios on the stability of this scheme has been analyzed for example in [19, 54].

5.2.3.2 Iterative scheme

The semi-implicit staggered scheme described above is efficient since it requires only two right hand side evaluations per time step. It is however un-

stable if the structure is too light compared to the fluid.

To alleviate this restriction, iterative schemes have been proposed, aiming at satisfying the coupling condition at the new time level as well. Most algorithms are based on fixed-point iterations (e.g., [89, 184]), or Newton iterations [64]. Van Brummelen showed analytically, using a simplified setup, that iterations can alleviate stability problems, but do not necessarily cure them under all conditions [169]. He found that the added mass of a compressible fluid scales with the time step, while it approaches a constant in incompressible fluids. As a consequence, even iterative, strong coupling schemes can sometimes be unstable, if incompressible fluid is considered. However, this type of scheme has been successfully applied to a wide range of problems, e.g., in [184] to elastically mounted cylinders or in [89] to flexible pipes. In compressible fluid–structure interaction, Storti et al. [157] used the fixed point iteration to compute a channel with flexible walls in the supersonic regime.

Algorithm 5.3 presents the iterative coupling scheme used in the present work. If only one iteration is performed, we recover the semi-implicit scheme described above, as it is the case for most iterative schemes of this type (e.g., [173]). The basic scheme thus consists of repeating the semi-implicit scheme, until convergence is achieved. However, poor convergence is observed in the literature [51, 89, 153, 169], if no relaxation is employed. The relaxation factor β is computed from Aitken's Δ^2 method [89], and is equal to one in the first iteration step.

Algorithm 5.3 Iterative scheme

Require: Fluid velocity field $\underline{u}^n$ and solid state $\mathcal{S}^n$ at time t^n

1: Initialization: set $\widetilde{\mathcal{S}}^{n+1} = \mathcal{S}^n$
2: **while** not converged **do**
3: Construct mask function $\widetilde{\chi}^{n+1}$ and solid velocity field $\widetilde{\underline{u}}_s^{n+1}$ from solid state $\widetilde{\mathcal{S}}^{n+1}$ (see Algorithm 5.1)
4: Compute source terms for the fluid $\underline{f}^n = \underline{f}(\underline{u}^n, \widetilde{\chi}^{n+1}, \widetilde{\underline{u}}_s^{n+1})$
5: Advance fluid to new time level $\underline{u}^n \to \widetilde{\underline{u}}^{n+1}$ using the AB2 scheme
6: Compute static pressure $\widetilde{p}^{n+1} = p(\widetilde{\underline{u}}^{n+1}, \widetilde{\chi}^{n+1}, \widetilde{\underline{u}}_s^{n+1})$
7: Interpolate pressure jump $\widetilde{[p]}_j^{\pm}(t^{n+1}) = \mathcal{I}(\widetilde{p}^{n+1})$ ▷ j is iteration index
8: Relaxation:
9: $\widetilde{\widetilde{[p]}}_j^{\pm} = (\beta)\widetilde{[p]}_j^{\pm} + (1-\beta)\widetilde{\widetilde{[p]}}_{j-1}^{\pm}$
10: Advance solid state $\mathcal{S}^n \to \widetilde{\mathcal{S}}^{n+1}$ using $\widetilde{\widetilde{[p]}}_j^{\pm}$ and the BDF2 scheme
11: Convergence test
12: **end while**

5.3 Validation tests of the fluid-structure interaction module

5.3.1 Energy budget

A first physically relevant test is whether our coupled solver artificially injects energy into the system. For computational simplicity, we consider a two–dimensional setup, consisting of a thin beam immersed in viscous fluid, and use the semi-implicit scheme. At startup, the beam is held fixed, and an impulsively started mean flow u_∞ is imposed. After the decay of the startup singularity, at $t = 2$, the beam is released and starts to perform damped oscillations. The two-dimensional domain is $\Omega = [0,4] \times [0,4]$, where linear dimensions are normalized by the beam length. The resolution is $N_x \times N_y = 2048 \times 2048$, the penalty parameter is $C_\eta = 10^{-4}$ and we consider two Reynolds numbers, $Re = u_\infty \ell/\nu = 200$ and 1900, which gives $K = 0.36$ and 0.12, respectively. The beams energy can be computed as the sum of the flexural, kinetic and potential energy:

$$E^s_{\text{solid}} = E^s_{\text{flex}} + E^s_{\text{kin}} + E^s_{\text{pot}} \tag{5.3.1}$$

where

$$E^s_{\text{flex}} = \frac{1}{2}\int_0^1 \eta \left(\frac{\partial \Theta}{\partial s}\right)^2 \mathrm{d}s \tag{5.3.2}$$

$$E^s_{\text{kin}} = \frac{1}{2}\mu \int_0^1 \underline{u}_s \cdot \underline{u}_s \,\mathrm{d}s \tag{5.3.3}$$

$$E^s_{\text{pot}} = \mu g \int_0^1 y \,\mathrm{d}s. \tag{5.3.4}$$

The elastic material is defined by $\mu = 0.57$ and $\eta = 0.065$. The energy budget for the penalized Navier–Stokes equation can be obtained by dot-multiplying eqn. (2.1.2a) by $\underline{u}$ and integrating over the computational domain $\Omega = \mathbb{T}^2$:

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}E^f_{\text{kin}} &= -\nu \int_\Omega |\nabla \underline{u}|^2 \,\mathrm{d}\Omega - \frac{1}{C_\eta}\int_\Omega \chi\,(\underline{u} - \underline{u}_s)\cdot \underline{u}\,\mathrm{d}\Omega \\
\dot{E}^f_{\text{kin}} &= -\nu \int_\Omega \underline{\omega}^2 \mathrm{d}\Omega - \frac{1}{C_\eta}\int_\Omega \chi \underline{u}_s \cdot (\underline{u} - \underline{u}_\infty - \underline{u}_s)\,\mathrm{d}\Omega \\
&\quad -\frac{1}{C_\eta}\int_\Omega \chi \underline{u}_\infty \cdot (\underline{u} - \underline{u}_s)\,\mathrm{d}\Omega - \frac{1}{C_\eta}\int_\Omega \chi\,(\underline{u} - \underline{u}_\infty - \underline{u}_s)^2\,\mathrm{d}\Omega \\
\dot{E}^f_{\text{kin}} &= \dot{E}^f_{\text{diss}} + \dot{E}^f_{\text{solid}} + \dot{E}^f_{\text{mean}} + \dot{E}^f_{\text{porous}}
\end{aligned} \tag{5.3.5}$$

where the penalization term has been split into three contributions: the energy that enters the fluid through the solid motion $\dot{E}^f_{\text{solid}}$ and the mean flow $\dot{E}^f_{\text{mean}}$, and the energy that is dissipated in the penalization term $\dot{E}^f_{\text{porous}}$, which can be interpreted as "porous dissipation".

For our numerical method, we require $\dot{E}^f_{\text{solid}} = \dot{E}^s_{\text{solid}}$, i.e., the coupling should not introduce artificial energy. Figure 5.3.1 illustrates a snapshot of the vorticity field (left) and the time series of the different terms in equation (5.3.5). One can observe good agreement between $\dot{E}^f_{\text{solid}}$ and $\dot{E}^s_{\text{solid}}$, the relative L_2 difference is about 7%. The remaining difference is attributed to finite-size effects in the model, since the underlying solid model is one-dimensional, but it is extended to a thin, yet two-dimensional structure.

5.3.2 Two-dimensional validation: Comparison with Turek et al.

To validate our numerical method quantitatively, we consider the benchmark proposed by Turek et al. [165, 166]. It consists of three stages, one of which has already been used to evaluate the model error of the beam equation. The remaining stages are the CFD test, in which a rigid obstacle is considered, and the FSI test with a flexible object. For each stage, three test cases are proposed, and here we consider only the CFD3 and FSI3 tests.

The setup is illustrated in figure 5.3.2 top. A circular cylinder of radius $R = 0.05$ with an appendage of length $\ell = 0.35$ and thickness $h = 0.02$ is immersed in a channel of size $L_x \times L_y = 2.5 \times 0.41$ with a parabolic velocity profile imposed at the inflow. The center of the cylinder is placed at $(0.2, 0.2)$, which intentionally yields an asymmetric setup. In our approach, the channel walls as well as the obstacle are modeled with the penalization method. The walls have the same thickness as the beam structure. The inflow condition is imposed directly using the volume penalization method.

We should note that in this setup, the distance between the object and the inflow is smaller than the upstream influence of the object itself. Thus, this configuration cannot be reproduced considering a long channel with a developed laminar flow, as it would be done experimentally.

First we consider the CFD3 test, where the obstacle is rigid. The Reynolds number is $\text{Re} = u_\infty \ell / \nu = 700$. The test is designed such that the flow reaches a periodic state, which is independent of the initial condition. As the channel walls are modeled with the penalization method, they cover a finite area, which has to be taken into account when forcing the mean velocity. To force unity mean flow in the channel, u_∞ is corrected using the channel walls height h_{chan}, thus $u_\infty = 1 - 2h_{\text{chan}}/L_y$.

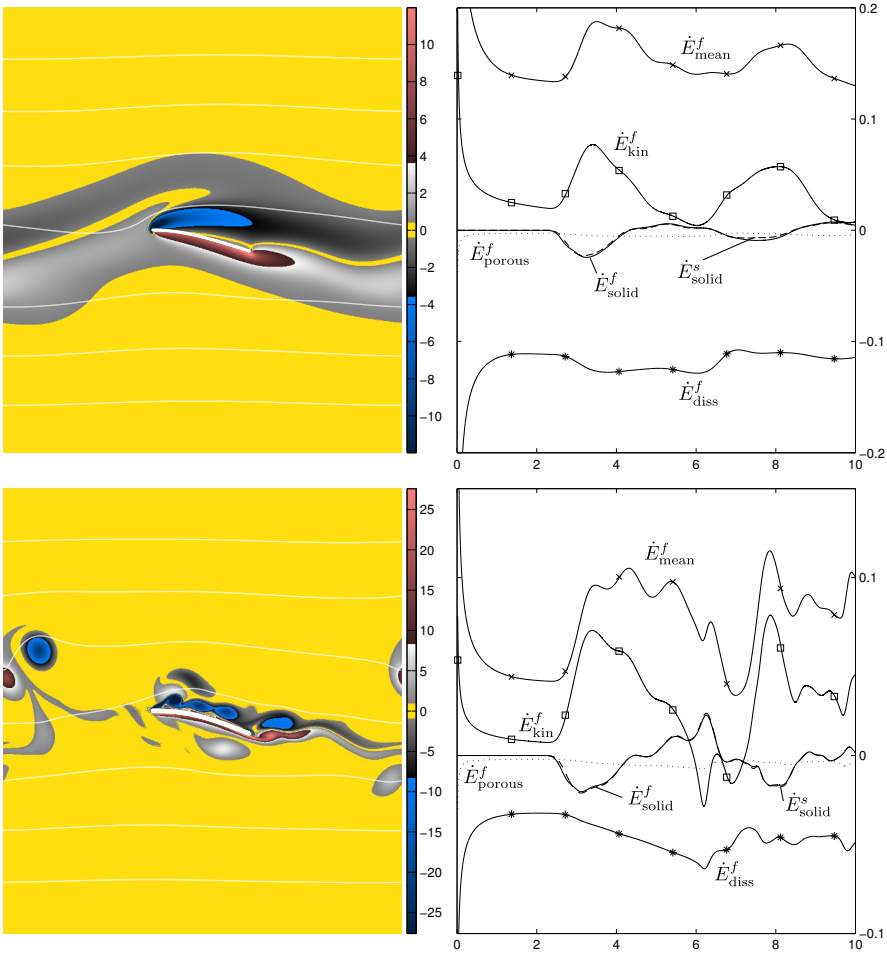


Figure 5.3.1 – Energy test case for the Reynolds numbers 200 *(top) and* 1900 *(bottom). Left column: Snapshot of the flow field, shown is vorticity with superimposed white streamlines. Right column: energy budget according to eqn. (5.3.5). The beam remains fixed until* $t = 2$*. The startup singularity is due to the impusively started flow. The curves for* $\dot{E}^f_{\text{solid}}$ *and* $\dot{E}^s_{\text{solid}}$ *show good agreement in both cases, indicating a good conservation of energy.*

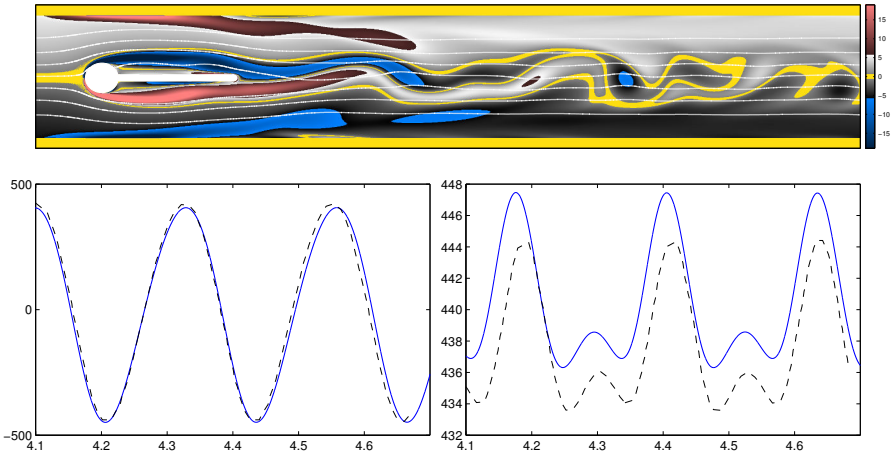


Figure 5.3.2 – Results for the CFD3 test with rigid appendage. Vorticity field with stream lines and the obstacle (top). Bottom: time series of the lift (left) and drag force (right) on the obstacle. Present results are represented by blue solid lines and the reference solution [165] by the dashed black line.

When starting with a uniform flow as initial condition, the developing wake behind the obstacle requires a considerable amount of time to destabilize and to reach the final periodic state. For this reason, all present simulations are started with the flow field at the end of the destabilization phase. The quantities for comparison are the lift and drag force in the developed periodic regime and the fundamental frequency of the lift force, which corresponds to the lowest significant frequency present in the spectrum. The forces are represented by their minimum and maximum values during one period. Due to transient effects, the min/max values are averaged over the last periods, and simulations are run long enough for the width of the 95% confidence interval to be smaller than 0.1% of the predicted value. This results in a computational time of about 30 period times.

In table 5.2, the present results for four levels of resolution are compared with the reference solution presented in [165]. The general accuracy of our method is satisfying, even though the maximum value of the lift force presents some noticeable difference with respect to the reference solution. In addition to the max/min values, the shape of the lift/drag curves are compared in figure 5.3.2, bottom. It can be observed that the curves are very similar. The drag curve exhibits an offset which is related to the smoothing layer in the χ-function, which can be interpreted as surface roughness. Due to the smoothing layer, our method generally overpredicts the drag force. On the

		Lift			
Resolution	C_η $(K = 0.43)$	Max		Min	
1300×288	$4 \cdot 10^{-3}$	423.66	-0.53%	-500.44	11.28%
2600×576	$1 \cdot 10^{-3}$	392.47	-7.85%	-464.57	3.31%
5200×1152	$2.5 \cdot 10^{-4}$	402.00	-5.62%	-452.64	0.65%
10400×2304	$6.3 \cdot 10^{-5}$	405.46	-4.80%	-446.48	-0.72%
Reference		425.92		-449.70	

	Drag					
Resolution	Max		Min		Frequency	
1300×288	512.62	15.18%	498.44	14.89%	4.46	1.47%
2600×576	475.70	6.88%	463.80	6.91%	4.41	0.33%
5200×1152	453.42	1.88%	442.14	1.92%	4.37	-0.58%
10400×2304	447.29	0.50%	436.15	0.53%	4.36	-0.81%
Reference	445.07		433.83		4.40	

Table 5.2 – Results of the CFD3 test with a fixed obstacle.

other hand, the offset is only 0.5% of the mean value and thus rather small. The corresponding vorticity field during the periodic state is shown in figure 5.3.2, top.

To conclude the quantitative validation we proceed to the FSI3 test case. Now, the appendage attached to the cylinder is flexible. As for the CFD3 test, some computational time can be saved by choosing a suitable initial condition. We use the same flow field as in the CFD3 test case, and the beam is initially at rest and undeformed. The non-dimensional parameters characterizing the solid are $\mu = 5.7143 \cdot 10^{-2}$ and $\eta = 2.5915 \cdot 10^{-2}$, and the Reynolds number is $\mathrm{Re} = u_\infty \ell / \nu = 700$.

As for the CFD3 test, the quantitative comparison with results from the literature is done using the max/min values of the respective time series. The fundamental frequency f_0 is that of the y-displacement d_y of the trailing edge. In the present work, the max/min value are obtained as stated previously for the CFD3 test, but interestingly the FSI3 test requires a shorter computational time to settle into its final periodic state. Here, about 25 oscillations were computed. The max/min values for the displacement d_y, the drag and lift force and the frequency f_0 for this test are presented in table 5.3. The dif-

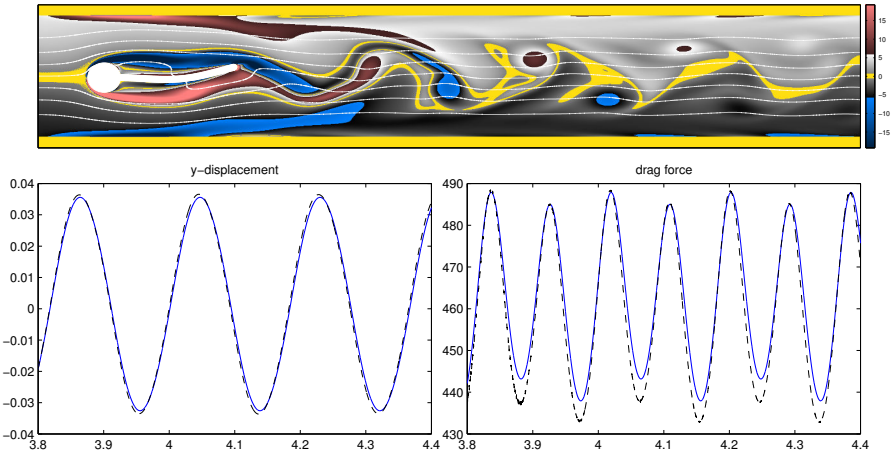


Figure 5.3.3 – Results for the FSI3 test with a flexible appendage. Top: snapshot of the vorticity field with the deformed obstacle. Bottom: time series of the y-displacement of the trailing edge and of the drag force acting on the entire obstacle.

ferent methods behind the numerical results are described in the respective papers and in the overview article [165].

For the y-displacement of the trailing edge, all methods except (5) and (7) agree to within 3.9%, and the present results fit well into this range of results. Unless otherwise stated, differences in this section are normalized with the averaged result. It is remarkable that the result for d_y does not significantly depend on the resolution level, as even the coarsest resolution yields an acceptable value. Compared to the initial computation from [165], present results differ by 0.6%.

For the drag force, the different methods are spread to a range over 7.5% of the averaged value. The drag computed with the present method is in excellent agreement for the two highest resolutions and again quite close to the original values from [165], with a difference of about 0.5%. The shapes of the time series of the displacement and the drag force are illustrated in figure 5.3.3 together with the original results presented in [165].

Concerning the lift force, the range of results is broader and varies up to 40% for method (7), 30% for method (5) and 7.5% for method (8). The present method seems to overpredict the amplitude of oscillations in the lift force, and the difference in both max and min value is about 20% that of the average.

In conclusion we find good quantitative agreement with the results found in the literature, for the solid and fluid solver alone, as well as for the coupled FSI algorithm.

Resolution	d_y $[\times 10^{-3}]$ max	d_y $[\times 10^{-3}]$ min	F_{drag} max	F_{drag} min	F_{lift} max	F_{lift} min	f_0
1300×288	36.47	-32.12	591.45	525.75	184.83	-189.29	5.89
2600×576	36.32	-32.98	516.73	455.39	187.83	-185.73	5.56
5200×1152	35.58	-32.59	487.87	437.93	190.50	-184.36	5.47
10400×2304	35.63	-32.71	481.20	432.50	188.52	-181.30	5.44
(1) Turek [165]	36.37	-33.45	487.81	432.79	156.13	-151.31	5.47
(2) Schäfer [156, 166]	36.73	-33.79	487.82	431.98	159.54	-155.86	5.82
(3) Rannacher [166]	35.89	-33.33	478.59	426.21	155.06	-150.34	5.42
(4) Turek [166]	36.46	-33.52	488.24	432.76	156.40	-151.40	5.47
(5) Breuer [166]	44.00	-41.00	507.00	428.00	204.90	-172.50	5.06
(6) Krafzyk [166]	36.58	-33.62	494.30	431.70	155.81	-152.19	5.50
(7) Wall [89, 166]	30.45	-27.55	451.50	416.50	91.13	-86.07	5.30
(8) Bletzinger [166]	38.18	-35.08	503.02	446.78	169.76	-162.04	5.50
Average	36.75	-33.81	488.14	431.62	158.62	-150.78	5.46

Table 5.3 – Results of the FSI3 test. Present computations using four levels of resolution on the top, compared to various results from the literature. For orientation the average over the different publications, excluding the present one, is shown.

5.3.3 Three-dimensional validation: Thrust generated by a heaving plate

As validation case for fluid–structure interaction in a three–dimensional setting we consider the numerical work by Yeh and Alexeev [186]. They consider a flexible panel with an imposed heaving motion at the leading edge with zero angle of attack (i.e., no pitching of the leading edge imposed). Owing to flexibility, the effective angle of attack is non-zero, and the foil produces a thrust force. Much like the three-dimensional FSI models in the present work, this work is intended to model swimming propulsion. The authors study the influence of the driving frequency, normalized by the eigenfrequency in fluid, $\phi = f/f_{1,f}$ by varying the elastic properties of the flexible plate. Two regimes are identified, which maximize either the velocity for $\phi \approx 1.1$ or the efficiency for $\phi \approx 1.7$.

The reference solution is computed using the Lattice Boltzmann Method,

which approximates incompressible fluid flow without solving an elliptical Poisson equation. An overset approach with a refined region in the vicinity of the plate and a coarser one in the far field are used. The flexible plate is modeled using the lattice-spring model, i.e., it is approximated as a system of mass points connected by springs.

Here, we consider only a single simulation out of the dataset presented in [186], arbitrarily fixing the added mass parameter (as defined in [186]) to $T = 1$ and the frequency ratio to $\phi = 1.1$. The former choice of a heavier swimmer is motivated by numerical stability, and thus all results are obtained using the semi-implicit FSI coupling scheme without iterations. With the stiffness parameters kindly provided[1], the non-dimensional solid properties in our model are

$$\mu = 0.4031 \qquad \eta = 1.6669.$$

Our normalization is based on the beam length ℓ, the driving frequency f and the fluid density ϱ_f. The plate's width is $w = 0.4$, and the prescribed heaving motion is given as

$$y = a_0 \cos(2\pi f t),$$

where $a_0 = 0.1$. The Reynolds number $Re = 2\pi f \ell a_0/\nu$ is equal to 250. The domain size is $4 \times 4 \times 2$ and it is discretized using $768 \times 768 \times 384$ points, with the penalization parameter equal to $C_\eta = 2 \cdot 10^{-4}$ ($K = 0.14$). We apply a vorticity sponge to remove the periodicity. The validation has to be performed in the steady cruising state, where the thrust generated by the plate is compensated by acceleratethe drag force. In our case, the plate remains anchored to the laboratory frame and accelerates the axial mean fluid flow u_∞ instead. To speed up the computation, we modify the mean flow equation according to eqn. (2.6.3), and set $m_{\text{fluid}} = 0.5$. In coarser resolution pre-computations, the mean flow was started from rest and the terminal value, 1.3, was set as initial condition $u_\infty(t = 0)$ in the high-resolution case, to further reduce the computational cost. In the high-resolution case, 11 cycles were performed.

Table (5.4) compares the results obtained during the last cycle with those given in [186]. The cruising speed is slightly reduced in the present work, but the agreement to within 5% is still tolerable. The aerodynamic power, computed as $P_{\text{aero}} = \int \underline{u}_s (\underline{u} - \underline{u}_s) / C_\eta$ (cf. section 3.2.5), is overpredicted by about 25%. Both, overprediction of power requirement and underprediction of cruising speed are related to an elevated drag coefficient owing to the smoothing layer in the χ-function. The trailing edge displacement d is very close to the reference solution (1%).

The flow field is visualized in figure 5.3.4. At each half-stroke, a vortex is shed, which travels perpendicular to the mean flow on a V-shaped path. The flow field is qualitatively similar to the reference computation.

[1] Private communication

	u_∞	P_{aero}	$\eta_{\text{eff}} = u_\infty / P_{\text{aero}}$	d
Yeh & Alexeev [186]	1.47	5.11	0.29	7.60
Present	1.40	6.53	0.22	7.68

Table 5.4 – Cycle-averaged results for the heaving plate.

In conclusion, the comparison yields reasonable agreement with the literature, and the remaining difference can be attributed to differences in modeling. The present plate is rigid in the lateral direction, while the reference data is obtained with a 2D flexible plate. This simplification has also an influence on the solid model parameters η and μ. The former is the non-dimensional rigidity, corrected by the Poisson ratio of the solid material [90]. Owing to this correction, the eigenfrequency in vacuum of the beam, $f_0 = 3.516/2\pi \sqrt{\eta/\mu}$ is slightly different (6%) from the corresponding eigenfrequency of the 2D flexible plate. However, the differences between both studies are reasonable and we thus conclude that our fluid–structure interaction module is validated also in the 3D case.

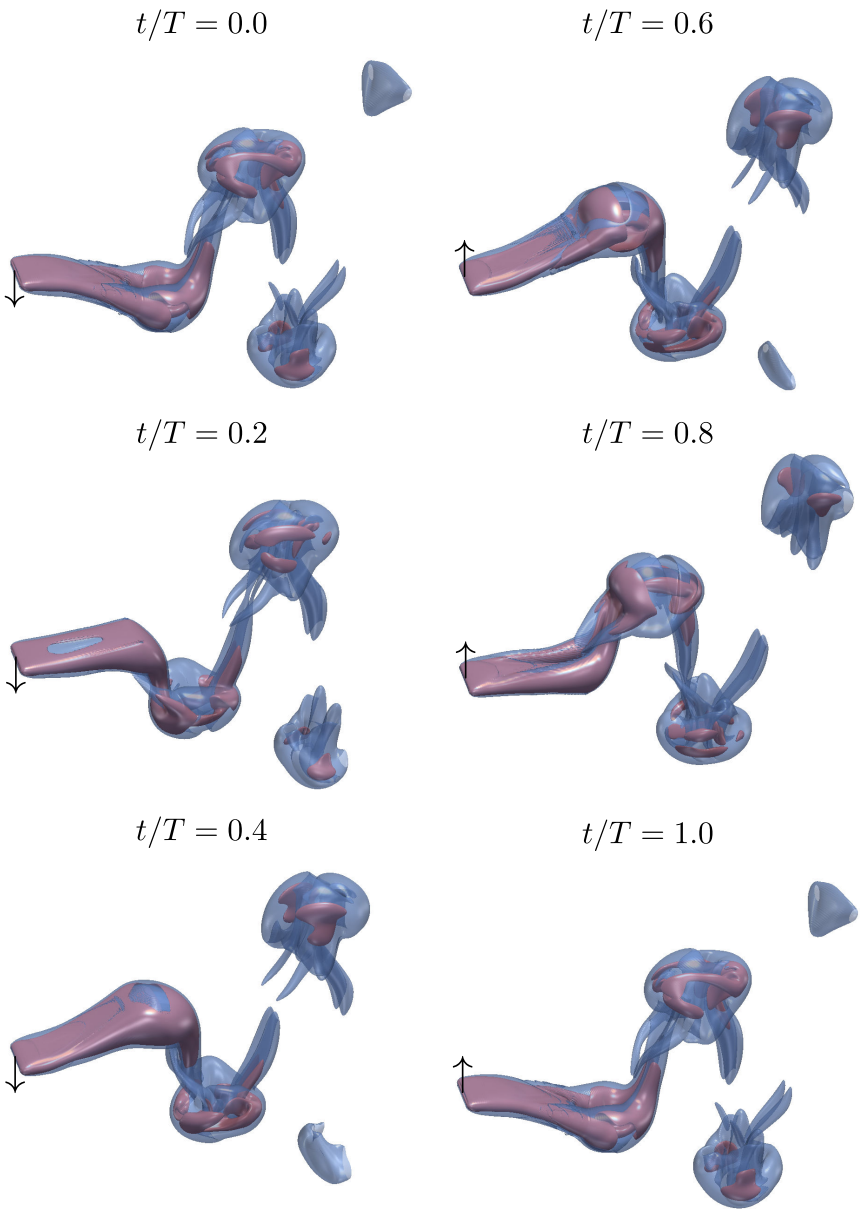


Figure 5.3.4 – Flow generated by the heaving plate, visualized by isosurfaces of vorticity magnitude. Blue semitransparent corresponds to $\|\underline{\omega}\| = 5$*, purple to* $\|\underline{\omega}\| = 20$*. The Reynolds number is 250.*

Chapter 6

Two-dimensional fluid-structure interaction

The content of this chapter is adopted from [44], [46] and [83].

This chapter is dedicated to fluid–structure interaction problems in a two–dimensional setting, which is a necessary step before passing to three–dimensional problems. We apply our method first to a canonical setup, where a beam is immersed in a fluid. No external driving force is applied. This problem is astonishingly rich in phenomena, as the system displays physical instabilities and transition to chaos, depending on the physical parameters. Flapping flyers have to avoid this regime. In the second section, we add an externally imposed heaving motion to the leading edge, and study the thrust production by this wing section. A short outlook on two wing sections heaving in tandem, like the wings of a dragonfly, is given.

6.1 Fluttering instability of flexible foils

The content of this section is adapted from [44].

In nature inspired fluid dynamics, the complex interaction of some deformable structures with an ambient flow is a commonplace problem. Whether it is gliding, swimming or flying, many types of animal locomotion strongly rely on this type of interaction [21]. The archetype problem of fluid-structure interaction is the flapping of a flag in the wind, attracting researchers due to its richness in phenomena. Indeed, a flag exhibits a large variety of possible regimes, depending on its material parameters and the surrounding flow. It

can be aligned with the flow in a stable state, or flap dynamically. The latter state can further be subdivided into highly regular and chaotic motion patterns. In some parameter ranges, also bistable behavior has been reported, where the dynamically selected state depends on the initial conditions.

Natural swimmers and flapping flyers exploit a combination of active and passive flapping to improve their flight performance [3, 41, 134], a source of inspiration for various investigations. Also, aside from locomotion, the fluttering instability occurs in other biological applications. Huang [71] first pointed out that flutter is encountered in the upper human airways, where the soft palate separates nasal and oral inflow. The instability manifests itself in the occurrence of snoring or, in severe cases, obstructive sleep apnoea/hypopnea.

We now apply the method described above to study the flutter problem. The domain size is $L \times H = 10 \times 4$ chord lengths, the vorticity sponge technique is applied at all four boundaries of the domain and thus prevents the wake from re-entering the domain. The spatial resolution is 2000×800 Fourier modes for $\mathrm{Re} \leq 200$, 2500×1000 for $\mathrm{Re} = 500$, 2800×1120 for $\mathrm{Re} = 750$ and finally 3200×1280 for $\mathrm{Re} = 1000$. The penalization parameter is fixed to $C_\eta = 10^{-4}$ and the smoothing is equal to two grid points, i.e., $C_{\mathrm{smth}} = 2$, in all simulations. The beam thickness is $4 \cdot 10^{-2}$, corresponding to eight grid points on the coarsest and 13 on the finest grid. To complete the picture and to check its validity, some runs with double resolution have also been performed, yielding a difference of less than 3% for the averaged end point deflection angle amplitude and about 8% for the lift force amplitude, with respect to the coarser simulations.

For the flutter phenomenon it is conventional to introduce a reduced free-stream velocity $u^\star = \sqrt{\mu/\eta} = u_\infty \ell \sqrt{\varrho_s / EI}$, as proposed in [42, 107, 150, 160]. There is no mathematical reason for this choice, but it is physically more intuitive to increase the flow velocity than decreasing the bending stiffness and hence change the material the beam is made of.

The setup is symmetric with respect to the beam, therefore the instability has to be triggered somehow. This is done by applying an external force, f_{ext} in equation 5.1.10, on the structure during a small time interval. At the end of this perturbation, the beam has moved downwards by about $2 \cdot 10^{-3}$ units and reaches a velocity of $2 \cdot 10^{-2}$. Note that applying an external force is possibly the easiest way to break the symmetry, as this is always compatible with the boundary conditions (5.1.10).

In this study we keep the mass ratio fixed at $\mu = 1/3$ and vary the Reynolds number and the reduced free-stream velocity. Figure 6.1.1 summarizes the parameters for which the different simulations have been performed, where the symbols indicate configurations found to be stable (□) or in a periodic

(◯) or chaotic (★) fluttering state.

We compare our results with two different computational approaches. The first one is the inviscid vortex shedding model presented by Michelin et al. [109]. This study uses the same model equation for the beam as we do and a simplified model for the fluid. Their results concerning the stability limit are in agreement with the existing experimental and linear stability results presented in [42]. For the mass ratio considered here, stability is lost above $u^\star_{\text{crit}} = 9.6$, according to their stability analysis. This stability limit is represented by the dash-dotted line in figure 6.1.1.

The second study we compare our results with the work of Connell and Yue [25], where a linear stability analysis for viscous flows and direct numerical simulations have been presented. The solid is also modelled using a nonlinear beam equation. The computational approach for these simulations relies on body-fitted, time-dependent grids with a resolution of 100×200 points. Spatial discretization is performed using second order finite differences. The linear stability analysis assumes a Blasius boundary layer on the beam to determine its internal tension, therefore the stabilizing force depends on the Reynolds number. Recasting the stability limit found by this analysis in our terminology, the beam becomes unstable at

$$u^\star_{\text{crit}} = 2\pi \sqrt{\frac{\mu}{\frac{\mu}{\mu\pi+1} - 1.3\,\text{Re}^{-1/2}}}, \tag{6.1.1}$$

represented by the solid line in figure 6.1.1. The numerical simulations performed in [25] focus mainly on the limit of vanishing bending stiffness, with the mass ratio being the parameter of interest, and are found to be in reasonable agreement with the analytical results.

6.1.1 Variation of the Reynolds number

In the first series of simulations, we focus on the influence of the Reynolds number for fixed values of μ and $u^\star$. In order to directly compare our results with the ones reported in [109], we fix $\mu = 1/3$ and further choose $u^\star = 14$. In the inviscid limit, this point was found to be in the chaotic regime. The considered Reynolds numbers are 25, 50, 100, 200, 500, 750 and 1000.

Figure 6.1.2 illustrates the computations for $\text{Re} = 25$, 100, 200 and 1000. The trajectory of the end point in the displacement-velocity plane illustrates the temporal behavior, and the vorticity field ω at a given time is visualized in the vicinity of the beam. Note that the figure is a zoom and the computational domain is about two times larger in both directions. The $\text{Re} = 750$ case is analyzed in detail in figure 6.1.3.

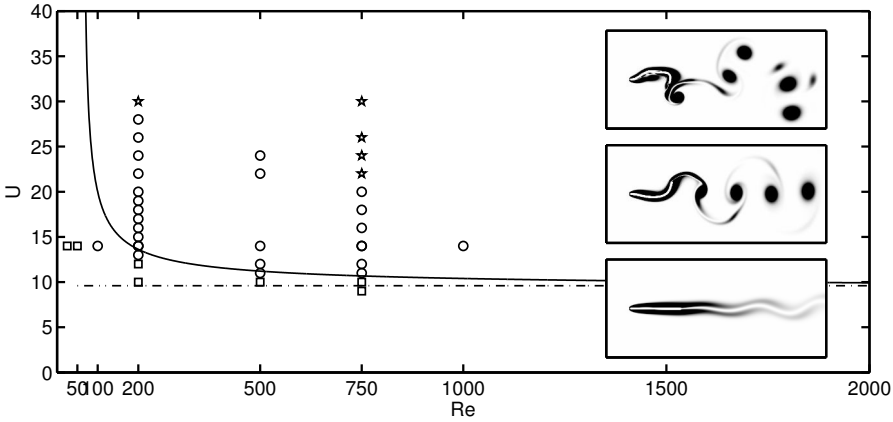


Figure 6.1.1 – Stability map in the Re-$u^\star$ plane for $\mu = 1/3$. Inviscid [109] $(-\cdot)$ and viscous linear stability analysis [25] $(-)$. Symbols indicate present simulations, found to be stable $(\square)$ or in an either periodic $(\bigcirc)$ or chaotic $(\bigstar)$ fluttering state. Insets show corresponding snapshots of the absolute vorticity $|\omega|$ at $\mathrm{Re} = 750$.

Re	$f_{\mathrm{disp}}\ [u_\infty/\ell]$	$a_{\mathrm{disp}}\ [\ell]$	$f_{\mathrm{drag}}\ [u_\infty/\ell]$	$f_{\mathrm{lift}}\ [u_\infty/\ell]$
100	0.4327	0.2020	0.8380	0.4138
200	0.5886	0.3368	1.1955	0.5886
500	0.6254	0.3186	1.2275	0.6254
750	0.6251	0.3142	1.2203	0.6251
1000	0.6161	0.3172	1.2321	0.6161

Table 6.1 – Amplitude and frequency of $y-$displacement and frequencies of drag and lift forces as a function of the Reynolds number. The weight ratio is $\mu = 1/3$ and the reduced inflow velocity is $u^\star = 14$.

Let us first discuss the lowest Reynolds numbers, illustrated exemplarily for $\mathrm{Re} = 25$ in the top row of figure 6.1.2. In the phase diagram, the initial perturbation is marked by the red part of the trajectory. The oscillation amplitude grows during the first half turn in phase space, and then continuously spirals down to the initial state of rest.

Configurations marked as stable in figure 6.1.1 were stopped if the beam's internal energy, consisting of the elastic and kinetic contribution, stayed below the initial perturbation energy for sufficiently long time. We used $T_{\mathrm{stop}} = 15$, which is larger than ten flapping cycles in a periodic case. In the vorticity snapshot, whose corresponding position in phase space is marked with a red star, a thick, attached boundary layer can be observed. The corresponding figures for $\mathrm{Re} = 50$ do not differ significantly, only the damping of the initial perturbation has become weaker and hence the typical spiral-like motion of the end point in phase space slower approaches its origin.

Increasing Re to 100, the behavior of the beam changes from stable to unstable. The phase diagram shows the growth in amplitude of the flutter oscillations, but also that this growth rate is very slow. Thus, we can conclude that the critical Reynolds number must be between 50 and 100, and that the distance to the upper limit of this range is smaller than the lower one. On the other hand, according to the viscous linear stability analysis, equation (6.1.1), the stability limit is predicted at $\mathrm{Re}_{\mathrm{crit}} \approx 183$. Hence we can confirm that the linear analysis tends to overpredict the stability threshold for low Reynolds numbers, a result that has been stated in [25] for a different parameter range.

When further increasing Re to 200, the growth rate of the oscillation becomes significantly higher, as can be seen in the phase diagram in figure 6.1.2 (third row from top). The vorticity field shows distinct, separated vortices, shed each time the beam tip reaches its maximum amplitude. Note that the flow field for $\mathrm{Re} = 100$ is illustrated during the amplification phase. Comparing $\mathrm{Re} = 100$ and 200, the flow field reveals different wake structures in the amplification phase and in the final periodic state: during the growth phase, no distinct vortices are shed at the trailing edge of the foil, the wake is formed of two coherent zones of positive and negative vorticity. Furthermore, the final periodic state differs in both cases, while this is not true when comparing higher Reynolds numbers among themselves. The case $\mathrm{Re} = 100$ is therefore different from the others.

For $\mathrm{Re} = 500$, the periodic state is reached sooner and the shed vortices become stronger and narrower, forming the characteristic von Kármán street, a trend that is preserved at $\mathrm{Re} = 750$ and 1000. Figure 6.1.3 illustrates the vortex shedding mechanism observed at $\mathrm{Re} = 750$. In snapshots A–B, the layer of negative vorticity on the top of the beam is sheared of the surface and elongated. The same holds for the layer of positive vorticity at the bottom side.

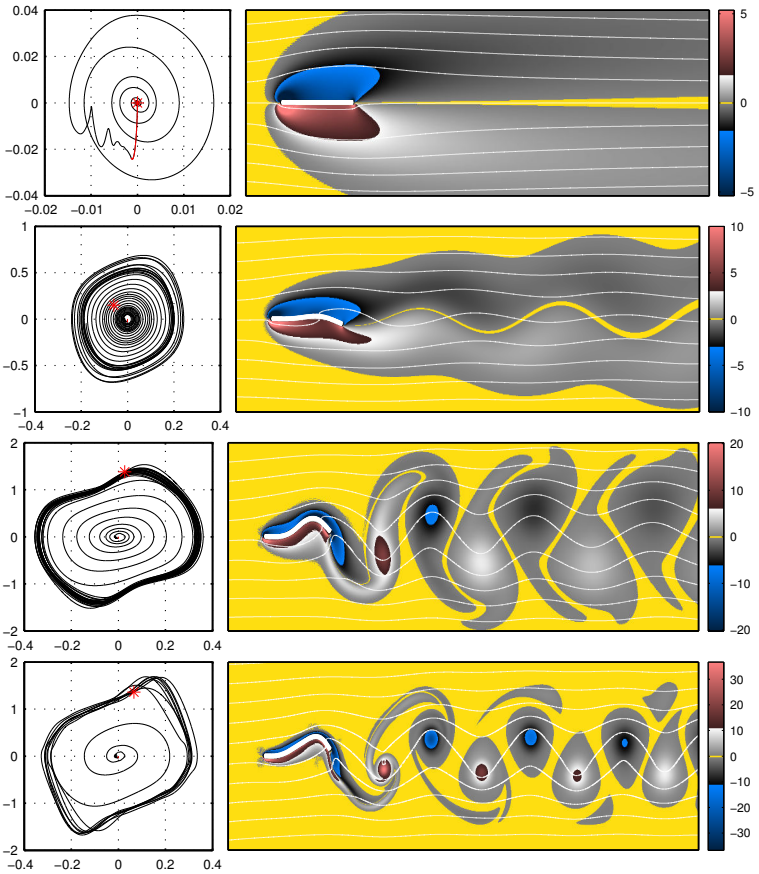


Figure 6.1.2 – The influence of the Reynolds number. Phase-diagram of the trailing edge in the displacement-velocity plane (y, u_y) *(left column) and a vorticity snapshot (right column) with superimposed white streamlines. Only part of the computational domain is shown. The Reynolds number is* $\mathrm{Re} = 25, 100, 200, 1000$ *(from top to bottom). The point in phase space belonging to the snapshot is marked with a red star.*

Between C and D, a single vortex begins to separate, still connected to the remainder of the vorticity layer on the top side. The vortex is advected, and the reverse process starts (F–G). For the instant G, the whole vorticity field is shown, illustrating how the vortices eventually separate and form the street of isolated vortices. The process is also illustrated in the phase diagram in figure 6.1.3 (bottom left). The phase trajectories are colored by the absolute acceleration of the beam. Note that the acceleration varies smoothly. The shape

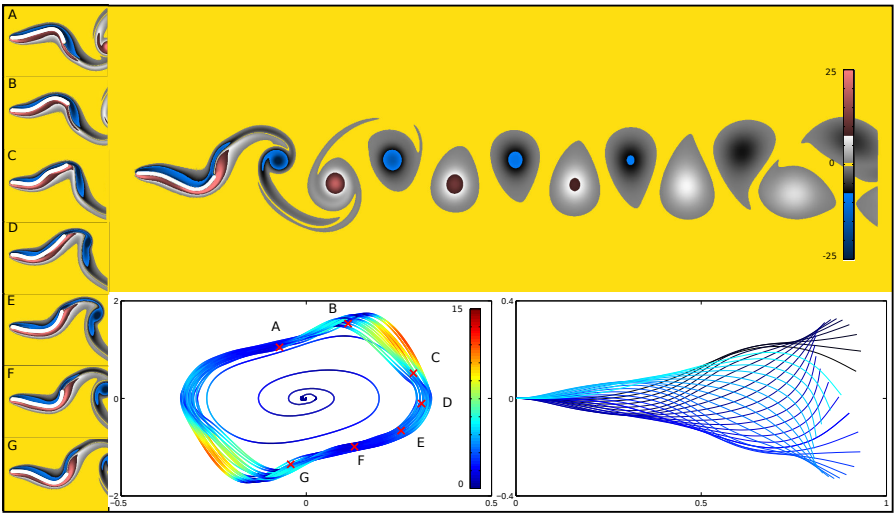


Figure 6.1.3 – Periodic state: Re=750 and $u^\star = 14$. Left side: (A–G) time history of a typical oscillation period with a snapshot every $\Delta t = 0.1$. Top: the whole vorticity field at $T = 84.5$ Bottom, middle: Phase diagram of the end point, colored with the absolute acceleration. Bottom, right: time history of the deflection line for one period of motion, with $\Delta t = 0.05$, color ranging from dark to light blue.

of the beam, illustrated in figure 6.1.3 (bottom, right), is in good agreement with [109].

Even though there are some changes in the shape of the phase trajectory, the flapping amplitude remains constant above $\mathrm{Re} = 200$. The same holds for the fundamental flapping frequency, that is the frequency of the trailing edge oscillations, as summarized in table 6.1. The frequencies are computed using the fast Fourier transform, and the amplitude of the displacement, a_{disp}, is the averaged value of all periods after the amplification phase. This averaging is done as the beam does not reach a perfect periodic state during the computation. That can also be seen from the phase diagrams, where the trajectories are more spread out.

Hence we can conclude that we do not observe a transition to a chaotic state for this choice of parameters, contrary to the result from the inviscid model [109]. This may also be an effect due to the finite simulation time, although the computation spans at least 20 oscillation periods.

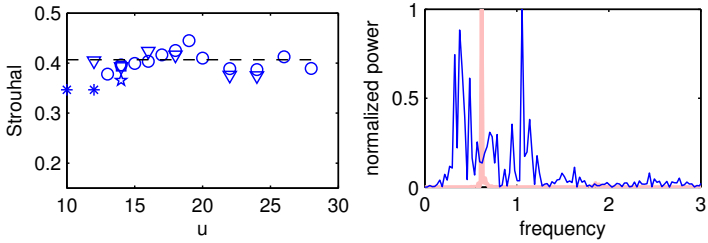


Figure 6.1.4 – Left: Strouhal number $\mathrm{St} = \frac{2Af}{u_\infty}$ *of the displacement, for simulations resulting in periodic behavior.* $\mathrm{Re} = 200$ ($\bigcirc$), $\mathrm{Re} = 500$ ($\square$), $\mathrm{Re} = 750$ ($\triangledown$), $\mathrm{Re} = 1000$ ($\star$) *and values presented by Michelin et al.* ($*$) *[109]. The dashed line corresponds to the average Strouhal number. Right: spectra of the integral lift force for* $\mathrm{Re} = 750$ *in a periodic* ($u^\star = 14$, *light red*) *and chaotic* ($u^\star = 30$, *blue*) *state.*

6.1.2 Influence of the stiffness and the transition to chaos

The first part of our parameter study dealt with the influence of the Reynolds number, now the influence of the stiffness is considered by varying the reduced inflow velocity. As illustrated in the stability diagram, figure 6.1.1, we focus on two different Reynolds numbers, 200 and 750, and increase the reduced inflow velocity from a stable state until we reach the region of chaotic flapping. A third series at $\mathrm{Re} = 500$ completes the picture.

Considering $\mathrm{Re} = 200$, the onset of flapping can be observed for $12 < u^\star_{\mathrm{crit}} < 13$, while equation (6.1.1) predicts that to happen at 13.62. This slight difference between our results and the analytical model persists at $\mathrm{Re} = 500$, where we find $10 < u^\star_{\mathrm{crit}} < 11$ versus 11.21 from the theory. When Re is increased further, our interval matches the analytical prediction. As stated previously, a simulation is considered as stable if the internal energy of the beam remains smaller than its initial perturbation energy for sufficiently long time, despite some small oscillations that are not yet completely damped.

For the stability limit we can hence conclude that our results are in reasonable agreement with the viscous linear stability analysis, the difference becoming smaller with increasing Reynolds number. Note also that the prediction from the inviscid stability analysis [109] is consistent with the viscous theory for sufficiently high Re.

The present results for the transition from a regular to an irregular flapping state cannot be compared with the results reported in [25], since therein the limit of small bending stiffness is considered. In this limit, the transition periodicity–chaos is found to be well approximated by $\mu_{\mathrm{chaos}} = \beta\,\mu_{\mathrm{crit}}$ with $\beta = 2.5$. In the case considered here, where the bending stiffness is not small,

such a proportionality cannot be observed.

When the flapping state is periodic, a single vortex is shed at each up- and downstroke, as discussed in section 6.1.1 and illustrated in figure 6.1.3. However, in the chaotic regime, the deformation at the trailing edge becomes larger (due to the increasing reduced inflow velocity), and now a pair of vortices of different sign is shed into the wake, see figure 6.1.5, labeled by X and Y. These dipoles propagate with an additional, self induced velocity, relative to the mean flow and thus form an irregular wake pattern. They can also travel in the cross-flow direction, as the dipole Y does. At most of the strokes, two vortices of approximately equal strength are shed (like the pair X).

However, the chaotic regime is also characterized by violent intermittent accelerations of the trailing edge, which are larger than the mean accelerations by an order of magnitude. In the phase diagram in figure 6.1.5, these violent snapping events are represented by strong peaks in the acceleration (indicated by the color). The mean absolute acceleration is found to be 5.7 and during snapping events peak values between 50 and 100 can be observed.

A single snapping event is illustrated in figure 6.1.5, left. The time instants A–G are equally spaced in time, with $\Delta t = 0.05$. The first three snapshots, A–C, show only minor evolution, illustrating the fact that the acceleration is small compared to the peak value between C and D, where the actual snapping takes place. Subsequently, a strong positive vortex is shed into the wake (E–G), followed by a less intense negative one (G).

This asymmetry in intensity of the shed dipole is also characteristic for the snapping event and distinguishes it from regular oscillations, where the dipole is approximately symmetric, compare dipole Z with X and Y in figure 6.1.5.

To further illustrate the chaotic nature of this regime, figure 6.1.5 contains also the time history of the beam deflection line. A distinct deflection line (Q) highlights another difference to the periodic state. The trailing edge of this deflection line is orientated against the flow direction, an event which does not occur in periodic simulations.

The differentiation between periodic and chaotic states, illustrated in figure 6.1.1, needs some further discussion. Several observations change during the transition. First, the relation between the frequencies of the displacement, and the lift and drag force differ in both flapping states. While we have $f_{\text{disp}} = f_{\text{lift}}$ and $f_{\text{drag}} = 2 f_{\text{disp}}$ in a periodic state (cf. table 6.1), these relations do no longer hold when the flapping is irregular. Moreover, we can no longer identify a dominant frequency in the spectra of the displacement and the hydrodynamic forces, as illustrated exemplarily by plotting the spectrum in figure 6.1.4 (right). This finding is consistent with the results reported in [25, 109].

We observe from our data that chaos does not necessarily need to occur di-

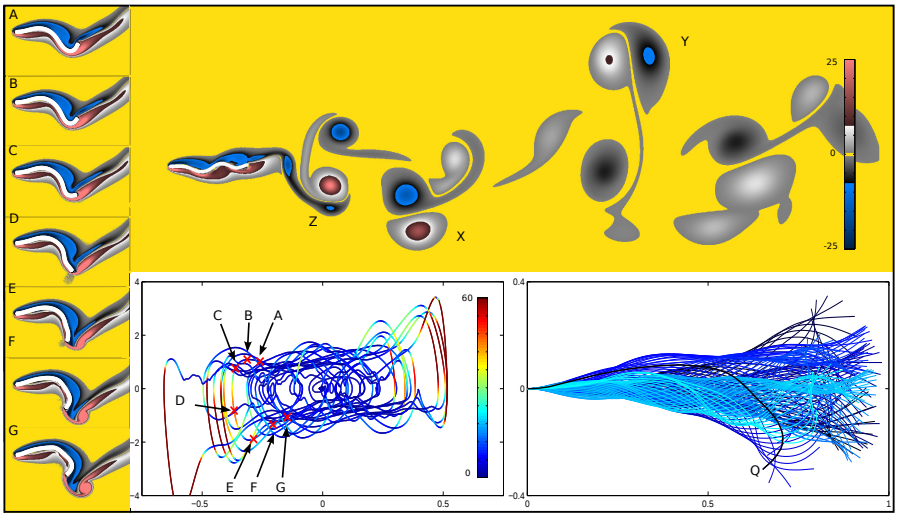


Figure 6.1.5 – Chaotic state: Re=750 and $u^\star = 30$. Left side: (A–G) time history of a snapping event with a snapshot every $\Delta t = 0.05$. Top: the whole vorticity field at $T = 11.70$, exhibiting the typical vortex pairs of different sign (X and Y), and the strongly asymmetric vortex pair resulting from the snapping event (Z). Bottom, middle: Phase diagram of the end point, colored with absolute acceleration. Bottom, right: time history of the deflection line with $\Delta t = 0.05$, color ranging from dark to light blue.

rectly after the breaking of symmetry. Effectively, the initial perturbation may be followed by a phase of regular flapping, before this regularity breaks down and the beam performs chaotic oscillations. Therefore it is difficult to judge whether a configuration is chaotic or periodic. The last simulations marked as periodic in figure 6.1.1 did not show a transition for a long time ($T_{\text{stop}} \geq 80$ corresponding to ≈ 50 periods), yet still it cannot be completely excluded that the periodicity will not break down. Note that in both [109] and [25] such a transitional behavior is not reported. However, [25] considers the limit of vanishing bending stiffness, and may therefore not be compared directly with the present results, and the method used in [109] relies on an inviscid approach. It can be conjectured that the finite Reynolds number inhibits the chaotic state for a certain time, and that this delay decreases with increasing Re.

To further investigate this transition, we apply the continuous wavelet transform (CWT) to the time evolution of the drag force. The CWT transforms the signal into the time-frequency domain [49]. We use the complex-valued Morlet wavelet with the wavenumber k_ψ denoting the barycenter of the wavelet

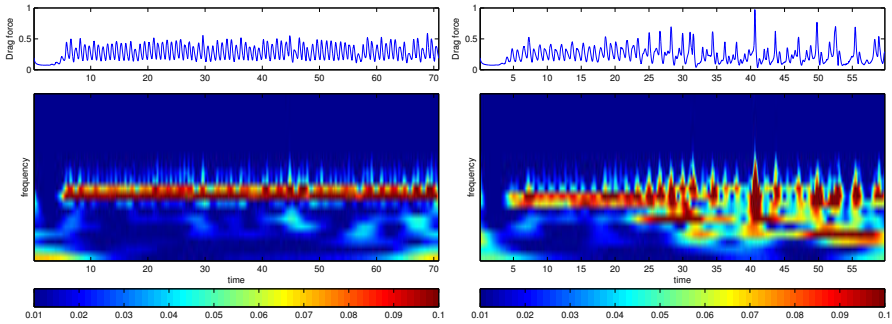


Figure 6.1.6 – Wavelet analyses of the drag force for case $u^\star = 20$ (left) and $u^\star = 22$ (right). The modulus of the complex valued Morlet wavelet coefficients are plotted as a function of position and scale. The original signal is plotted on the top.

support in Fourier space. Here we use $k_\psi = 5$.

Figure 6.1.6 shows the modulus of the wavelet-coefficients as a function of time and scaling factor (which is related to frequency), and the time history of the underlying drag signal on top, for Re $= 750$ and $u^\star = 20$ and 22. The former case is the last value of $u^\star$ found to be in a periodic state and the latter is the first chaotic one. The modulus of the wavelet-coefficients yields an energy distribution in the scale-time domain.

For the $u^\star = 20$ case, one distinct dominant frequency can be found, corresponding to a horizontal band in wavelet-space. This frequency does not alter significantly, albeit some deviations from a purely periodic state can be observed. Note that the values for the lowest frequencies close to the beginning and the end of the signal are due to boundary effects.

Considering the $u^\star = 22$ case, a different behavior can be observed. The drag force exhibits a periodic behavior until $T \approx 22$, similar to the $u^\star = 20$ case. After this time, a broad band of frequencies can be observed, characteristic for a chaotic state.

In periodic states, neither the flapping amplitude nor the frequency undergo a significant change when $u^\star$ is increased, hence the Strouhal number St $= 2\,A\,f\,/\,u_\infty$ remains practically constant, as illustrated in figure 6.1.4. The standard deviation from the mean value, represented by the dash-dotted line, is 4.5% and there is no distinct trend visible. The comparison with the values reported in [109] yields reasonable agreement. In particular, the constant flapping frequency matches well the experimental results reported by Shelly et al. [150], where a linear dependency on the inflow velocity has been found. Therefore the frequency is a constant when normalized with u_∞. The Strouhal number is constant, and does in particular not approach a “nat-

ural frequency of the fluid dynamics defined by a universal Strouhal number" [25] of St $\approx$ 0.2. This result seems to somewhat contradict those of Connell and Yue [25]. They explained the transition to chaos with a resonance phenomenon when the flapping frequency matches this universal Strouhal number. A possible explanation may be that [25] considered the limit of vanishing bending stiffness, while here we deal with a finite one. We conjecture that the transition mechanism is different when the bending stiffness becomes larger.

6.2 Thrust generation by flexible heaving foils

The content of this section is adapted from [46] and [83].

6.2.1 Introduction

In forward flight, flapping wings generate the desired thrust by performing up- and downstrokes combined with the supination and pronation movements that set the angle of attack. A canonical two–dimensional approximation of this process is the heaving and pitching motion of a foil, as illustrated in figure 6.2.1. If the wing is assumed to be rigid, the time evolution of the positional angle ϕ and the feathering angle α must be imposed, as described in chapter 3, cf. Fig. 6.2.1(a). In the two- dimensional approximation, shown in Fig. 6.2.1(b), this is equivalent to prescribing a heave motion $y_{\mathrm{le}}(t)$ and a pitch motion $\alpha(t)$ (see, e.g., [164]). However, if the wing is flexible and clamped at the leading edge, the effective angle of incidence varies passively due to the action of inertial and aerodynamic forces (Fig. 6.2.1c), and this mechanism also generates thrust with only the heave motion imposed (see, e.g, [81, 134]). Note that there are several distinct effects associated with the spanwise and the chordwise flexibility [81, 120, 121], and this section focuses on the chordwise flexibility only.

The setup is inspired by the experimental work presented in [134], where a "mechanical flapper" was presented. The device consists of two (virtually) rigid bars with attached elastic foils. The rigid bars virtually eliminate spanwise deformation, and thus the foil can be considered as flexible only in the chordwise direction. The whole robot was mounted on a "merry-go-round", i.e., a long rod that could rotate freely and thus emulate forward flight. An extension of this work was presented in [66], where four wings were considered, similar to the dragonfly in Fig. 6.2.1.

6.2.2 Single wing section

We now consider a single flexible foil immersed in a two-dimensional setup as illustrated in Fig. 6.2.2A. Its mechanical constants are derived from [134], where semi-circular wings of diameter $D = 0.06\,\mathrm{m}$ and mid-span chord length $c = 0.03\,\mathrm{m}$ were considered. The wing model chosen here as reference has a bending rigidity of $EI = 1.83 \cdot 10^{-4}\,\mathrm{Nm}$[1], mass per unit area $\varrho_s = 10.63 \cdot 10^{-2}\,\mathrm{kg/m^2}$, thickness $h = 0.078\,\mathrm{mm}$ and a relaxation frequency of $f^\star = 34.2\,\mathrm{Hz}$. This yields the following constants describing the solid model:

$$\mu = 2.8902 \qquad \eta = 5.5284$$

[1] Note that EI is stiffness per unit depth

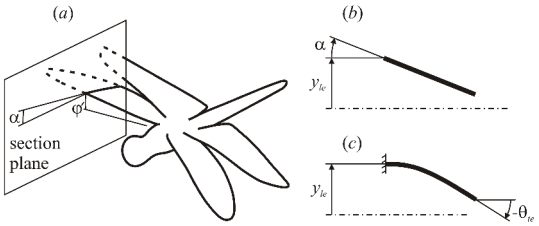


Figure 6.2.1 – Schematic illustration of the chordwise section of a three–dimensional wing (a), a two–dimensional approximation using a rigid heaving and pitching foil (b) and a flexible foil clamped at its leading edge (c).

f/f_0	1.29	1.19	1.08	1.00	0.97	0.92	0.86	0.80
N	25	25	25	25	25	25	25	25

f/f_0	0.76	0.65	0.54	0.43	0.32
N	25	25	20	12	9

Table 6.2 – Physical parameters: the heaving frequency normalized by the first resonant and the number of strokes computed in each simulation.

where the density of air is assumed to be $\varrho_f = 1.226\,\mathrm{kg/m^3}$. The corresponding first linear eigenfrequency is $f_0 = 3.516/2\pi \cdot \sqrt{\eta/\mu} = 0.77$. The leading edge is actuated using

$$y_{\mathrm{le}} = a_0 \sin(2\pi f t)$$

where $a_0 = 0.5$ and the heaving frequency f will be the main parameter of investigation. The Reynolds number, based on the free-stream velocity and the length of the foil, $Re = u_\infty \ell/\nu$ is set to 300, which is lower than the experimental value of 2000, but higher than in the preliminary publication [83]. Meanwhile, similarities between the present simulations and experimental findings suggest that Reynolds number effects are not dominant in this particular problem.

Initially, at time $t = 0$, the fluid velocity is equal to the imposed mean flow u_∞. Then, the heaving is set in motion, while a startup conditioner is applied during the first stroke to avoid problems with the pressure singularity at $t = 0$ due to the impulsively started motion. The resulting trailing edge deflection is depicted in Fig. 6.2.2B for the case $f/f_0 = 0.8$. A slight phase shift between the leading and trailing edge deflection can be observed, and the deflection of the trailing edge is quite significant. Owing to the dynamic deformation, the panel produces thrust and lift forces, illustrated in 6.2.2C-D. The averaged lift

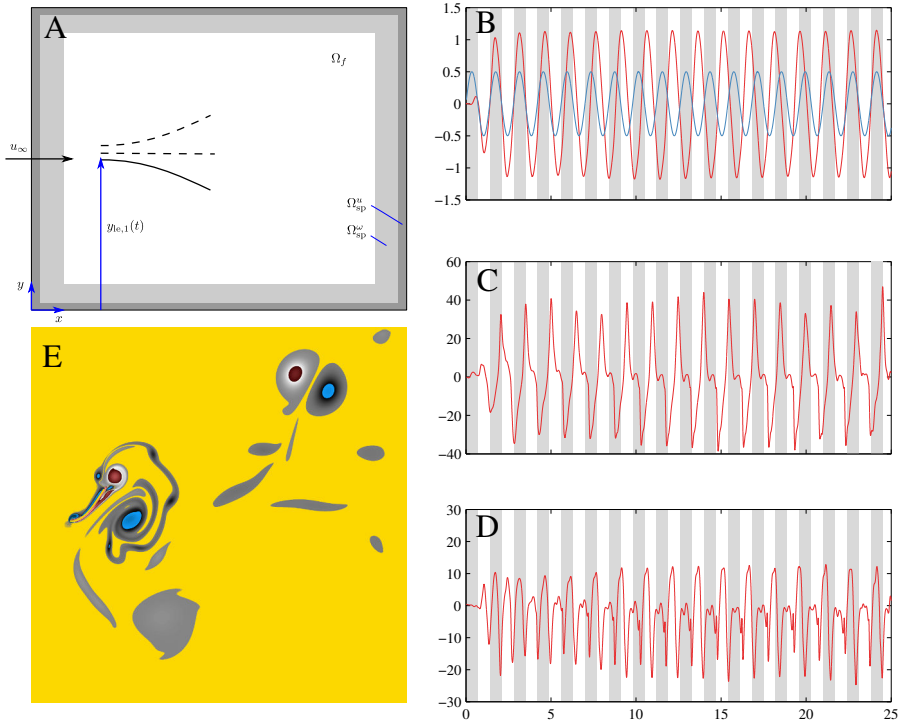


Figure 6.2.2 – Single wing section. The flexible foil follows a prescribed heaving motion (A) in a computational domain of size 10×10*, where the length of the foil is* 1*. The outflow boundary is modeled with a vorticity sponge* ($\Omega^\omega_{\text{sp}}$)*, and the remaining structures are destroyed by the velocity sponge* (Ω^u_{sp})*. The axial mean flow* u_∞ *is fixed to unity. (B) Prescribed heaving motion at the leading edge (blue) and resulting displacement of the trailing edge (red), in the case* $f/f_0 = 0.8$*. Gray shaded areas correspond to the first half of every full period* $T = 1.4$*. (C-D) Time series of the lift force (C) and the thrust force (D). The flow field is visualized in (E) by the vorticity, which is scalar-valued in the 2D case.*

force is zero, though the cycle averages over individual strokes may have non-zero values. This latter can cause slow fluctuations of the center of gravity in freely flying insects. The cycle-averaged axial force is negative, meaning that indeed thrust and not drag is produced. A typical snapshot of the vortical flow field is illustrated in Fig. 6.2.2E, showing the formation of typical vortex pairs.

We now turn to studying the effects of the prescribed heaving frequency f, varying it between $0.32 f_0$ and $1.29 f_0$ using 13 intermediate values, see table 6.2. All other problem parameters remain constant. For higher frequencies, the flow gets more erratic and we therefore raise the number of strokes, N, in these cases from 9 to 25. All simulations are discretized using high resolution (3072×3072) with a penalization parameter of $C_\eta = 6.25 \cdot 10^{-5}$ ($K = 0.14$). In [46] we also showed data obtained at coarser resolution, indicating validity of the results. Fig. 6.2.3 depicts the averaged values for thrust, efficiency and trailing edge angle. The thrust force is simply the axial force, which is averaged over $N - 2$ computed cycles, where the first two cycles are excluded because of the nonphysical initial condition. In agreement with the experiments [134] and our preliminary data [83], the maximum thrust is obtained at $f/f_0 = 0.86$, which is below the resonant frequency in vacuum. At the lower end of the parameter range, virtually no thrust is produced. The value drops rapidly after the maximum and crosses zero thrust near the resonant frequency. Even higher frequencies then produce drag instead of thrust. Because of the erratic flow nature, thrust is given together with the 95% confidence interval.

The efficiency is computed as the ratio of input power, i.e., the work per time of the external forces $[p]^{\pm}$ and $[\tau]^{\pm}$, and the output power $F_{\text{thrust}} \cdot u_\infty$. Note the input power corresponds to the aerodynamic power defined in eqn. (3.2.8). In figure 6.2.3B, a broad plateau with values around 0.1 is observed; the maximum thrust generated at $f/f_0 = 0.86$ also lies in this region of maximum efficiency. The rapid drop in thrust and efficiency in the vicinity of the resonant frequency can possibly be explained by the trailing edge deflection angle shown in Fig. 6.2.3C. It reaches values near 90° at the same frequency where the thrust is maximized. Further increase in frequency result in larger angles. This indicates that at the high-frequency end of the parameter range, fluid momentum is directed forwards and thus decreases the thrust force.

6.2.3 Two wing sections

It is an appealing feature of the present method that it can be extended to take several obstacles into account, a task that may present some extra difficulties when applying body-fitted grids to this problem. Here, we present some preliminary results for two wing sections in interaction. Thus, we place

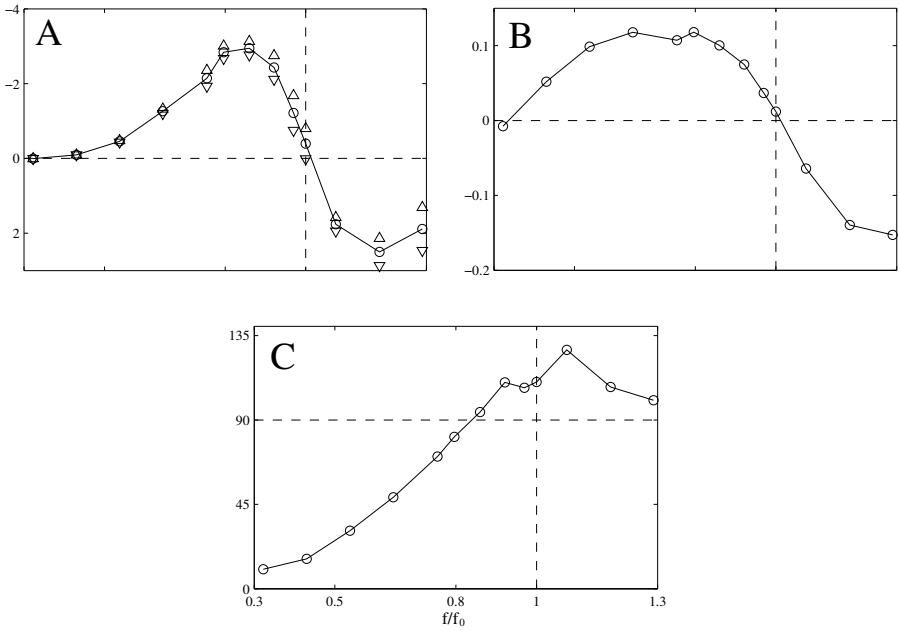


Figure 6.2.3 – Averaged thrust (A), efficiency (B) and trailing edge angle (C) as a function of the heaving frequency normalized with the first linear resonant frequency in vacuum. The thrust is given with its 95% confidence interval indicated by triangles.

a second beam behind the first one, that follows the same imposed motion, but with a phase shift δ, $y_{\mathrm{le},2} = y_0 + y_f \sin(2\pi f t + \delta)$. We fix the heaving frequency to $f/f_0 = 0.80$, which is the last datapoint before the maximum thrust in figure 6.2.3.

Inspired by the paper of Usherwood and Lehmann [168], we investigate the influence of the phase shift on the wake structures for two values of δ, namely $\delta = 0.75\pi$ and $\delta = 1.25\pi$. As we are interested in the wake structure, the computational domain is $L_x \times L_y = 30 \times 20$ chord lengths, and the resolution is 4608×3072 nodes. The penalization parameter is $C_\eta = 6.25 \cdot 10^{-5}$. The same sponge technique as for the single foils is applied, see figure 6.2.2.

In figure 6.2.4 (left column), the instantaneous vorticity fields for both cases are shown. They reveal that the $\delta = 1.25\pi$ case (bottom) yields a top/bottom asymmetry in the wake, as stronger dipoles are shed downwards while only little vortices are shed on the upper part of the wake. The time averaged vorticity field on the other hand reveals a more symmetric structure, indicating that the wake settles in either direction during the simulation. The time

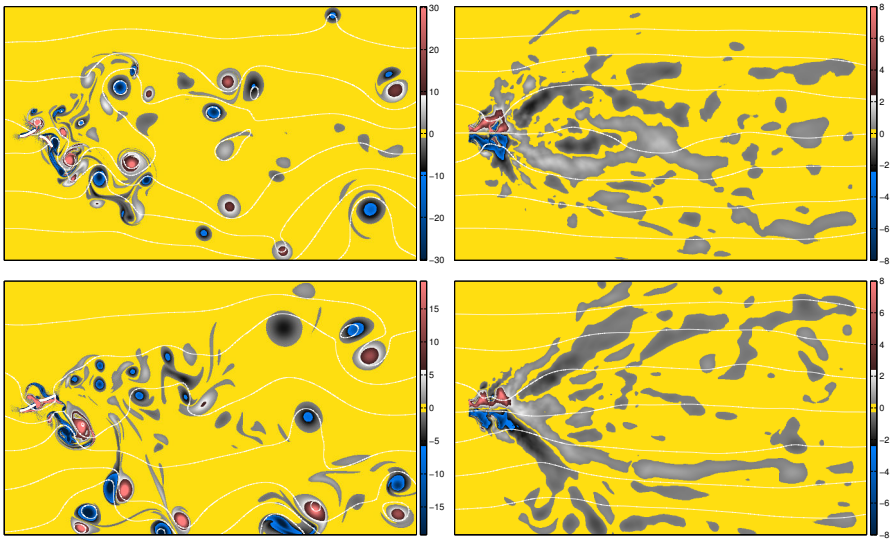


Figure 6.2.4 – Two heaving foils in tandem, instantaneous (left) and time-averaged (right) vorticity fields. For visibility, only part of the computational domain is shown. The phase shift is $\delta = 0.75\pi$ (top) and $\delta = 1.25\pi$ (bottom).

averaged vorticity in the $\delta = 0.75\pi$ case (top, right) also shows a symmetric structure, but the vorticity is overall spread less widely.

Chapter 7

Three-dimensional fluid-structure interaction: Application to swimming

Preliminary results of this section have been published in [47].

Swimming organisms exploit bending waves to produce propulsive force, an effect which has been extensively studied. Predicting the cruising velocity, however, remains challenging, as the drag force has to be taken into account. In this chapter, we numerically simulate simplified "swimmers", which consist of a chordwise flexible plate undergoing a driven pitching motion at the leading edge, immersed in a viscous incompressible fluid. The solid is actively coupled with the fluid, i.e., we deal with a fluid–structure interaction problem. The emphasis is placed on the longitudinal tip vortices, which result from the finite span, and their contribution to the drag force.

The usage of flexible foils for thrust generation as a simplified model for swimming organisms is common in both experimental and numerical contributions. Dewey et al. [28] for instance studied the flexible pitching panels experimentally. They find the efficiency, i.e., the ratio of thrust to power coefficient, to be maximized if the Strouhal number is in the optimal range $0.25 < St < 0.35$ and the pitching frequency is tuned to the structural resonant frequency of the foil. The former finding is supported by a variety of contributions [41, 163, 164]. The connection between the driving frequency f and the resonant frequency f_0 is subject of a controversy in the community. Kang et al. [81] state that operating at or near a structural resonant will result in enhanced performance, and that this fact is widely accepted. However, different studies found the precise relation f/f_0 to vary apprecia-

bly. In section 6.2, we already discussed the result from [134], which states optimal performance around $f/f_0 = 0.7$. Our 2D data points in the same direction, although the difference to the resonant is smaller. Yeh and Alexeev [186] found two regimes which maximize cruising speed and efficiency at $f/f_0 \approx 1.1$ and 1.6, respectively, as discussed briefly in section 5.3.3. However, they normalize by the resonant frequency in fluid, which can be derived analytically [48]. Contrarily to these findings, Vanella et al. [171] provides evidence for peak efficiency in flexible insect wings around 0.33. The invoked argument is the usage of superharmonic resonances, as also stated in [134]. Collectively, these findings indeed suggest an important role of the resonant frequency, though the exact relation remains not fully understood.

The total drag acting on these swimming organisms or robots can be decomposed in the contributions of the friction drag and the vortex induced drag. The former contribution has been relatively well explored. Theoretical studies have considered the laminar boundary layer, which is either compressed or stretched by the undulatory motion of the swimmer [95]. This effect is usually referred to as the "Lighthill boundary-layer thinning hypothesis". More recently, Ehrenstein et al. [35] employed high-quality numerical simulations using body-fitted meshes to quantify and verify this hypothesis.

The vortex induced drag, which may play a significant role, has only recently gained the attention of experimentalists. In the context of simplified mechanical swimming robots, Raspa et al. [135] established a basic model to explain the influence of the finite aspect ratio by the formation of trailing longitudinal tip-vortices. The present numerical study is inspired by these experiments, and should be seen as complimentary approach, given the difficulty of experimentally measuring the instantaneous flow field appropriately. In a first step, using rectangular swimmers, we will reproduce some experimental results and confirm the interpretation that the tip vortices play a major role in the drag force of the swimmer. In a second step, we move on and modify the swimmer's shape and find that a contracting form may be advantageous in terms of terminal cruising speed.

7.1 Materials

The setup is illustrated in figure 7.1.1. The swimmer is tethered to the computational domain and accelerates the mean fluid flow instead. In the first part of the study, we will keep the mean flow fixed, i.e., we consider all swimmers at the same speed. This simplified the comparison. In a second step, the mean flow is released and computed dynamically from the thrust force.

The material properties reported in [135] yield $\mu^{\text{exp}} = 0.0012$ and $\eta = 0.0134$. For numerical stability reasons, we set $\mu = 0.0096$ instead, as will be explained

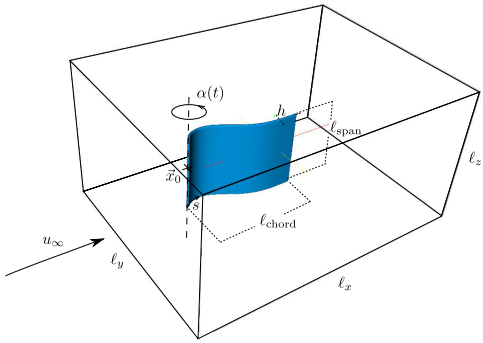


Figure 7.1.1 – Setup used in the present work. The swimmer, which consists of a spanwise flexible plate undergoing an imposed pitching motion $\alpha(t)$, is immersed in a viscous incompressible fluid with imposed mean flow $\vec{u}_\infty$.

later. Two-dimensional simulations confirmed that the solution is not very sensitive to the value of μ in this regime. The swimmer's length $\ell_{\text{chord}} = 0.15\,\text{m}$, the fluid density $\varrho_f = 1000\,\text{kg/m}^3$, a time scale $T = 1\,\text{s}$ and the velocity scale $U = \ell_{\text{chord}}/T$ have been used for normalization. The imposed pitching motion is sinusoidal, $\alpha = \alpha_{\max}\sin(2\pi f t)$ with $\alpha_{\max} = 50°$. Contrary to the experiment, we do not vary the driving frequency f but keep it fixed at unity (thus $f = 1\,\text{Hz}$).

7.2 Rectangular swimmers: influence of the aspect ratio

In this section we present the results obtained for rectangular plates. The Reynolds number is $Re = U\ell_{\text{chord}}/\nu = 1000$, the swimmer is computed in a box of size $2.66 \times 2.00 \times 1.33$ wherein its leading edge at mid-span is located at $\underline{x}_0 = \begin{pmatrix} 0.5 & 1.0 & 0.66 \end{pmatrix}$. At this Reynolds number, we found a resolution of $512 \times 384 \times 256$ to be sufficient. The original experiment is performed at much higher Reynolds number of $Re = 22500$, which is currently out of scope for direct numerical simulations. The value of the penalization parameter is $C_\eta = 10^{-3}$ ($K = 0.19$). The constant mean flow $\underline{u}_\infty = \begin{pmatrix} 0.5 & 0 & 0 \end{pmatrix}$ is impulsively started at $t = 0$ (thus the initial condition is $\underline{u}(\underline{x}, t = 0) = \underline{u}_\infty$), and we computed a total of 5 periods. Since our discretization is periodic, a vorticity sponge term is applied to all faces of the domain to prevent vortices from re-entering the domain, with a parameter of $\varepsilon_{\text{sp}} = 10^{-1}$, see section 2.5 for more information about that boundary condition. During the first pe-

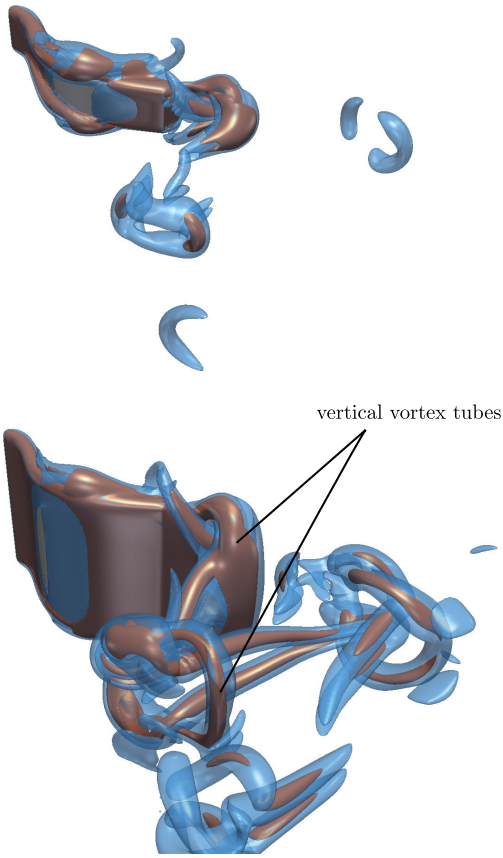


Figure 7.2.1 – Isosurfaces of vorticity $\|\underline{\omega}\| = 18$ *(copper) and* $\|\underline{\omega}\| = 9$ *(blue) for aspect ratios* $AR = 0.2$ *(top),* $AR = 0.7$ *(bottom). For* $AR = 0.2$*, the tip vortices dominate the flow. In the* $AR = 0.7$ *case, vertical vortex tubes are visible.*

riod, the imposed pitching angle is multiplied by a startup conditioner, i.e., $\alpha\,(t<1) = \left(-20t^7+70t^6-84t^4+35t^4\right)\alpha_{\max}\sin\left(2\pi f t\right)$, in order to avoid an impulsively started motion, which would yield a pressure singularity. We carry out five simulations with varying aspect ratio, $AR = \ell_{\text{span}}/\ell_{\text{chord}} = \left\{\begin{array}{ccccc} 0.2 & 0.3 & 0.5 & 0.7 & \infty \end{array}\right\}$. The latter value indeed corresponds to a 2D simulation, because the vortical structures are stable at Re=1000, which was verified by simulating a plate that extends periodically throughout the z-direction.

The vortical structure of the flow field is visualized in figure 7.2.1 for the

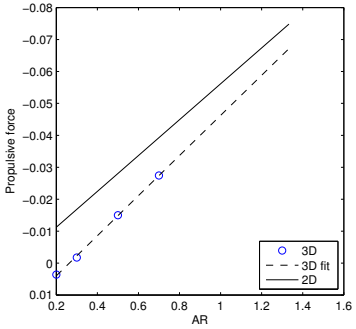


Figure 7.2.2 – Thrust force as a function of the aspect ratio. The solid line represents the force predicted by the 2D approximation and the dashed line is a linear least-squares fit through the available data points from the 3D simulations.

smallest and largest value of AR, at the beginning of the fourth stroke $t = 4.05$. In the $AR = 0.7$ case, the vertical vortex tubes can be observed. These tubes correspond to the vortices shed in the 2D case, where $AR = \infty$. They connect to the tip vortices and form ring-like structures, propagating perpendicular to the mean flow which also advects them downstream. It is visible that, in the $AR = 0.2$ case, the tip vortices actually dominate the wake structure—the vertical vortex tubes are not clearly distinguishable.

The cycle-averaged thrust force, that is the x-component of the hydrodynamic force, is shown in figure 7.2.2. Note that thrust points in negative x-direction. The solid line represents the prediction based on 2D simulations. The four 3D simulations are marked by circles. It can be observed that the thrust scales almost linearly with the aspect ratio. Assuming linearity, we make the ansatz

$$F_x^{3D} = F_{\text{thrust}} \cdot AR + F_{\text{tip}}$$

and fit the coefficients $F_{\text{thrust}} = -0.0628$ and $F_{\text{tip}} = 0.0165$ using least squares to the data points. In the case of 2D simulations, we found $F_x^{2D} = F_{\text{thrust}}^{2D} \cdot AR$ where the thrust per unit span is $F_{\text{thrust}}^{2D} = -0.0561$. We can thus observe that the values for the thrust per unit span are quite similar in both 3D and 2D cases, $F_{\text{thrust}} \approx F_{\text{thrust}}^{2D}$, and that the tip vortices indeed act like a constant offset. We can thus numerically confirm the experimental results presented by Raspa et al. [135].

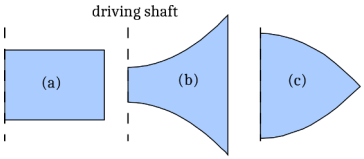


Figure 7.3.1 – Different shapes investigated here, termed (a) "rectangular", (b) "expanding" and (c) "contracting". All three shapes have the same surface.

7.3 Non-rectangular swimmers: influence of the fin shape

The results for rectangular plates illustrate the importance of tip vortices for the total drag. Actual fish, however, have of course non-rectangular caudal fins, with possible consequences for the vortical structures in the wake. We choose an additional set of two different shapes, an expanding and a contracting form, to study their influence on the cruising speed. All shapes have the same surface and follow the same imposed driving motion. For simplicity, we still assume the mechanical structure to be one-dimensional and with constant μ and η, although the varying $\ell_{\text{span}}(s)$ suggests that both should depend on s. This is a first order approximation, since both μ, η are linear in ℓ_{span}, but as $\eta \propto h^3$ the dominant effect of the stiffness η is captured and as μ is small anyways (light swimmer), this assumption seems justified. The non-rectangular shapes are defined as

$$\begin{aligned} \ell_{\text{span}}^{\text{exp}}(s) &= 2\left(\frac{0.35}{2} + 0.525s^2\right) \\ \ell_{\text{span}}^{\text{contr}}(s) &= 2\left(\frac{1.05}{2} - 0.525s^2\right) \end{aligned}$$

and are illustrated to scale in figure 7.3.1.

Since our swimmers remain anchored to the laboratory frame and instead move the surrounding fluid, it can be time-consuming to compute the steady state velocity, depending on the fluid domain size. In order to access the cruising speed, we modify the mean flow equation according to eqn. (2.6.3), with the initial condition $u_\infty(t=0) = 0$. The mean flow is then set to $\underline{u}_\infty = \begin{pmatrix} u_\infty & 0 & 0 \end{pmatrix}^T$. The fluid mass to be accelerated is set to a relatively small value, $m_{\text{fluid}} = 0.1235$, in order to speed up the computation because the steady state is reached sooner than with larger values. Without modification, the fluid mass would be equal to the volume of the computational domain, $\ell_x \ell_y \ell_z = 7.07$. Figure 7.3.2 illustrates the result obtained for all swimmers. They all reach their steady state (to a good approximation) within

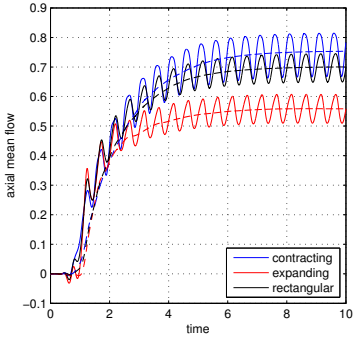


Figure 7.3.2 – Axial mean flow over time for the three different swimmers from figure 7.3.1. Solid lines is instantaneous data, dashed lines is a moving average over the duration of one stroke.

10 strokes, but the resulting cycle-averaged cruising speed and power expenditures significantly depend on the swimmer's shape. The contracting shape ($u_\infty = 0.75$) outruns both the rectangular ($u_\infty = 0.70$) and the expanding ($u_\infty = 0.55$) shapes in terms of the cruising speed, but requires the smallest aerodynamic power expenditures, computed according to eqn. (3.2.8). Another remarkable difference between the three simulations is that the expanding one has the smallest trailing edge displacement d, which is due to the larger concentration of area there. The pressure acting on the tail is thus much higher in that case, reducing the deflection amplitude.

All swimmers have finite span and thus exhibit tip vortices, and again these vortices offer a potential explanation for the higher cruising velocity of the contracting shape. *A priori*, one might think the expanding form is advantageous, since the larger trailing edge will produce larger vertical tube vortices (cf figure 7.2.1), and thus reduce the spurious three-dimensional effects.

However, the opposite is true. Figure 7.3.3 shows the vortical structures for the contracting and expanding shape at the same time, which is during the steady cruising state. The tip vortices, shed in both configurations around mid-chord, are advected downstream due to the mean flow, and they can be associated with a zone of lower pressure. This drop in pressure creates a local net force pointing in the direction of the vortex core, and part of which contributes to the total drag force (depending on the orientation of the surface normal relative to the x-direction). Visibly, in the contracting case, the tip vortex quickly looses contact with the actual swimmer – its parasitic drag is thereby reduced. The opposite is true for the expanding type swimmer: not only does the tip vortex not loose contact with the swimmer, it does instead even increase the portion of the swimmer influenced by the tip vortices, com-

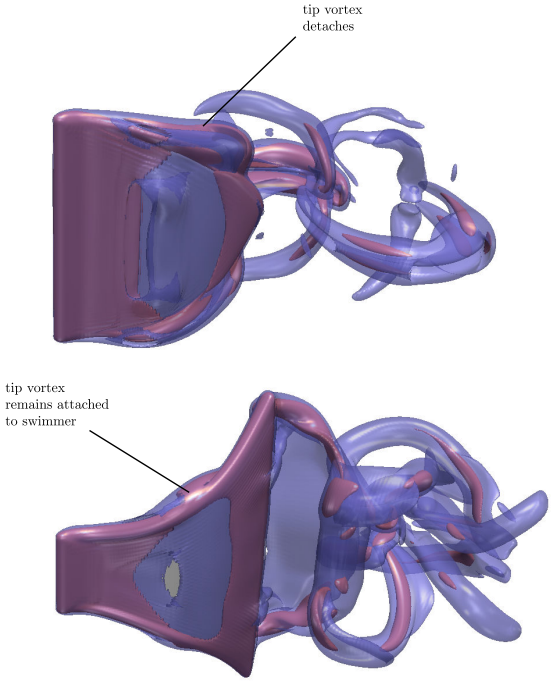


Figure 7.3.3 – Isosurfaces of vorticity magnitude for contacting (top) and expanding shapes (bottom). Blue $\|\underline{\omega}\| = 18$, purple: $\|\underline{\omega}\| = 9$. In the contracting case, the tip vortices rapidly detach from the swimmer, while they remain close to the surface in the expanding one. This finding may explain the observed difference in performance.

pared to the rectangular swimmer. It can also be noted that the total mean enstrophy,

$$\langle Z \rangle = \left\langle \iiint \|\underline{\omega}\|^2 \, d\underline{x} \right\rangle,$$

which is a measure for the dissipation in the fluid wake, is significantly higher in the expanding than in the contracting case, $\langle Z \rangle_{\text{contracting}} = 98.7$ versus $\langle Z \rangle_{\text{expanding}} = 127.7$, indicating a higher dissipation rate in the expanding case.

Shape	u_∞	P_{aero}	d
contracting	0.75	0.16	0.62
rectangular	0.70	0.18	0.48
expanding	0.56	0.21	0.42

Table 7.1 – Cycle-averaged data (cruising speed, aerodynamic power and trailing edge displacement amplitude) for the three swimmers.

7.4 Conclusion

We numerically simulated the flow past simplified elastic swimmer models. These models consist of flexible plates that have a driven pitching angle at their leading edges. In the first part, we simulated rectangular swimmers, that are inspired by the experimental work presented in [135]. We confirmed the finding that the tip vortices are a significant contribution to the total drag, and thus should be taken into account when predicting the cruising speed of these swimmers. We showed that indeed the 3D nature of these flows acts like a parasitic drag that is virtually independent of the aspect ratio.

In a second step, with the insight gained in the first one, we investigated shapes other than rectangular, namely a contracting and an expanding one, and compared their cruising velocities. We found that the contracting shape is the best, and postulate that this may possibly be explained by the tip vortices quickly "loosing touch" to the swimmer, which reduces their influence on the drag.

Part IV

Conclusion and perspectives

Conclusions This work is placed in the interdisciplinary field of biologically-inspired fluid dynamics, focusing on numerical simulations. Its contribution is therefore twofold, with the development of numerical methods on the one hand and applications to insect flight and swimming on the other.

The numerical method is based on the work of Kolomenskiy and Schneider, who previously developed two– and three–dimensional codes, employing Fourier pseudospectral methods combined with the volume penalization method to take obstacles into account. In various contributions, this combination proved fruitful for the simulation of fixed and moving rigid obstacles. The present work adopts the Fourier discretization and the volume penalization method from its precursors, but modifies the algorithm used for moving obstacles. In particular, this algorithm allows simulating flexible, deforming obstacles as well, but it is also found advantageous when rigid obstacles are considered. The basic idea is to rely on the signed distance function to move the Lagrangian mask function χ and resampling it on the Eulerian grid. This algorithm requires no additional Fourier transformations and is simpler to implement. The numerical code, `flusi`[1], is open source and freely available on the internet[2]. The implementation based on Fourier methods for the fluid has advantages and drawbacks. The numerical properties are favorable in terms of absent numerical diffusion and dispersion, and the incompressibility constraint can be handled in a simple and efficient manner. In fact, the most striking feature is simplicity; using tuned libraries allows to design simple and efficient codes, and to exploit massively parallel supercomputers. The drawback of the approach is the equidistant grid; if grid stretching were applied, the inversion of the Laplacian would have become a cumbersome problem. The constraint of equidistant grids impacts performance to a variable degree: for turbulence simulations, this grid is virtually optimal, while for simulations with more confined phenomena, it may effectively limit the domain size.

The first application, presented in part II, considers insects with rigid wings. Compared to previous work, the new algorithm allows to take into account complete insects, including body, head, wings as well as details such as legs and antennae. Furthermore, arbitrary motion of wings and body can be computed, as well as free–flight simulations with some or all degrees of freedom. Benchmarking of the code is done using configurations found in the literature, showing excellent agreement. Subsequently, we use our model to study a bumblebee in tethered forward flight, under laminar and turbulent inflow conditions. For the latter, homogeneous isotropic turbulence fields, also computed with the same code, are used as upstream perturbations to study the impact of turbulence on flapping flight. Our data highlight that the fundamental aerodynamic mechanisms, most importantly the leading edge vortex,

[1] Acronyme for **flu**id-**s**tructure **i**nteraction

[2] https://github.com/pseudospectators/FLUSI/

are not destroyed by turbulent perturbations. Therefore, the averaged flight characteristics are the same as in laminar flight, even in background turbulence as intense as 99% with respect to mean flow. This is a striking difference to man-made aircrafts, where the laminar-turbulent transition in the boundary layer can, for certain airfoils, be dramatically altered by inflow perturbations as small as 1%. Their aerodynamic characteristics are thus very different under laminar and turbulent conditions. From estimations of the impact of fluctuations in stroke-averaged forces and moments, we conclude that flapping flight in turbulence is a control problem rather than an aerodynamic one. Future work will consider untethered insects to further explore the effect of turbulence on flight stability.

The third part of this thesis deals with flexible solids interacting with a fluid. The collective long-time goal of these efforts is to replace the rigid insect wings by flexible ones. A one–dimensional solid model has been developed and implemented, which is used as flexible beam in two–dimensional setups and as a plate with one rigid direction in the three-dimensional case. The two–dimensional simulations are applied to the problem of thrust production through flexibility, where a heaving wing section has been considered. The pure heaving motion produces a net thrust force, owing to the non-zero effective pitching angle resulting from the deformation. We explored the link between the structural resonant and the heaving frequency, finding, similarly to experimental work, a peak efficiency below the first resonant frequency. We then proceed and consider three-dimensional flows interacting with plates with one rigid and one flexible direction. Validation tests confirm that our method is working, and we apply it then to the problem of a plate with imposed pitching motion. This configuration is intended to model undulatory swimming. Based on the simulations with rectangular plates, we eventually modify the lateral shape of the swimmer model, to take into account shapes typically observed at the caudal fin of swimming fish. This way, our model is interpreted more in the direction of carrangi- or thunniform swimming, rather than undulatory. The main finding is that, surprisingly, a contracting shape outperforms an expanding one in terms of both cruising speed and power expenditure, though the latter form is much more typical for actual fish fins.

Perspectives It is planned to complement the Fourier based solver by a wavelet-based adaptive solver. Especially for fluid–structure interaction, but also for a variety of problems currently out of scope for the spectral code, like insect swarming, this method could provide novel opportunities. The idea has been around for an extended period of time, however, at the end of this thesis, a robust idea is conceived on how to implement this type of numerical method. Essentially, the method of artificial compressibility will be combined with the volume penalization method. The former allows purely

explicit schemes well suited for adaptive solvers. It introduces a new parameter, the “artificial speed of sound”, which allows to control to what extend the divergence-free constraint is satisfied. First, extended testing has shown on uniform grids that the combination allows to balance the errors made in the boundary conditions and the incompressibility. The Fourier method heavily favors the latter, as all our flow fields are divergence-free to machine precision. With the method of artificial compressibility, this constraint can be relaxed and this may provide new opportunities to improve the overall error significantly.

On the physics side, the three–dimensional fluid–structure interaction tools are now sophisticated enough to gain valuable insight into the phenomena. We mainly intent to apply our methods to several heaving foils in tandem, in order to investigate the role of wing-wing interaction in four-winged insects like dragonflies. Preliminary two–dimensional simulations as well as experimental findings by Godio-Diana are promising.

The third path to be explored is the finding the optimal wing kinematics for a model insect using genetic algorithms. In fact, the Fourier method can be fast enough to compute thousands of realizations of, e.g., a fruit fly, and in combination with fewer, well resolved simulations, can find the best possible kinematics. It is of considerable interest to see the result of this optimization for a variety of flight modes, like hovering and forward flight, and we already implemented the automatized computational facility to perform this optimization.

Part V

Appendices

Appendix A

Parallel implementation

The results presented in this thesis are obtained by using high-performance computing. In this appendix, some aspects of parallel computation and efficiency are discussed. We focus on the computation of three-dimensional flows, because their demand for resources is considerably higher.

A.1 Data distribution

Modern high-performance computing relies on massively parallel machines, like IDRIS' IBM Blue Gene/Q machine "turing", on which a big part of the present computations have been performed. It has a total 98 304 CPU cores, with 1 GB memory per core, yielding 96 TB memory in total. Each CPU can only hold a portion of the data, as the total data exceeds the locally available memory by orders of magnitude. For example, a $1152 \times 768 \times 768$ simulation using the FLUSI code allocates a total of 154 GB.

The Fourier discretization requires us to perform fast Fourier transforms (FFT) in all three directions. However, as explained in 2.7, the Fourier transform does not have a finite support in physical space (while it is perfectly localized in Fourier space, see, e.g., [72]). Thefore, to perform an FFT in one direction, say the x-direction, all data in physical space in that direction is required. FFTs are therfore performed with the help of MPI transpose routines, meaning that the data is redistributed among MPI processes in order to make the desired direction contiguous, i.e., not split among CPUs. Our code performs a 1D decomposition along the third index if the number of CPU is smaller or equal the datapoints in z-direction, or a 2D decomposition along the second and thrid index, if the number of CPU exceeds the number of points in the z-

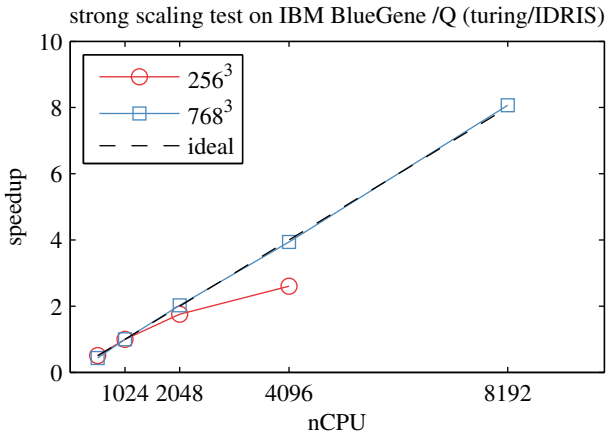


Figure A.2.1 – Parallel scaling: strong scaling tests of flusi code on IBM BlueGene / Q machine at IDRIS, Paris.

direction. The code can thus run on $n_y n_z$ CPUs at most, which however is not a limitation in practice.

A.2 Parallel scaling

Figure A.2.1 hows the speedup, i.e., $s = t_{ref}/t$, where t_{ref} and t is the elapsed time in the reference and actual simulation, respectively. The case chosen is a simulation of isotropic turbulence, but the speedup has been found the same in the case of insects. Two simulation sizes, 256^3 and 768^3 are tested using up to 8192 CPU cores. The reference simulation is run on 1024 cores. The small resolution case has a speedup of about 3 at 4096 cores, while the higher resoltion case is seen to scale almost perfectly up to 8192 cores. Many simulations in this theses are even larger than 768^3 and can therefore by run on 16384 CPU without a loss of efficiency.

Appendix B

Details on the solid model and its numerical solution

B.1 Beam equation with non-constant coefficients

In the derivation of the beam equation we assumed $\eta = \mu = \text{const}$, which is not striktly true if the shape of the plate varies. We here give the result of the derivation if both μ and η depend on s, and further simplify the notation by denoting derivatives with indices, i.e., $\partial^3\theta/\partial s^3 = \theta_{sss}$. Terms marked in red arise from the non-constant coefficients. The defining equation (5.1.6) for the tension becomes

$$\begin{aligned} T_{ss} - \theta_s^2 T - \frac{\mu_s}{\mu} T_s &= -\theta_s p - \eta\theta_{ss}^2 - 2\eta\theta_s\theta_{sss} - \mu\left(\dot{\alpha} + \dot{\theta}\right)^2 - \tau_s \\ &\quad -2\eta_{ss}\theta_s^2 - 5\eta_s\theta_s\theta_{ss} + \frac{\mu_s}{\mu}\left(\eta\theta_s\theta_{ss} + \tau + \eta_s\theta_s^2\right) \end{aligned} \tag{B.1}$$

For the evolution equation, we find instead of eqn. (5.1.7):

$$\begin{aligned} \mu\ddot{\theta} &= -\ddot{\alpha}\mu - p_s - \eta\theta_{ssss} + \theta_s\tau + 2T_s\theta_s + \theta_{ss}T + \eta\theta_s^2\theta_{ss} \\ &\quad -3\eta_s\theta_{sss} - \theta_s\eta_{sss} - 3\eta_{ss}\theta_{ss} + \eta_s\theta_s^3 \\ &\quad +\frac{\mu_s}{\mu}\left(p + 2\eta_s\theta_{ss} + \theta_s\eta_{ss} + \theta_{sss}\eta - \theta_s T\right), \end{aligned} \tag{B.2}$$

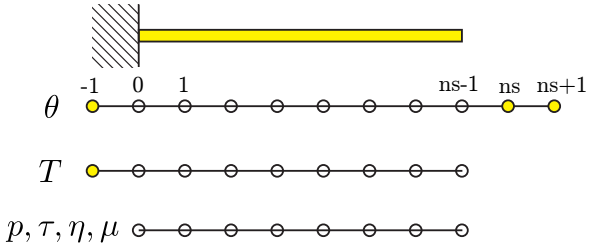


Figure B.2.1 – Spatial discretization of the beam. The yellow nodes are ghost nodes added to satisfy the non-linear boundary conditions.

which we denote as $\ddot{\theta} = f(\theta, T)$. The boundary conditions at the leading edge, $s = 0$, read (clamped+imposed acceleration):

$$\theta = 0 \tag{B.3}$$
$$T_s + \eta_s \theta_s^2 + \theta_s \theta_{ss} \eta = K_1 \tag{B.4}$$
$$\theta_s T - 2\eta_s \theta_{ss} - \theta_s \eta_{ss} - \theta_{sss} \eta = K_2, \tag{B.5}$$

where we introduced the constants

$$\begin{aligned} \mu(0)\left(\ddot{x}_0 \cos(\alpha) + \ddot{y}_0 \sin(\alpha) + g \sin(\alpha)\right) - \tau(0) &:= K_1 \\ \mu(0)\left(-\ddot{x}_0 \sin(\alpha) + \ddot{y}_0 \cos(\alpha) + g \cos(\alpha)\right) + p(0) &:= K_2. \end{aligned}$$

Finally, at the trailing edge, the boundary conditions for the free end read ($s = 1$, $N = M = T = 0$, no tensions):

$$T = 0 \tag{B.6}$$
$$\eta_s \theta_s + \eta \theta_{ss} = 0 \tag{B.7}$$
$$\eta \theta_s = 0 \tag{B.8}$$

B.2 Discretization in space and time

Spatially, the above set of equations is discretized with finite differences. Four ghost points are added to take the four non-Dirichlet conditions into acount, as illustrated in figure B.2.1. The discretized boundary conditions at the lead-

ing edge read

$$\begin{aligned}\frac{T_1 - T_{-1}}{2\Delta s} + \eta_0 \frac{(\theta_{-1} - 2\theta_0 + \theta_1)(\theta_1 - \theta_{-1})}{2\Delta s^3} + \eta_{s,0}\left(\frac{\theta_1 - \theta_{-1}}{2\Delta s}\right)^2 &= K_1 \\ T_0 \frac{\theta_1 - \theta_{-1}}{2\Delta s} - \eta_0 \frac{-3\theta_{-1} + 10\theta_0 - 12\theta_1 + 6\theta_2 - \theta_3}{2\Delta s^3} \cdots & \\ -2\eta_{s,0}\frac{\theta_{-1} - 2\theta_0 + \theta_1}{\Delta s^2} - \eta_{ss,0}\frac{\theta_1 - \theta_{-1}}{2\Delta s} &= K_2\end{aligned}$$

and their counterparts at the trailing edge, where we employ a fourth order discretization to take both additional nodes into account, read

$$\begin{aligned}\eta_{s,ns-1}\frac{\theta_{ns} - \theta_{ns-2}}{2\Delta s} + \eta_{ns-1}\frac{-\frac{1}{12}\theta_{ns-3} + \frac{4}{3}\theta_{ns-2} - \frac{5}{2}\theta_{ns-1} + \frac{4}{3}\theta_{ns} - \frac{1}{12}\theta_{ns+1}}{\Delta s^2} &= 0 \\ \eta_{ns-1}\frac{\frac{1}{12}\theta_{ns-3} - \frac{2}{3}\theta_{ns-2} + \frac{2}{3}\theta_{ns} - \frac{1}{12}\theta_{ns+1}}{\Delta s} &= 0.\end{aligned}$$

In the interior region of the beam, we use the following second order approximations for the derivatives

$$\begin{aligned}\theta_{s,i} &= \frac{-\frac{1}{2}\theta_{i-1} + \frac{1}{2}\theta_{i+1}}{\Delta s} \\ \theta_{ss,i} &= \frac{\theta_{i-1} - 2\theta_i + \theta_{i+1}}{\Delta s^2} \\ \theta_{sss,i} &= \frac{-\frac{1}{2}\theta_{i-2} + \theta_{i-1} - \theta_{i+1} + \frac{1}{2}\theta_{i+2}}{\Delta s^3} \\ \theta_{ssss,i} &= \frac{\theta_{i-2} - 4\theta_{i-1} + 6\theta_i - 4\theta_{i+1} + \theta_{i+2}}{\Delta s^4}.\end{aligned}$$

The stencil thus extends by two grid points in each direction. Therefore, the first and last point in eqn. (B.1) and (B.2), respectively, require special treatment with modified stencils.

To obtain the complete discrete beam equation, the second time derivative is rewritten as a first order system by introducing $\underline{w} = \left(\theta, \dot{\theta}\right)^T$. The differential equation then reads $\dot{\underline{w}} = \underline{w}$. The general time marching scheme is written as

$$\underline{w}^{n+1} = C_3\underline{w}^n + C_4\underline{w}^{n-1} + \frac{\Delta t}{C_1}\left(\underline{f}\left(\underline{w}^{n+1}\right) + C_2\underline{f}\left(\underline{w}^n\right)\right), \tag{B.1}$$

where the coefficients C_i allow switching between a first order Euler implicit (EI1), a second order Crank-Nicolson (CN2) and a second order backward-differentiation scheme (BDF2). The following table defines the constants for each time marching scheme.

	C_1	C_2	C_3	C_4
EI1	1	0	1	0
CN2	2	1	1	0
BDF2	$\frac{1+2\xi}{1+\xi}$	0	$\frac{(1+\xi)^2}{1+2\xi}$	$-\frac{\xi^2}{1+2\xi}$

All schemes are implicit and thus alleviate the time step restriction imposed by the fourth derivative. Equation (B.1) is a non-linear system of equations for the solution at the new time level, $\underline{w}^{n+1}$. Recasting it as a root problem reads

$$0 = C_3\underline{w}^n - \underline{w}^{n+1} + C_4\underline{w}^{n-1} + \frac{\Delta t}{C_1}\left(\underline{F}\left(\underline{w}^{n+1}\right) + C_2\underline{F}\left(\underline{w}^n\right)\right),$$

yielding

$$\begin{aligned} 0 \quad = \quad & \begin{pmatrix} C_3\theta^n - \theta^{n+1} + C_4\theta^{n-1} \\ C_3\dot{\theta}^n - \dot{\theta}^{n+1} + C_4\dot{\theta}^{n-1} \end{pmatrix} \\ & + \frac{\Delta t}{C_1}\left(\begin{pmatrix} \dot{\theta}^{n+1} \\ f\left(\theta^{n+1}, T^{n+1}\right) \end{pmatrix} + C_2\begin{pmatrix} \dot{\theta}^n \\ f\left(\theta^n, T^n\right) \end{pmatrix}\right). \end{aligned} \tag{B.2}$$

Thus, the angular velocity at the new time level can simply be determined as

$$\dot{\theta}^{n+1} = \frac{C_1}{\Delta t}\left(\theta^{n+1} - C_3\theta^n - C_4\theta^{n-1}\right) - C_2\dot{\theta}^n.$$

The tension at the old time level, T^n, can be computed from θ^n and is thus known. The semi-discrete evolution eqn. reads

$$\begin{aligned} 0 \quad = \quad & \underbrace{C_3\dot{\theta}^n + C_4\dot{\theta}^{n-1} + \frac{\Delta t}{C_1}C_2 f\left(\theta^n, T^n\right)}_{:=K_3} - \dot{\theta}^{n+1} + \frac{\Delta t}{C_1}f\left(\theta^{n+1}, T^{n+1}\right) \\ = \quad & K_3 - \frac{C_1}{\Delta t}\left(\theta^{n+1} - C_3\theta^n - C_4\theta^{n-1}\right) + C_2\dot{\theta}^n + \frac{\Delta t}{C_1}f\left(\theta^{n+1}, T^{n+1}\right) \\ = \quad & \underbrace{K_3 + C_2\dot{\theta}^n + \frac{C_1}{\Delta t}\left(C_3\theta^n + C_4\theta^{n-1}\right)}_{:=K_4} - \frac{C_1}{\Delta t}\theta^{n+1} + \frac{\Delta t}{C_1}f\left(\theta^{n+1}, T^{n+1}\right) \\ = \quad & K_4 - \frac{C_1}{\Delta t}\theta^{n+1} + \frac{\Delta t}{C_1}f\left(\theta^{n+1}, T^{n+1}\right). \end{aligned} \tag{B.3}$$

For brevity, we denote the non-linear root problem as

$$\underline{F}\left(\underline{x}\right) = 0,$$

where

$$\underline{x} = \left(\begin{array}{cccccccccc} \theta_{-1}^{n+1} & \theta_{0}^{n+1} & \cdots & \theta_{ns-1}^{n+1} & \theta_{ns}^{n+1} & \theta_{ns+1}^{n+1} & T_{-1}^{n+1} & T_{0}^{n+1} & \cdots & T_{ns-1}^{n+1} \end{array} \right),$$

and we thus have $2ns+4$ unknowns. The first six equations of $\underline{F}$ are the discrete boundary conditions (we drop the time index $n+1$ from now on to shorten the lengthy equations, all time-dependent quantities without index are at the new time level):

$$\begin{aligned}
F_1 &= \theta_0 \\
F_2 &= \frac{T_1 - T_{-1}}{2\Delta s} + \eta_0 \frac{(\theta_{-1} - 2\theta_0 + \theta_1)(\theta_1 - \theta_{-1})}{2\Delta s^3} \\
&\quad + \eta_{s,0} \left(\frac{\theta_1 - \theta_{-1}}{2\Delta s} \right)^2 - K_1 \\
F_3 &= T_0 \frac{\theta_1 - \theta_{-1}}{2\Delta s} - \eta_0 \frac{-3\theta_{-1} + 10\theta_0 - 12\theta_1 + 6\theta_2 - \theta_3}{2\Delta s^3} \\
&\quad - 2\eta_{s,0} \frac{\theta_{-1} - 2\theta_0 + \theta_1}{\Delta s^2} - \eta_{ss,0} \frac{\theta_1 - \theta_{-1}}{2\Delta s} - K_2 \\
F_4 &= T_{ns-1} \\
F_5 &= \eta_{s,ns-1} \frac{\theta_{ns} - \theta_{ns-2}}{2\Delta s} + \eta_{ns-1} \frac{-\frac{1}{12}\theta_{ns-3} + \frac{4}{3}\theta_{ns-2} - \frac{5}{2}\theta_{ns-1} + \frac{4}{3}\theta_{ns} - \frac{1}{12}\theta_{ns+1}}{\Delta s^2} \\
F_6 &= \eta_{ns-1} \frac{\frac{1}{12}\theta_{ns-3} - \frac{2}{3}\theta_{ns-2} + \frac{2}{3}\theta_{ns} - \frac{1}{12}\theta_{ns+1}}{\Delta s}
\end{aligned}$$

We then have the $(ns-2)$ regular interior points for the evolution eqn. (B.3),

hence $i = 1, ns-2$. Note we skip F_7 and F_8 for a moment.

$$
\begin{aligned}
F_{8+i} &= K_{4,i} - \frac{C_1}{\Delta t}\theta_i + \frac{\Delta t}{C_1}\left[f_i\left(\theta^{n+1}, T^{n+1}\right)\right] \\
&= K_{4,i} - \frac{C_1}{\Delta t}\theta_i + \frac{\Delta t}{\mu_i C_1}\left[-\eta_i \frac{\theta_{i-2} - 4\theta_{i-1} + 6\theta_i - 4\theta_{i+1} + \theta_{i+2}}{\Delta s^4}\right. \\
&\quad + \tau_i \frac{-\frac{1}{2}\theta_{i-1} + \frac{1}{2}\theta_{i+1}}{\Delta s} + 2\frac{-\frac{1}{2}T_{i-1} + \frac{1}{2}T_{i+1}}{\Delta s}\frac{-\frac{1}{2}\theta_{i-1} + \frac{1}{2}\theta_{i+1}}{\Delta s} \\
&\quad + T_i \frac{\theta_{i-1} - 2\theta_i + \theta_{i+1}}{\Delta s^2} + \eta_i\left(\frac{-\frac{1}{2}\theta_{i-1} + \frac{1}{2}\theta_{i+1}}{\Delta s}\right)^2 \frac{\theta_{i-1} - 2\theta_i + \theta_{i+1}}{\Delta s^2} \\
&\quad - 3\eta_{s,i}\frac{-\frac{1}{2}\theta_{i-2} + \theta_{i-1} - \theta_{i+1} + \frac{1}{2}\theta_{i+2}}{\Delta s^3} - \eta_{sss,i}\frac{-\frac{1}{2}\theta_{i-1} + \frac{1}{2}\theta_{i+1}}{\Delta s} \\
&\quad - 3\eta_{ss,i}\frac{\theta_{i-1} - 2\theta_i + \theta_{i+1}}{\Delta s^2} + \eta_{s,i}\left(\frac{-\frac{1}{2}\theta_{i-1} + \frac{1}{2}\theta_{i+1}}{\Delta s}\right)^3 \\
&\quad + \frac{\mu_{s,i}}{\mu_i}\left(p_i + 2\eta_{s,i}\frac{\theta_{i-1} - 2\theta_i + \theta_{i+1}}{\Delta s^2} + \right. \\
&\quad + \eta_{ss,i}\frac{-\frac{1}{2}\theta_{i-1} + \frac{1}{2}\theta_{i+1}}{\Delta s} + \eta_i\frac{-\frac{1}{2}\theta_{i-2} + \theta_{i-1} - \theta_{i+1} + \frac{1}{2}\theta_{i+2}}{\Delta s^3} \\
&\quad \left.\left. - T_i\frac{-\frac{1}{2}\theta_{i-1} + \frac{1}{2}\theta_{i+1}}{\Delta s}\right) - \ddot{\alpha}\mu_i^2 - p_{s,i}\right]
\end{aligned}
$$

and then the interior points for the tension eqn. (B.1) $(i = 1, ns-1)$

$$
\begin{aligned}
F_{7+ns+i} \quad = \quad & \frac{T_{i-1} - 2T_i + T_{i+1}}{\Delta s^2} - \left(\frac{-\frac{1}{2}\theta_{i-1} + \frac{1}{2}\theta_{i+1}}{\Delta s} \right)^2 T_i \\
& + p_i \frac{-\frac{1}{2}\theta_{i-1} + \frac{1}{2}\theta_{i+1}}{\Delta s} + \eta_i \left(\frac{\theta_{i-1} - 2\theta_i + \theta_{i+1}}{\Delta s^2} \right)^2 \\
& + 2\eta_i \frac{-\frac{1}{2}\theta_{i-1} + \frac{1}{2}\theta_{i+1}}{\Delta s} \frac{-\frac{1}{2}\theta_{i-2} + \theta_{i-1} - \theta_{i+1} + \frac{1}{2}\theta_{i+2}}{\Delta s^3} \\
& + \mu_i \left(\dot{\alpha} + \frac{C_1}{\Delta t} \left(\theta_i - C_3 \theta_i^n - C_4 \theta_i^{n-1} \right) - C_2 \dot{\theta}_i^n \right)^2 \\
& + \tau_{s,i} + 2\eta_{ss,i} \left(\frac{-\frac{1}{2}\theta_{i-1} + \frac{1}{2}\theta_{i+1}}{\Delta s} \right)^2 - \frac{\mu_{s,i}}{\mu_i} \frac{-\frac{1}{2}T_{i-1} + \frac{1}{2}T_{i+1}}{\Delta s} \\
& + 5\eta_{s,i} \frac{-\frac{1}{2}\theta_{i-1} + \frac{1}{2}\theta_{i+1}}{\Delta s} \frac{\theta_{i-1} - 2\theta_i + \theta_{i+1}}{\Delta s^2} \\
& - \frac{\mu_{s,i}}{\mu_i} \left(\eta_i \frac{-\frac{1}{2}\theta_{i-1} + \frac{1}{2}\theta_{i+1}}{\Delta s} \frac{\theta_{i-1} - 2\theta_i + \theta_{i+1}}{\Delta s^2} \right. \\
& \left. + \tau_i + \eta_{s,i} \left(\frac{-\frac{1}{2}\theta_{i-1} + \frac{1}{2}\theta_{i+1}}{\Delta s} \right)^2 \right)
\end{aligned}
$$

As mentioned previously, two points near the boundary are modified since the stencils otherwise would exceed the grid. The first point for the T-equation

reads (green terms are modified)

$$
\begin{aligned}
F_7 \overset{i=0}{=}\ & \frac{T_{i-1}-2T_i+T_{i+1}}{\Delta s^2}-\left(\frac{-\frac{1}{2}\theta_{i-1}+\frac{1}{2}\theta_{i+1}}{\Delta s}\right)^2 T_i \\
& +p_i\frac{-\frac{1}{2}\theta_{i-1}+\frac{1}{2}\theta_{i+1}}{\Delta s}+\eta_i\left(\frac{\theta_{i-1}-2\theta_i+\theta_{i+1}}{\Delta s^2}\right)^2 \\
& +2\eta_i\frac{-\frac{1}{2}\theta_{i-1}+\frac{1}{2}\theta_{i+1}}{\Delta s}\frac{-\frac{5}{2}\theta_i+9\theta_{i+1}-12\theta_{i+2}+7\theta_{i+3}-\frac{3}{2}\theta_{i+4}}{\Delta s^3} \\
& +\mu_i\left(\dot{\alpha}+\frac{C_1}{\Delta t}\left(\theta_i-C_3\theta_i^n-C_4\theta_i^{n-1}\right)-C_2\dot{\theta}_i^n\right)^2 \\
& +\tau_{s,i}+2\eta_{ss,i}\left(\frac{-\frac{1}{2}\theta_{i-1}+\frac{1}{2}\theta_{i+1}}{\Delta s}\right)^2-\frac{\mu_{s,i}}{\mu_i}\frac{-\frac{1}{2}T_{i-1}+\frac{1}{2}T_{i+1}}{\Delta s} \\
& +5\eta_{s,i}\frac{-\frac{1}{2}\theta_{i-1}+\frac{1}{2}\theta_{i+1}}{\Delta s}\frac{\theta_{i-1}-2\theta_i+\theta_{i+1}}{\Delta s^2} \\
& -\frac{\mu_{s,i}}{\mu_i}\left(\eta_i\frac{-\frac{1}{2}\theta_{i-1}+\frac{1}{2}\theta_{i+1}}{\Delta s}\frac{\theta_{i-1}-2\theta_i+\theta_{i+1}}{\Delta s^2}\right. \\
& \left.+\tau_i+\eta_{s,i}\left(\frac{-\frac{1}{2}\theta_{i-1}+\frac{1}{2}\theta_{i+1}}{\Delta s}\right)^2\right)
\end{aligned}
$$

and eventually, the last point for the evolution equation (green terms are

modified)

$$
\begin{aligned}
F_8 \overset{i=ns-1}{=} \; & K_{4,i} - \frac{C_1}{\Delta t}\theta_i + \frac{\Delta t}{\mu_i C_1}\left[-\eta_i \frac{\theta_{i-2} - 4\theta_{i-1} + 6\theta_i - 4\theta_{i+1} + \theta_{i+2}}{\Delta s^4} \right. \\
& + \tau_i \frac{-\frac{1}{2}\theta_{i-1} + \frac{1}{2}\theta_{i+1}}{\Delta s} + 2\frac{-\frac{1}{2}T_{i-2} + 2T_{i-1} + \frac{3}{2}T_i}{\Delta s}\frac{-\frac{1}{2}\theta_{i-1} + \frac{1}{2}\theta_{i+1}}{\Delta s} \\
& + T_i \frac{\theta_{i-1} - 2\theta_i + \theta_{i+1}}{\Delta s^2} + \eta_i \left(\frac{-\frac{1}{2}\theta_{i-1} + \frac{1}{2}\theta_{i+1}}{\Delta s} \right)^2 \frac{\theta_{i-1} - 2\theta_i + \theta_{i+1}}{\Delta s^2} \\
& - 3\eta_{s,i} \frac{-\frac{1}{2}\theta_{i-2} + \theta_{i-1} - \theta_{i+1} + \frac{1}{2}\theta_{i+2}}{\Delta s^3} - \eta_{sss,i} \frac{-\frac{1}{2}\theta_{i-1} + \frac{1}{2}\theta_{i+1}}{\Delta s} \\
& - 3\eta_{ss,i} \frac{\theta_{i-1} - 2\theta_i + \theta_{i+1}}{\Delta s^2} + \eta_{s,i} \left(\frac{-\frac{1}{2}\theta_{i-1} + \frac{1}{2}\theta_{i+1}}{\Delta s} \right)^3 \\
& + \frac{\mu_{s,i}}{\mu_i} \left(p_i + 2\eta_{s,i} \frac{\theta_{i-1} - 2\theta_i + \theta_{i+1}}{\Delta s^2} + \right. \\
& + \eta_{ss,i} \frac{-\frac{1}{2}\theta_{i-1} + \frac{1}{2}\theta_{i+1}}{\Delta s} + \eta_i \frac{-\frac{1}{2}\theta_{i-2} + \theta_{i-1} - \theta_{i+1} + \frac{1}{2}\theta_{i+2}}{\Delta s^3} \\
& \left. \left. - T_i \frac{-\frac{1}{2}\theta_{i-1} + \frac{1}{2}\theta_{i+1}}{\Delta s} \right) - \ddot{\alpha}\mu_i^2 - p_{s,i} \right]
\end{aligned}
$$

The non-linear root problem $\underline{F}(\underline{x}) = 0$ is now defined, and we employ Newton-Raphson iterations to solve it, which yields the following solution procedure

1. Given $\underline{x}^n$, guess $\underline{x}_0^{n+1}$ using an explicit time marching scheme
2. Calculate $\underline{F}\left(\underline{x}_i^{n+1}\right)$
3. Calculate the Jacobian $\underline{\underline{J}} = J_{kl} = \frac{\partial F_k\left(\underline{x}_i^{n+1}\right)}{\partial x_l}$
4. Solve the linear system $\underline{\underline{J}} \cdot \delta\underline{x}_i = -\underline{F}\left(\underline{x}_i^{n+1}\right)$ to get the increment $\delta\underline{x}_i^{n+1}$
5. Set the new vector $\underline{x}_{i+1}^{n+1} = \underline{x}_i^{n+1} + \delta\underline{x}_i^{n+1}$
6. If $\left|\delta\underline{x}_i^{n+1}\right|$ is sufficiently small (we used $\left|\delta\underline{x}_i^{n+1}\right| < \varepsilon = 10^{-10}$) then abort, else go back to 1

It is possible to construct the Jacobian using finite differences for $\underline{x}$, but this requires $(2ns+4)^2$ evaluations of the function $\underline{F}$ and is thus not suitable. We

thus choose to analytically derive the Jacobian, which is then hard-coded in the numerical code. It is defined as

$$J_{ij} = \frac{\partial F_i}{\partial x_j} = \begin{pmatrix} \frac{\partial F_1}{\partial \theta_{-1}} & \cdots & \frac{\partial F_1}{\partial \theta_{ns+1}} & \frac{\partial F_1}{\partial T_{-1}} & \cdots & \frac{\partial F_1}{\partial T_{ns-1}} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial F_{2ns+4}}{\partial \theta_{-1}} & \cdots & \frac{\partial F_{2ns+4}}{\partial \theta_{ns+1}} & \frac{\partial F_{2ns+4}}{\partial T_{-1}} & \cdots & \frac{\partial F_{2ns+4}}{\partial T_{ns-1}} \end{pmatrix}$$

For the first equation the derivatives yield

$$\begin{aligned} \frac{\partial F_1}{\partial \theta_{-1}} &= 0 \\ \frac{\partial F_1}{\partial \theta_0} &= 1 \end{aligned}$$

For the second equation the derivatives yield

$$\begin{aligned} \frac{\partial F_2}{\partial \theta_{-1}} &= \eta_0 \frac{\theta_0 - \theta_{-1}}{\Delta s^3} - \eta_{s,0} \frac{\theta_1 - \theta_{-1}}{2\Delta s^2} \\ \frac{\partial F_2}{\partial \theta_0} &= -\eta_0 \frac{\theta_1 - \theta_{-1}}{\Delta s^3} \\ \frac{\partial F_2}{\partial \theta_1} &= -\eta_0 \frac{\theta_0 - \theta_{-1}}{\Delta s^3} + \eta_{s,0} \frac{\theta_1 - \theta_{-1}}{2\Delta s^2} \\ \frac{\partial F_2}{\partial \theta_2} &= 0 \\ \frac{\partial F_2}{\partial \theta_3} &= 0 \\ \frac{\partial F_2}{\partial T_{-1}} &= -\frac{1}{2\Delta s} \\ \frac{\partial F_2}{\partial T_0} &= 0 \\ \frac{\partial F_2}{\partial T_1} &= \frac{1}{2\Delta s} \end{aligned}$$

For the third equation the derivatives yield

$$\begin{aligned}
\frac{\partial F_3}{\partial \theta_{-1}} &= -\frac{2\Delta s\eta_{s,0} - \frac{3}{2}\eta_0 + \frac{\Delta s^2}{2}\left(T_0 - \eta_{ss,0}\right)}{\Delta s^3} \\
\frac{\partial F_3}{\partial \theta_0} &= -\frac{5\eta_0 - 4\Delta s\eta_{s,0}}{\Delta s^3} \\
\frac{\partial F_3}{\partial \theta_1} &= \frac{6\eta_0 - 2\Delta s\eta_{s,0} + \frac{\Delta s^2}{2}\left(T_0 - \eta_{ss,0}\right)}{\Delta s^3} \\
\frac{\partial F_3}{\partial \theta_2} &= -3\frac{\eta_0}{\Delta s^3} \\
\frac{\partial F_3}{\partial \theta_3} &= \frac{\eta_0}{2\Delta s^3} \\
\frac{\partial F_3}{\partial T_0} &= \frac{\theta_1 - \theta_{-1}}{2\Delta s}
\end{aligned}$$

For the fourth equation the derivatives yield

$$\frac{\partial F_4}{\partial T_{ns-1}} = 1$$

For the fifth equation the derivatives yield

$$\begin{aligned}
\frac{\partial F_5}{\partial \theta_{ns-3}} &= -\frac{\eta_{ns-1}}{12\Delta s^2} \\
\frac{\partial F_5}{\partial \theta_{ns-2}} &= \frac{8\eta_{ns-1} - 3\Delta s\eta_{s,ns-1}}{6\Delta s^2} \\
\frac{\partial F_5}{\partial \theta_{ns-1}} &= -\frac{5\eta_{ns-1}}{2\Delta s^2} \\
\frac{\partial F_5}{\partial \theta_{ns}} &= \frac{8\eta_{ns-1} + 3\Delta s\eta_{s,ns-1}}{6\Delta s^2} \\
\frac{\partial F_5}{\partial \theta_{ns+1}} &= -\frac{\eta_{ns-1}}{12\Delta s^2}
\end{aligned}$$

For the sixth equation the derivatives yield

$$\begin{aligned}
\frac{\partial F_6}{\partial \theta_{ns-3}} &= \frac{\eta_{ns-1}}{12\Delta s} \\
\frac{\partial F_6}{\partial \theta_{ns-2}} &= -\frac{2\eta_{ns-1}}{3\Delta s} \\
\frac{\partial F_6}{\partial \theta_{ns}} &= \frac{2\eta_{ns-1}}{3\Delta s} \\
\frac{\partial F_6}{\partial \theta_{ns+1}} &= -\frac{\eta_{ns-1}}{12\Delta s}
\end{aligned}$$

After the first six equations, which represent the boundary conditions, we treat the equations 7 and 8, which are like the regular interior points, but with some terms modified in order to take the ghost nodes into account, while not exceeding the grid. For equation seven we find

$$
\begin{aligned}
\frac{\partial F_7}{\partial \theta_{i-1}} &= \frac{\mu_{s,i}}{\mu_i}\left(\eta_i \frac{\frac{1}{2}\theta_{i-1} - \frac{1}{2}\theta_{i+1}}{\Delta s^3} - \eta_{s,i}\frac{\frac{1}{2}\theta_{i-1} - \frac{1}{2}\theta_{i+1}}{\Delta s^2} + \eta_i \frac{\theta_{i-1} - 2\theta_i + \theta_{i+1}}{2\Delta s^3}\right) \\
&\quad - \frac{p_i}{2\Delta s} + \eta_i \frac{\frac{5}{2}\theta_i - 9\theta_{i+1} + 12\theta_{i+2} - 7\theta_{i+3} + \frac{3}{2}\theta_{i+4}}{\Delta s^4} \\
&\quad + \eta_i \frac{2\theta_{i-1} - 4\theta_i + 2\theta_{i+1}}{\Delta s^4} - T_i \frac{\frac{1}{2}\theta_{i-1} - \frac{1}{2}\theta_{i+1}}{\Delta s^2} - 5\eta_{s,i}\frac{\frac{1}{2}\theta_{i-1} - \frac{1}{2}\theta_{i+1}}{\Delta s^3} \\
&\quad + 2\eta_{ss,i}\frac{\frac{1}{2}\theta_{i-1} - \frac{1}{2}\theta_{i+1}}{\Delta s^2} - 5\eta_{s,i}\frac{\theta_{i-1} - 2\theta_i + \theta_{i+1}}{2\Delta s^3} \\
\frac{\partial F_7}{\partial \theta_i} &= \left(5\eta_{s,i} - \eta_i \frac{\mu_{s,i}}{\mu_i}\right)\frac{\theta_{i-1} - \theta_{i+1}}{\Delta s^3} - \eta_i \frac{3\theta_{i-1} - 16\theta_i + 13\theta_{i+1}}{2\Delta s^4} \\
&\quad - 2C_1\mu_i \frac{C_1C_3\theta_i^n - \dot{\alpha}\Delta t - C_1\theta_i + C_1C_4\theta_i^{n-1} + C_2\Delta t\dot{\theta}_i^n}{\Delta t^2} \\
\frac{\partial F_7}{\partial \theta_{i+1}} &= \frac{p_i}{2\Delta s} - \left(5\eta_{s,i} - \eta_i\frac{\mu_{s,i}}{\mu_i}\right)\frac{\theta_i - \theta_{i+1}}{\Delta s^3} + \left(T_i - 2\eta_{ss,i} + \eta_{s,i}\frac{\mu_{s,i}}{\mu_i}\right)\frac{\theta_{i-1} - \theta_{i+1}}{2\Delta s^2} \\
&\quad - \eta_i \frac{13\theta_i + 14\theta_{i-1} - 40\theta_{i+1} + 24\theta_{i+2} - 14\theta_{i+3} + 3\theta_{i+4}}{2\Delta s^4} \\
\frac{\partial F_7}{\partial \theta_{i+2}} &= 12\eta_i \frac{\theta_{i-1} - \theta_{i+1}}{\Delta s^4} \\
\frac{\partial F_7}{\partial \theta_{i+3}} &= -7\eta_i \frac{\theta_{i-1} - \theta_{i+1}}{\Delta s^4} \\
\frac{\partial F_7}{\partial \theta_{i+4}} &= 3\eta_i \frac{\theta_{i-1} - \theta_{i+1}}{2\Delta s^4} \\
\frac{\partial F_7}{\partial T_{i-1}} &= \frac{\Delta s\frac{\mu_{s,i}}{\mu_i} + 2}{2\Delta s^2} \\
\frac{\partial F_7}{\partial T_i} &= -\frac{(\theta_{i-1} - \theta_{i+1})^2}{4\Delta s^2} - \frac{2}{\Delta s^2} \\
\frac{\partial F_7}{\partial T_{i+1}} &= -\frac{\Delta s\frac{\mu_{s,i}}{\mu_i} - 2}{2\Delta s^2}
\end{aligned}
$$

And for equation eight

$$
\begin{aligned}
\frac{\partial F_8}{\partial \theta_{i-2}} &= -\frac{\Delta t \eta_i - \frac{1}{2}\Delta s \Delta t \left(3\eta_{s,i} - \eta_i \frac{\mu_{s,i}}{\mu_i}\right)}{C_1 \mu_i \Delta s^4} \\
\frac{\partial F_8}{\partial \theta_{i-1}} &= \Delta t \frac{\eta_{sss,i} - \tau_i + T_i \frac{\mu_{s,i}}{\mu_i} - \eta_{ss,i} \frac{\mu_{s,i}}{\mu_i}}{2C_1 \mu_i \Delta s} \\
&\quad -\Delta t \frac{24\eta_{s,i} - 8\eta_i \frac{\mu_{s,i}}{\mu_i} + 3\eta_{s,i}\theta_{i-1}^2 + 3\eta_{s,i}\theta_{i+1}^2 - 6\eta_{s,i}\theta_{i-1}\theta_{i+1}}{8C_1 \Delta s^3 \mu_i} \\
&\quad -\Delta t \frac{T_i + 4T_{i-1} - T_{i-2} + 6\eta_{ss,i} - 4\eta_{s,i}\frac{\mu_{s,i}}{\mu_i}}{2C_1 \Delta s^2 \mu_i} \\
&\quad -\Delta t \eta_i \frac{4\theta_i\theta_{i-1} - 4\theta_i\theta_{i+1} + 2\theta_{i-1}\theta_{i+1} - 3\theta_{i-1}^2 + \theta_{i+1}^2 - 16}{4C_1 \Delta s^4 \mu_i} \\
\frac{\partial F_8}{\partial \theta_i} &= -\frac{C_1}{\Delta t}\Delta t \frac{6\eta_i + \frac{1}{2}\eta_i\theta_{i-1}^2 + \frac{1}{2}\eta_i\theta_{i+1}^2 - \eta_i\theta_{i-1}\theta_{i+1}}{C_1 \Delta s^4 \mu_i} \\
&\quad +\Delta t \Delta s^2 \frac{2T_i - 6\eta_{ss,i} + 4\eta_{s,i}\frac{\mu_{s,i}}{\mu_i}}{C_1 \Delta s^4 \mu_i} \\
\frac{\partial F_8}{\partial \theta_{i+1}} &= \Delta t \frac{32\eta_i - 2\eta_i\theta_{i-1}^2 + 6\eta_i\theta_{i+1}^2 + 8\eta_i\theta_i\theta_{i-1} - 8\eta_i\theta_i\theta_{i+1} - 4\eta_i\theta_{i-1}\theta_{i+1}}{8C_1 \Delta s^4 \mu_i} \\
&\quad +\Delta t \frac{24\eta_{s,i} - 8\eta_i \frac{\mu_{s,i}}{\mu_i} + 3\eta_{s,i}\theta_{i-1}^2 + 3\eta_{s,i}\theta_{i+1}^2 - 6\eta_{s,i}\theta_{i-1}\theta_{i+1}}{8C_1 \Delta s^3 \mu_i} \\
&\quad -\Delta t \frac{4\eta_{sss,i} - 4\tau_i + 4T_i \frac{\mu_{s,i}}{\mu_i} - 4\eta_{ss,i}\frac{\mu_{s,i}}{\mu_i}}{8C_1 \Delta s \mu_i} \\
&\quad +\Delta t \frac{20T_i + 16T_{i-1} - 4T_{i-2} - 24\eta_{ss,i} + 16\eta_{s,i}\frac{\mu_{s,i}}{\mu_i}}{8C_1 \Delta s^2 \mu_i} \\
\frac{\partial F_8}{\partial \theta_{i+2}} &= -\frac{\Delta t \eta_i + \frac{1}{2}\Delta s \Delta t \left(3\eta_{s,i} - \eta_i \frac{\mu_{s,i}}{\mu_i}\right)}{C_1 \Delta s^4 \mu_i} \\
\frac{\partial F_8}{\partial T_{i-2}} &= \Delta t \frac{\theta_{i-1} - \theta_{i+1}}{2C_1 \Delta s^2 \mu_i} \\
\frac{\partial F_8}{\partial T_{i-1}} &= -2\Delta t \frac{\theta_{i-1} - \theta_{i+1}}{C_1 \Delta s^2 \mu_i} \\
\frac{\partial F_8}{\partial T_i} &= -\frac{\frac{1}{2}\Delta t \left(4\theta_i + \theta_{i-1} - 5\theta_{i+1}\right) - \frac{1}{2}\Delta s \Delta t \frac{\mu_{s,i}}{\mu_i}\left(\theta_{i-1} - \theta_{i+1}\right)}{C \Delta s^2 \mu_i}
\end{aligned}
$$

We now treat the interior points for the evolution equation

$$
\begin{aligned}
\frac{\partial F_{8+i}}{\partial \theta_{i-2}} &= -\frac{\Delta t \eta_i - \frac{1}{2}\Delta s \Delta t \left(3\eta_{s,i} - \eta_i \frac{\mu_{s,i}}{\mu_i}\right)}{C_1 \Delta s^4 \mu_i} \\
\frac{\partial F_{8+i}}{\partial \theta_{i-1}} &= \Delta t \frac{2T_i + T_{i-1} - T_{i+1} - 6\eta_{ss,i} + 4\eta_{s,i}\frac{\mu_{s,i}}{\mu_i}}{2C_1 \Delta s^2 \mu_i} \\
&\quad -\Delta t \frac{24\eta_{s,i} - 8\eta_i \frac{\mu_{s,i}}{\mu_i} + 3\eta_{s,i}\theta_{i-1}^2 + 3\eta_{s,i}\theta_{i+1}^2 - 6\eta_{s,i}\theta_{i-1}\theta_{i+1}}{8C_1 \Delta s^3 \mu_i} \\
&\quad +\Delta t \frac{\eta_{sss,i} - \tau_i + T_i \frac{\mu_{s,i}}{\mu_i} - \eta_{ss,i}\frac{\mu_{s,i}}{\mu_i}}{2C_1 \Delta s \mu_i} \\
&\quad -\Delta t \eta_i \frac{4\theta_i\theta_{i-1} - 4\theta_i\theta_{i+1} + 2\theta_{i-1}\theta_{i+1} - 3\theta_{i-1}^2 + \theta_{i+1}^2 - 16}{4C_1 \Delta s^4 \mu_i} \\
\frac{\partial F_{8+i}}{\partial \theta_i} &= -\frac{C_1}{\Delta t} - \Delta t \frac{6\eta_i + \frac{1}{2}\eta_i\theta_{i-1}^2 + \frac{1}{2}\eta_i\theta_{i+1}^2 - \eta_i\theta_{i-1}\theta_{i+1}}{C_1 \Delta s^4 \mu_i} \\
&\quad -\Delta t \frac{2T_i - 6\eta_{ss,i} + 4\eta_{s,i}\frac{\mu_{s,i}}{\mu_i}}{C_1 \Delta s^2 \mu_i} \\
\frac{\partial F_{8+i}}{\partial \theta_{i+1}} &= \Delta t \frac{2T_i - T_{i-1} + T_{i+1} - 6\eta_{ss,i} + 4\eta_{s,i}\frac{\mu_{s,i}}{\mu_i}}{2C_1 \Delta s^2 \mu_i} \\
&\quad +\Delta t \frac{24\eta_{s,i} - 8\eta_i \frac{\mu_{s,i}}{\mu_i} + 3\eta_{s,i}\theta_{i-1}^2 + 3\eta_{s,i}\theta_{i+1}^2 - 6\eta_{s,i}\theta_{i-1}\theta_{i+1}}{8C_1 \Delta s^3 \mu_i} \\
&\quad -\Delta t \frac{\eta_{sss,i} - \tau_i + T_i \frac{\mu_{s,i}}{\mu_i} - \eta_{ss,i}\frac{\mu_{s,i}}{\mu_i}}{2C_1 \Delta s \mu_i} \\
&\quad +\Delta t \eta_i \frac{4\theta_i\theta_{i-1} - 4\theta_i\theta_{i+1} - 2\theta_{i-1}\theta_{i+1} - \theta_{i-1}^2 + 3\theta_{i+1}^2 + 16}{4C_1 \Delta s^4 \mu_i} \\
\frac{\partial F_{8+i}}{\partial \theta_{i+2}} &= -\Delta t \frac{\eta_i + \frac{1}{2}\Delta s \left(3\eta_{s,i} - \eta_i \frac{\mu_{s,i}}{\mu_i}\right)}{C_1 \Delta s^4 \mu_i} \\
\frac{\partial F_{8+i}}{\partial T_{i-1}} &= \Delta t \frac{\theta_{i-1} - \theta_{i+1}}{2C_1 \Delta s^2 \mu_i} \\
\frac{\partial F_{8+i}}{\partial T_i} &= \frac{\frac{\Delta t}{\Delta s^2}\left(\theta_{i-1} - 2\theta_i + \theta_{i+1}\right) + \frac{\mu_{s,i}}{\mu_i}\frac{\frac{1}{2}\theta_{i-1} - \frac{1}{2}\theta_{i+1}}{\Delta s}}{C_1 \mu_i} \\
\frac{\partial F_{8+i}}{\partial T_{i+1}} &= -\Delta t \frac{\theta_{i-1} - \theta_{i+1}}{2C_1 \Delta s^2 \mu_i}
\end{aligned}
$$

And eventually the coeffiecents for the Tension-equation are

$$
\begin{aligned}
\frac{\partial F_{7+ns+i}}{\partial \theta_{i-2}} &= \eta_i \frac{\theta_{i-1} - \theta_{i+1}}{2\Delta s^4} \\
\frac{\partial F_{7+ns+i}}{\partial \theta_{i-1}} &= \left(5\eta_{s,i} - \eta_i \frac{\mu_{s,i}}{\mu_i}\right) \frac{\theta_i - \theta_{i-1}}{\Delta s^3} - \frac{p_i}{2\Delta s} \\
&\quad - \left(T_i - 2\eta_{ss,i} + \eta_{s,i} \frac{\mu_{s,i}}{\mu_i}\right) \frac{\theta_{i-1} - \theta_{i+1}}{2\Delta s^2} \\
&\quad - \eta_i \frac{8\theta_i - \theta_{i-2} - 8\theta_{i+1} + \theta_{i+2}}{2\Delta s^4} \\
\frac{\partial F_{7+ns+i}}{\partial \theta_i} &= 10\eta_{s,i} \frac{\frac{1}{2}\theta_{i-1} - \frac{1}{2}\theta_{i+1}}{\Delta s^3} \\
&\quad - \eta_i \frac{4\theta_{i-1} - 8\theta_i + 4\theta_{i+1}}{\Delta s^4} \\
&\quad - 2C_1\mu_i \frac{C_2\dot{\theta}_i^n - \dot{\alpha} + \frac{C_1}{\Delta t}\left(C_3\theta_i^n - \theta_i + C_4\theta_i^{n-1}\right)}{\Delta t} \\
&\quad - 2\eta_i \frac{\mu_{s,i}}{\mu_i} \frac{\frac{1}{2}\theta_{i-1} - \frac{1}{2}\theta_{i+1}}{\Delta s^3} \\
\frac{\partial F_{7+ns+i}}{\partial \theta_{i+1}} &= \frac{p_i}{2\Delta s} - \left(5\eta_{s,i} - \eta_i \frac{\mu_{s,i}}{\mu_i}\right) \frac{\theta_i - \theta_{i+1}}{\Delta s^3} \\
&\quad + \left(T_i - 2\eta_{ss,i} + \eta_{s,i} \frac{\mu_{s,i}}{\mu_i}\right) \frac{\theta_{i-1} - \theta_{i+1}}{2\Delta s^2} \\
&\quad - \eta_i \frac{8\theta_i - 8\theta_{i-1} + \theta_{i-2} - \theta_{i+2}}{2\Delta s^4} \\
\frac{\partial F_{7+ns+i}}{\partial \theta_{i+2}} &= -\eta_i \frac{\theta_{i-1} - \theta_{i+1}}{2\Delta s^4} \\
\frac{\partial F_{7+ns+i}}{\partial T_{i-1}} &= \frac{\Delta s \frac{\mu_{s,i}}{\mu_i} + 2}{2\Delta s^2} \\
\frac{\partial F_{7+ns+i}}{\partial T_i} &= -\frac{(\theta_{i-1} - \theta_{i+1})^2}{4\Delta s^2} - \frac{2}{\Delta s^2} \\
\frac{\partial F_{7+ns+i}}{\partial T_{i+1}} &= -\frac{\Delta s \frac{\mu_{s,i}}{\mu_i} - 2}{2\Delta s^2}
\end{aligned}
$$

Appendix C

Curriculum Vitae

Personal Information

Name	Thomas Sebastian Engels
Address	Thaerstraße 41, 10249 Berlin, Germany
Telephone	+49 17683252194
Email	thomas.engels@mailbox.tu-berlin.de
Date and place of birth	01.02.1986 in Düren, Nordrhein-Westfalen, Germany
Languages	German (Native), English (fluent), French (fluent)

Education and degrees

10/2005 – 4/2008

RWTH Aachen. Student in mechanical engineering, Degree: Intermediate Diploma with overall mark "good".

since 4/2008

Technical University Berlin. Student (Diplom) in "Physical Engineering" with specialties "Fluid mechanics" and "Thermodynamics". The finalization of the degree "Dipl. Ing." is pending.

9/2009 – 3/2010

Université de la Méditerranée. Student (Master) in "Mechanics, Physics and Engineeing", specialty "Fluid mechanics and nonlinear Physics", Degree: Master of Science with overall mark "good" (14.7/20), ranked second in class.

06/2010 – 12/2010

CNRS (Centre National de la Recherche Scientifique) Research internship and master's thesis at the laboratory M2P2 ("Mécanique, Modélisation et Procédés Propres"). Title of the work: "Numerical Modelling of Flexible Foils In Incompressible Viscous Fluids". Supervisors: K. Schneider and D. Kolomenskiy. Thesis defended on 06/30/2011 with mark "very good" (17/20)

10/2011 – 10/2014

Université de Provence (now Aix-Marseille Université). PhD student financed by the French Ministry of Education, French-German PhD program jointly supervised by K. Schneider (Aix-Marseille Université) and J. Sesterhenn (Technical University Berlin), half-time in both countries. Title of the dissertation: "Numerical Modeling of Fluid-Structure Interaction in Bio-Inspired Propulsion".

10/2014 – 12/2015

Technische Universität Berlin. PhD Student financed by Technical University Berlin, topic unaltered. Thesis defence scheduled for 12/10/2015

Publications

Peer-reviewed articles

[44] **T. Engels**, D. Kolomenskiy, K. Schneider and J. Sesterhenn, "Two-dimensional simulation of the fluttering instability using a pseudospectral method with volume penalization", *Computers & Structures* 122 (2012), pp. 101-112.

[46] **T. Engels**, D. Kolomenskiy, K. Schneider and J. Sesterhenn, "Numerical Simulation of Fluid-Structure Interaction with the Volume Penalization Method", *J. Comput. Phys.* 281 (2015), pp. 96-115.

[47] **T. Engels**, D. Kolomenskiy, K. Schneider and J. Sesterhenn, "A numerical study of vortex-induced drag of Elastic swimmer models", *Proceedings 6th International Symposium on Aero-aqua Bio-Mechanisms, November 13.-16., Honolulu, Hawaii, USA* (2014).

[43] **T. Engels**, D. Kolomenskiy, K. Schneider, J. Sesterhenn and F.-O. Lehmann, "Bumblebee flight in heavy turbulence", *Phys. Rev. Letters* (under review, 9/2015).

[45] **T. Engels**, D. Kolomenskiy, K. Schneider and J. Sesterhenn, "FluSI: A novel parallel simulation tool for flapping insect flight using a Fourier method with volume penalization", arXiv: 1506.06513, *SIAM J. Sci. Computing* (under rewiev 6/2015).

[83] D. Kolomenskiy, **T. Engels** and K. Schneider, "Numerical Modelling of Flexible Heaving Foils", *J. Aero Aqua Bio-mechanisms* 3 (2013), pp. 22-28.

[84] D. Kolomenskiy, M. Maeda, **T. Engels**, H. Liu, K. Schneider and J.-C. Nave, "Aerodynamic ground effect in fruitfly sized insect takeoff", arXiv: 1504.04484, *Bioinspir. Biomim.* (under review 07/2015).

Conferences

Engels, T., Kolomenskiy, D., Schneider, K. and Sesterhenn, J. - Numerical modeling of flexible insect wings using volume penalization. Fluids & Elasticity, La Jolla, California, USA, 14.-16.11.2012.

Engels, T., Kolomenskiy, D., Schneider, K. and Sesterhenn, J. - Numerical modeling of flexible insect wings using volume penalization. 65th Annual Conference, Division of Fluid Dynamics, American Physical Society, San Diego, 18.-20.11.2012 (USA). Bull. Amer. Phys. Soc., (17), 52, 2012.

Sheng, J.X., **Engels, T.**, Kolomenskiy D. and Schneider, K. - Fluid structure interaction with low and high order flexibility using volume penalization. 65th Annual Conference, Division of Fluid Dynamics, American Physical Society, San Diego, 18.-20.11.2012 (USA). Bull. Amer. Phys. Soc., (17), 52, 2012.

Engels, T., Kolomenskiy, D., Schneider, K. and Sesterhenn, J. - A numerical study of vortex-induced drag of elastic swimmer models. ISABMEC14, Honululu, Hawaii, USA, 11/2014 .

Related professional experience

05/2007 – 02/2008

Student teaching assistant at RWTH Aachen, Institut für Allgemeine Mechanik. Courses taught: Mechanics for Engineers I

10/2008 - 07/2010

Student teaching assistant at Technical University Berlin, Institut für Mechanik. Courses taught: Mechanics for Engineers I-III

02/2011 - 10/2011

Student research assistant at Technical University Berlin, Institut für Strömungsmechanik und technische Akustik, chair of experimental fluid mechanics. Assistant to Sebastian Göke, working on experimental studies on wet combustion.

01/10/2012 – 05/10/2012

Teaching assistant for the workshop „Fourier, Wavelet and Wavelet Packet Transforms: theory and applications to signal and image processing“ in Gif-sur-Yvette (Paris), assistant to Prof. Marie Farge and Prof. Kai Schneider.

Appendix D

List of publications

[44] **T. Engels**, D. Kolomenskiy, K. Schneider and J. Sesterhenn, "Two-dimensional simulation of the fluttering instability using a pseudospectral method with volume penalization", *Computers & Structures* 122 (2012), pp. 101-112.

[46] **T. Engels**, D. Kolomenskiy, K. Schneider and J. Sesterhenn, "Numerical Simulation of Fluid-Structure Interaction with the Volume Penalization Method", *J. Comput. Phys.* 281 (2015), pp. 96-115.

[47] **T. Engels**, D. Kolomenskiy, K. Schneider and J. Sesterhenn, "A numerical study of vortex-induced drag of Elastic swimmer models", *Proceedings 6th International Symposium on Aero-aqua Bio-Mechanisms, November 13.-16., Honolulu, Hawaii, USA* (2014).

[43] **T. Engels**, D. Kolomenskiy, K. Schneider, J. Sesterhenn and F.-O. Lehmann, "Bumblebee flight in heavy turbulence", *Phys. Rev. Lett.* 116 (2016), 028103.

[45] **T. Engels**, D. Kolomenskiy, K. Schneider and J. Sesterhenn, "FluSI: A novel parallel simulation tool for flapping insect flight using a Fourier method with volume penalization", arXiv: 1506.06513, *SIAM J. Sci. Computing* (under rewiev 6/2015).

[83] D. Kolomenskiy, **T. Engels** and K. Schneider, "Numerical Modelling of Flexible Heaving Foils", *J. Aero Aqua Bio-mechanisms* 3 (2013), pp. 22-28.

[84] D. Kolomenskiy, M. Maeda, **T. Engels**, H. Liu, K. Schneider and J.-C. Nave, "Aerodynamic ground effect in fruitfly sized insect takeoff", arXiv: 1504.04484, *PLoS ONE*, in press

Bibliography

[1] University of Minnesota Insect Collection. http://insectcollection.umn.edu. Accessed: 2014-05-14.

[2] K. Akselvoll and P. Moin. Large-eddy simulation of turbulent confined coannular jets. *J. Fluid Mech.*, 315:387–411, 1996.

[3] S. Alben. Optimal flexibility of a flapping appendage in an inviscid fluid. *J. Fluid Mech.*, 614:355–380, 2008.

[4] P. Angot, C. Bruneau, and P. Fabrie. A penalization method to take into account obstacles in incompressible viscous flows. *Numer. Math.*, 81:497–520, 1999.

[5] E. Arquis and J.-P. Caltagirone. Sur les conditions hydrodynamiques au voisinage d'une interface milieu fluide milieu poreux: application à la convection naturelle. *C. R. Acad. Sci. Paris, Sér. II*, 299, 1984.

[6] S. Badia, F. Nobile, and C. Vergara. Fluid–structure partitioned procedures based on robin transmission conditions. *J. Comput. Phys.*, 227:7027–7051, 2008.

[7] G. K. Batchelor. *An introduction to fluid dynamics*. Cambridge University Press, 1967.

[8] K.-J. Bathe and H. Zhang. A mesh adaptivity procedure for CFD and fluid-structure interactions. *Computers & Structures*, 87:604–617, 2009.

[9] J. Becker. A second order backward difference method with variable steps for a parabolic problem. *BIT Numer. Math.*, 38:644–662, 1998.

[10] M. Belliard and C. Fournier. Penalized direct forcing and projection schemes for Navier–Stokes. *C. R. Acad. Sci. Paris, Ser. I*, 348:1133–1136, 2010.

[11] M. Bergmann and A. Iollo. Modeling and simulation of fish-like swimming. *J. Comput. Phys.*, 230:329–348, 2011.

[12] G. J. Berman and Z. J. Wang. Energy-minimizing kinematics in hovering insect flight. *J. Fluid Mech.*, 582:153–168, 2007.

[13] G. Bimbard, D. Kolomenskiy, O. Bouteleux, J. Casas, and R. Godoy-Diana. Force balance in the take-off of a pierid butterfly: relative importance and timing of leg impulsion and aerodynamic forces. *J. Exp. Biol.*, 216:3551–3563, 2013.

[14] O. Boiron, G. Chiavassa, and R. Donat. A high-resolution penalization method for large mach number flows in the presence of obstacles. *Computers & Fluids*, 38:703–714, 2009.

[15] H. J. Bungartz, M. Mehl, and M. Schäfer, editors. *Fluid-Structure Interaction II: Modelling, Simulation, Optimization*. Number 73 in Lecture Notes in Computational Science and Engineering. Springer Berlin / Heidelberg, 2010.

[16] Hans-Joachim Bungartz and Michael Schäfer. *Fluid-structure interaction: modelling, simulation, optimisation*, volume 1. Springer Science & Business Media, 2006.

[17] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T.A. Zang. *Spectral Methods in Fluid Dynamics*. Springer Verlag, 1986.

[18] G. Carbou and P. Fabrie. Boundary layer for a penalization method for viscous incompressible flow. *Adv. Diff. Equ.*, 8:1453–2480, 2003.

[19] P. Causin, J.F. Gerbeau, and F. Nobile. Added-mass effect in the design of partitioned algorithms for fluid-structure problems. *Computer Methods in Applied Mechanics and Engineering*, 194:4506–4527, 2005.

[20] Y. H. Chen, M. Skote, Y. Zhao, and Huang W. M. Dragonfly (sympetrum flaveolum) flight: Kinematic measurement and modelling. *J. Fluids Struct.*, 40:115–126, 2013.

[21] S. Childress, A. Hosoi, W. W. Schultz, and Z. J. Wang, editors. *Natural Locomotion in Fluids and on Surfaces: Swimming, Flying and Sliding*. Springer, 2012.

[22] S. A. Combes and T. L. Daniel. Flexural stiffness in insect wings I. scaling and the influence of wing venation. *J. Exp. Biol.*, 206:2979–2987, 2003.

[23] S. A. Combes and T. L. Daniel. Flexural stiffness in insect wings II. Spatial distribution and dynamic wing bending. *J. Exp. Biol.*, 206:2989–2997, 2003.

[24] S. A. Combes and R. Dudley. Turbulence-driven instabilities limit insect flight performance. *Proc. Natl. Acad. Sci. USA*, 106:9105–9108, 2009.

[25] B. S. H. Connell and D. K. P. Yue. Flapping dynamics of a flag in a uniform stream. *J. Fluid Mech.*, 581:33–67, 2007.

[26] J. W. Cooley and J. W. Tukey. An algorithm for the machine calculation of complex Fourier series. *Math. Comput.*, 19:297–301, 1965.

[27] J. Craske and M. van Reeuwijk. Robust and accurate open boundary conditions for incompressible turbulent jets and plumes. *Computers & Fluids*, 86:284–297, 2013.

[28] P. A. Dewey, B. M. Boschitsch, K. W. Moored, H. A. Stone, and A. J. Smits. Scaling laws for the thrust production of flexible pitching panels. *J. Fluid Mech.*, 732:29–46, 2013.

[29] M. H. Dickinson, F.-O. Lehmann, and S. P. Sane. Wing rotation and the aerodynamic basis of insect flight. *Science*, 284:1954–1960, 1999.

[30] W. B. Dickson, P. Polidoro, M. M. Tanner, and M .H. Dickinson. A linear systems analysis of the yaw dynamics of a dynamically scaled insect model. *J. Exp. Biol.*, 213:3047–3081, 2010.

[31] E. H. Dowell and K. C. Hall. Modelling of fluid-structure-interaction. *Annu. Rev. Fluid Mech.*, 33:445–90, 2001.

[32] G. Du and M. Sun. Effects of wing deformation on aerodynamic forces in hovering hoverflies. *J. Exp. Biol.*, 213(13):2273–2283, 2010.

[33] R. Dudley and C. P. Ellington. Mechanics of forward flight in bumblebees I. kinematics and morphology. *J. Exp. Biol.*, 148:19–52, 1990.

[34] R. Dudley and C. P. Ellington. Mechanics of forward flight in bumblebees II. quasi-steady lift and power requirements. *J. Exp. Biol.*, 148:53–88, 1990.

[35] U. Ehrenstein, M. Marquillie, and C. Eloy. Skin friction on a flapping plate in uniform flow. *Phil. Trans. R. Soc. A*, 372, 2014.

[36] C. P. Ellington. The aerodynamics of hovering inset flight. III: Kinematics. *Phil. Trans. R. Soc. Lond. B*, 305:41–78, 1984.

[37] C. P. Ellington. The aerodynamics of hovering inset flight. VI: Lift and power requirements. *Phil. Trans. R. Soc. Lond. B*, 305:145–181, 1984.

[38] C. P. Ellington. Limitations on animal flight performance. *J. Exp. Biol.*, 160:71–91, 1991.

[39] C. P. Ellington, C. van den Berg, A. P. Willmott, and A. L. R. Thomas. Leading-edge vortices in insect flight. *Nature*, 384:626–630, 1996.

[40] C.P. Ellington. The novel aerodynamics of insect flight: applications to micro-air vehicles. *J. Exp. Biol.*, 202:3439–3448, 1999.

[41] C. Eloy. Optimal strouhal number for swimming animals. *J. Fluids Struct.*, 30:205–218, 2012.

[42] C. Eloy, C. Souilliez, and L. Schouveiler. Aeroelastic instability of a flexible plate in a uniform flow. *J. Fluid Mech.*, 611:97–106, 2008.

[43] T. Engels, D. Kolomenskiy, K. Schneider, F.-O. Lehmann, and J. Sesterhenn. Bumblebee flight in heavy turbulence. *Phys. Rev. Lett.*, 116:028103, Jan 2016.

[44] T. Engels, D. Kolomenskiy, K. Schneider, and J. Sesterhenn. Two-dimensional simulation of the fluttering instability using a pseudospectral method with volume penalization. *Computers & Structures*, 122:101–112, 2012.

[45] T. Engels, D. Kolomenskiy, K. Schneider, and J. Sesterhenn. FluSI: A novel parallel simulation tool for flapping insect flight using a Fourier method with volume penalization. *arXiv:1506.06513, SIAM J. Sci. Comput., under revision*, 2015.

[46] T. Engels, D. Kolomenskiy, K. Schneider, and J. Sesterhenn. Numerical simulation of fluid-structure interaction with the volume penalization method. *J. Comput. Phys.*, 281:96–115, 2015.

[47] T. Engels, D. Kolomenskiy, K. Schneider, and J.L. Sesterhenn. A numerical study of vortex-induced drag of elastic swimmer models. In *Proceedings 6th International Symposium on Aero-aqua Bio-Mechanisms, November 13.-16., Honolulu, Hawaii, USA.*, 2014.

[48] C. A. Van Eysden and J. E. Sader. Frequency response of cantilever beams immersed in viscous fluids with applications to the atomic force microscope: Arbitrary mode order. *J. Appl. Phys.*, 101:044908, 2007.

[49] M. Farge and K. Schneider. Wavelets: Application to turbulence. In J.-P. Françoise, G. L. Naber, and T. S. Tsun, editors, *Encyclopedia of Mathematical Physics*, pages 408–420. Academic Press, Oxford, 2006.

[50] C. Farhat, K. Vanderzee, and P. Geuzaine. Provably second-order time-accurate loosely-coupled solution algorithms for transient nonlinear computational aeroelasticity. *Comput Methods Appl Mech Eng*, 195:1973–2001, 2006.

[51] M. A. Fernández. Coupling schemes for incompressible fluid–structure interaction: implicit, semi-implicit and explicit. *SeMA Journal*, 55:59–108, 2011.

[52] M.J. Fernández, R. Dudley, and F. Bozinovic. Comparative energetics of the giant hummingbird (patagona gigas). *Phyiol. Biochem. Zool.*, 84(3):333–40, 2011.

[53] J. Ferziger and M. Peric. *Computational Methods for Fluid Dynamics*. Springer, 3rd edition, 2002.

[54] C. Foerster, W. Wall, and E. Ramm. Artificial added mass instabilities in sequential staggered coupling of nonlinear structures and incompressible viscous flows. *Comput. Methods Appl. Mech. Eng.*, 196:1278–1293, 2007.

[55] M. Frigo and S. G. Johnson. The design and implementation of FFTW3. *Proc. IEEE*, 94:216–231, 2005.

[56] S. N. Fry, R. Sayaman, and M .H. Dickinson. The aerodynamics of free-flight maneuvers in Drosophila. *Science (New York, N.Y.)*, 300(5618):495–8, April 2003.

[57] S. N. Fry, R. Sayaman, and M .H. Dickinson. The aerodynamics of hovering flight in Drosophila. *J. Exp. Biol.*, 208(Pt 12):2303–18, June 2005.

[58] J. R. Garratt and P. A. Taylor, editors. *Boundary-Layer Meteorology 25th Anniversary Volume, 1970–1995*. Springer Netherlands, Dordrecht, 1996.

[59] R. Gautier, D. Biau, and E. Lamballais. A reference solution of the flow over a circular cylinder at re = 40. *Computers & Fluids*, 75:103–111, 2013.

[60] M. Gazzola, B. Hejazialhosseini, and P. Koumoutsakos. reinforcement learning and wavelet adapted vortex methods for simulations of self-propelled swimmers. *SIAM J. Sci. Comput.*, 36:B622–B639, 2014.

[61] M. Gazzola, W. M. Van Rees, and P. Koumoutsakos. C-start: optimal start of larval fish. *J. Fluid Mech.*, pages 1–14, 2012.

[62] M. Gee. Truly monolithic algebraic multigrid for fluid–structure interaction. *Int. J. Numer. Methods Eng.*, 85:987–1016, 2011.

[63] Thomas Geijtenbeek, Michiel van de Panne, and A Frank van der Stappen. Flexible muscle-based locomotion for bipedal creatures. *ACM Transactions on Graphics (TOG)*, 32(6):206, 2013.

[64] J. F. Gerbeau and M. Vidrascu. A quasi-Newton algorithm based on a reduced model for fluid–structure interactions problems in blood flows. *Math. Model. Num. Anal.*, 37:631–648, 2003.

[65] S. A. Ghaffari, S. Viazzo, K. Schneider, and P. Bontoux. Simulation of forced deformable bodies interacting with two-dimensional incompressible flows: Application to fish-like swimming. *Int. J. Heat Fluid Flow*, 51:88–109, 2015.

[66] R. Godoy-Diana, P. Jain, M. Centeno, A. Weinreb, and B. Thiria. Four-winged flapping flyer in forward flight. In Jaime Klapp, Gerardo Ruíz Chavarría, Abraham Medina Ovando, Abel López Villa, and Leonardo Di G. Sigalotti, editors, *Selected Topics of Computational and Experimental Fluid Mechanics*, Environmental Science and Engineering, pages 147–158. Springer International Publishing, 2015.

[67] J. Pereira Gomes, S. Yigit, H. Lienhart, and M. Schäfer. Experimental and numerical study on a laminar fluid-structure interaction reference test case. *Journal of Fluids and Structures*, 27(1):43–61, 2011.

[68] J. L. Guermond, P. D. Minev, and J. Shen. An overview of projection methods for incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 195:6011–6045, 2006.

[69] C.R. Hardie and P. Raghu. Visual transduction in Drosophila. *Nature*, 413:186–193, 2001.

[70] M.M. Hejlesen, P. Koumoutsakos, A. Leonard, and J.H. Walther. Iterative Brinkman penalization for remeshed vortex methods. *J. Comput. Phys.*, 280:547–562, 2015.

[71] L. Huang. Flutter of cantilevered plates in axial flow. *J. Fluids Struct.*, 9:127–147, 1995.

[72] Barbara Burke Hubbard. *The World According to Wavelets: The Story of a Mathematical Technique in the Making*. A K Peters/CRC Press, 1998.

[73] B. Hübner, E. Walhorn, and D. Dinkler. A monolithic approach to fluid–structure interaction using space–time finite elements. *Comput. Methods Appl. Mech. Engrg.*, 193:2087–2104, 2004.

[74] Sarah Miriam Iams. *Characterizing Mosquito Flight Using Measurement and Simulation*. PhD thesis, Cornell University, 2014.

[75] C. Introini, M. Belliard, and C. Fournier. A second order penalized direct forcing for hybrid Cartesian/immersed boundary flow simulations. *Computers & Fluids*, 90:21–41, 2014.

[76] T. Ishihara, T. Gotoh, and Y. Kaneda. Study of high–Reynolds number isotropic turbulence by direct numerical simulation. *Annu. Rev. Fluid Mech.*, 41:165–180, 2009.

[77] Takashi Ishihara and Yukio Kaneda. High resolution dns of incompressible homogeneous forced turbulence—time dependence of the statistics—. In *Statistical Theories and Computational Approaches to Turbulence*, pages 177–188. Springer, 2003.

[78] C. Ji, A. Munjiza, and J.J.R. Williams. A novel iterative direct-forcing immersed boundary method and its finite volume applications. *J. Comput. Phys.*, 231:1797–1821, 2012.

[79] J. Jiménez, A. A. Wray, P. G. Saffman, and R. S. Rogallo. The structure of intense vorticity in isotropic turbulence. *J. Fluid Mech.*, 255:65–90, 1993.

[80] B. Kadoch, D. Kolomenskiy, P. Angot, and K. Schneider. A volume penalization method for incompressible flows and scalar advection-diffusion with moving obstacles. *J. Comput. Phys.*, 231:4365–4383, 2012.

[81] C.-K. Kang, H. Aono, C. E. S. Cesnik, and W. Shyy. Effects of flexibility on the aerodynamic performance of flapping wings. *J. Fluid Mech.*, 689:32–74, 2011.

[82] J. Kim and P. Moin. Application of a fractional-step method to incompressible Navier-Stokes equations. *J. Comput. Phys.*, 59(2):308–323, 1985.

[83] D. Kolomenskiy, T. Engels, and K. Schneider. Numerical modelling of flexible heaving foils. *J. Aero Aqua Bio-mechanisms*, 3:22–28, 2013.

[84] D. Kolomenskiy, M. Maeda, T. Engels, H. Liu, K. Schneider, and J.-C. Nave. Aerodynamic ground effect in fruitfly sized insect takeoff. *PLoS ONE, in press, arXiv:1504.04484*, 2016.

[85] D. Kolomenskiy, H. K. Moffatt, M. Farge, and K. Schneider. The Lighthill–Weis-Fogh clap-fling-sweep mechanism revisited. *J. Fluid Mech.*, 676:572–606, 2011.

[86] D. Kolomenskiy, H. K. Moffatt, M. Farge, and K. Schneider. Two- and three-dimensional numerical simulations of the clap-fling-sweep of hovering insects. *J. Fluids Struct.*, 27:784–791, 2011.

[87] D. Kolomenskiy and K. Schneider. A Fourier spectral method for the Navier-Stokes equations with volume penalization for moving solid obstacles. *J. Comput. Phys.*, 228:5687–5709, 2009.

[88] J. Kweon and H. Choi. Sectional lift coefficient of a flapping wing in hovering motion. *Phys. Fluids*, 22:071703, 2010.

[89] U. Küttler and W. Wall. Fixed-point fluid-structure interaction solvers with dynamic relaxation. *Computational Mechanics*, 43:61–72, 2008.

[90] L.D. Landau and E.M. Lifshitz. *Theory of Elasticity*, volume 7 of *Theoretical Physics*. Butterworth-Heinemann, 3rd edition, 1986.

[91] F. O. Lehmann, S. Gorb, N. Nasir, and P. Schützner. Elastic deformation and energy loss of flapping fly wings. *J. Exp. Biol.*, 214:2949–2961, 2011.

[92] F.O. Lehmann and S. Pick. The aerodynamic benefit of wing–wing interaction depends on stroke trajectory in flapping insect wings. *J. Exp. Biol.*, 210:1362–1377, 2007.

[93] D. Lentink and M.H. Dickinson. Rotational accelerations stabilize leading edge vortices on revolving fly wings. *J. Exp. Biol.*, 212:2705–2719, 2009.

[94] M. Lesieur, O. Métais, and P. Comte. *Large-Eddy Simulations of Turbulence*. Cambridge University Press, 2005.

[95] M. J. Lighthill. Large-amplitude elongated-body theory of fish locomotion. *Proc. R. Soc. Lond. B*, 179:125–138, 1971.

[96] H. Liu. Integrated modeling of insect flight: From morphology, kinematics to aerodynamics. *J. Comput. Phys.*, 228:439–459, 2009.

[97] H. Liu and K. Kawachi. A numerical study of insect flight. *J. Comput. Phys.*, 146:124–156, 1998.

[98] Q. Liu and O. Vasilyev. A Brinkman penalization method for compressible flows in complex geometries. *J. Comput. Phys.*, 227:946–966, 2007.

[99] T. Luu, A. Cheung, D. Ball, and M. V. Srinivasan. Honeybee flight: a novel 'streamlining' response. *J. Exp. Biol.*, 214:2215–2225, 2011.

[100] K. Y. Ma, S. B. Fuller P. Chirarattananon, and R. J. Wood. Control flight of a bibiological inspired, insect-scale robot. *Science*, 340:603–607, 2013.

[101] L. Machiels. Predictability of small-scale motion in isotropic fluid turbulence. *Phys. Rev. Lett.*, 79:18, 1997.

[102] M. Maeda, N. Gao, N. Nishihashi, and H. Liu. A free-flight simulation of insect flapping flight. *J. Aero Aqua Bio-mech.*, 1(1):71–79, 2010.

[103] M. Maeda and H. Liu. Ground effect in fruit fly hovering: A three-dimensional computational study. *J. Biomech. Sc. Engin.*, 8:344–355, 2013.

[104] K. Mahesh, S. K. Lele, and P. Moin. The influence of entropy fluctuations on the interaction of turbulence with a shock wave. *J. Fluid Mech.*, 334:353–379, 1997.

[105] H. Matthies, R. Niekamp, and J. Steindorf. Algorithms for strong coupling procedures. *Computer Methods in Applied Mechanics and Engineering*, 195:2028–2049, 2006.

[106] W. J. Maybury and F.-O. Lehmann. The fluid dynamics of flight control by kinematic phase lag variation between two robotic insect wings. *J. Exp. Biol.*, 207:4707–4726, 2004.

[107] S. Michelin. *Falling, flapping, flying, swimming,...: high-Re fluid-solid interactions with vortex shedding*. PhD thesis, University of California, San Diego, 2009.

[108] S. Michelin and S. G. Llewellyn Smith. Resonance and propulsion performance of a heaving flexible wing. *Phys. Fluids*, 21:071902–1–071902–15, 2009.

[109] S. Michelin, S. Llewellyn Smith, and B. Glover. Vortex shedding model of a flapping flag. *J. Fluid Mech.*, 617:1–10, 2008.

[110] L. A. Miller and C. S. Peskin. A computational fluid dynamics of 'clap and fling' in the smallest insects. *J. Exp. Biol.*, 208:195–212, 2005.

[111] L. A. Miller and C. S. Peskin. Flexible clap and fling in tiny insect flight. *J. Exp. Biol.*, 212:3076–3090, 2009.

[112] R. Mittal and G. Iaccarino. Immersed boundary methods. *Annu. Rev. Fluid Mech.*, 37:239–261, 2005.

[113] A.S. Monin and A.M. Yaglom. *Statistical Fluid Mechanics: Mechanics of Turbulence*. MIT Press, Cambridge, MA, 1971.

[114] J. Morales, W.J.T. Bos, K. Schneider, and D.C. Montgomery. The effect of toroidicity on reversed field pinch dynamics. *Plasma Phys. Control. Fusion*, 56:095024, 2014.

[115] J. Morales, M. Leroy, W.J.T. Bos, and K. Schneider. Simulation of confined magnetohydrodynamic flows using a pseudo-spectral method with volume penalization. *J. Comput. Phys.*, 274:64–94, 2014.

[116] N. Mordant and J.-F. Pinton. Velocity measurement of a settling sphere. *Eur. Phys. J. B*, 18:343–352, 2000.

[117] T.J. Mueller, L.J. Pohlen, P.E. Conigliaro, and B.J. Jansen. The influence of free-stream disturbances on low Reynolds number airfoil experiments. *Exp. Fluids*, 1:3–14, 1983.

[118] F. T. Mujires, M. J. Elzinga, J. M. Melis, and M. H. Dickinson. Flies evade looming targets by executing rapid visually directed banked turns. *Science*, 344:172–177, 2014.

[119] J. Murtis, J.S. Elkinton, and R. T. Carde. Odor plumes and how insects use them. *Annu. Rev. Entomol.*, 37:505–532, 1992.

[120] T. Nakata and H. Liu. Aerodynamic performance of a hovering hawkmoth with flexible wings: a computational approach. *Proceedings of the Royal Society B: Biological Sciences*, 279(1729):722–731, 2012.

[121] T. Nakata and H. Liu. A fluid-structure interaction model of insect flight with flexible wings. *J. Comput. Phys.*, 231(4):1822–1847, 2012.

[122] R. Nguyen van yen, D. Kolomenskiy, and K. Schneider. Approximation of the Laplace and Stokes operators with Dirichlet boundary conditions through volume penalization: A spectral viewpoint. *Numer. Math.*, 128:301–338, 2014.

[123] Steven A. Orszag and G. S. Patterson. Numerical simulation of three-dimensional homogeneous isotropic turbulence. *Phys. Rev. Lett.*, 28:76–79, Jan 1972.

[124] V. M. Ortega-Jimenez, J. S. M. Greeter, R. Mittal, and T. L. Hedrick. Hawkmoth flight stability in turbulent vortex streets. *J. Exp. Biol.*, 216:4567–4579, 2013.

[125] V. M. Ortega-Jimenez, R. Mittal, and T. L. Hedrick. Hawkmoth flight performance in tornado-like whirlwind vortices. *Bioinspir. Biomim.*, 9(2):025003, 2014.

[126] V. M. Ortega-Jimenez, N. Sapir, M. Wolf, E. A. Variano, and R. Dudley. Into turbulent air: size-dependent effects of von Kármán vortex streets on hummingbird flight kinematics and energetics. *Proc. Biol. Sci.*, 281(1783):20140180, 2014.

[127] S. Osher and R. Fedkiw. *Level Set Methods and Dynamic implicit Surfaces*. Springer, 2003.

[128] R. Pasquetti, R. Bwemba, and L. Cousin. A pseudo-penalization method for high reynolds number unsteady flows. *App. Num. Math.*, 58:946–954, 2008.

[129] D. Pekurovsky. P3DFFT: a framework for parallel computations of Fourier transforms in three dimensions. *SIAM J. Sci. Comput.*, 34:C192–C209, 2012.

[130] C. S. Peskin. Numerical analysis of blood flow in the heart. *J. Comput. Phys.*, 25:220–252, 1977.

[131] C. S. Peskin. The immersed boundary method. *Acta Numerica*, 11:479–517, 2002.

[132] R. Peyret. *Spectral Methods for Incompressible Viscous Flow*. Springer Berlin / Heidelberg, 2002.

[133] R. Ramamurti and W. C. Sandberg. A three-dimensional computational study of the aerodynamic mechanisms of insect flight. *J. Exp. Biol.*, 205:1507–1518, 2002.

[134] S. Ramananarivo, R. Godoy-Diana, and B. Thiria. Rather than resonance, flapping wing flyers may play on aerodynamics to improve performance. *Proc. Natl. Acad. Sci. USA*, 108:5964–5969, 2011.

[135] V. Raspa, S. Ramananarivo, B. Thiria, and R. Godoy-Diana. Vortex-induced drag and the role of aspect ratio in undulatory swimmers. *Phys. Fluids*, 26(4), 2014.

[136] S. Ravi, J. D. Crall, L. McNeilly, S. F. Gagliardi, A. A. Biewener, and S. A. Combes. Hummingbird flight stability and control in freestream turbulent winds. *J. Exp. Biol.*, 218:1444–1452, 2015.

[137] S. Ravi, J.D. Crall, A. Fisher, and S. A. Combes. Rolling with the flow: bumblebees flying in unsteady wakes. *J. Exp. Biol.*, 216:4299–4309, 2013.

[138] L. Ristroph, A. J. Bergou, G. Ristroph, K. Coumes, G. J. Berman, J. Guckenheimer, Z. J. Wang, and I. Cohen. Discovering the flight autostabilizer of fruit flies by inducing aerial stumbles. *Proc. Natl. Acad. Sci. USA*, 107(11):4820–4824, 2010.

[139] L. Ristroph, G. Ristroph, S. Morozova, A.J. Bergou, S. Chang, J. Guckenheimer, Z.J. Wang, and I. Cohen. Active and passive stabilization of body pitch in insect flight. *J. R. Soc. Interface*, 10:20130237, 2013.

[140] R. S. Rogallo. Numerical experiments in homogeneous turbulence. *NASA Technical Memorandum*, 81315:1–93, 1981.

[141] R.L. Sani and P.M. Gresho. Résumé and remarks on the open boundary condition minisymposium. *Int. J. Numer. Methods Fluids*, 18:983–1008, 1994.

[142] T. Sawada and T. Hisada. Fluid-structure interaction analysis of the two-dimensional flag-in-wind problem by an interface-tracking ALE finite element method. *Computers & Fluids*, 36:136–146, 2007.

[143] P. Schlatter, N.A. Adams, and L. Kleiser. A windowing method for periodic inflow/outflow boundary treatment of non-periodic flows. *J. Comput. Phys.*, 206:505–535, 2005.

[144] K. Schneider. Numerical simulation of the transient flow behaviour in chemical reactors using a penalisation method. *Computers & Fluids*, 34:1223–1238, 2005.

[145] K. Schneider and M. Farge. Decaying two-dimensional turbulence in a circular container. *Phys. Rev. Lett.*, 95(24):244502, 2005.

[146] K. Schneider and M. Farge. Numerical simulation of the transient flow behaviour in tube bundles using a volume penalization method. *J. Fluids Struct.*, 20:555–566, 2005.

[147] K. Schneider, D. Kolomenskiy, and E. Deriaz. Is the CFL condition sufficient? some remarks. In Carlos A. de Moura and Carlos S. Kubrusly, editors, *The Courant-Friedrichs-Lewy (CFL) Condition. 80 Years After Its Discovery*, pages 139–146, 2013.

[148] J. Schulze and J. Sesterhenn. Optimal distribution of porous media to reduce trailing edge noise. *Computers & Fluids*, 78(0):41 – 53, 2013.

[149] J. Sesterhenn. A characteristic-type formulation of the Navier–Stokes equations for high order upwind schemes. *Computers & fluids*, 30(1):37–67, 2000.

[150] M. Shelley, N. Vandenberghe, and J. Zhang. Heavy flags undergo spontaneous oscillations in flowing water. *Phys. Rev. Lett.*, 94, 2005.

[151] A. Shishkin, P. Schuetzner, C. Wagner, and F.-O. Lehmann. Experimental quantification and numerical simulation of unsteady flow conditions during free flight maneuvers in insects. In C. Tropea and H. Bleckmann, editors, *Nature-Inspired Fluid Mechanics*, pages 81–99. Berlin Heidelberg: Springer., 2012.

[152] W. Shyy, H. Aono, S. K. Chimakurthi, P. Trizila, C.-K. Kang, C. E. S. Cesnik, and H. Liu. Recent progress in flapping wing aerodynamics and aeroelasticity. *Progress in Aerospace Sciences*, 46(7):284–327, 2010.

[153] F. Sotiropoulos and X. Yang. Immersed boundary methods for simulating fluid–structure interaction. *Prog. Aerospace Sci.*, 65:1–21, 2014.

[154] R. B. Srygley and A. L. R. Thomas. Unconventional lift-generating mechanisms in free-flying butterflies. *Nature*, 420:660–664, 2002.

[155] J. Steindorf. *Partitionierte Verfahren fur Probleme der Fluid-Struktur Wechselwirkung*. PhD thesis, Technische Universität Braunschweig, Fachbereich für Mathematik und Informatik, 2002.

[156] D.C. Sternel, M.Schäfer, M. Heck, and S. Yigit. Efficiency and accuracy of fluid–structure interaction simulations using an implicit partioned approach. *Computational Mechanics*, 43:103–113, 2008.

[157] M. A. Storti, N. M. Nigro, R. R. Paz, and L. D. Dalcín. Strong coupling strategy for fluid–structure interaction problems in supersonic regime via fixed point iteration. *J. Sound Vibr.*, 320:859–877, 2009.

[158] K. Suzuki, K. Minami, and T. Inamuro. Lift and thrust generation by a butterfly-like flapping wing-body model: immersed boundary-lattice Boltzmann simulations. *J. Fluid Mech.*, 767:659–695, 2015.

[159] S. M. Swartz, K. S. Breuer, and D. J. Willis. Aeromechanics in aeroecology: flight biology in the aerosphere. *Integr. Comp. Biol.*, 48(1):85–98, 2008.

[160] L. Tang, M. P. Païdoussis, and J. Jiang. Cantilevered flexible plates in axial flow: Energy transfer and the concept of flutter-mill. *J. Sound Vib.*, 326:263–276, 2009.

[161] G. J. Taylor, T. Luu, D. Ball, and M. V. Srinivasan. Vision and air flow combine to streamline flying honeybees. *Sci. Rep.*, 3:2614, 2013.

[162] J. F. Thompson, Z.U.A Warsi, and C. W. Mastin. *Numerical grid generation: Foundations and Applications*. North-Holland Amsterdam, 1985.

[163] G. S. Triantafyllou, M. S. Triantafyllou, and M. A. Grosenbaugh. Optimal thrust development in oscillating foils with application to fish propulsion. *J. Fluid Struct.*, 7:205–224, 1993.

[164] M.S. Triantafyllou, A.H. Techet, and F.S.Hover. Review of experimental work in biomimetic foils. *IEEE J. Oceanic Eng.*, 29:585–594, 2004.

[165] S. Turek and J. Hron. Proposal for numerical benchmarking of fluid-structure interaction between an elastic object and laminar incompressible flow. In H.J. Bungartz and M. Schäfer, editors, *Fluid-Structure Interaction: Modelling, Simulation, Optimisation*, volume 53 of *Lecture Notes in Computational Science and Engineering*, pages 371–385. Springer Berlin Heidelberg, 2006.

[166] S. Turek, J. Hron, M. Razzaq, H. Wobker, and M. Schäfer. Numerical benchmarking of fluid-structure interaction: A comparison of different discretization and solution approaches. In Hans-Joachim Bungartz, Miriam Mehl, and Michael Schäfer, editors, *Fluid Structure Interaction II*, volume 73 of *Lecture Notes in Computational Science and Engineering*, pages 413–424. Springer Berlin Heidelberg, 2010.

[167] M. Uhlmann. An immersed boundary method with direct forcing for the simulation of particulate flows. *J. Comput. Phys.*, 209:448–476, 2005.

[168] James R. Usherwood and Fritz-Olaf Lehmann. Phasing of dragonfly wings can improve aerodynamic efficiency by removing swirl. *J. R. Soc. Interface*, 5:1303–1307, 2008.

[169] E. H. van Brummelen. Added mass effects of compressible and incompressible flows in fluid-structure interaction. *J. Appl. Mech.*, 76:021206, 2009.

[170] J. T. Vance, I. Faruque, and J. S. Humbert. Kinematic strategies for mitigating gust perturbations in insects. *Bioinspir. Biomim.*, 8(1):016004, 2013.

[171] M. Vanella, T. Fitzgerald, S. Preidikman, E. Balaras, and B. Balachandran. Influence of flexibility on the aerodynamic performance of a hovering wing. *J. Exp. Biol*, 121:95–105, 2009.

[172] A. Vincent and M. Meneguzzi. The satial structure and statistical properties of homogeneous turbulence. *Journal of Fluid Mechanics*, 225:1–20, 4 1991.

[173] M. von Scheven. *Effiziente Algorithmen für die Fluid-Struktur-Wechselwirkung*. PhD thesis, Universität Stuttgart - Institut für Baustatik und Baudynamik, 2009.

[174] Z. J. Wang. Dissecting insect flight. *Annu. Rev. Fluid Mech.*, 37:183–210, 2005.

[175] D. R. Warrick, B. W. Tobalske, and D. R. Powers. Lift production in the hovering hummingbird. *Proceedings of the Royal Society of London B: Biological Sciences*, 276(1674):3747–3752, 2009.

[176] T. Weis-Fogh. Quick estimates of flight fitness in hovering animals, including novel mechanisms for lift production. *J. Exp. Biol.*, 59:169–230, 1973.

[177] T. Wick. Fluid-structure interactions using different mesh motion techniques. *Computers & Structures*, 89(13-14):1456 – 1467, 2011.

[178] T. J. Wolf, C. P. Ellington, and I. S. Begley. Foraging costs in bumblebees: field conditions cause large individual differences. *Insectes Sociaux*, 46(3):291–295, 1999.

[179] J. H. Wu and M. Sun. Unsteady aerodynamic forces of a flapping wing. *J. Exp. Biol.*, 207:1137–1150, 2004.

[180] J.-H. Wu and M. Sun. Control of flight forces and moments by flapping wings of model bumblebee. *Appl. Math. Mech. -Engl. Ed.*, 29:333–350, 2008.

[181] J. C. Wyngaard. *Turbulence in the Atmosphere*. Cambridge University Press, 2010.

[182] Y. Xiong and M. Sun. Dynamic flight stability of a bumblebee in forward flight. *Acta Mech. Sin.*, 24:25–36, 2008.

[183] N. Xu and M. Sun. Lateral dynamic flight stability of a model bumblebee in hovering and forward flight. *J. Theor. Biol.*, 319:102–115, 2013.

[184] J. Yang, S. Preidikman, and E. Balaras. A strongly coupled, embedded-boundary method for fluid–structure interactions of elastically mounted rigid bodies. *J. Fluids Struct.*, 24:167–182, 2008.

[185] X. Yang, X. Zhang, Z. Li, and G.-W. He. A smoothing technique for discrete delta functions with application to immersed boundary method in moving boundary simulations. *J. Comput. Phys.*, 228:7821–7836, 2009.

[186] Peter Derek Yeh and Alexander Alexeev. Free swimming of an elastic plate plunging at low Reynolds number. *Phys. Fluids*, 26(5):–, 2014.

[187] P. K. Yeung, X. M. Zhai, and K. R. Sreenivasan. Turbulence at high resolution: intense events in dissipation, enstrophy and acceleration. In *67th Annual Meeting of the APS Division of Fluid Dynamics*, 2014.

[188] N. Yokoyama, K. Senda, M. Iima, and N. Hirai. Aerodynamic forces and vortical structures in flapping butterfly's forward flight. *Phys. Fluids*, 25:021902, 2013.

[189] J. Young, S. M. Walker, R. J. Bomphrey, G. K. Taylor, and A. L. R. Thomas. Details of insect wing design and deformation enhance aerodynamic function and flight efficiency. *Science*, 325, 2009.

[190] F. A. Zabala, G. M. Card, E. I. Fontaine, and M .H. Dickinson. Flight dynamics and control of evasive maneuvers: The fruit fly's takeoff. *IEEE Transactions on Biomedical Engineering*, 56, 2009.

Flying and swimming animals have developed efficient ways to produce the fluid flow that generates the desired forces for their locomotion. These bio-inspired problems couple fluid dynamics and solid mechanics with complex geometries and kinematics. The present thesis is placed in this interdisciplinary context and uses numerical simulations to study these fluid--structure interaction problems with applications in insect flight and swimming fish. Based on existing work on rigid moving obstacles, using an efficient Fourier discretization, a numerical method has been developed, which allows the simulation of flexible, deforming obstacles as well, and provides enhanced versatility and accuracy in the case of rigid obstacles. The method relies on the volume penalization method and the fluid discretization is still based on a Fourier discretization. The code, designed to run on massively parallel supercomputers, is entirely open source and freely available on the internet. We first apply this method to insects with rigid wings, where the body and other details, such as the legs and antennae, can be included. After presenting detailed validation tests, we proceed to studying a bumblebee model in fully developed turbulent flow. Our simulations show that turbulent perturbations affect flapping insects in a different way than human-designed fixed-wing aircrafts. While in the latter, upstream perturbations can cause transitions in the boundary layer, the former do not present systematical changes in aerodynamic forces. We conclude that insects rather face control problems in a turbulent environment than a deterioration in force production. In the next step, we design a solid model, based on a one--dimensional beam equation, and simulate coupled fluid--solid systems. Applications deal, in a two-dimensional setup, with insect flight, but also with simplified three--dimensional models for swimming fish. In these 'swimmers', consisting of a flexible plate with one rigid direction, we study the influence of the shape on the hydrodynamic efficiency. A contracting shape, as found in some amphibians, is found to swim faster and require less power than an expanding shape, which is more similar to most caudal fins observed in fish. We present evidence that this finding can be explained by a favorable interaction with the tip-vortices in the case of the contracting shape.