

MASTER OF SCIENCE THESIS

---

# **Hybrid Eulerian-Lagrangian Vortex Particle Method**

**A fast and accurate numerical method for 2D Vertical-Axis  
Wind Turbine**

**L. Manickathan B.Sc.**

---

Date TBD

Faculty of Aerospace Engineering · Delft University of Technology



# **Hybrid Eulerian-Lagrangian Vortex Particle Method**

**A fast and accurate numerical method for 2D Vertical-Axis  
Wind Turbine**

MASTER OF SCIENCE THESIS

For obtaining the degree of Master of Science in Aerospace  
Engineering at Delft University of Technology

L. Manickathan B.Sc.

Date TBD



Copyright © L. Manickathan B.Sc.  
All rights reserved.

DELFT UNIVERSITY OF TECHNOLOGY  
DEPARTMENT OF  
AERODYNAMICS AND WIND ENERGY

The undersigned hereby certify that they have read and recommend to the Faculty of Aerospace Engineering for acceptance a thesis entitled "**Hybrid Eulerian-Lagrangian Vortex Particle Method**" by **L. Manickathan B.Sc.** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: Date TBD

Head of department:

---

prof.dr.ir. G.J.W. van Bussel

Academic Supervisor:

---

dr.ir. C.J. Simao Ferreira

Academic Supervisor:

---

dr.ir. A. Palha da Silva Clerigo



---

# **Summary**

This is the summary of the thesis.



---

# Acknowledgements

I wish to thank the following persons...

Delft, The Netherlands  
Date TBD

L. Manickathan B.Sc.



---

# Contents

<b>Summary</b>	v
<b>Acknowledgements</b>	vii
<b>List of Figures</b>	xiii
<b>List of Tables</b>	xv
<b>1 Verification and Validation of Hybrid Method</b>	1
1.1 Lamb-Oseen Vortex Evolution . . . . .	2
1.1.1 Problem Definition . . . . .	2
1.1.2 Results and Discussion . . . . .	3
1.1.3 Conclusion . . . . .	12
1.2 Clercx-Bruneau Dipole Convection . . . . .	13
1.2.1 Problem Definition . . . . .	13
1.2.2 Results and Discussion . . . . .	14
1.2.3 Conclusion . . . . .	17
1.3 Clercx-Bruneau Dipole Collision . . . . .	17
1.3.1 Problem Definition . . . . .	17
1.3.2 Results and Discussion . . . . .	19
1.3.3 Conclusion . . . . .	22
1.4 Impulsively Started Cylinder at $Re = 550$ . . . . .	22
1.4.1 Problem Definition . . . . .	22
1.4.2 Results and Discussion . . . . .	23
1.4.3 Conclusion . . . . .	26
1.5 Stalled Elliptic airfoil at $Re = 5000$ . . . . .	30
1.5.1 Problem Definition . . . . .	30
1.5.2 Results and Discussion . . . . .	30
1.6 Multi-body problem . . . . .	33

<b>Nomenclature</b>	<b>1</b>
<b>2 Conclusion and Recommendation</b>	<b>35</b>
2.1 Conclusion . . . . .	35
2.1.1 Lagrangian domain . . . . .	35
2.1.2 Eulerian domain . . . . .	35
2.1.3 Hybrid method . . . . .	35
2.2 Recommendations . . . . .	35
2.2.1 Lagrangian domain . . . . .	35
2.2.2 Eulerian domain . . . . .	35
2.2.3 Hybrid method . . . . .	35
<b>A pHyFlow Code Structure</b>	<b>37</b>
<b>References</b>	<b>37</b>

---

## List of Figures

1.1	(Not to scale) The domain decomposition for the Lamb-Oseen vortex problem, $\Omega_E \subseteq \Omega_L$ . The Eulerian domain is defined as $\Omega_E = [-0.5, 0.5] \times [-0.5, 0.5]$ with Dirichlet boundary $\partial\Omega_{dirichlet}$ [—, solid red]. The parameters of the discretization are tabulated in table 1.1. . . . .	3
1.2	Initial relative error at $t = 0$ inside the Eulerian domain $\Omega_E$ . The figure depicts (a) the relative error in velocity $\mathbf{u}$ , and (b) the relative error in vorticity $\omega$ . . . . .	5
1.3	Evolution of the maximum relative error in velocity (dashed) and the maximum relative error in vorticity (solid), equation ??, from $t = 0$ to $t = 1$ , using the parameters tabulated in table 1.1. The figure compares uncoupled case (black) vs. the one-way coupled case (blue) vs. the fully coupled case (red). . . . .	5
1.4	Plot of the relative error in velocity ( <i>left</i> ) and relative error in vorticity ( <i>right</i> ) in the Eulerian domain $\Omega_E$ at $t = 1$ . The figure compares the error between (a)(b) the uncoupled, (c)(d) the one-way coupled, and (e)(f) the fully coupled cases. . . . .	6
1.5	Plot of the relative error in velocity (left) and the relative error in vorticity (right) in the Eulerian domain $\Omega_E$ at $t = 1$ . The figure compares the error between (a)(b) without conservation of circulation, and (c)(d) with the conservation of circulation. . . . .	8
1.6	Plot of the maximum relative error in vorticity $\epsilon_\omega$ [ - -, dashed] and maximum relative error in vorticity $\epsilon_u$ [ —, solid], equation ??, from $t = 0$ to $t = 1$ , using the parameters tabulated in table 1.1. The figure compares the coupling scheme with conservation of circulation (red) vs. the coupling scheme without conservation of circulation (green). . . . .	9
1.7	Plot of the error in total circulation $\epsilon_\Gamma$ of the Lagrangian method from $t = 0$ to $t = 1$ . The figure compares the scheme with conservation of circulation [ —, solid red], and the scheme without conservation of circulation [ —, solid green]. . . . .	9
1.8	Evolution of the maximum relative error for various nominal blob spacing $h = [0.01, 0.02, 0.05, 0.1]$ from $t = 0$ to $t = 1$ . The figures shows the maximum relative error in vorticity [ - -, dashed] and maximum relative error in vorticity [ —, solid]. . . . .	10

1.9	Convergence of the maximum relative error in vorticity due to the nominal blob spacing $h = [0.01, 0.02, 0.05, 0.1]$ . . . . .	11
1.10	Evolution of the maximum relative error for various overlap ratios $\lambda = [0.5, 0.75, 1.0, 1.5]$ from $t = 0$ to $t = 1$ . The figures shows the maximum relative error in vorticity [ - -, dashed] and the maximum relative error in vorticity [ —, solid]. . . . .	11
1.11	Evolution of the maximum relative error from $t = 0$ to $t = 1$ for various number of Eulerian sub-steps $k_E = [1, 2, 5, 10]$ , modifying (a) the Lagrangian time step $\Delta t_L$ , and (b) the Eulerian time step $\Delta t_E$ . The figures shows the maximum relative error in vorticity [ - -, dashed] and maximum relative error in vorticity [ —, solid]. . . . .	11
1.12	( <i>Not to Scale</i> ) The domain decomposition for the Clercx-Bruneau convection problem, with the positive pole located at $p_+ = (x_1, y_1) = (-1, 0.1)$ and negative pole located at $p_- = (x_2, y_2) = (-1, -0.1)$ . The parameters of the simulation are tabulated in table 1.2. . . . .	13
1.13	Plot of the Clercx-Bruneau dipole at $t = [0, 0.2, 0.4, 0.7]$ using parameters tabulated in table 1.2. The figure compares the hybrid simulation (top halves) against the FE only simulation (bottom halves). . . . .	15
1.14	Evolution of the maximum vorticity $\max\{\omega\}$ from $t = 0$ to $t = 0.7$ . The solutions are compared with the benchmark results of FE only [—, solid black], and VPM only [- -, dashed black] simulations. The figure depicts (a) the maximum vorticity in the Eulerian and the Lagrangian sub-domain of the hybrid method, and (b) the maximum vorticity of hybrid method with nominal blob spacing $h = 0.01$ and $h = 0.005$ . . . . .	16
1.15	Vorticity contour plots of the dipole with levels ..., -50, -30, -10, 10, 30, 50, ... of the Eulerian and the Lagrangian sub-domains. The figure highlights the effect of the artificial vorticity at the boundary of the Eulerian domain. . . . .	16
1.16	Evolution of the (a) kinetic energy $E$ and (b) enstrophy for the nominal blob spacing $h = 0.01$ and $h = 0.005$ . . . . .	16
1.17	[ <i>Not to Scale</i> ] The domain decomposition for the Clercx-Bruneau dipole collision problem, with the positive pole at $p_+ = (x_1, y_1) = (0.1, 0)$ and negative pole at $p_- = (x_2, y_2) = (-0.1, 0)$ . The parameters of the simulation are tabulated in table 1.3. . . . .	18
1.18	Plot of the dipole at $t = [0, 0.2, 0.4, 0.6, 0.8, 1]$ , comparing the hybrid simulation (left half) and FE only simulation (right half). . . . .	20
1.19	Comparison of the vorticity contours at $t = 1$ . The figure compares the plot obtained by (a) literature, Clercx and Bruneau [?], and (b) the present study, the hybrid and FE only simulation. . . . .	21
1.20	Variation in the fluid parameters from $t = 0$ to $t = 1$ . The figure compares the hybrid results [—, solid blue] with the FE only [—, solid black] results. . . . .	21
1.21	Compares the vorticity generated at the bottom-left wall ( $y = -1, -0.6 \leq x \leq 0$ ) at $t = 0.4$ [—, solid blue], $t = 0.6$ [—, solid red] and $t = 1$ [—, solid green]. . . . .	22
1.22	[ <i>Not to Scale</i> ] The domain decomposition for the Impulsively started cylinder. The parameters of the domain are tabulated in table 1.4. . . . .	23
1.23	Eulerian method and Lagrangian method resolutions . . . . .	24
1.24	Comparison of the vorticity contours for (a) $t = 1$ , (b) $t = 3$ , (c) $t = 5$ and (d) $t = 7$ with contour levels $[-7, \dots, -3, -2, -1, 0.5, -0.2, -0.1, 0.1, 0.2, 0.5, 1, 2, 3, \dots, 7]$ . The figures compares the hybrid simulation (top half) with FE only simulation (bottom half). . . . .	25

1.25 Evolution of the drag coefficient during the initial stages $t = 0$ to $t = 4$ with total drag coefficient $C_d$ (solid), pressure drag coefficient $C_{d_{pres}}$ (dashed) and friction drag coefficient $C_{d_{frie}}$ (dotted). The figure compares results of hybrid simulation ( <b>blue</b> ), FE only simulation ( <b>red</b> ) and reference data ( <b>black</b> ) of Koumoutsakos and Leonard [?]	27
1.26 Parameters sensitivity analysis on the drag evolution of the cylinder from $t = 0$ to $t = 4$ , compared with literature data ( <b>black</b> ) obtained from Koumoutsakos and Leonard [?]	27
1.27 Evolution of the lift and drag coefficient from $t = 0$ to $t = 40$ with artificial perturbation [?]. The figure compares hybrid ( <b>blue</b> ), FE only ( <b>red</b> ), and the reference data ( <b>black</b> ) from RosenFeld et al. [?]	28
1.28 hybrid cylinder LongRun contourfComparison	28
1.29 First dipole	29
1.30 [ <i>Not to Scale</i> ]	30
1.31 hybrid ellipse HybridvsFE contourDeviation	31
1.32 hybrid ellipse Hybrid contours	32
1.33 Forces	32
1.34 [ <i>Not to Scale</i> ]	33
1.35 hybrid ellipse Hybrid contours	34



---

## List of Tables

1.1	Summary of the parameters for the Lamb-Oseen vortex evolution. . . . .	4
1.2	Summary of the parameters for the Clercx-Bruneau dipole convection problem. . . . .	14
1.3	Summary of the parameters for the Clercx-Bruneau dipole collision. . . . .	18
1.4	Summary of the parameters of the hybrid simulation for the Impulsively started cylinder test case for $Re = 550$ . . . . .	24
A.1	Attributes of <code>Blobs</code> class and their description. . . . .	39
A.2	Attributes of <code>Panels</code> class and their description. . . . .	42
A.3	Attributes of <code>LagrangianSolver</code> class and their description. . . . .	43
A.4	Attributes of <code>EulerianSolver</code> class and their description. . . . .	46
A.5	Attributes of <code>HybridSolver</code> class and their description. . . . .	49



---

# Chapter 1

---

## Verification and Validation of Hybrid Method

The verification and validation of the hybrid method focuses on 4 aspects:

- Lamb-Oseen vortex evolution at  $Re = 1000$ (section 1.1)
- Clercx-Bruneau dipole convection at  $Re = 625$  (section 1.2)
- Clercx-Bruneau dipole collision at  $Re = 550$  (section 1.3)
- Stalled elliptical airfoil at  $Re = 5000$  (section 1.5)
- Multi-body problem at  $Re = 550$  (section 1.6)

This chapter focuses on the verification and the validation of the hybrid method. We investigated several test-cases: the Lamb-Oseen vortex problem, the Clercx-Bruneau dipole collision by Clercx and Bruneau [?], the impulsively started cylinder test case, an elliptical airfoil at  $Re = 5000$  and a multi-body problem.

The implementation of the hybrid method was verified using the analytical solutions of the Lamb-Oseen vortex problem. The analytical solution of the problem enabled us to quantify the error in the scheme.

The validation of the hybrid solver was performed using the test cases provided from literature. The Clercx-Bruneau dipole collision problem of Clercx and Bruneau [?], provide a detailed analysis on the evolution of the vorticity field and the generation of vorticity from the boundary. The impulsively started cylinder problem investigated by Koumoutsakos and Leonard [?], and RosenFeld et al. [?] showed the evolution of the forces acting on an immersed cylinder in a free-stream flow.

With a proper verification and validation of the hybrid solver, we were able to perform simulation of more complex problems. We were able to perform simulation of a stalled flow past a thin airfoil using an elliptical airfoil at  $Re = 5000$ . Furthermore, a simulation

of two cylinder in tandem showed the feasibility of simulation a multi-body problem. Both these simulations served as a proof of concept for a simulation of a full VAWT in future research.

## 1.1 Lamb-Oseen Vortex Evolution

The Lamb-Oseen vortex test case simulates the evolution of a laminar vortex core in an unbounded domain. In section ??, we used this test case to verify and validate the implementation of the vortex blobs of the Lagrangian solver and in section ??, we used it to verify the implementation of the Eulerian solver. Therefore, in a similar fashion we employed this test case to verify the coupling of the hybrid solver.

The unbounded nature of the problem helps us to neglect the influence of the solid boundary (i.e the wall). Therefore, this test case did not require the panel solver in the Lagrangian solver as we are only concerned with the coupling of the vortex blobs to the Eulerian solver. Thus, we primarily focused on the vorticity field interpolation error discussed in section ???. Furthermore, with the analytical solution, we were able to quantitatively present the importance of ensuring conservation of circulation.

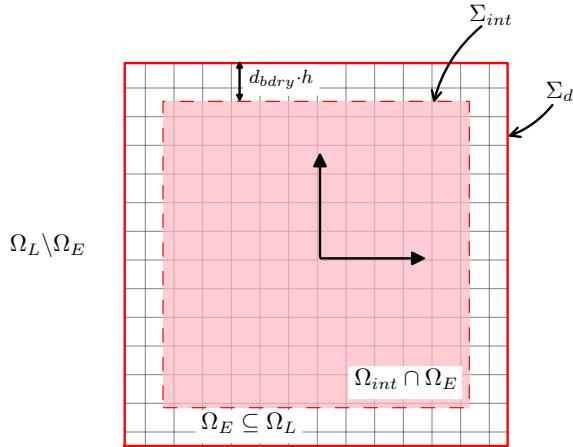
Moreover, we were able to quantify the influences of the discretization on the accuracy of the coupling. A parameter sensitivity analysis was therefore performed to determine their effects on the coupling error. The parameters that determine the spatial discretization of the vortex blobs are the nominal particle spacing  $h$ , and the overlap ratio  $\lambda$  (see figure ??). The spatial discretization of the Eulerian solver is regarded as a control variable for this test case as its impact was concluded in section ???. The parameters that determine the temporal discretization of the hybrid method are the time step size of the Eulerian solver  $\Delta t_E$ , and the time step size of the Lagrangian solver  $\Delta t_L$ . These are depended according to equation ??, with  $k_E$  being the number of Eulerian sub-steps.

The coupling error was quantified by determining the growth of maximum relative error in vorticity  $\epsilon_\omega$  given by equation ??, the approach used in section ?? and in section ??.

### 1.1.1 Problem Definition

The Lamb-Oseen Vortex problem is defined by the vorticity field and the velocity field, equations ?? and ??, respectively. The hybrid solver is initialized by first assigning the strengths of the vortex blobs using equation ???. The Eulerian domain  $\Omega_E$  is then initialized using the solution of the Lagrangian solver. Daeninck [?] used this approach to enhance the coupling between the methods ensuring minimum interpolation error.

Figure 1.1 shows the hybrid domain configuration for the Lamb-Oseen vortex problem with the Lagrangian domain  $\Omega_L$  spanning the full fluid domain. The Eulerian domain  $\Omega_E$  only resolves the center of the Lamb-Oseen core,  $\Omega_E \subseteq \Omega_L$ , bounding  $[-0.5, -0.5] \times [-0.5, -0.5]$ . The boundary of the Eulerian domain  $\partial\Omega_E$  is a Dirichlet velocity boundary  $\partial\Omega_E = [\Sigma_d]$  where the velocity boundary condition is applied, as described in section ???. The correction of the Lagrangian domain is performed in the interpolation domain  $\Omega_{int}$  according to the procedures described in section ???. The boundary of the interpolation domain  $\Sigma_{int}$  is defined with an offset  $d_{bdry}$  from the Eulerian boundary  $\Sigma_d$  by a distance



**Figure 1.1:** (Not to scale) The domain decomposition for the Lamb-Oseen vortex problem,  $\Omega_E \subseteq \Omega_L$ . The Eulerian domain is defined as  $\Omega_E = [-0.5, 0.5] \times [-0.5, 0.5]$  with Dirichlet boundary  $\partial\Omega_{dirichlet}$  [—, solid red]. The parameters of the discretization are tabulated in table 1.1.

$d_{bdry} = 2 \cdot h$ , where  $h$  is the nominal blob spacing. Similar choice was made by Stock [?], and ensures that the potential inaccuracies at the outer Eulerian boundary is ignored during the interpolation procedure.

The spatial discretization of the Eulerian domain  $\Omega_E$  is regarded as a control (i.e fixed) variable. Therefore, the parameter sensitivity analysis is performed by varying the discretization of the Lagrangian method only. The Eulerian domain is discretized with an unstructured mesh formulation using GMSH (see section ??) having  $N_{cells} = 26448$  unstructured cells and grid size  $h_{grid}$  ranging from 0.007 to 0.0016.

The Lamb-Oseen vortex problem is defined according to the parameters tabulated in table 1.1. The center of the core is located at  $(x, y) = (0, 0)$ , where the Eulerian domain  $\Omega_E$  is also centered. The parameters are chosen such that vorticity  $\omega$  and velocity  $\mathbf{u}$  is non-zero at the boundary of the Eulerian domain  $\Sigma_d$ , figure 1.1. With the Lamb-Oseen time constant  $\tau = 100$ , this can be ensured.

The evolution of the Lagrangian solver and the Eulerian solver is performed according to sections ?? and ?? respectively. The Lagrangian solver performs TRS for diffusion of the vortex blobs, see section ???. The scheme requires vortex blob redistribution at every step,  $f_{redis} = 1$ . In conjunction with the redistribution, the population control is also performed at every step,  $f_{pc} = 1$  with  $(\Gamma_{loc}, \Gamma_{glob})$  as tabulated in table 1.1.

### 1.1.2 Results and Discussion

The investigation of the Lamb-Oseen vortex problem is divided into three parts. The first part of the investigation concerns with comparing several stages of the hybrid coupling, defined in section 1.1.2. We compared the uncoupled scheme with the one-way coupled scheme and with the fully coupled scheme. These successive coupling investigation helped us determine the source of the error, and furthermore quantify the errors in the coupling process. The second part of the investigation, section 1.1.2 focuses on importance of

**Table 1.1:** Summary of the parameters for the Lamb-Oseen vortex evolution.

Parameters	Value	Unit	Description
$\Gamma_c$	1	$\text{m}^2 \text{s}^{-1}$	Core strength
$\Omega$	$[-0.5, 0.5] \times [-0.5, 0.5]$	m	Eulerian domain bounds
$\nu$	0.001	$\text{kg s}^{-1} \text{m}^{-1}$	Kinematic viscosity
$\tau$	100	s	Lamb-Oseen time constant
$\lambda$	1	-	Overlap ratio
$h$	0.01	m	Nominal blob spacing
$(\Gamma_{loc}, \Gamma_{glob})$	$(1 \times 10^{-14}, 1 \times 10^{-14})$	-	Population control threshold
$h_{grid}$	0.007 to 0.016	m	FE cell diameter span
$N_{\text{cells}}$	26448	-	Number of mesh cells
$\Delta t_L$	0.001	s	Lagrangian time step size
$\Delta t_E$	0.001	s	Eulerian time step size
$k_E$	1	-	Eulerian sub-steps
$N_{\text{t-steps}}$	1000	-	Number of time integration steps
$t$	0 to 1	s	Simulation time span
$d_{bdry}$	$2 \cdot h$	m	Interpolation boundary offset

conservation of circulation that was discussed in section ???. The results of the non-conserved and conserved scheme are compared to conclude the importance of conservation of circulation. During these two investigations, the parameters tabulated in table 1.1 are used.

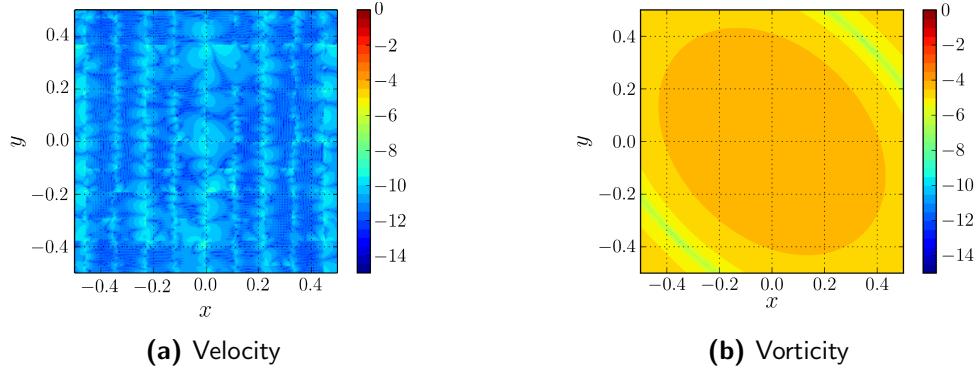
The third and final investigation is dedicated to the parameter sensitivity analysis, section 1.1.2. The parameters that determine the spatial and temporal discretization of the scheme is investigated to verify the convergence of scheme.

### Uncoupled vs. One-way Coupled vs. Fully Coupled

The several stages of the hybrid coupling with the fully Eulerian test case, to verify the implementation of the hybrid algorithm. The three stages of the coupling are as follows:

- **Uncoupled:** The uncoupled test case involves only the Eulerian solver and serves as a benchmark to quantify the error in the coupling. The boundary conditions are determined directly from the analytical formulation, equation ??.
- **One-way coupled:** The one-way coupled test case is a partially coupled hybrid test case where the Eulerian method is evolved using the Lagrangian solution. The correction of the Lagrangian solution is not performed in this scenario. Thus, this case helped us determine the error in the evolution of the Eulerian method using the Lagrangian solution.
- **Fully coupled:** The fully coupled test case performs the full coupling strategy described in section ???. The Eulerian method is evolved using the Lagrangian solution. At the end of each time step, the Lagrangian solution inside the interpolation

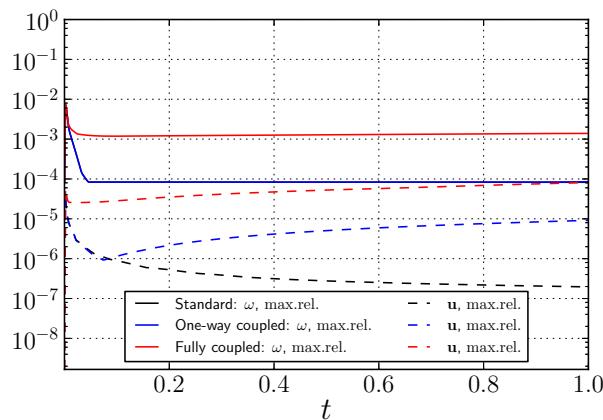
domain  $\Omega_{int}$ , figure 1.1, is corrected. This test case helped us quantify the error in transferring the Eulerian solution to the Lagrangian method.



**Figure 1.2:** Initial relative error at  $t = 0$  inside the Eulerian domain  $\Omega_E$ . The figure depicts (a) the relative error in velocity  $\mathbf{u}$ , and (b) the relative error in vorticity  $\omega$ .

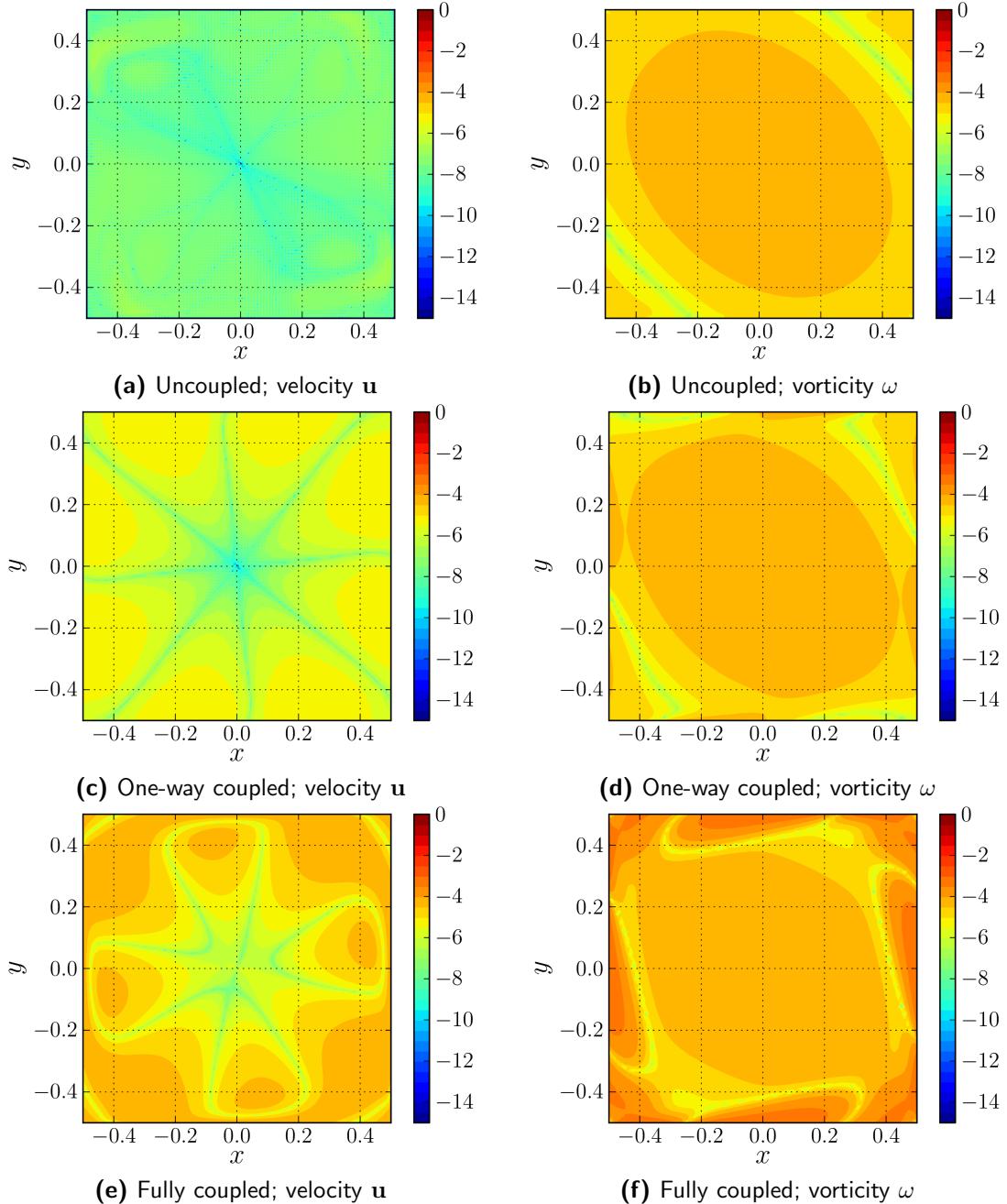
Figure 1.2 depicts the initial relative error in velocity and vorticity inside the Eulerian domain  $\Omega_E$ . The relative error in velocity is near machine epsilon  $\epsilon \leq 10^{-10}$ , but the error in vorticity is in the order of  $10^{-5}$ . A similar observation was made in section ??, and it concluded that the source of the error is the projection error of the finite element when determining the vorticity  $\omega$  from velocity  $\mathbf{u}$ , described in section ??.

The simulation was evolved from  $t = 0$  to  $t = 1$  with  $N_{t-steps} = 1000$  Lagrangian time steps with  $k_E = 1$  Eulerian sub-steps. A detailed summary of the time step parameters is tabulated in table 1.1. Figure 1.3 shows the evolution of maximum relative error in vorticity  $\omega$  and velocity  $\mathbf{u}$  of the uncoupled, one-way coupled and the fully coupled cases inside the Eulerian domain  $\Omega_E$  w.r.t. the analytical solution, equation ???. The initial observation shows that the error in velocity is two to three orders of magnitude lower than the error in vorticity (due to the projection error). The figure shows that the uncoupled



**Figure 1.3:** Evolution of the maximum relative error in velocity (dashed) and the maximum relative error in vorticity (solid), equation ??, from  $t = 0$  to  $t = 1$ , using the parameters tabulated in table 1.1. The figure compares uncoupled case (black) vs. the one-way coupled case (blue) vs. the fully coupled case (red).

scheme has the lowest error in vorticity and velocity. As the boundary condition is directly obtained from the analytical solution, the error only arises from the FE discretization of the Eulerian method. As the time progresses, the error in velocity converges near  $10^{-7}$  and the error in vorticity converges near  $10^{-4}$ .



**Figure 1.4:** Plot of the relative error in velocity (*left*) and relative error in vorticity (*right*) in the Eulerian domain  $\Omega_E$  at  $t = 1$ . The figure compares the error between (a)(b) the uncoupled, (c)(d) the one-way coupled, and (e)(f) the fully coupled cases.

The one-way coupled case shows an increase in the relative error in velocity field  $\mathbf{u}$  inside the Eulerian domain  $\Omega_E$ . However, the difference is negligible at the initial stages of the simulation. This states that the analytical solution was well represented by the vortex blobs with a small discretization error. However by  $t = 1$ , the error in velocity increases by two orders of magnitude from  $10^{-7}$  to  $10^{-5}$ , w.r.t to the uncoupled scheme. This implies that the source of the error is due to the time integration error of the Lagrangian method. In section ??, we observed similar trend of growth in error due to time-marching of the vortex blobs.

The fully coupled case demonstrates that there is an additional increase in the error. Unlike the one-way coupled case, the increase in the error is observed from the start of the simulation, implying that the increase in the error is due to the correction of the strengths of the vortex blobs. In section ??, we discussed that the re-initialization of the vortex blobs introduces smoothing error in the vorticity field (i.e the Gaussian blurring of the vorticity field). This causes the Lagrangian solution to further deviate from the analytical solution of the Lamb-Oseen vortex. The consequence of this was that at  $t = 1$ , the error in vorticity  $\epsilon_\omega$  increased from  $10^{-4}$  to  $10^{-3}$  and the error in velocity increased from  $10^{-5}$  to  $10^{-4}$ , w.r.t the one-way coupled case.

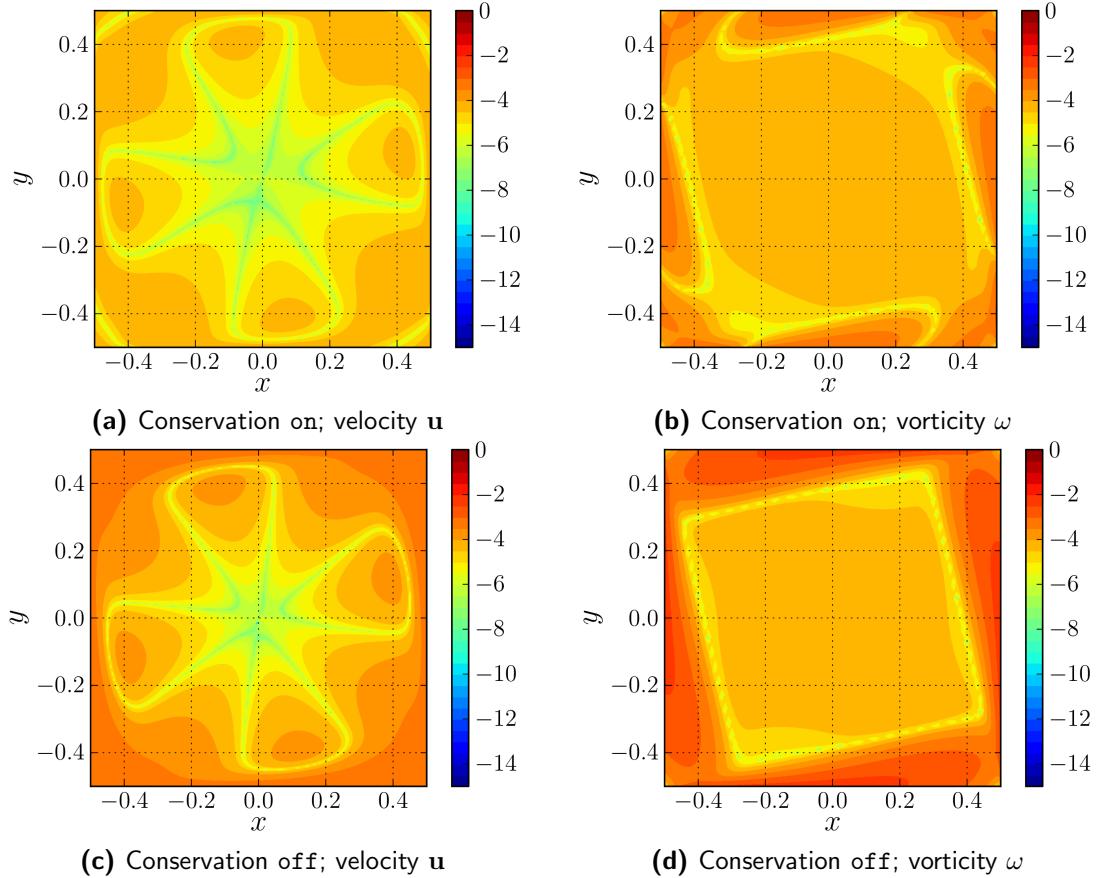
Figure 1.4 shows the relative error in velocity and vorticity inside the Eulerian domain  $\Omega_E$  at  $t = 1$ , for the three stages of coupling. We observe that there is an increase in error when going from the uncoupled scheme to the one-way coupled scheme to the fully coupled. Comparing the uncoupled scheme to the one-way coupled scheme, an increase in error is observed at the boundary of the Eulerian domain  $\Sigma_d$ . Comparing the one-way coupled to fully coupled case, we see that there is an additional increase in the error from the boundary. Therefore, the artificial vorticity generated from the boundary is due to the mismatch in the solutions of the Eulerian and the Lagrangian method. A larger mismatch in the solution will introduce strong artificial vorticity from the boundary.

The strength of the artificial vorticity at the boundary is proportional to the error in the coupling and to ensure an accurate coupling scheme, we have ensure this vorticity does not corrupt the characteristics of the original vorticity field. Therefore, we modified the discretization of the Lagrangian field such that the artificial vorticity generated from the boundary is less the threshold of influence (i.e  $< 1\%$  of the maximum vorticity  $\max\{\omega\}$ ).

### Conservation of circulation

The approach for ensuring conservation of circulation during the coupling process was discussed in section ???. To validate the importance of conservation of circulation, we ran two simulation with and without the conservation of circulation during the transfer of vorticity from the Eulerian method to the Lagrangian method. The control variables of the simulation are the parameters tabulated in table 1.1.

Figure 1.5 compares the error in the Eulerian domain  $\Omega_E$  of the coupling approach without the conservation circulation against the approach satisfying the conservation of circulation, at  $t = 1$ . We see that the scheme without the conservation has significantly larger error than the scheme with conservation. The maximum error is near the Eulerian boundary  $\Sigma_d$  and shows an increase in the artificial vorticity emanating from the bound-

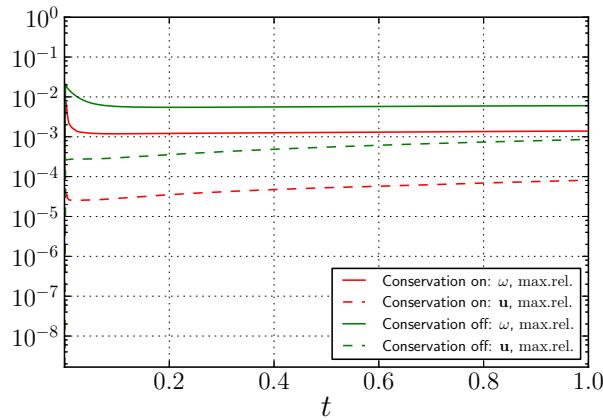


**Figure 1.5:** Plot of the relative error in velocity (left) and the relative error in vorticity (right) in the Eulerian domain  $\Omega_E$  at  $t = 1$ . The figure compares the error between **(a)(b)** without conservation of circulation, and **(c)(d)** with the conservation of circulation.

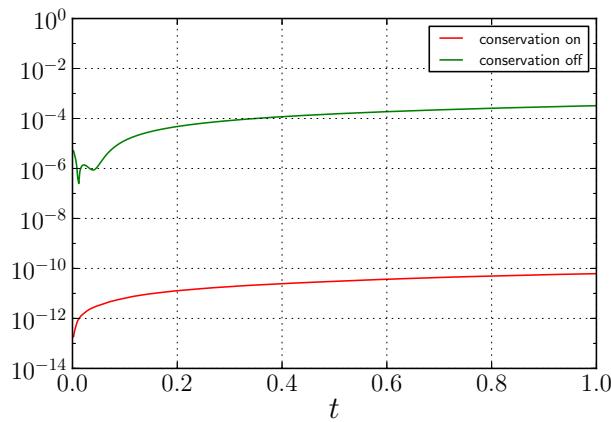
ary due to the larger mismatch in the solutions, figure 1.5d. However, when we ensure that circulation is conserved, figure 1.5b, the boundary produces significantly less error.

Figure 1.6 shows the evolution of the maximum relative error from  $t = 0$  to  $t = 1$ , comparing the results of with and without the conservation of circulation. Observing the difference in the relative error in velocity and vorticity, we see that the scheme without the conservation of circulation produces larger error at all times  $t$ . At  $t = 1$ , we observe that scheme without the conservation of circulation has a relative error in vorticity near  $10^{-2}$ , whereas with the conservation of circulation, the relative error is an order of magnitude lower, reaching only  $10^{-3}$ . Similarly for velocity, the scheme without the conservation of circulation has the relative error approaching  $10^{-3}$ , whereas with the conservation enabled, the error only reaches  $10^{-4}$ .

Figure 1.7 shows the change in total circulation from  $t = 0$  to  $t = 1$  for the non-conserved and conserved scheme. It is apparent that without the conservation of circulation, the error in total circulation is significantly larger and approaches  $10^{-3}$ . If the circulation is not conserved explicitly, the transfer of vorticity from the Eulerian method to the Lagrangian method introduces additional error in total circulation. By ensuring conservation of circulation, as described in section ??, we see that the error in total circulation is significantly



**Figure 1.6:** Plot of the maximum relative error in vorticity  $\epsilon_\omega$  [ - -, dashed] and maximum relative error in vorticity  $\epsilon_u$  [ —, solid], equation ??, from  $t = 0$  to  $t = 1$ , using the parameters tabulated in table 1.1. The figure compares the coupling scheme with conservation of circulation (red) vs. the coupling scheme without conservation of circulation (green).



**Figure 1.7:** Plot of the error in total circulation  $\epsilon_\Gamma$  of the Lagrangian method from  $t = 0$  to  $t = 1$ . The figure compares the scheme with conservation of circulation [ —, solid red], and the scheme without conservation of circulation [ —, solid green].

smaller, near  $10^{-10}$ .

In summary, we have determined that to ensure minimum error during the transfer of Eulerian solution to the Lagrangian solution, we have to ensure that the total circulation of the Lagrangian method is conserved.

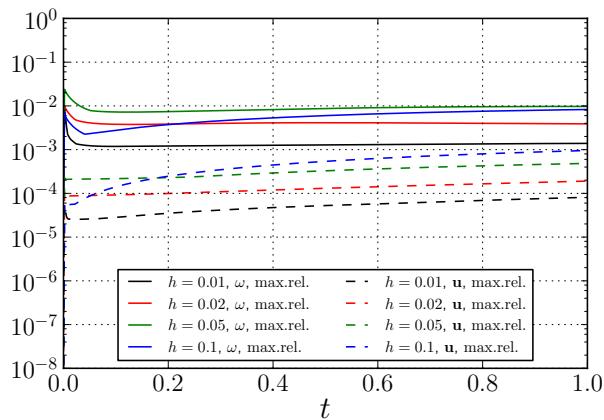
### Parameter sensitivity analysis

The parameter sensitivity analysis is the last stage of the Lamb-Oseen vortex investigation. The Lamb-Oseen vortex test case was ideal to determining the effects of the temporal and the spatial discretization of the hybrid method on the accuracy of the coupling.

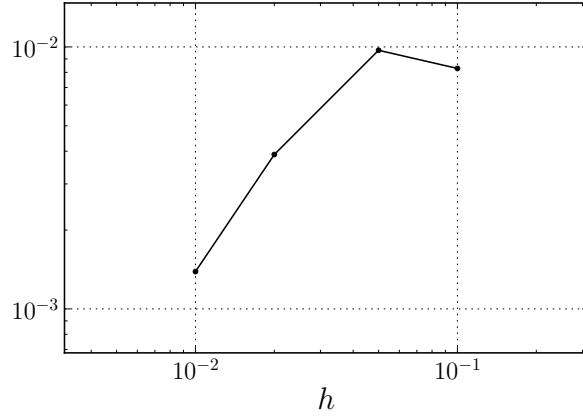
To investigate the effects of the spatial discretization on the accuracy of the coupling, we ran several test cases varying the nominal blob spacing  $h$ , and test cases varying the overlap ratio  $\lambda$ . To investigate the effects of temporal discretization, we modified the Lagrangian time step size  $\Delta t_L$  w.r.t the Eulerian time step size  $\Delta t_E$ . The control variables of the simulations are the ones tabulated in table 1.1.

Figure 1.8 shows the impact of varying the nominal blob spacing  $h$  on the coupling. The maximum relative error in vorticity and the maximum relative error in velocity is plotted from  $t = 0$  to  $t = 1$  for nominal blob spacing  $h = [0.01, 0.02, 0.05, 0.1]$ . The figure shows that increasing the spatial resolution of the Lagrangian method reduces the error. At  $t = 1$ , the minimum error is observed for  $h = 0.01$  with the relative error in velocity at  $10^{-4}$  and the relative error in vorticity at  $10^{-3}$ . The maximum relative error is observed for  $h = 0.1$ , with  $10^{-3}$  for relative error in velocity and  $10^{-2}$  for relative error in vorticity. This implies that the growth in error is of order one. Figure 1.9 shows the variation in the maximum relative error in vorticity at  $t = 1$  for various nominal blob spacing  $h$ . The figure indeed shows that the change in error due to the spatial discretization is of order one.

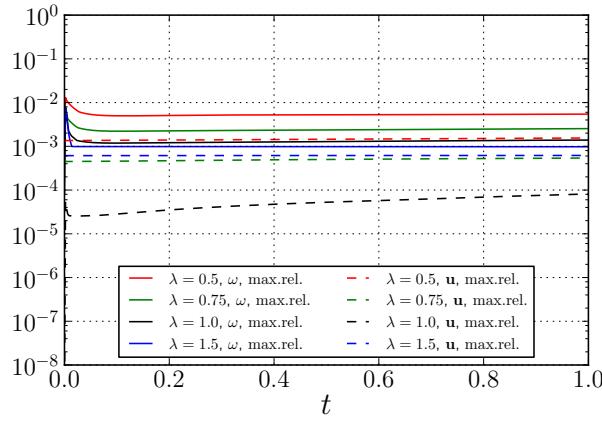
Figure 1.10 compares the evolution of the maximum relative error for various overlap ratios,  $\lambda = [0.5, 0.75, 1.0, 1.5]$ . We see that the minimum error in velocity and vorticity is



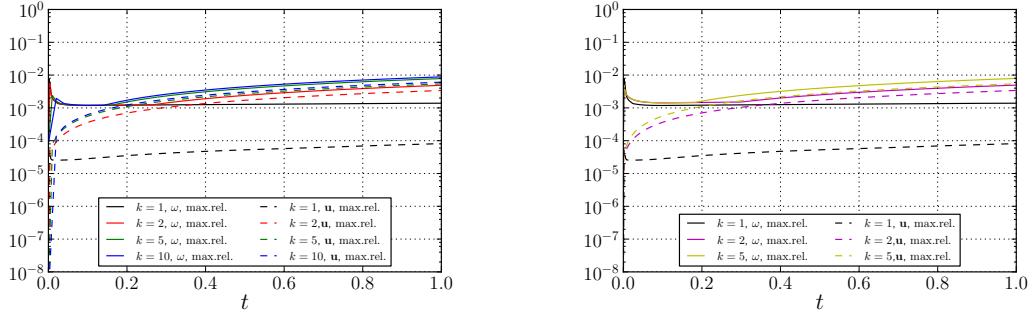
**Figure 1.8:** Evolution of the maximum relative error for various nominal blob spacing  $h = [0.01, 0.02, 0.05, 0.1]$  from  $t = 0$  to  $t = 1$ . The figures shows the maximum relative error in vorticity [ - -, dashed] and maximum relative error in vorticity [ —, solid].



**Figure 1.9:** Convergence of the maximum relative error in vorticity due to the nominal blob spacing  $h = [0.01, 0.02, 0.05, 0.1]$ .



**Figure 1.10:** Evolution of the maximum relative error for various overlap ratios  $\lambda = [0.5, 0.75, 1.0, 1.5]$  from  $t = 0$  to  $t = 1$ . The figures shows the maximum relative error in vorticity [ - -, dashed] and the maximum relative error in vorticity [ —, solid].



(a)  $\Delta t_L = [0.001, 0.002, 0.005, 0.01]$ ,  $\Delta t_E = 0.001$       (b)  $\Delta t_E = [0.001, 0.0005, 0.0002]$ ,  $\Delta t_L = 0.001$

**Figure 1.11:** Evolution of the maximum relative error from  $t = 0$  to  $t = 1$  for various number of Eulerian sub-steps  $k_E = [1, 2, 5, 10]$ , modifying (a) the Lagrangian time step  $\Delta t_L$ , and (b) the Eulerian time step  $\Delta t_E$ . The figures shows the maximum relative error in vorticity [ - -, dashed] and maximum relative error in vorticity [ —, solid].

observed for the overlap ratio  $\lambda = 1$ . As we move from this value, an increase in the error is observable. In section ??, we determined that to reduce the Gaussian blurring of the vorticity field from the Gaussian vortex kernels, we require an overlap ratio  $\lambda = 1$  and a small nominal blob spacing  $h$ . The parameter sensitivity analysis on the spatial discretization validates this observation and states that to ensure minimum error in coupling, these criteria has to be satisfied.

Figure 1.11 shows the impact of varying the temporal discretization of the Lagrangian method and the Eulerian method w.r.t each other. The relation of the Eulerian time step size  $\Delta t_E$  to the Lagrangian time step size  $\Delta t_L$  is described in section ???. Figure 1.11a shows the effect of modifying the Lagrangian time step size  $\Delta t_L$  w.r.t to a fixed Eulerian time step size. With  $\Delta t_E = 0.001$  and the number of Eulerian sub-steps  $k_E = [1, 2, 5, 10]$ , we have  $\Delta t_L = k_E \cdot \Delta t_E = [0.001, 0.002, 0.005, 0.01]$ . Similarly, figure 1.11b shows the effect of modifying the Eulerian time step size  $\Delta t_E$  w.r.t to the Lagrangian time step. With  $\Delta t_L = 0.001$  and  $k_E = [1, 2, 5]$ , we have  $\Delta t_E = \Delta t_L/k_E = [0.001, 0.0005, 0.0002]$ . We see that the minimum error occurs when the time steps match (i.e  $\Delta t_L = \Delta t_E$ ). However if increase the number of Eulerian time steps from  $k_E = 1$  to  $k_E = 2$ , there an substantial increase in the relative error in velocity. This observation states that the linear interpolation used for sub-stepping process, has potential for improvement. A possible solution might be to employ a higher-order interpolation method for determining the Eulerian Dirichlet boundary condition at the sub-steps.

### 1.1.3 Conclusion

In section 1.1.2, we observed that moving from uncoupled to one-way coupled case, increases the relative error in velocity. The growth in error was mainly due to the time integration error of the Lagrangian method. When moving from one-way coupled to fully coupled scheme, there is a tangible increase in the relative error in vorticity and an additional increase in the error in velocity. The increase in this error was due to the re-initialization of the vortex blobs introducing an additional smoothing error at each correction step.

In section 1.1.2, we observed that conservation of circulation is vital in ensuring an accurate coupling strategy. The transfer of vorticity from the Eulerian method to the Lagrangian method, must be performed with a focus on the conservation of circulation, to ensure that the artificial vorticity at the Eulerian Dirichlet boundary  $\Sigma_d$  is minimal.

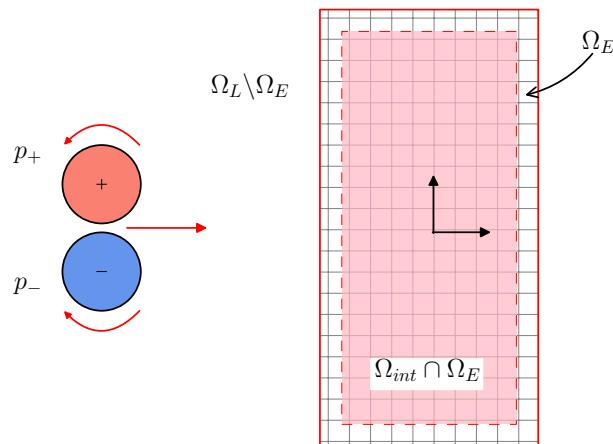
In section 1.1.2, we investigated the impact of varying the spatial and temporal discretization on the accuracy of the coupling. We determined there is an increase in error, if the Lagrangian method is spatially under-resolved w.r.t to the Eulerian method. An overlap ratio of  $\lambda = 1$  was shown to have the minimum error during the coupling, as it ensures minimum Gaussian blurring. Varying the number of Eulerian sub-steps  $k_E$ , showed that the linear interpolation for the Dirichlet boundary condition is potential source of improvement.

## 1.2 Clercx-Bruneau Dipole Convection

In section 1.1, we determine the effects of transferring the Lagrangian solution to the Eulerian method, and transferring the Eulerian solution back to the Lagrangian method on an Lamb-Oseen vortex test case. However, an important aspect of domain decomposition method is the entering and the exiting of a vortex core in the Eulerian domain  $\Omega_E$ . We used the Clercx-Bruneau dipole [?] to simulate the convection of vortex an Eulerian domain  $\Omega_E$ .

### 1.2.1 Problem Definition

The hybrid domain decomposition of this investigation is depicted in figure 1.12. The Eulerian domain  $\Omega_E$  is finite with bounds  $[-0.25, 0.25] \times [-0.5, 0.5]$ . The Clercx-Bruneau dipole, defined by equation 1.12, is initialized outside the Eulerian domain  $\Omega_L \setminus \Omega_E$  at  $(x_1, y_1) = (-1, 0.1)$  and  $(x_2, y_2) = (-1, -0.1)$ , corresponding to the positive and negative cores, respectively. As the simulation progresses, the dipole convects along the  $x$ -axis, passing through the Eulerian domain.



**Figure 1.12:** (Not to Scale) The domain decomposition for the Clercx-Bruneau convection problem, with the positive pole located at  $p_+ = (x_1, y_1) = (-1, 0.1)$  and negative pole located at  $p_- = (x_2, y_2) = (-1, -0.1)$ . The parameters of the simulation are tabulated in table 1.2.

The Eulerian and the Lagrangian domain is discretized according to the parameters shown in table 1.2. The focus of this simulation is the entry and the exit of vorticity from in and out of the Eulerian domain and its impact on the solution. The simulation was first benchmarked using a Finite Element (FE) only simulation, and a Vortex Particle Method (VPM) only simulation, providing an basis for hybrid simulation. We can assume that the FE and VPM simulation are valid as we have already verified and validated their implementation in section ?? and ??, respectively.

To ensure that FE only simulation was valid, the Eulerian domain  $\Omega_E$  stretched up to the far-field of the dipole, where the vorticity and the induced velocity is zero. The Eulerian domain  $\Omega_E$  of the FE only simulation spanned  $[-3, 3] \times [-2, 2]$ . For a valid comparison,

**Table 1.2:** Summary of the parameters for the Clercx-Bruneau dipole convection problem.

Parameters	Value	Unit	Description
$\Omega_E$	$[-0.25, 0.25] \times [-0.5, 0.5]$	m	Eulerian domain bounds
$Re$	625	-	Reynolds number
$U$	1	$\text{m s}^{-1}$	Characteristic velocity
$W$	1	m	Characteristic Length
$\nu$	$1.6 \times 10^{-3}$	$\text{kg s}^{-1} \text{ m}^{-1}$	Kinematic viscosity
$(x, y)_{1,2}$	$(-1, \pm 0.1)$	m	Initial location of the monopoles
$\omega_e$	299.528385375226 <sup>a</sup>	-	Characteristic vorticity of the monopole
$\lambda$	1	-	Overlap ratio
$h$	0.005	m	Nominal blob spacing
$h_{grid}$	$\approx 0.007$	m	FE cell diameter
$N_{\text{cells}}$	40000	-	Number of mesh cells
$\Delta t_L$	$2.5 \times 10^{-4}$	s	Lagrangian time step size
$\Delta t_E$	$2.5 \times 10^{-5}$	s	Eulerian time step size
$k_E$	10	-	Eulerian sub-steps
$N_{\text{t-steps}}$	2800	-	Number of time integration steps
$t$	0 to 0.7	-	Simulation time
$d_{bdry}$	$2 \cdot h$	m	Interpolation boundary offset

<sup>a</sup> Obtained from Renac et al. [?]

these benchmark simulations followed similar parameters as the ones tabulated in table 1.2.

### 1.2.2 Results and Discussion

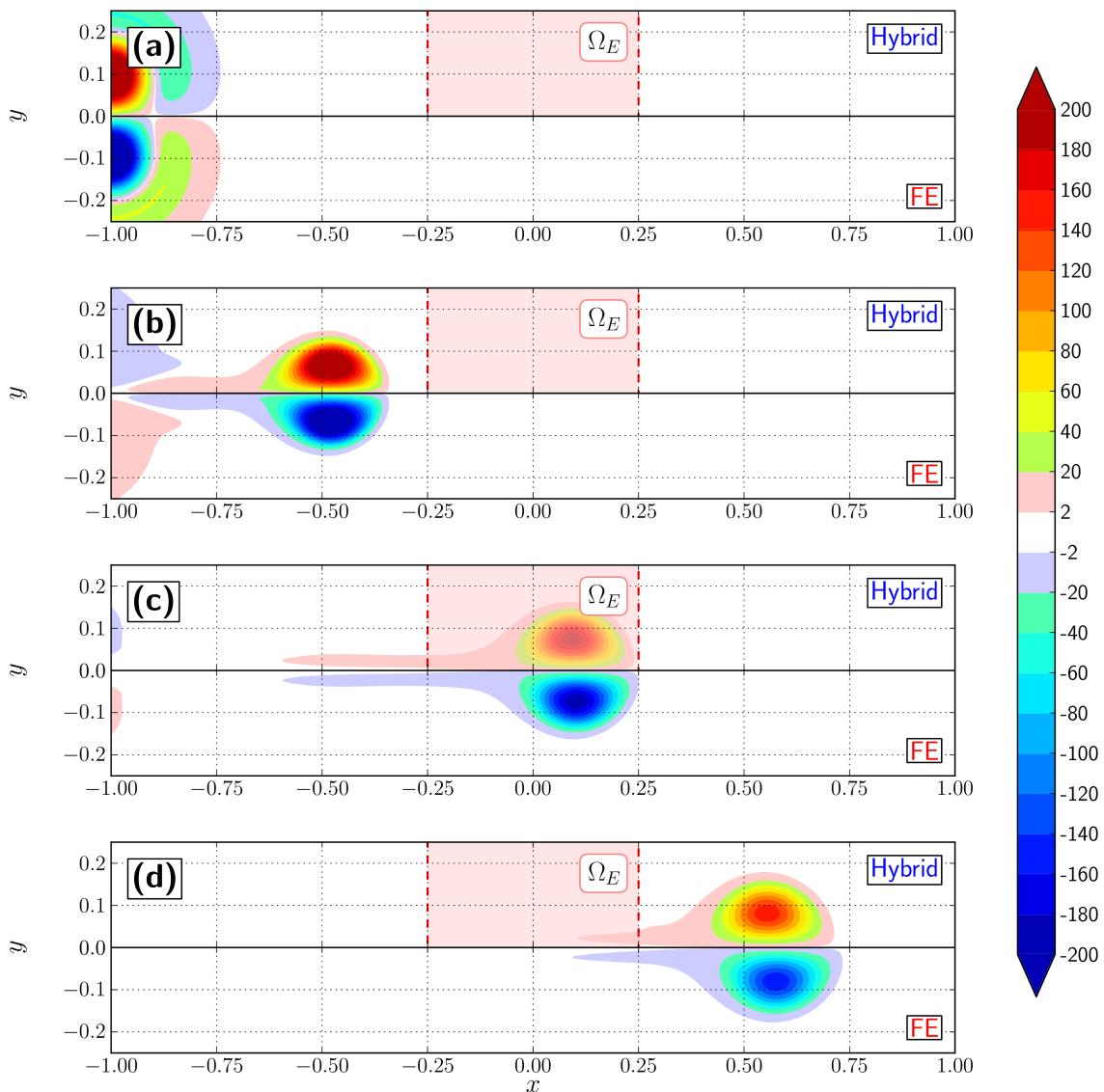
Figure 1.13 compares the vorticity field of the FE simulation and the hybrid simulation at various instances,  $t = [0, 0.2, 0.4, 0.6]$ . The top half of each subplot belongs to the hybrid simulation, whereas the bottom half to the FE only simulation. It was determined that the cores of the dipole enters the Eulerian domain at  $t = 0.26$  and exits the domain at  $t = 0.45$ . The figure shows that solution of both simulation matches up to the entry of the dipole, figure 1.13a and figure 1.13b. However in figure 1.13c, the solutions start to deviate, and at  $t = 0.7$ , figure 1.13d, it is apparent that the dipole in the hybrid method is lagging w.r.t to the FE only simulation. This would imply that the passage of the vortex through the domain has a stalling influence on the vorticity evolution.

To investigate further on this influence of the passage, we analyzed variation in maximum vorticity  $\omega_{max}$ . Figure 1.14a shows evolution of the maximum vorticity  $\omega_{max}$  from  $t = 0$  to  $t = 0.7$  in the Eulerian and the Lagrangian sub-domain of the hybrid simulation, the FE only simulation, and the VPM only simulation. We observe there is a slight difference in the maximum vorticity for the FE only and the VPM only simulation, which increases as the time progresses.

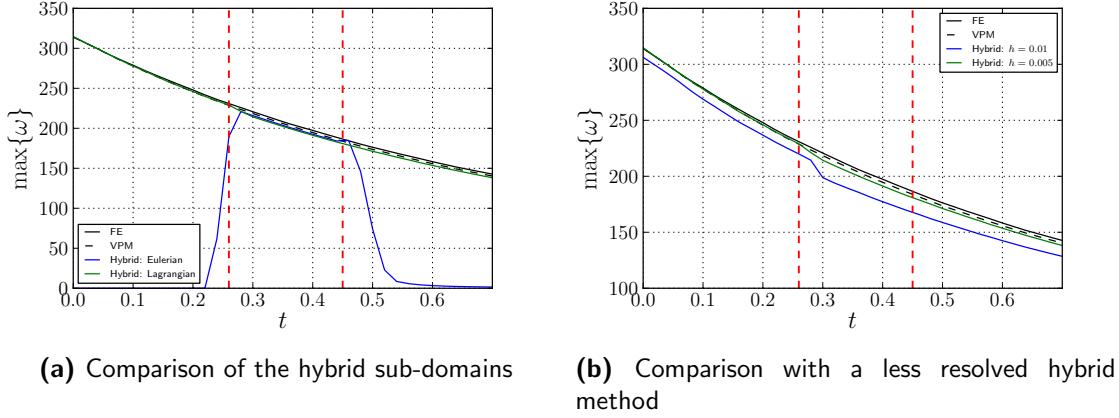
At  $t = 0.26$ , the maximum vorticity in the Eulerian sub-domain starts to increase, signifying the entering of the vortex core. Similarly, at  $t = 0.45$ , the maximum vorticity starts to decrease, signifying the exiting of the vortex core. As the dipole enters the sub-domain,

there is slight drop in the maximum vorticity. Similarly, as the dipole exits, there is a slight peak in the solution of the Eulerian sub-domain. The explanation to the increases in the vorticity in the Eulerian domain is the generation of the artificial vorticity. When the core of the dipole is right next to the Eulerian boundary  $\Sigma_d$ , the influence of the vorticity on the boundary is larger. However, with the implementation of Stocks interpolation tolerance at the boundary, this artificial vorticity is somewhat neglected.

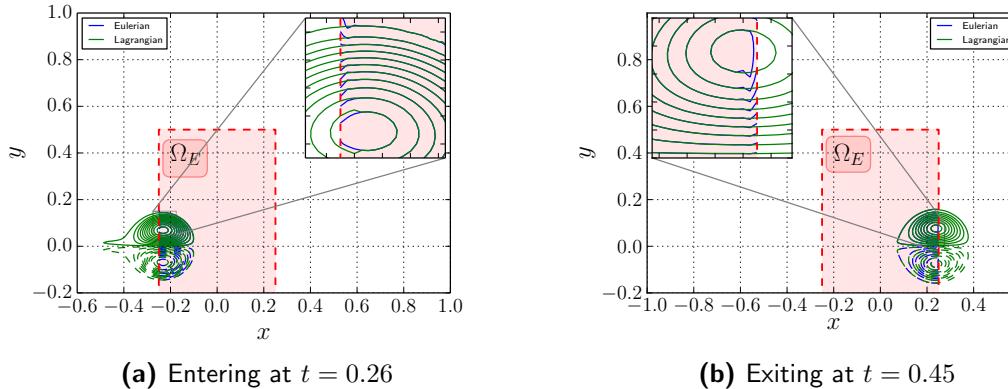
Figure 1.15a shows a vorticity contour plot of the Lagrangian method and the Eulerian method, at  $t = 0.28$  when the dipole has just entered the Eulerian domain. We see that there is a mismatch in the vorticity at the boundary of the Eulerian domain. Similarly, there is a slight mismatch in the vorticity field when the dipole leaves the Eulerian domain,



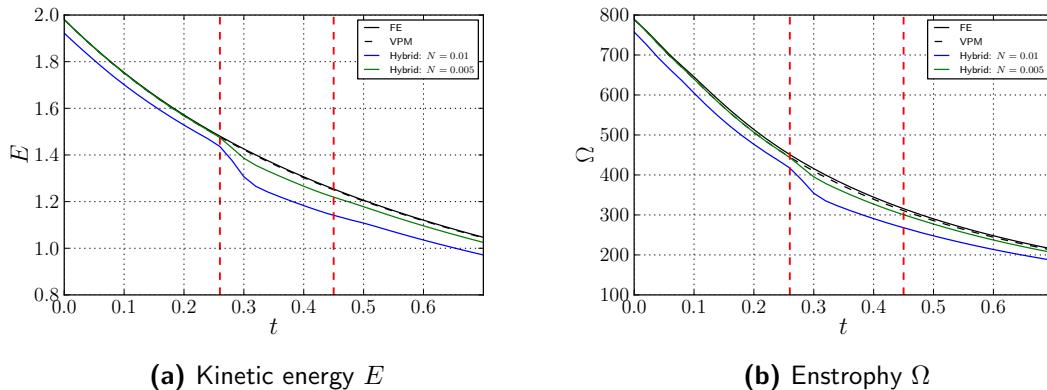
**Figure 1.13:** Plot of the Clercx-Bruneau dipole at  $t = [0, 0.2, 0.4, 0.7]$  using parameters tabulated in table 1.2. The figure compares the hybrid simulation (top halves) against the FE only simulation (bottom halves).



**Figure 1.14:** Evolution of the maximum vorticity  $\max\{\omega\}$  from  $t = 0$  to  $t = 0.7$ . The solutions are compared with the benchmark results of FE only [—, solid black], and VPM only [- -, dashed black] simulations. The figure depicts (a) the maximum vorticity in the Eulerian and the Lagrangian sub-domain of the hybrid method, and (b) the maximum vorticity of hybrid method with nominal blob spacing  $h = 0.01$  and  $h = 0.005$ .



**Figure 1.15:** Vorticity contour plots of the dipole with levels ..., -50, -30, -10, 10, 30, 50, ... of the Eulerian and the Lagrangian sub-domains. The figure highlights the effect of the artificial vorticity at the boundary of the Eulerian domain.



**Figure 1.16:** Evolution of the (a) kinetic energy  $E$  and (b) enstrophy for the nominal blob spacing  $h = 0.01$  and  $h = 0.005$ .

figure 1.15b.

A simulation with lower Lagrangian resolution was ran to verify this theory. Figure 1.14b compares the evolution of maximum vorticity  $\omega_{max}$  for nominal blob spacing  $h = 0.01$  and  $h = 0.005$ . The less resolved simulation shows a larger drop in maximum vorticity during the entry of the dipole. However, at  $t = 0.45$ , the exiting of the dipole seems has no effect on the maximum vorticity, due to the interpolation tolerance from the boundary.

The evolution of the kinetic energy  $E$  and the enstrophy  $\Omega$  shows the same behavior, figures 1.16a and 1.16b, respectively. It shows that during the entry there is larger change in the kinetic energy and the enstrophy of the flow. The artificial vorticity causes an increased diffusion of the dipole. With the reduced strength of the vortex core, the dipole is weaker in energy and travels a shorter distance, as observed in figure 1.13. The effect is more sever for a lower resolved Lagrangian method, as seen for the simulation with  $h = 0.01$ .

### 1.2.3 Conclusion

In conclusion, we see that a high resolution discretization of the Lagrangian method inside the Eulerian domain  $\Omega_L \cap \Omega_E$  is paramount for accurate transfer of information to and from the Eulerian method. For a lower resolved Lagrangian method in this region introduces artificial vorticity at the boundary of the Eulerian domain  $\Sigma_d$ , corrupting the solution of the coupling.

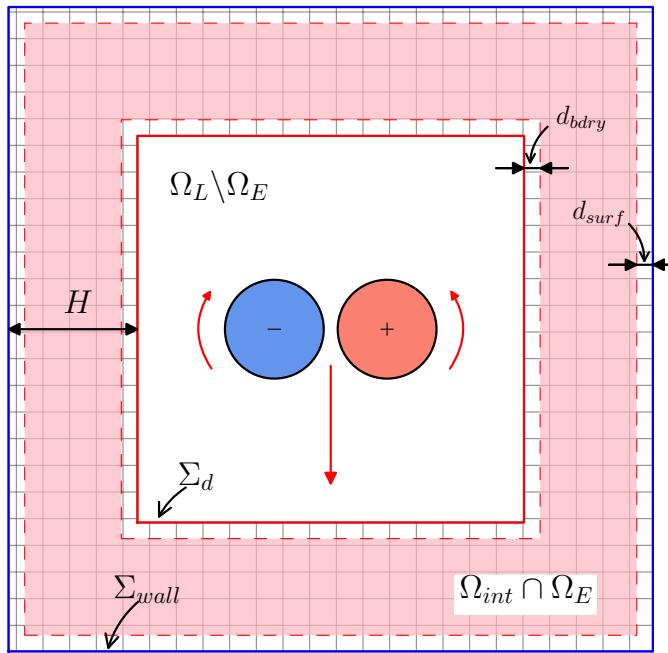
## 1.3 Clercx-Bruneau Dipole Collision

In this section, we study the Clercx-Bruneau dipole colliding with a solid wall. A Finite Element (FE) only investigate was first performed in section ?? to form a benchmark for further investigation, validated against the study of Clercx and Bruneau [?]. The main goal of the Clercx-Bruneau dipole collision test case was to investigate how the hybrid method deal with a wall bounded problem.

### 1.3.1 Problem Definition

A detailed description of the Clercx-Bruneau dipole collision problem is given in section ???. Figure 1.17 shows the step-up of the hybrid simulation, with the Eulerian sub-domain  $\Omega_E$  resolving the near-wall region, and the Lagrangian sub-domain domain resolving the complete fluid domain. The fluid domain is bounded by the no-slip wall  $\Sigma_{wall}$  (shown in blue). The Eulerian domain  $\Omega_E$  extends from the wall  $\Sigma_{wall}$  to the boundary  $\Sigma_d$ , where the velocity boundary condition from the Lagrangian method is prescribed. The parameters of the simulation are tabulated in table 1.3.

As we are dealing with the wall-bounded problem, we require the vortex panel method to enforce the boundary condition in the Lagrangian method. In section ??, we described the decomposition of the Lagrangian domain  $\Omega_L$  to the vortex blob domain  $\Omega_b$  and the vortex panel domain  $\Omega_p$ . Therefore, this decomposition was applied to this problem.



**Figure 1.17:** [Not to Scale] The domain decomposition for the Clercx-Bruneau dipole collision problem, with the positive pole at  $p_+ = (x_1, y_1) = (0.1, 0)$  and negative pole at  $p_- = (x_2, y_2) = (-0.1, 0)$ . The parameters of the simulation are tabulated in table 1.3.

**Table 1.3:** Summary of the parameters for the Clercx-Bruneau dipole collision.

Parameters	Value	Unit	Description
$\Omega$	$[-1, 1]^2$	m	Extend of Eulerian domain from wall $\Sigma_d$
$H$	0.2	m	Eulerian domain width
$Re$	625	-	Reynolds number
$U$	1	$\text{m s}^{-1}$	Characteristic velocity
$W$	1	m	Characteristic Length
$\nu$	$1.6 \times 10^{-3}$	$\text{kg s}^{-1} \text{m}^{-1}$	Kinematic viscosity
$(x, y)_{1,2}$	$(\pm 0.1, 0)$	m	Initial location of the dipole
$\omega_e$	299.528385375226 <sup>a</sup>	-	Characteristic vorticity of the monopole
$\lambda$	1	-	Overlap ratio
$h$	0.003	m	Nominal blob spacing
$N_{\text{panels}}$	400	-	Number of panels
$h_{\text{grid}}$	0.005 to 0.01	m	FE cell diameter
$N_{\text{cells}}$	58272	-	Number of mesh cells
$\Delta t_L$	$2.5 \times 10^{-4}$	s	Lagrangian time step size
$\Delta t_E$	$2.5 \times 10^{-5}$	s	Eulerian time step size
$k_E$	10	-	Eulerian sub-steps
$N_{\text{t-steps}}$	4000	-	Number of time integration steps
$t$	0 to 1	-	Simulation time
$d_{\text{bdry}}$	$2 \cdot h$	m	Interpolation domain offset from $\Sigma_d$
$d_{\text{surf}}$	$3 \cdot h$	m	Interpolation domain offset from $\Sigma_{\text{wall}}$

<sup>a</sup> Obtained from Renac et al. [?]

The dipole is initialized in the center of the domain, in the Lagrangian only domain  $\Omega_L \setminus \Omega_E$  at the locations  $(x, y)_{1,2}$ . As the simulation progresses, the dipole travels along the negative  $y$ -axis, entering the Eulerian domain  $\Omega$  and colliding with the no-slip wall  $\Sigma_{wall}$ .

### 1.3.2 Results and Discussion

Figure 1.18 shows the state of the dipole at  $t = [0, 0.2, 0.4, 0.6, 0.8, 1]$ . The figure compares the hybrid simulation (left half) with the FE only simulation (right half), investigated in section ???. Once the dipole enters the Eulerian domain, at  $t = 0.4$ , we observe that there is a slight difference in the solution. We observe a slight artificial vorticity emanating from the Eulerian boundary  $\Sigma_d$ , corrupting the vorticity field. The corruption of the vorticity field is apparent when observing the hybrid plot at  $t = 0.4$ , at the location  $x = -0.2$  and  $y = -0.8$ . The artificial vorticity is in order of  $< \pm 5$ . Note that with the maximum vorticity in the fluid near  $\max\{\omega\} = 300$  is near 2% of the maximum vorticity.

Figure 1.19 compares the vorticity contour at  $t = 1$  against the FE only simulation. We see there is a slight difference in the vorticity contour lines of hybrid solution, figure 1.19b. The shape of the contour lines near the wall is slightly different, and furthermore, the location of the core is also shifted slightly.

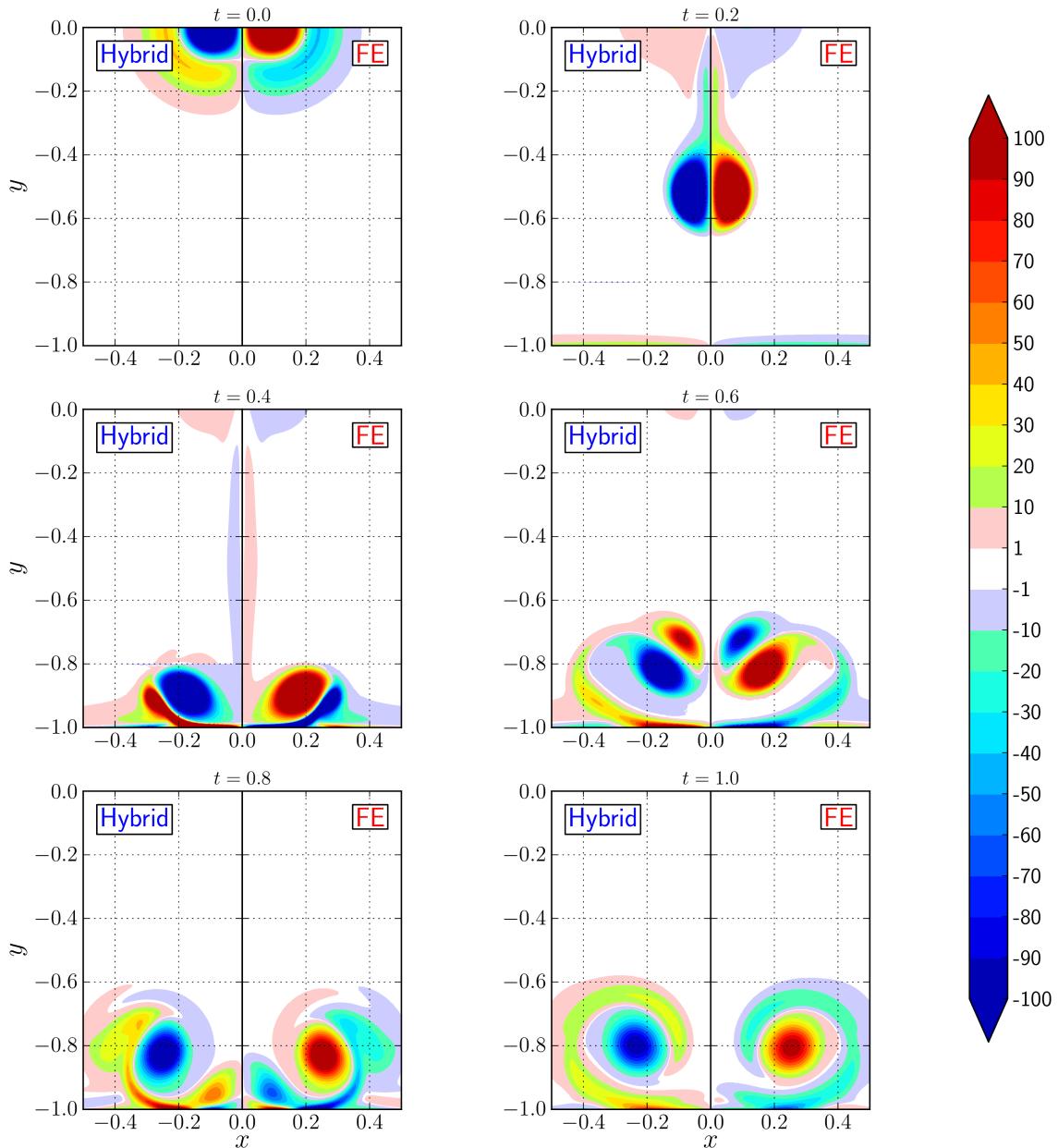
To investigate further on the cause of this difference, we studied the change in maximum vorticity  $\omega_{max}$ , the kinetic energy  $E$  and the enstrophy  $\Omega$ , and the palinstrophy  $P$ , shown in figure 1.20d. The variation in maximum vorticity, figure 1.20a, shows that the first peak in the hybrid is slightly lower than the FE only simulation, at  $t \approx 0.35$ . However, the second peak in vorticity, at  $t \approx 0.65$  is higher than the standard simulation. Between  $t = 0.4$  and  $t = 0.6$ , the dipole exits and re-enters the Eulerian domain, as seen in figure 1.18. The artificial vorticity generated during this time, causes a detrimental effect on overall solution.

Figure 1.20b shows that, as the dipole leaves the Eulerian domain  $\Omega_E$  from  $t = 0.4$ , the kinetic energy  $E$  reduces less, and is higher than the FE only simulation at  $t \leq 0.4$ . Therefore, the core that is leaving and re-entering the Eulerian domain  $\Omega_E$  has a higher kinetic energy  $E$ . This could be cause of deviation seen at  $t = 1$ , figure 1.19.

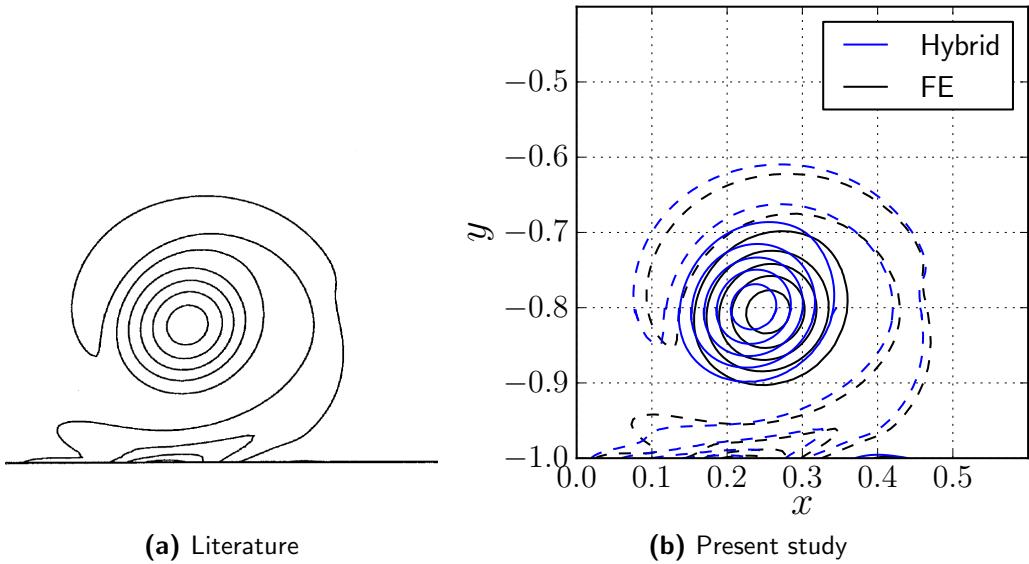
Figure 1.20c shows that the enstrophy  $\Omega$  matches reasonable well with the FE only simulation and we see that there is a slight difference at the peaks. Similarly, figure 1.20d, shows the variation in palinstrophy  $P$ . The solution stars to deviate from  $t \approx 0.5$ , after the vortex core re-enters the domain.

Figure 1.21 shows the vorticity at the boundary, along  $y = -1$ . We observe that for  $t = 0.4$  the solution matches, at  $t = 0.6$  the peak in vorticity is larger for the hybrid simulation, and for  $t = 1$ , the peak has a larger span. The increased kinetic energy  $E$  of the vortex could provide a possible explanation for this. With a higher kinetic energy, the wall generates stronger vorticity to repel the dipole.

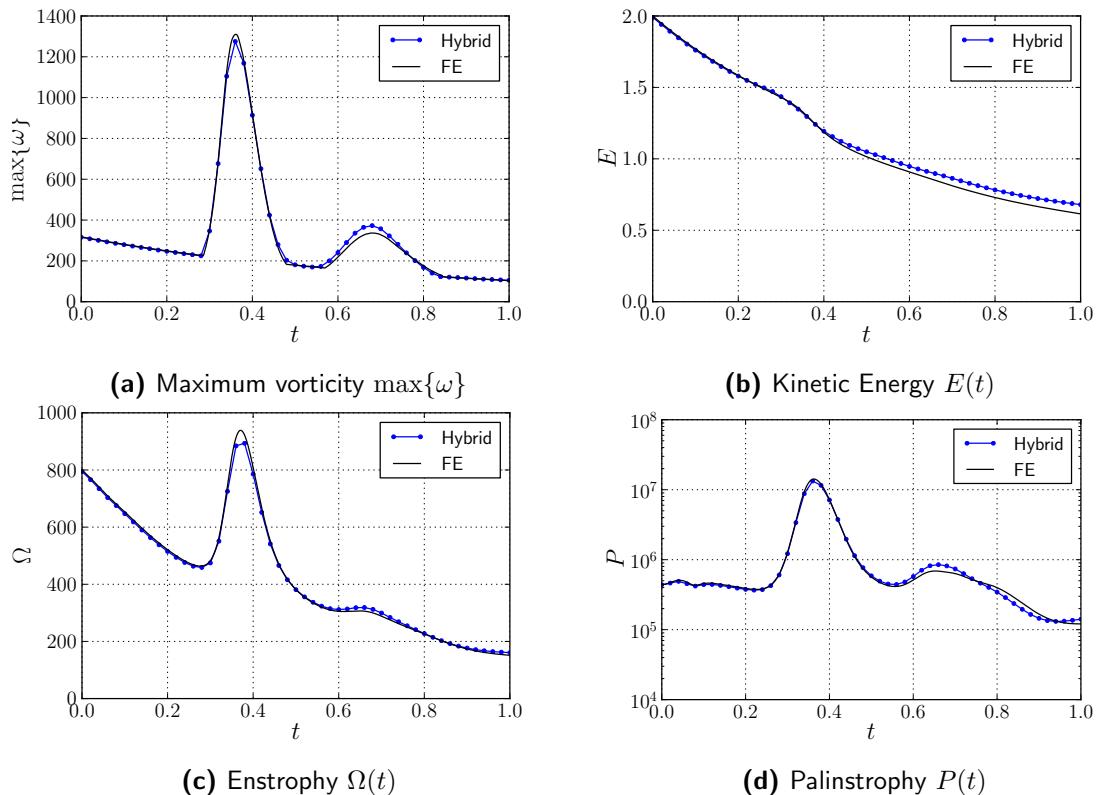
We investigated further with a higher resolved Lagrangian method, with a smaller nominal blob spacing  $h$ , a larger number of panels  $N_{panels}$ , and a smaller Lagrangian time step size  $\Delta t_L$ . However, these simulation did not provide significant improvement to the present results.



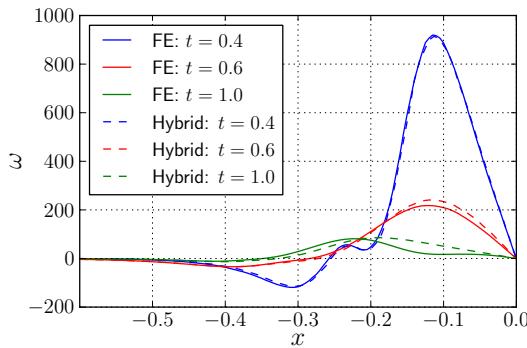
**Figure 1.18:** Plot of the dipole at  $t = [0, 0.2, 0.4, 0.6, 0.8, 1]$ , comparing the hybrid simulation (left half) and FE only simulation (right half).



**Figure 1.19:** Comparison of the vorticity contours at  $t = 1$ . The figure compares the plot obtained by **(a)** literature, Clercx and Bruneau [?], and **(b)** the present study, the hybrid and FE only simulation.



**Figure 1.20:** Variation in the fluid parameters from  $t = 0$  to  $t = 1$ . The figure compares the hybrid results [—, solid blue] with the FE only [—, solid black] results.



**Figure 1.21:** Compares the vorticity generated at the bottom-left wall ( $y = -1$ ,  $-0.6 \leq x \leq 0$ ) at  $t = 0.4$  [—, solid blue],  $t = 0.6$  [—, solid red] and  $t = 1$  [—, solid green].

### 1.3.3 Conclusion

In conclusion, we determined that there exists a slight difference in the geometry of the vorticity contours and the location of the dipole at the end of the simulation. The deviation of the dipole starts as the dipole enters the Eulerian domain. The entering and the re-entering process of the dipole introduces artificial vorticity from the Dirichlet boundary  $\Sigma_d$ , increasing the overall kinetic energy  $E$  of the problem. This intern has an influence on the position of the dipole at  $t = 1$ . Increasing the resolution of the Lagrangian solution only minimally increases the accuracy of the results.

## 1.4 Impulsively Started Cylinder at $Re = 550$

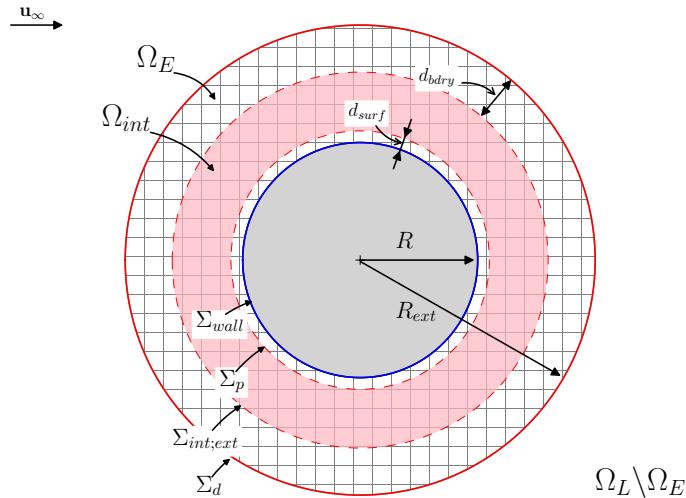
In this section, we will study the flow around an impulsively started cylinder at  $Re = 550$ . The purpose of this test case is to ensure we are able to correctly predict the forces acting on the body. In section ??, a FE only simulation was run, and we were able to determine the performance of the FE simulation w.r.t to the literature data provided by Koumoutsakos and Leonard [?], and the data provided by Rosenfeld et al. [?]. These investigations will be used as the benchmark for the upcoming study.

### 1.4.1 Problem Definition

The description of the impulsively started test case was given in section ???. For the hybrid simulation, we performed similar investigations and compared with the validation data. The parameters of the simulation are tabulated in table 1.4.

Figure 1.22 shows the domain decomposition of the hybrid simulation. The Eulerian domain  $\Omega_E$  is bounded by boundary  $\partial\Omega_E$ , where  $\partial\Omega_E = \Sigma_d \cup \Sigma_{wall}$ , where  $\Sigma_d$  is the external Dirichlet boundary and  $\Sigma_{wall}$  is the no-slip wall. The Lagrangian domain  $\Omega_L$  resolves the full fluid domain. The interpolation region  $\Omega_{int}$ , where we correct the particle strengths, is within the Eulerian domain  $\Omega_E$ , such that  $\Omega_{int} \subset \Omega_E$ . The interpolation region  $\Omega_{int}$  is bounded by  $\partial\Omega_{int}$  where  $\partial\Omega_{int} = \Sigma_p \cup \Sigma_{int;ext}$ . The vortex panel boundary  $\Sigma_p$  has an offset  $d_{surf} = 3 \cdot h$  from the wall  $\Sigma_{wall}$ . The offset is chosen according to stock,

see section ???. The exterior boundary  $\Sigma_{int;ext}$  of the interpolation region  $\Omega_{int}$  is defined with a larger offset  $d_{bdry} = 0.1 \cdot R$ . We observed in the previous sections that the main error of the hybrid scheme is the artificial vorticity generated at the  $\Sigma_d$  and to reduce this error, we chose the larger offset.



**Figure 1.22:** [Not to Scale] The domain decomposition for the impulsively started cylinder. The parameters of the domain are tabulated in table 1.4.

The initial boundary conditions of the Eulerian Dirichlet boundary  $\Sigma_d$ , is the velocity field induced by the vortex panels. At time progress, the vorticity is generated from the Eulerian boundary  $\Sigma_{wall}$ , transferring to the vortex blobs inside the interpolation region  $\Omega_{int}$ .

Two investigation where performed with the impulsively started problem. The first study focused on the impact of parameters on the lift and drag acting on the cylinder. The parameters of interest during this parameter sensitivity analysis were the number of vortex panels  $N_{panels}$ , nominal blob spacing  $h$ , time step size of the Lagrangian method  $\Delta t_L$ .

The second focus of the investigation was the long run performance of the forces acting of the cylinder. Artificial perturbation was induced as described in section ?? to initiate vortex shedding at the initial stages of the simulation.

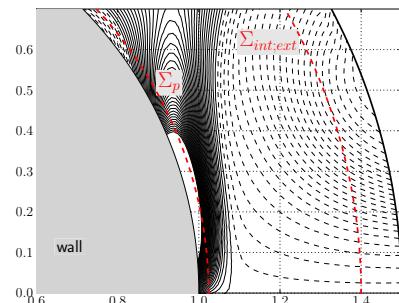
#### 1.4.2 Results and Discussion

Figure 1.24 shows the vorticity contour at the initial stages of the simulation,  $t = [1, 3, 5, 7]$ . The plot compares the hybrid simulation (top half) with the FE only simulation (bottom half). The hybrid half of the plot also depicts the Eulerian domain  $\Omega_E$ , and is bounded to the cylinder, resolving the near-wall region of the problem.

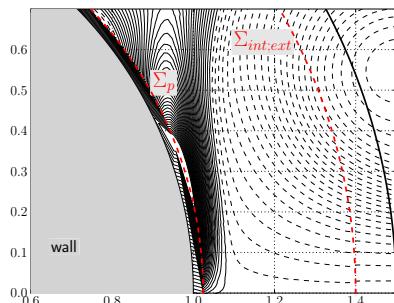
We observe that the vorticity contours of the hybrid simulation matches with the FE only simulation. The figure is nearly symmetric expect the artificial vorticity emanating from the Dirichlet boundary  $\Sigma_d$ , convecting with the free-stream. This magnitude of this artificial vorticity is within  $|\omega| \leq 0.2$  where the maximum vorticity in the domain is  $\max\{\Omega\} = 32$ . Therefore, the relative vorticity generated from the boundary with less than 1% of the maximum vorticity in the fluid.

**Table 1.4:** Summary of the parameters of the hybrid simulation for the Impulsively started cylinder test case for  $Re = 550$ .

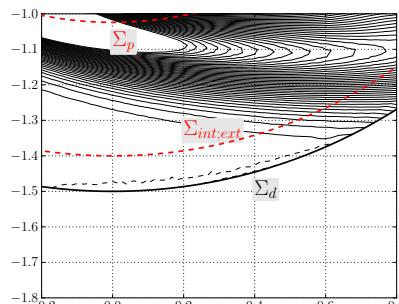
Parameters	Value	Unit	Description
$Re$	550	-	Reynolds number
$\mathbf{u}_\infty$	[1, 0]	$\text{m s}^{-1}$	Free-stream velocity
$R$	1	m	Radius of cylinder
$R_{ext}$	1.5	m	Radius of Eulerian domain $\Omega_E$
$\nu$	$3.6 \times 10^{-3}$	$\text{kg s}^{-1} \text{m}^{-1}$	Kinematic viscosity
$\lambda$	1	-	Overlap ratio
$h$	0.008	m	Nominal blob spacing
$h_{grid}$	0.008 to 0.04	m	FE cell diameter
$N_{\text{cells}}$	32138	-	Number of mesh cells
$N_{\text{panels}}$	100	-	Number of panels
$\Delta t_L$	0.005	s	Lagrangian time step size
$\Delta t_E$	0.001	s	Eulerian time step size
$k_E$	5	-	Eulerian sub-steps
$N_{\text{t-steps}}$	40000	-	Number of time integration steps
$t$	0 to 40	-	Simulation time
$d_{bdry}$	$0.1 \cdot R$	m	Interpolation offset from boundary $\Sigma_d$
$d_{surf}$	$3 \cdot h$	m	Interpolation offset from boundary $\Sigma_{wall}$



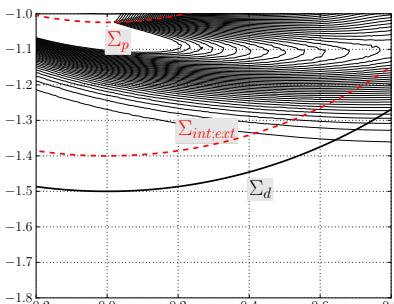
(a) Wall region: Eulerian method



(b) Wall region: Lagrangian method

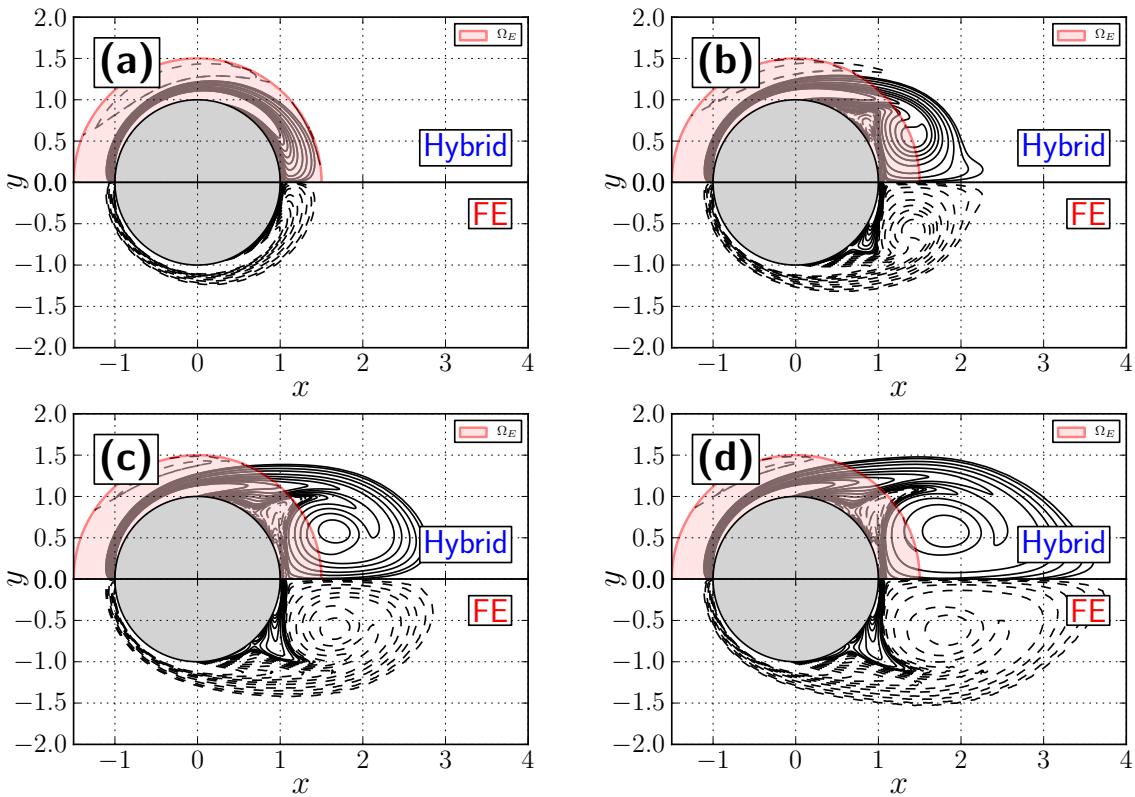


(c) Boundary region: Eulerian method



(d) Boundary region: Lagrangian method

**Figure 1.23:** Eulerian method and Lagrangian method resolutions



**Figure 1.24:** Comparison of the vorticity contours for (a)  $t = 1$ , (b)  $t = 3$ , (c)  $t = 5$  and (d)  $t = 7$  with contour levels  $[-7, \dots, -3, -2, -1, 0.5, -0.2, -0.1, 0.1, 0.2, 0.5, 1, 2, 3, \dots, 7]$ . The figures compares the hybrid simulation (top half) with FE only simulation (bottom half).

To investigate the effect of this error, we determined the error in the evolution of the drag during the initial stages of the simulation. Figure 1.25 shows the evolution of the drag coefficient  $C_d$ , friction drag  $C_{d_{fric}}$ , and the pressure drag  $C_{d_{pres}}$ , comparing the hybrid simulation, FE only simulation and the reference data obtained from Koumoutsakos and Leonard [?]. Observing the figure, we see that the hybrid simulation has a larger difference with the reference data. The error is due to the difference in the pressure drag  $C_{d_{pres}}$ . We see that the hybrid simulation generally has a larger drag coefficient  $C_d$ . Furthermore, at the at  $t < 0.3$ , we see that there is a slight difference in the initial drag trend.

To investigate further on the causes of this trends, we performed a parameter sensitivity analysis. Figure 1.26 shows the impact of varying the resolution of the Lagrangian method w.r.t to the Eulerian method on the accuracy of the drag coefficient calculated.

Figure 1.26a investigates the effect of changing the Lagrangian time step size  $\Delta t_L$  on the drag coefficient. The Lagrangian time step was varied with setting the number of Eulerian sub-steps  $k_E$  to  $k_E = 1$  and  $k_E = 5$ . With fixed Eulerian time step size  $\Delta t_E = 0.001$ , the Lagrangian time step sizes were  $\Delta t_L = 0.001$  and  $\Delta t_L = 0.005$ , respectively. The figure shows that reducing the Lagrangian time-step size has only a minimal improvement.

Figure 1.26b shows the effect of varying the nominal blob spacing  $h$  from  $h = 0.008$  to  $h = 0.005$ . The figure shows that increasing the resolution of the blobs as significant

improvement on the drag coefficient. Furthermore, the initial trend at  $t < 0.3$  matches more accurately with higher resolution. This investigation shows the to have an accurate result, we require a finer resolution of the Lagrangian field near the Eulerian domain  $\Omega_E$ .

Figure 1.26c shows the effect of varying the number of vortex panels  $N_{panels}$  from  $N = 100$  to  $N = 400$ . The improvement with higher resolved vortex sheet is smaller than the improvement obtained by varying the blob resolution.

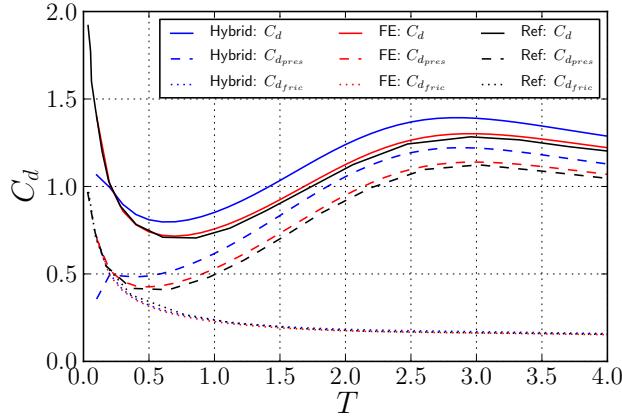
The second focus of the impulsively started cylinder is the long run,  $t = 0$  to  $t = 40$ , evolution of the drag and lift of the cylinder. We performed similar comparison as done in section ???. An artificial perturbation was induced according to Leocointe & Piquet [?]. Figure 1.27 shows the evolution of the lift coefficient  $C_l$ , and the drag coefficient  $C_d$  of hybrid simulation, FE only simulation, and the reference data from RosenFeld et al. [?].

Investigating the evolution of drag shows that the hybrid simulation has higher drag. After  $t = 5$ , there is slight mismatch in the oscillation of the drag. However observing the amplitude fluctuation, we see that the simulation tend to fluctuate around  $C_d = 1.4$ . Observing the evolution of lift shows that the hybrid simulation has a larger initial amplitude. Furthermore, there exist a negative phase shift in the amplitude. However, at time progress,  $t > 20$ , we see that the frequency and the amplitude of the oscillation is similar to the reference data.

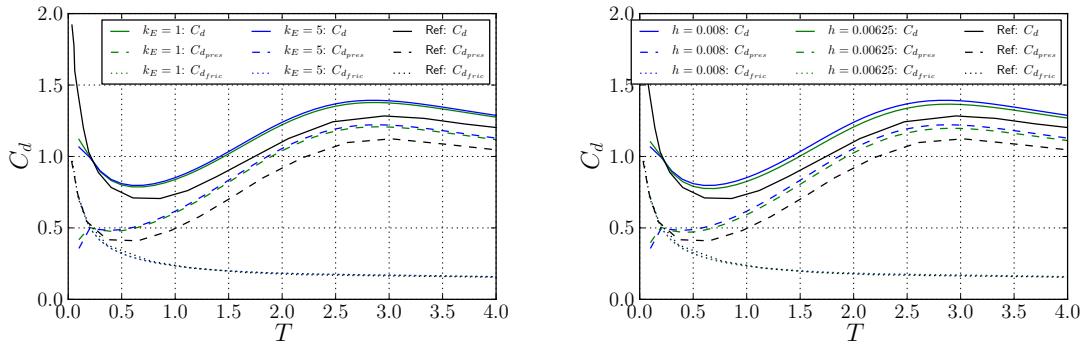
Figure 1.28 compares the vorticity field of the hybrid simulation, and the FE only simulation at  $t = [10, 20, 30, 40]$ . The shed vorticity of the hybrid simulation matches reasonably well with the FE only with a slight dif

A through investigation of the oscillation requires a longer simulation where the amplitude of the oscillation would become fixed. However, due to the lack of computational resources, a longer simulation than  $t = 40$  with the current simulation parameters was not feasible.

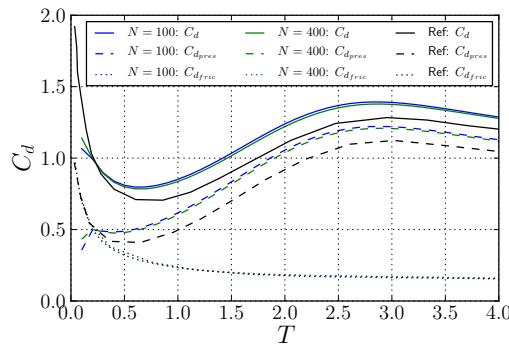
#### 1.4.3 Conclusion



**Figure 1.25:** Evolution of the drag coefficient during the initial stages  $t = 0$  to  $t = 4$  with total drag coefficient  $C_d$  (solid), pressure drag coefficient  $C_{d,pres}$  (dashed) and friction drag coefficient  $C_{d,fric}$  (dotted). The figure compares results of hybrid simulation (blue), FE only simulation (red) and reference data (black) of Koumoutsakos and Leonard [?]

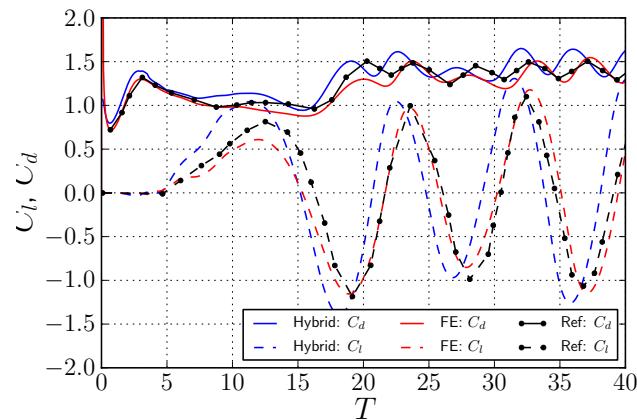


**(a)** Variation in Lagrangian time step size  $\Delta t_L$ : **(b)** Variation in nominal blob spacing  $h$ :  $h = k_E = 1$  with  $\Delta t_L = \Delta t_E$  and  $k_E = 5$  with  $\Delta t_L = 5 \cdot \Delta t_E$

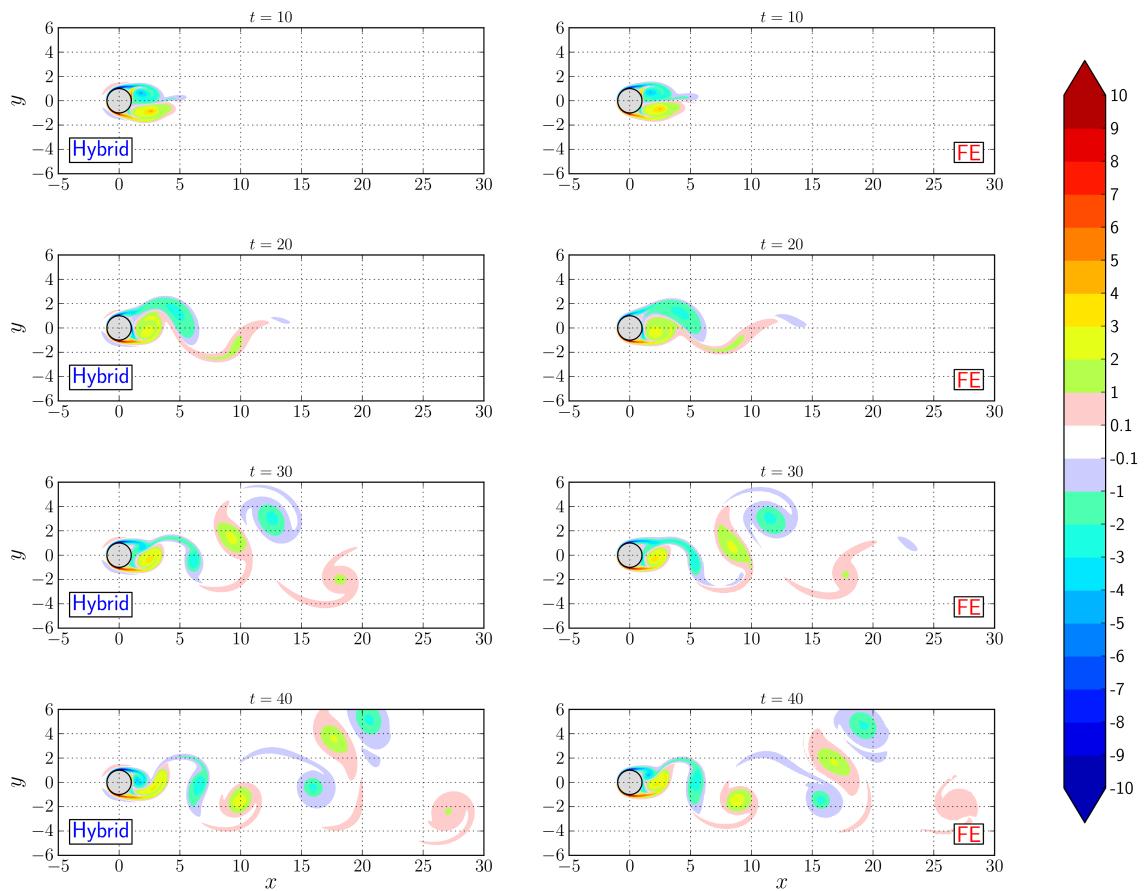


**(c)** Variation in number of panels  $N_{panels}$ :  $N = 100$  and  $N = 400$

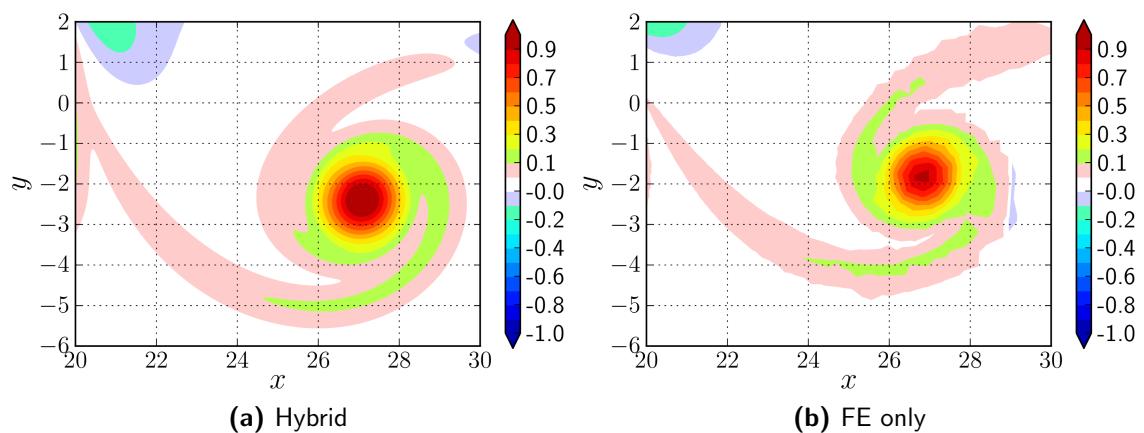
**Figure 1.26:** Parameters sensitivity analysis on the drag evolution of the cylinder from  $t = 0$  to  $t = 4$ , compared with literature data (black) obtained from Koumoutsakos and Leonard [?]



**Figure 1.27:** Evolution of the lift and drag coefficient from  $t = 0$  to  $t = 40$  with artificial perturbation [?]. The figure compares hybrid (blue), FE only (red), and the reference data (black) from RosenFeld et al. [?].



**Figure 1.28:** hybrid cylinder LongRun contourfComparison



**Figure 1.29:** First dipole

## 1.5 Stalled Elliptic airfoil at $Re = 5000$

### 1.5.1 Problem Definition

The stalled airfoil performed similar to the impulsively started cylinder.

The parameters are tabulated below.

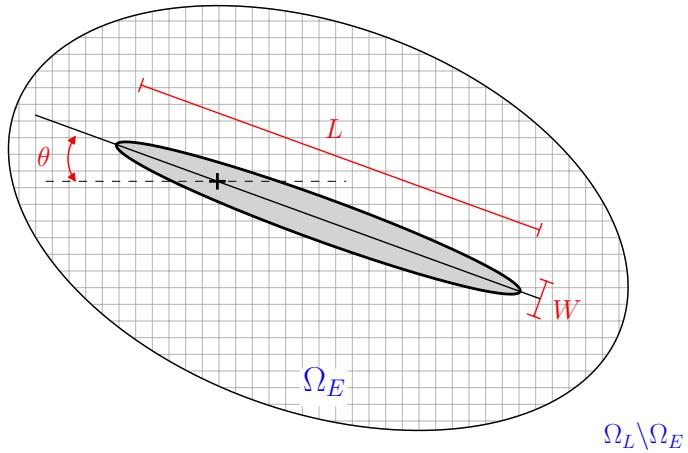
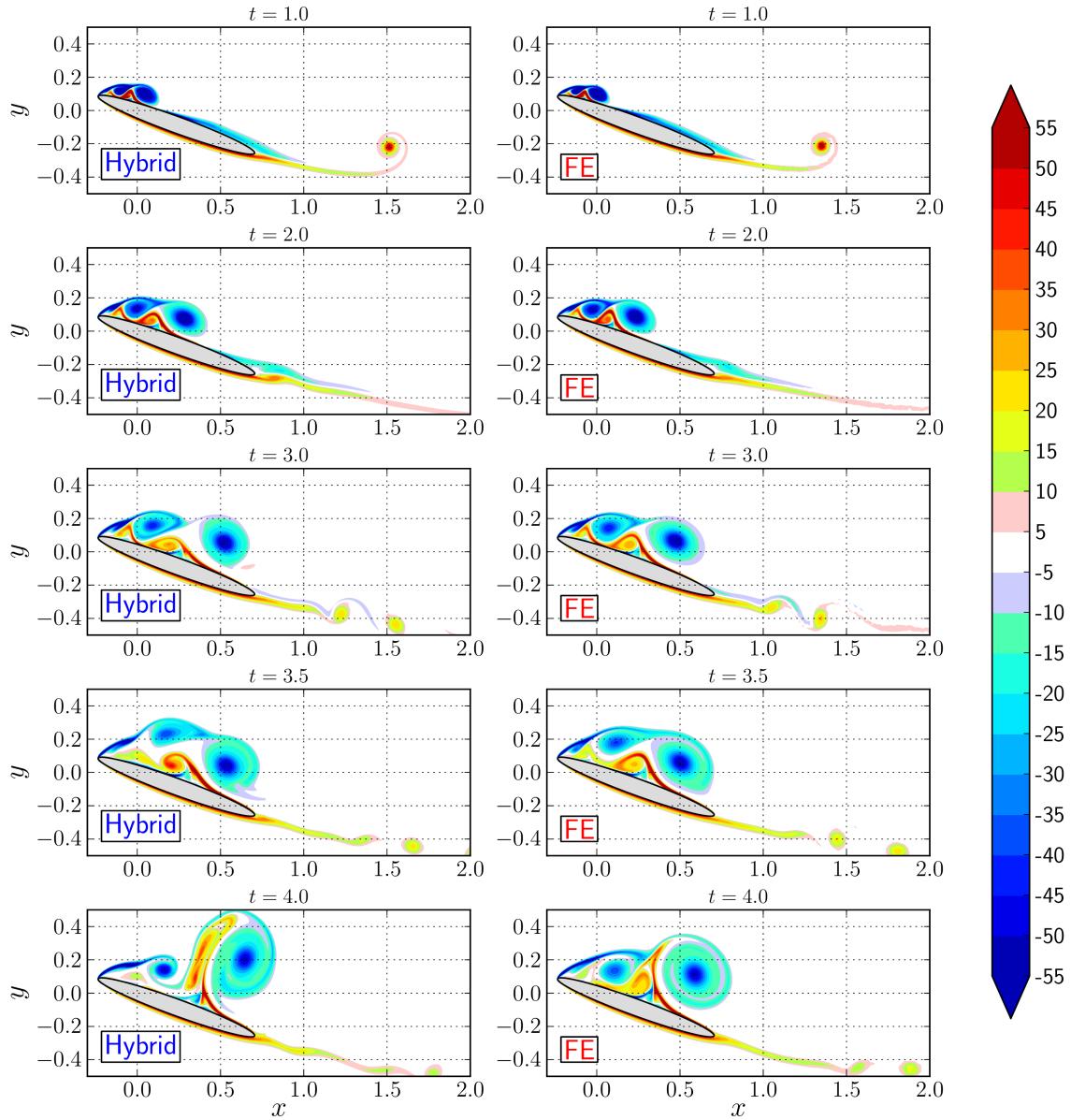


Figure 1.30: [Not to Scale]

### 1.5.2 Results and Discussion



**Figure 1.31:** hybrid ellipse HybridvsFE contourDeviation

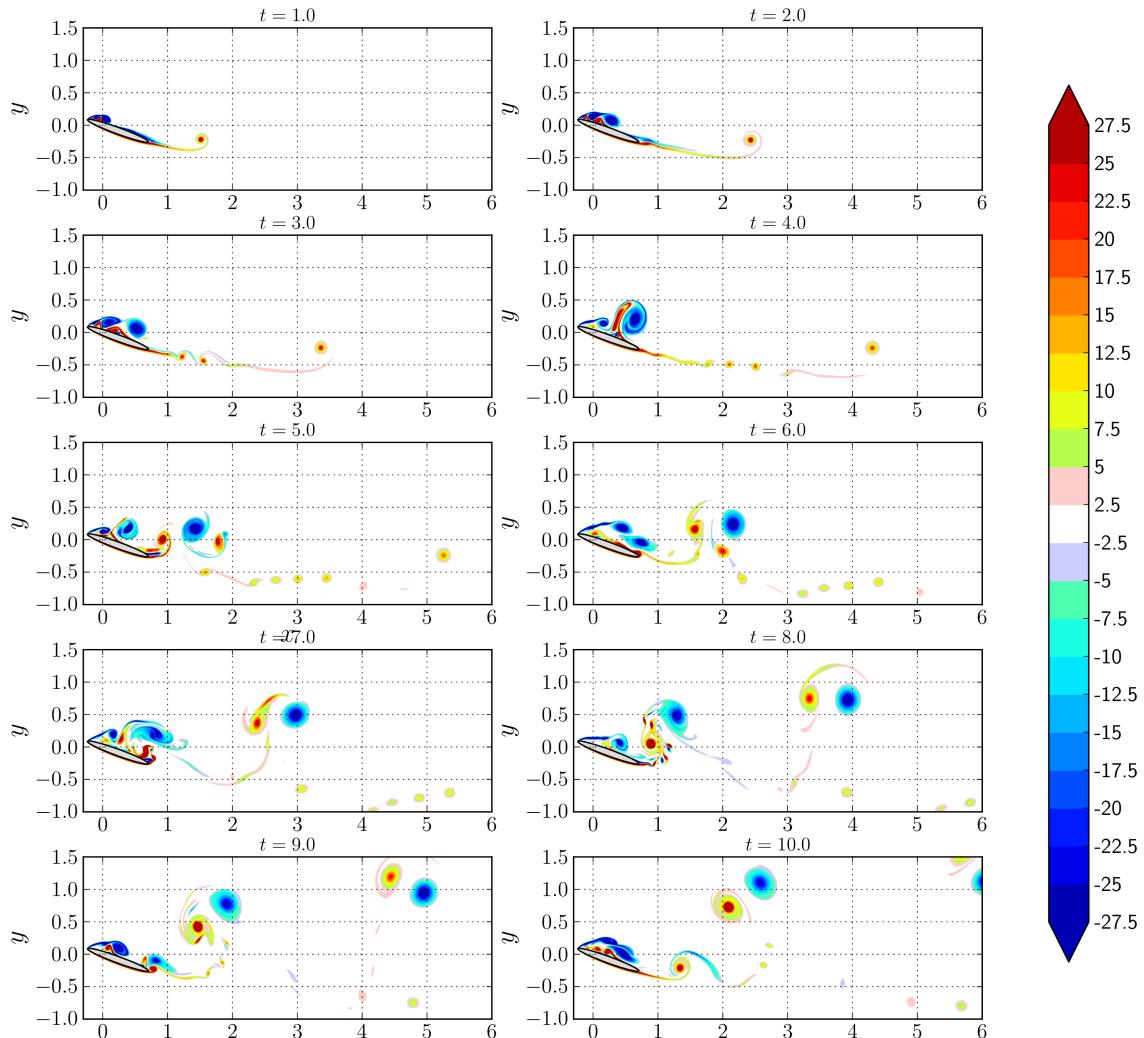


Figure 1.32: hybrid ellipse Hybrid contours

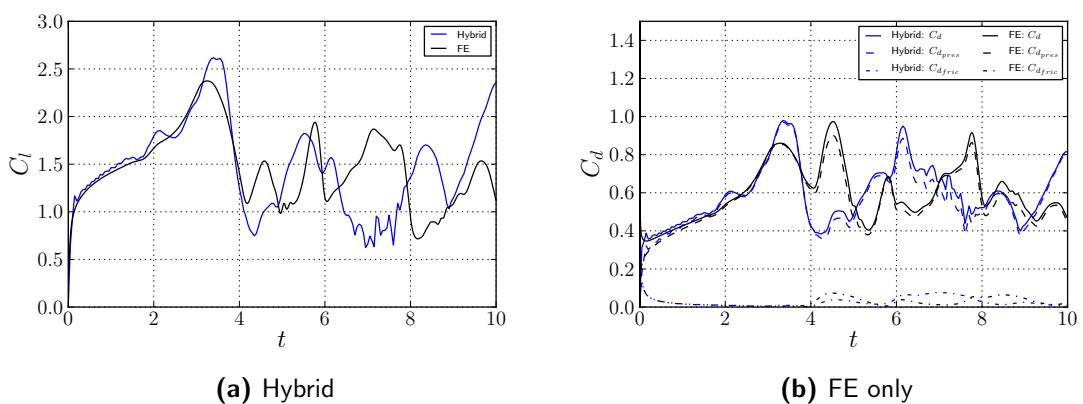


Figure 1.33: Forces

## 1.6 Multi-body problem

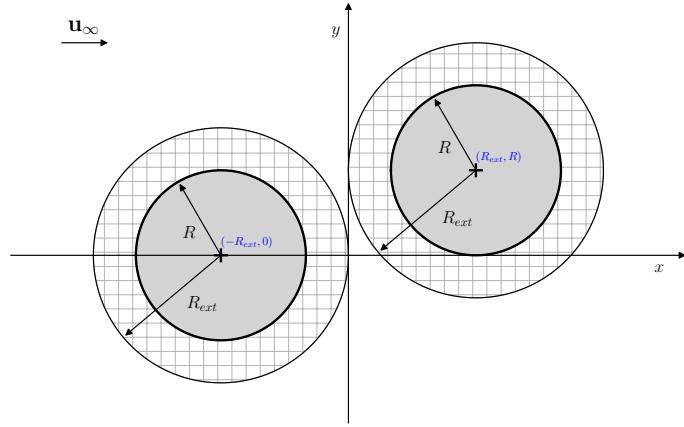
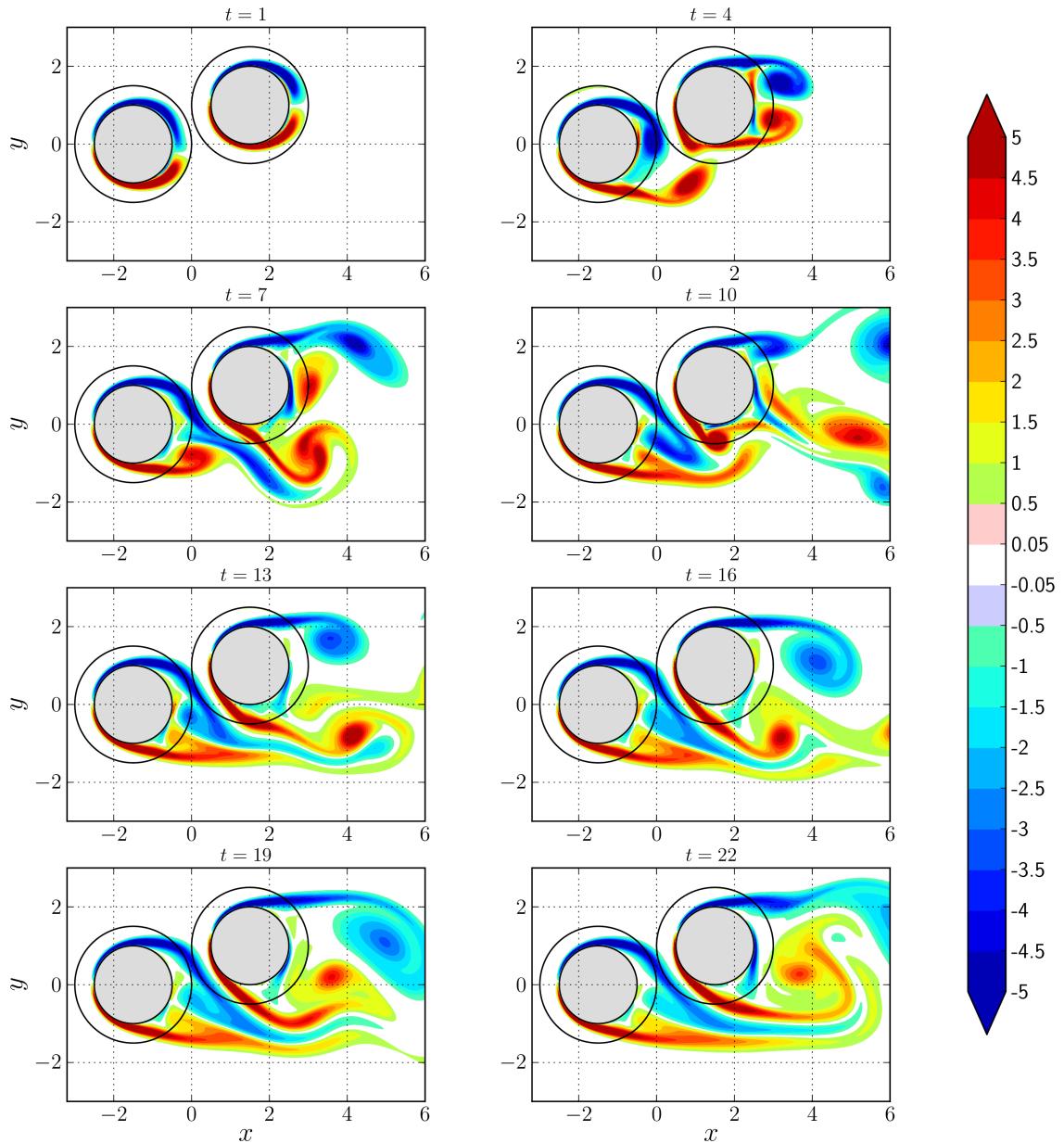


Figure 1.34: [Not to Scale]



**Figure 1.35:** hybrid ellipse Hybrid contours

---

## Chapter 2

---

# Conclusion and Recommendation

## 2.1 Conclusion

### 2.1.1 Lagrangian domain

### 2.1.2 Eulerian domain

### 2.1.3 Hybrid method

- Sub-stepping: This observation states that the linear interpolation used for sub-stepping process, has potential for improvement. A possible solution might be to employ a higher-order interpolation method for determining the Eulerian Dirichlet boundary condition at the sub-steps.
- Adaptive discretization of blobs: In conclusion, we see that a high resolution discretization of the Lagrangian method inside the Eulerian domain  $\Omega_L \cap \Omega_E$  is paramount for accurate transfer of information to and from the Eulerian method. For a lower resolved Lagrangian method in this region introduces artificial vorticity at the boundary of the Eulerian domain  $\Sigma_d$ , corrupting the solution of the coupling.

## 2.2 Recommendations

### 2.2.1 Lagrangian domain

### 2.2.2 Eulerian domain

### 2.2.3 Hybrid method



---

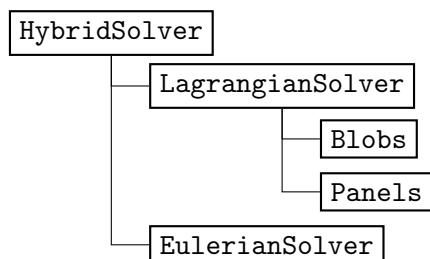
## Appendix A

---

# pHyFlow Code Structure

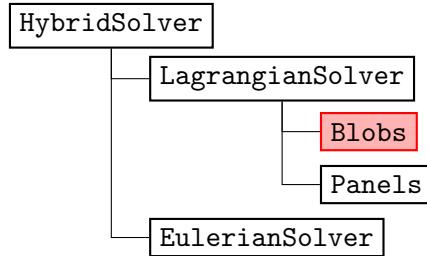
The document outlines the pHyFlow code structure. The pHyFlow functions are organized into several classes. The functions related to the vortex particles are placed inside the `Blobs` class. The functions related to the panel problem are inside `Panels` class. The `LagrangianSolver` class is made to couple the functions of the vortex blobs and the vortex panel together. The functions of the Eulerian domain are placed inside the `EulerianSolver` class, where the Navier-stokes grid problem is solved. Finally, coupling of all the problems are done with the `HybridSolver` class. Note, all the classes are capable of handling multi-body / multi-domain problem within them and `LagrangianSolver` class and the `HybridSolver` class only couples methods together.

pHyFlow Structure:

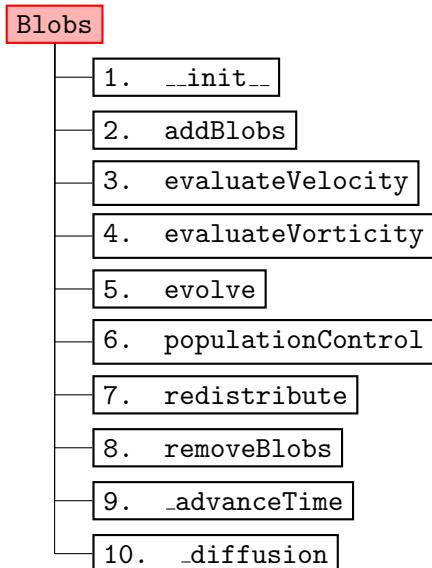


## Blobs Class

The main structure of the `Blobs` class. This class contains all the function related to the calculation of the vortex blobs.



### Class structure:



### Attributes:

Attributes	Description
<code>blobControlParams</code>	The diffusion parameters. It is a dictionary containing all the parameters of the diffusion method used for the simulation. Contains: <code>stepRedistribution</code> , <code>stepPopulationControl</code> , <code>gThresholdLocal</code> , <code>gThresholdGlobal</code> .
<code>computationMethod</code>	<code>computationMethod</code> (tuple) with the type of Biot-Savart solver ( <code>direct</code> , <code>fmm</code> ) and the type of hardware to use ( <code>cpu</code> , <code>gpu</code> ).
<code>deltaTc</code>	The size of the convective time step $\Delta t_c$
<code>deltaTd</code>	The size of the convective time step $\Delta t_d$
<code>diffusionParams</code>	A dictionary containing all the parameters related to the computation of the diffusion step. Specifies the diffusion scheme and other specific parameters. Contains: <code>method</code> , <code>c2</code> .
<code>g</code>	The strength of the vortex blobs $\alpha$ .

<code>gThresholdGlobal</code>	Maximum value of variation of total vorticity due to the removal of blobs during population control.
<code>gThresholdLocal</code>	Minimum value of circulation to consider for each vortex blob when selecting blobs to remove during population control.
<code>h</code>	The size of the cell associated to the vortex blobs. Corresponds to the minimum spacing between the core of two neighboring cells. It is related to the core size of the blob, $\sigma$ , and to the spacing $h$ by the expression $Ov = h/\sigma$ .
<code>integrationMethod</code>	<code>integrationMethod (fe, rk4)</code> the type of time integrator used: <code>fe</code> forward Euler, <code>rk4</code> Runge-Kutta 4 <sup>th</sup> order.
<code>nu</code>	The fluid kinematic viscosity, used to calculate the diffusion coefficient: <code>c2</code> and diffusion time step <code>deltaTd</code> , $\Delta t_d$ .
<code>numBlobs</code>	The number of blobs.
<code>overlap</code>	The overlap ratio between neighboring blobs.
<code>plotVelocity</code>	A flag that defines if velocity is to be plotted or not.
<code>sigma</code>	The core size of the vortex blobs.
<code>stepDiffusion</code>	The frequency of diffusion steps.
<code>stepPopulationControl</code>	The frequency of population control.
<code>stepRedistribution</code>	The frequency of redistribution of blobs.
<code>timeIntegrationParams</code>	A dictionary containing all time integration parameters of the simulation. Contains the definition of the time integration scheme possibly additional parameters specific to the scheme.
<code>t</code>	The current time of the simulation.
<code>tStep</code>	The current time step of the simulation.
<code>velocityComputationParams</code>	A dictionary containing all the parameters related to the computation of induced velocities. Specifies computation scheme (direct or fmm) and hardware to use (cpu or gpu).
<code>vInf</code>	The free stream velocity.
<code>x</code>	The $x$ coordinates of the vortex blobs.
<code>y</code>	The $y$ coordinates of the vortex blobs.

**Table A.1:** Attributes of `Blobs` class and their description.**`--init--`**

**Description:** Initialize the `Blobs` class with either the given input parameters or by reading a file containing all the necessary parameters.

**Input Parameters**

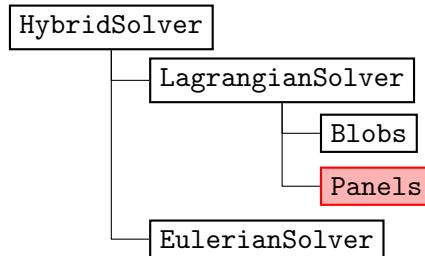
<code>File Name</code>	Containing all the parameters to re-initialize the class.
— or —	
<code>Parameters</code>	
Vorticity Field	: { <code>xBlob</code> , <code>yBlob</code> , <code>gBlob</code> } or { <code>wFunction</code> , <code>xBounds</code> , <code>yBounds</code> }
Blob parameters	: <code>overlap</code> , <code>h</code>
Time Step parameters	: <code>deltaTc</code> , <code>nu</code> , <code>stepRedistribution</code> , <code>integrationMethod</code> , <code>computationMethod</code>
Population control parameters	: <code>stepPopulationControl</code> , <code>gThreshold</code>

**Descriptions of the parameters:**

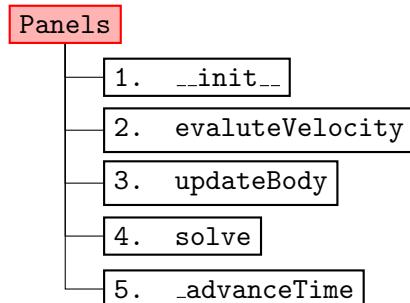
<i>Vorticity field</i>		<i>Default</i>
xBlob,yBlob	: the $x, y$ blob coordinates.	-
gBlob	: the circulation $\Gamma_i$ associated to each of the vortex blobs.	-
— or —		
wExactFunction	: the function that returns the exact value of vorticity $\omega$ at any given $x, y$ coordinates. <b>Input parameters</b> : xEval,yEval <b>Assigns</b> : - <b>Returns</b> : wEval	1.0
xBounds, yBounds	: the $x, y$ bounds of the domain where the particles was originally distributed.	-
<i>Blob parameters</i>		<i>Default</i>
overlap	: the overlap ratio $h/\sigma$ .	1.0
h	: the size of the cell $h$ associated to the blobs. <i>Note:</i> Cells are square.	-
<i>Time step parameters</i>		<i>Default</i>
deltaTc	: the size of the convective time step $\Delta t_c$ .	-
nu	: the fluid kinematic viscosity $\nu$ , used to calculate the diffusion coefficient $c^2$ and diffusion time step size $\Delta T_d$ .	-
stepRedistribution	: the redistribution step frequency.	1
integrationMethod	: the time integration method (FE: Forward euler , RK4: 4 <sup>th</sup> order Runge-Kutta).	RK4
computationMethod	: the calculation method to evolve the blobs, (Direct: Direct Method, FMM: Fast-Multipole Method) using (CPU, GPU).	{FMM, GPU}.
<i>Population control parameters</i>		<i>Default</i>
stepPopulationControl	: population control step frequency	1.
gThreshold	: the tuple with minimum <b>and</b> maximum value of the circulation $\Gamma_{min}$ .	-
<i>Free stream velocity</i>		<i>Default</i>
vInf	: The free-stream velocity function, returning the velocity action on the vortex blobs. <b>Input parameters</b> : t <b>Assigns</b> : - <b>Returns</b> : vx,vy	-

## Panels class

The main structure of the panel method class **Panels**. This class contains all the functions related to the calculation of panels.



### Class structure:



### Attributes:

Attributes	Description
A	The inter-induction matrix <b>A</b> , the LHS of the problem.
cmGlobal	The global position vector for each of the <b>N</b> body, refining the position of the local panel $(0, 0)$ in the global coordinate system.
deltaT	The simulation time step size $\Delta T$
geometryKeys	The dictionary containing all the parameters of the geometry. Contains: <b>xPanel</b> (the $x$ coordinate of the <b>M</b> panel corners.), <b>yPanel</b> (The $y$ coordinate of the <b>M</b> panel corners), <b>cmGlobal</b> , <b>thetaLocal</b> , <b>dPanel</b> (The off-set of the panel collocation point from the panel mid-point).
nBodies	The number of panel bodies.
norm	The $x, y$ normal vector of each panel.
normCat	The global concatenated $x, y$ component of the panel normal vector at each collocation points.
nPanels	The number of panels in each body/geometry.
nPanelsTotal	The total number of panels.
panelKernel	A string defining panel kernel type.
problemType	A string defining the panel problem is of a <b>moving</b> type or of a <b>fixed</b> type.
solverCompParams	The dictionary containing solver computation parameters.
sPanel	The vortex sheet strengths $\gamma$ of <b>M</b> panels.
t	The current time $t$ of the simulation.
tang	The $x, y$ tangent vector of each panel.

<code>tangCat</code>	The global concatenated $x, y$ component of the panel normal vector at each collocation points.
<code>thetaLocal</code>	The local rotation angle $\theta$ w.r.t to the local coordinate system. The rotational will be performed around the local reference point $(0, 0)$ , i.e around the global center of rotation point <code>cmGlobal</code> .
<code>tStep</code>	The current step of the simulation.
<code>velCompParams</code>	A dictionary containing the velocity computation parameters, method and hardware.
<code>xyCPGlobal</code>	The global $x, y$ coordinate of the panel collocation points.
<code>xyCPGlobalCat</code>	The global concatenated $x, y$ coordinate of the panel collocation points.
<code>xyPanelGlobal</code>	The global $x, y$ coordinate of the panel bodies.
<code>xyPanelGlobalCat</code>	The global concatenated $x, y$ coordinate of the panel bodies.
<code>xyPanelLocal</code>	The local $x, y$ coordinate of the panel bodies.

**Table A.2:** Attributes of Panels class and their description.`__init__`**Input Parameters**

<i>File Name</i>	Containing all the parameters to re-initialize the class.
<i>Parameters</i>	Panel coordinates : { <code>xCP</code> , <code>yCP</code> , <code>xPanel</code> , <code>yPanel</code> , <code>cmGlobal</code> , <code>thetaLocal</code> }
	External velocity : <code>externVel</code>

**Description:** Initialize the `panels` class with the given input parameters. In the case of a multibody problem, a list of panel coordinates can be given and internally it takes care of the inter-coupling.

*Panel coordinates*

<code>xCP</code> , <code>yCP</code>	: the local $x, y$ -coordinates of the panel collocation points.
<code>xPanel</code> , <code>yPanel</code>	: the local coordinate of the panel edges. <i>Note:</i> Should have a closed loop (end with initial point coordinates).
<code>cmGlobal</code>	: the position of reference points of a given panel body.
<code>thetaLocal</code>	: the rotational angles of the panel body axes w.r.t to the global $x$ -axis.

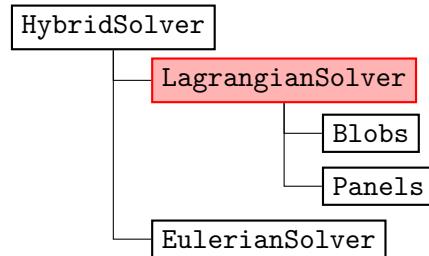
*External velocity*

<code>externVel</code>	: Reference to an external velocity <b>function</b> acting of the panels. For the panel case, the external velocity will be the induced velocity of the blobs + freestream <code>vortexBlob.evaluateVelocity</code> .
------------------------	---

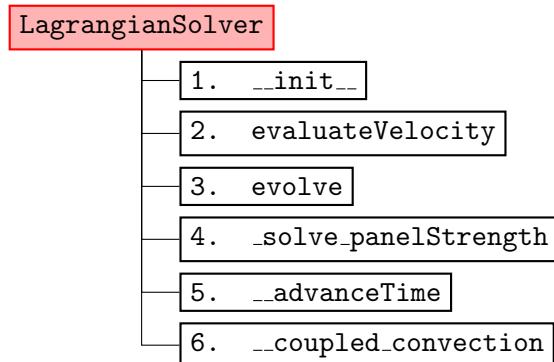
**Input parameters** : `xCP`,`yCP`**Assigns** : -**Returns** : `vxCP`,`vyCP`

## LagrangianSolver Class

The main structure of the **Blobs + Panels** (LagrangianSolver) class. This class contains all the function related to the calculations of panel with vortex blobs.



### Class structure:



### Attributes:

Attributes	Description
<code>deltaT</code>	The inter-induction matrix $\mathbf{A}$ , the LHS of the problem.
<code>gTotal</code>	The total circulation of the Lagrangian domain.
<code>t</code>	The current time $t$ of the simulation.
<code>tStep</code>	The current step of the simulation.
<code>vInf</code>	The $x, y$ component of the free-stream velocity.
<code>Blobs</code>	The vortex blobs class <code>Blobs</code> .
<code>Panels</code>	The vortex panels class <code>Panels</code> .

**Table A.3:** Attributes of LagrangianSolver class and their description.

### `__init__`

#### Input Parameters

<i>File Name</i>	Containing all the parameters to re-initialize the class.
<i>Parameters</i>	<code>vortexBlobs</code> : {vortexBlobs} class. <code>panels</code> : panels class.

**Description:** Initialize the `vortexMethod` class using `vortexBlob+panelMethod` classes.

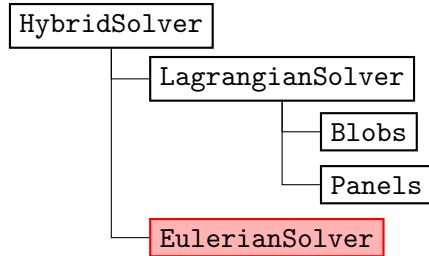
**Input parameters:**

`Blobs`: vortex particle class

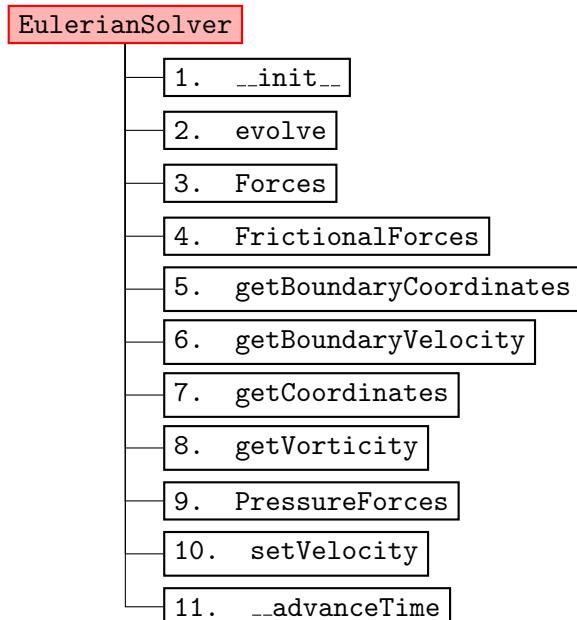
`Panels`: panel method class

## EulerianSolver

The main structure for the Navier-stokes class `EulerianSolver`. This class contains all the functions related to computation of the Navier-stokes problem. Below is set of functions that acts as the interface to the class.



### Class structure:



### Attributes:

Attributes	Description
<code>deltaT</code>	The time step size $\Delta t$ .
<code>deltaTMax</code>	The maximum allowable time step size $\max\{\Delta t\}$ .
<code>cfl</code>	The CourantFriedrichsLowy condition stability number CFL.
<code>cmGlobal</code>	The $x, y$ position of the mesh local reference point $(0, 0)$ in the global coordinates.
<code>hMin</code>	The minimum mesh cell size.
<code>nu</code>	The fluid kinematic viscosity $\nu$ .
<code>probeGridMesh</code>	The <i>local</i> $x, y$ coordinates of the probe grid mesh.
<code>probeGridParams</code>	The dictionary containing all the parameters of the probe grid for extracting the vorticity data.

<code>solverParams</code>	The dictionary file containing all the solver parameters.
<code>t</code>	The current time of the simulation.
<code>thetaLocal</code>	The local rotational angle $\theta$ of the mesh domain. Therefore, the rotation will be done about local reference point (0, 0), i.e <code>cmGlobal</code> in the global coordinate system.
<code>tStep</code>	The current step of the simulation.
<code>uMax</code>	The maximum fluid velocity $\max\{\mathbf{u}\}$ .

**Table A.4:** Attributes of `EulerianSolver` class and their description.

`__init__`

**Description:** Initialize the `navierStokes` class either using a `fileName` containing all the necessary parameter for initialization or by explicitly inputting the parameters.

### Input Parameters

<i>File Name</i>	Containing all the parameters to re-initialize the class.
— or —	
<i>Parameters</i>	Mesh data : <code>mesh</code> , <code>boundaryDomains</code>
	Geometry position : <code>cmGlobal</code> , <code>thetaLocal</code>
	Fluid parameters : <code>uMax</code> , <code>nu</code>
	Solver options : <code>cfl</code>
	Probe grid parameters : <code>x0</code> , <code>y0</code> , <code>Lx</code> , <code>Ly</code> , <code>hx</code> , <code>hy</code>

### Description of the parameters:

#### *Mesh data*

- `mesh` : the mesh data file.  
`boundaryDomains` : the boundary mesh domain data file.

#### *Geometry position*

- `cmGlobal` : the  $x, y$  position of the geometry in global coordinates.  
`thetaGlobal` : the rotation angle (in  $rad$ ) of the geometry in global coordinate system.

#### *Fluid parameters*

- `uMax` : the maximum fluid velocity  $U_{max}$ . Used to determine the maximum time step size  $\Delta t_{max}$ .  
`nu` : the fluid kinematic viscosity  $\nu$ , for incompressible navier-stokes problem.

#### *Solver options*

- `cfl` : the *CFL* stability parameter. If explicit time marching scheme,  $CFL < 1$ .

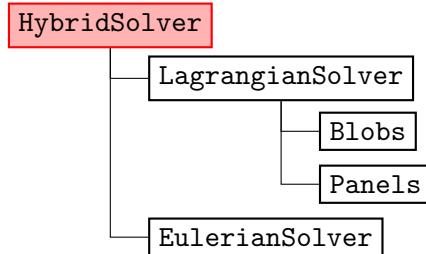
*Probe grid parameters*

---

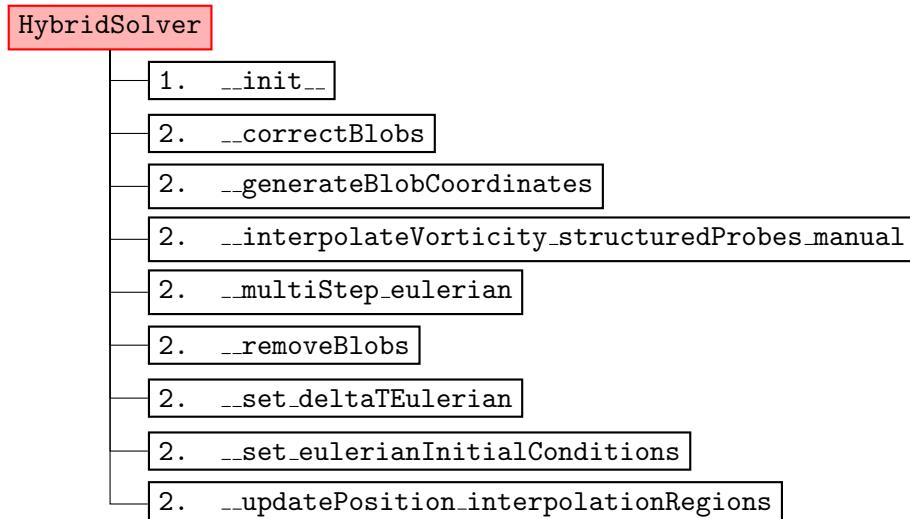
<b>x0,y0</b>	: the $x, y$ coordinate of the origin of the probe grid.
<b>Lx,Ly</b>	: the $x, y$ size (width and height) of the probing grid.
<b>hx,hy</b>	: the $x, y$ spacing of the probe grid cell.

## HybridSolver Class

The main structure for the hybrid class `HybridSolver`. This class contains all the functions related to computation of the hybrid problem.



### Class structure:



### Attributes:

Attributes	Description
<code>deltaTEulerian</code>	The time step size of the Eulerian sub-domain $\Delta t_E$ .
<code>deltaTLagrangian</code>	The time step size of the Lagrangian sub-domain $\Delta t_L$ .
<code>nu</code>	The fluid kinematic viscosity $\nu$ .
<code>t</code>	The current time $t$ of the simulation.
<code>tStep</code>	The current step of the simulation.
<code>vInf</code>	The $x, y$ component of the free-stream velocity.
<code>interpolationRegion</code>	The dictionary containing the <code>surfacePolygon</code> and <code>boundaryPolygon</code> defining the boundaries of the interpolation region for each Eulerian sub-domains. The geometry is identified by the keys of the Eulerian sub-domain found in <code>multiEulerian</code> . The coordinates are defined in local coordinate system of the Eulerian grid and will be transformed (rotated + moved) during the evolution step.
<code>lagrangian</code>	The Lagrangian solver class contains all the parameters related to simulation the flow in lagrangian sub-domain.

<code>multiEulerian</code>	The <code>multiEulerian</code> is solver class containing all the Eulerian sub-domains of the hybrid problem.
----------------------------	---

**Table A.5:** Attributes of `HybridSolver` class and their description.

`__init__`

### Input Parameters

<i>File Name</i>	Containing all the parameters to re-initialize the class.
<i>Parameters</i>	vortexMethod : {vortexMethod} class.
	navierStokes : navierStokes class.
	Interpolation region : <code>xPolygon, yPolygon</code>
	Motion functions : <code>T, cmGlobal, thetaGlobal, cmDotGlobal, thetaDotGlobal</code>

**Description:** Initialize the `hybrid` class using `LagrangianSolver + EulerianSolver` classes.

### Input parameters:

**LagrangianSolver:** The vortex method containing **Blobs** and **Panels** classes which can already handle the multi-body problem.

**EulerianSolver:** The Navier-Stokes grid solver class (if multiple: list of `EulerianSolver` classes). The number of navier-stokes class has to be same as the number of vortex panels.

**Interpolation Region:** the Navier-Stokes class (if multiple: list of `EulerianSolver` classes). Should be equal to number of Navier-Stokes classes. The interpolation region should be defined as list of  $x, y$  coordinates of the polygon of the interpolation region.

**Motion function:** the function describing the motion of all the geometries in the hybrid class.

### Interpolation Regions

**xPolygon, yPolygon:** the new  $x, y$  coordinate of the polygons description the interpolation region. The polygon should have a closed loop (end with starting coordinates) before continuing to the next polygon. In the case of multiple polygons, a list of `xPolygon, yPolygon` should be given and should be as many as the number of navier-stokes domain.

*Motion function*

---

T	: the current time.
cmGlobal	: a list of new positions of the geometries in the hybrid problem.
thetaGlobal	: a list of new rotational angle of the geometries in the hybrid problem.
cmDotGlobal	: a list of current displacement velocity of the geometries in the hybrid problem.
thetaDotGlobal	: a list of current rotational velocity of the geometries in the hybrid problem.

**Input parameters** : T

**Assigns** : -

**Returns** : cmGlobal,thetaGlobal,cmDotGlobal,thetaDotGlobal