Fast Motion of Heaving Airfoils

Siddhartha Verma, Ivica Kicic, Guido Novati, Flavio Noca and Petros Koumoutsakos
To Flap, or not to Flap…

• Low Reynolds number flight
  • Birds & insects: flapping => unsteady effects
    McCroskey (Ann. Rev. 1982), Read et al. (J Fluids Struct. 2003)

• Designing better Micro Aerial Vehicles (MAVs)
  Rozhdestvensky & Ryzhov (Prog. Aero. Sci. 2003),
  Shyy et al. (Prog. Aero. Sci. 2010)

• Several advantages of flapping at low Re
  • High Manoeuvrability - quick response to gusts
  • Less noise than a propeller or jet
  • Short take-off & landing
  • Generate lift effectively at variable speeds
  • Low speed flight - risk of stall with fixed wing
    • Explore difficult terrain - under jungle canopies, caves, crevasses, collapsed buildings …
Heaving Airfoils

- Purely heaving airfoils generate thrust
- Knoller-Betz-Katzmayr effect
Numerical methods

- Naiver-Stokes: Remeshed vortex methods (2D)
  - Solve vorticity form of incompressible Navier-Stokes

- Brinkman penalization
  - Accounts for fluid-solid interaction
    - Angot et al., Numerische Mathematik (1999)

- 2D: Wavelet-based adaptive grid
  - Cost-effective compared to uniform grids

- Surface-force distribution
  - Force exerted by fluid-flow on the body
    - Verma et al., Int J. Num. Meth. Fluids (2017)
Quantities of Interest

- Average horizontal velocity
- Outcome of flow-body interactions

- Net horizontal force
- Outcome of flow-body interactions

- Power input: \[ P_{\text{Input}} = F_y \cdot u_y \]
  - Effort by external agent in heaving the airfoil vertically

- Cost of Transport
  - Energy consumed for travelling a unit distance
  - Car equivalent: Litres/km

0.1 \leq \text{Amplitude/L} \leq 0.4
0.1Hz \leq \text{Frequency} \leq 3Hz
\nu = 6e-5
\text{Re} \leq 5000

L = \text{Chord Length} = .2
\text{(domain = 1)}
Airfoil Dynamics

- \( T = 1.33 \) (Startup)
  - Max \( u_y \), vortex shedding at LE => low press.
  - Max thrust, but Max \( P_{\text{Input}} \)

- \( T = 9.0 \) (Steady)
  - \( u_y = 0, \ \alpha_{\text{effective}} = 0 \)
  - Flow head-on, Max drag

- \( T = 9.5 \)
  - Max \( u_y \), Low press. LE, Max thrust
FTLE and particle tracking

- Finite Time Lyapunov Exponent
  - Identify regions of flow separation (fluid doesn’t mix)
  - Integration time = 0.3 periods

- Particle tracking
  - Points seeded on opposite sides of FTLE ridges diverge (red vs green/black)
Optimization: Parallelized NSGA-II

Basic structure of Genetic Algorithms:

- Population Initialization
- Fitness Evaluation
- Selection of the Fittest
- Mating & Mutation
- convergence?

Multi-objective selection of non-dominated individuals:
- Maintain a diverse approximation of the Pareto front
  - NSGA-II algorithm (Deb et al., 2002)
- Scheduling of evaluations to use comp. resources efficiently
  - TORC library (Hadjidoukas et al., 2012)
Given 2N individuals (current population + offspring), keep only the N best.

Sort individuals by rank and crowding distance

\[ i < j \text{ if } (\text{rank}_i < \text{rank}_j) \]
Parallelization

- Parallelization with \( \Pi^4 U \): minimal effort
- No details of parallel machinery required in coding
  

- Essentially, we took Deb et al.’s C-code, and coupled with \( \Pi^4 U \)

```c
void task(double *x, double *y)
{
    *y = x[0] + x[1];
}
```

```
int main(int argc, char *argv[])
{
    double result[100];
    for (int i=0; i<100; i++)
    {
        double d[2] = {drand48(), drand48()};
        task(d, &result[i]);
    }
    return 0;
}
```

```
void task(double *x, double *y)
{
    *y = x[0] + x[1];
}
```

```
int main(int argc, char *argv[])
{
    double result[100];
    #pragma omp parallel for
    for (int i=0; i<100; i++)
    {
        double d[2] = {drand48(), drand48()};
        task(d, &result[i]);
    }
    return 0;
}
```

```
int main(int argc, char *argv[])
{
    double result[100];
    torc_init(argc, argv, MODE_MW);
    for (int i=0; i<100; i++)
    {
        double d[2] = {drand48(), drand48()};
        torc_task(-1, task, 2, 
            2, MPI_DOUBLE, CALL_BY_COP, // IN (COPY) 
            1, MPI_DOUBLE, CALL_BY_RES, // OUT 
            d, &result[i]);
    }
    torc_waitall();
    return 0;
}
```
Coupling Framework

- **Simulation**
  - MRAG-I2D [Rossinelli et al., JPC 2015]
  - 2D multi-core adaptive solver for incompressible fluids

- **Optimization**
  - NSGA2 [Deb et al., IEEE TEC 2002]
  - Multi-objective genetic algorithm

- **Scheduling**
  - TORC library [Hadjidoukas et al, ECPP 2012]
  - Platform agnostic task-based parallelism
  - Multi-level parallelism
  - Transparent load balancing
The Pareto Front

- Pareto Front: Lists ‘non-dominated’ individuals
  - No other individual better than these in both objectives
  - Tradeoff between conflicting objectives, critical for survival
- Advantage: Pick parameters most suitable for mission requirements
  - High speed
  - Balanced performance
  - High efficiency
Performance Metrics

- High velocity => high cost
  (and vice versa)
- Almost linear relationship between the conflicting metrics

<table>
<thead>
<tr>
<th>Mean QoI</th>
<th>Efficient</th>
<th>Generalist</th>
<th>Fast</th>
</tr>
</thead>
<tbody>
<tr>
<td>Velocity (L/s)</td>
<td>0.74</td>
<td>3.81</td>
<td>6.32</td>
</tr>
<tr>
<td>CoT</td>
<td>1.07</td>
<td>8.25</td>
<td>15.84</td>
</tr>
<tr>
<td>Velocity (rel)</td>
<td>19%</td>
<td>100%</td>
<td>166%</td>
</tr>
<tr>
<td>CoT (rel)</td>
<td>13%</td>
<td>100%</td>
<td>192%</td>
</tr>
</tbody>
</table>
- Optimal airfoils *avoid* large Amplitudes (A < 20% of chord length)
- Higher Freq. preferred
Parameters vs. Parameters

- **Efficient**
  - Low Freq. ($v < 1.0$) | Amp. varies ($0.07 < A/L < 0.2$)

- **Generalist**
  - Both vary: Freq. ($1.0 < v < 2.0$) | Amp.: ($0.07 < A/L < 0.2$)

- **Fast**
  - High Freq. ($1.5 < v < 2.0$) | Specific Amp. ($A \approx 0.12L$)

Empty region since simulations thresholded to $Re < 5000$
Objectives vs. Parameters

- **Efficient**
  - Low Freq. \((v < 1.0)\) | Amp. varies \((0.07 < A/L < 0.2)\)

- **Generalist**
  - Both vary: Freq. \((1.0 < v < 2.0)\) | Amp.: \((0.07 < A/L < 0.2)\)

- **Fast**
  - High Freq. \((1.5 < v < 2.0)\) | Specific Amp. \((A \approx 0.12L)\)

- Optimal individuals:
  - Both Velocity and CoT proportional to Freq.
  - Amplitude - no consistent trend \((A/L < 0.2)\)

- Two distinct ‘branches’ for generalist group
Distinct Behaviour, Similar Outcome

- Close neighbours on the Pareto front
- Markedly different parameter values
- 2X difference in $A_{\text{heave}}$ & $F_{\text{heave}}$

<table>
<thead>
<tr>
<th></th>
<th>Foil 1</th>
<th>Foil 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A_{\text{heave}}/L$</td>
<td>0.10</td>
<td>0.18</td>
</tr>
<tr>
<td>$F_{\text{heave}}$</td>
<td>1.26</td>
<td>0.67</td>
</tr>
</tbody>
</table>

![Graph showing velocity vs. cost and frequency for Foil 1 and Foil 2.](#)
• Max thrust for heaving airfoil => low pressure at Leading Edge
  • \(u_y\) and \(P_{input}\) are both max

• Optimal individuals prefer large Freq. range, but limited Amp. range
  • **Efficient**: Low Freq., varying Amp.
  • **Generalist**: High Freq., varying Amp.
  • **Fast**: High Freq., Specific Amp \((A\approx0.12L)\)

• Optimisation discovers similar performance, with very different heaving parameters
Backup
Algorithm 1: NSGA-II

1. Initialize population $P^{(0)}$ of size $\mu$ and set generation $g = 0$.
2. while No stopping criterion is met do
3. \hspace{1em} Generate an offspring population $\bar{P}^{(g+1)}$ from the parent population $P^{(g)}$ using crossover and mutation.
4. \hspace{1em} Combine offspring $\bar{P}^{(g+1)}$ and parent population $P^{(g)}$ into a mixed population and sort it according to non-domination rank and crowding distance.
5. \hspace{1em} Select the best $\mu$ individuals for the next parent generation $P^{(g+1)}$.
6. \hspace{1em} $g = g + 1$
7. end
8. Output the current population.
Parameters

L = Chord Length = .2
(domain = 1)

.1 ≤ Amplitude / L ≤ 1

.1 ≤ Frequency ≤ 1

ν = 6e-5

Re ≤ 5000

0 ≤ Angle of attack ≤ 90°

0 ≤ Angular / Heave freq ≤ 10

0 ≤ Relative phase ≤ 360°

0 ≤ Mean angle ≤ 90°
Comparison of 2, 3 and 6 parameters

- Fixed angle of 10° decreases velocity by half. Also greatly increases oscillations.

- Pitching case has no additional oscillations, but still low velocity.

- These were manually selected parameters. Optimization will give more insight.

<table>
<thead>
<tr>
<th></th>
<th>Heave</th>
<th>+angle</th>
<th>+pitch</th>
</tr>
</thead>
<tbody>
<tr>
<td>Velocity</td>
<td>100%</td>
<td>50%</td>
<td>69%</td>
</tr>
<tr>
<td>CoT</td>
<td>100%</td>
<td>119%</td>
<td>103%</td>
</tr>
</tbody>
</table>