

# Tdyn

Environment for Multi-Physics simulation, including Fluid Dynamics, Turbulence, Heat Transfer, Advection of Species, Structural mechanics, Free surface and user defined PDE solvers.



---

**Tdyn Theoretical Background**

© COMPASS Ingeniería y Sistemas SA  
info@compassis.com  
6<sup>th</sup> Revision Tdyn 5.0 September 2006

# TDYN THEORETICAL BACKGROUND

## *Table of Contents*

<b>INTRODUCTION.....</b>	<b>3</b>
Navier Stokes Solver .....	3
Heat Transfer Solver .....	4
Species Advection Solver .....	5
<b>STABILITY OF CFD ALGORITHMS .....</b>	<b>6</b>
<b>BASIC THEORY .....</b>	<b>6</b>
Finite Calculus (FIC) Formulation .....	6
Stabilised Navier Stokes Equations .....	8
Stabilized integral forms.....	10
Convective and pressure gradient projections.....	12
Monolithic time integration scheme .....	13
Compressible flows.....	17
Stabilised Heat Transfer Equations.....	17
<b>TURBULENCE SOLVERS .....</b>	<b>18</b>
Zero equation (algebraic) model: The Smagorinsky model.....	20
One-equation models: The kinetic energy model (k-model) .....	21
Two-equation models: The k- $\epsilon$ model .....	22
<b>IMPLICIT LES TURBULENCE MODEL .....</b>	<b>23</b>
Computation of the characteristic lengths.....	24
<b>OVERLAPPING DOMAIN DECOMPOSITION LEVEL SET.....</b>	<b>26</b>

Tdyn Theoretical Background

**STATEMENT OF THE PROBLEM.....26**

**REFERENCES .....32**

## Introduction

This document gives a short overview of the theoretical principles that Tdyn is based on.

### **Navier Stokes Solver**

The incompressible Navier-Stokes-Equations in a given three-dimensional domain  $\Omega$  and time interval  $(0, t)$  can be written as:

$$\left. \begin{aligned} \rho \left( \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) + \nabla p - \nabla \cdot (\mu \nabla \mathbf{u}) &= \rho \mathbf{f} \quad \text{in } \Omega \times (0, \mathbf{t}) \\ \nabla \cdot \mathbf{u} &= 0 \quad \text{in } \Omega \times (0, \mathbf{t}) \end{aligned} \right\}$$

#### **Eq. 1 Incompressible Navier Stokes equations**

where  $\mathbf{u} = \mathbf{u}(\mathbf{x}, t)$  denotes the velocity vector,  $p = p(\mathbf{x}, t)$  the pressure field,  $\rho$  the (constant) density,  $\mu$  the dynamic viscosity of the fluid and  $\mathbf{f}$  the volumetric acceleration. The above equations need to be combined with the following boundary conditions:

$$\begin{aligned} \mathbf{u} &= \mathbf{u}_c \quad \text{in } \Gamma_D \times (0, t) \\ p &= p_c \quad \text{in } \Gamma_P \times (0, t) \\ \mathbf{n} \cdot \boldsymbol{\sigma} \cdot \mathbf{g}_1 &= 0, \quad \mathbf{n} \cdot \boldsymbol{\sigma} \cdot \mathbf{g}_2 = 0, \quad \mathbf{n} \cdot \mathbf{u} = \mathbf{u}_M \quad \text{in } \Gamma_M \times (0, T) \\ \mathbf{u}(\mathbf{x}, 0) &= \mathbf{u}_\theta(\mathbf{x}) \quad \text{in } \Omega_D \times \{0\} \\ p(\mathbf{x}, 0) &= p_\theta(\mathbf{x}) \quad \text{in } \Omega_D \times \{0\} \end{aligned}$$

#### **Eq. 2 Navier Stokes boundary conditions**

In the above equations,  $\Gamma := \partial\Omega$  denotes the boundary of the domain  $\Omega$ , with  $\mathbf{n}$  the normal unit vector, and  $\mathbf{g}_1, \mathbf{g}_2$  the tangent vectors of the boundary surface  $\partial\Omega$ .  $\mathbf{u}_c$  is the velocity field on  $\Gamma_D$  (the part of the boundary of Dirichlet type, or prescribed velocity type),  $p_c$  the prescribed pressure on  $\Gamma_P$  (prescribed

pressure boundary).  $\sigma$  is the stress field,  $\mathbf{u}_M$  the value of the normal velocity and  $\mathbf{u}_0, p_0$  the initial velocity and pressure fields. The union of  $\Gamma_D, \Gamma_P$  and  $\Gamma_M$  must be  $\Gamma$ ; their intersection must be empty, as a point of the boundary can only be part of one of the boundary types, unless it is part of the border between two of them.

The spatial discretisation of the Navier-Stokes equations has been done by means of the finite element method, while for the time discretisation an iterative algorithm that can be consider as an implicit two steps "Fractional Step Method" has been used. Problems with dominating convection are stabilised by the so-called "Finite Increment Calculus" method, presented below.

### **Heat Transfer Solver**

Tdyn solves the transient Heat Transfer Equations in a given domain  $\Omega := \{\Omega_F \cup \Omega_S\}$  (being  $\Omega_F$  the fluid domain and  $\Omega_S$  the solid domain) and time interval  $(0, t)$ :

$$\rho C_p \left( \frac{\partial \theta}{\partial t} + (\mathbf{u} \cdot \nabla) \theta \right) - \nabla \cdot (k_F \nabla \theta) = \rho_F q_F \quad \text{in } \Omega_F \times (0, t)$$

$$\rho_s C \frac{\partial \theta}{\partial t} - \nabla \cdot (k_s \nabla \theta) = \rho_s q_s \quad \text{in } \Omega_s \times (0, t)$$

#### **Eq. 3 Heat transfer equations**

where  $\theta = \theta(\mathbf{x}, t)$  denotes the temperature field,  $\rho_s$  the (constant) solid density,  $k_s$  the solid thermal conduction tensor,  $k_F$  the fluid thermal conduction constant,  $C_p$  the fluid specific heat constant,  $C$  the solid specific heat constant and  $q_s$  and  $q_F$  the solid and fluid, respectively, volumetric heat source distributions. The above equations need to be combined with the following boundary conditions:

## Tdyn Theoretical Background

$$\begin{aligned}\theta &= \theta_c & \text{in } \Gamma_\theta \times (0, t) \\ nk_F \nabla \theta &= f_F & \text{in } \Gamma_F \times (0, T) \\ nk_S \nabla \theta &= f_S & \text{in } \Gamma_S \times (0, T) \\ \theta(x, 0) &= \theta_0(x) & \text{in } \Omega \times \{0\}\end{aligned}$$

### ***Eq. 4 Heat transfer boundary conditions***

In the above equations,  $\Gamma := \partial\Omega$  denotes the boundary of the domain  $\Omega$ , with  $\mathbf{n}$  the normal unit vector,  $\theta_c$  is the temperature field on  $\Gamma_\theta$  (the part of the boundary of Dirichlet type, or prescribed temperature type),  $f_F$  the prescribed heat flux on  $\Gamma_F$  (prescribed heat flux fluid boundary),  $f_S$  the prescribed heat flux on  $\Gamma_S$  (prescribed heat flux solid boundary) and  $\theta_0$  the initial temperature field.

The spatial discretisation of the Heat transfer equations is done by means of the finite element method, while for the time discretisation implicit first and second order schemes have been implemented.

Problems with dominating convection are stabilised by the so-called "Finite Increment Calculus" method, presented above.

### ***Species Advection Solver***

Tdyn solves the transient Species Advection Equations in a given fluid domain  $\Omega_F$  for a number of different species, and time interval  $(0, t)$ :

$$\frac{\partial \varphi}{\partial t} + (\mathbf{u} \cdot \nabla) \varphi + (c \cdot \varphi \cdot \mathbf{g} \cdot \nabla) \varphi - \nabla \cdot (k_p \nabla \varphi) = q_p \quad \text{in } \Omega_F \times (0, t)$$

### ***Eq. 5 Species advection equations***

where  $\varphi = \varphi(\mathbf{x}, t)$  denotes the concentration of species field,  $c$  the decantation coefficient,  $k_S$  the total diffusion coefficient (including turbulent effects) and  $q_P$  volumetric source. The above equations need to be combined with the standard boundary conditions.

As in the previous examples, the spatial discretisation of the Species Advection equations has been done by means of the finite element method,

while for the time discretisation implicit first and second order schemes have been implemented. Problems with dominating convection are stabilised by the so-called "Finite Increment Calculus" method, presented below.

## Stability of CFD Algorithms

Using the standard Galerkin method to discretise the incompressible Navier-Stokes equations leads to numerical instabilities coming from two sources. Firstly, owing to the advective-diffusive of the governing equations, oscillations appear in the solution at high Reynolds numbers, when the convection terms become dominant. Secondly, the mixed character of the equations limits the choice of velocity and pressure interpolations such that equal order interpolations cannot be used.

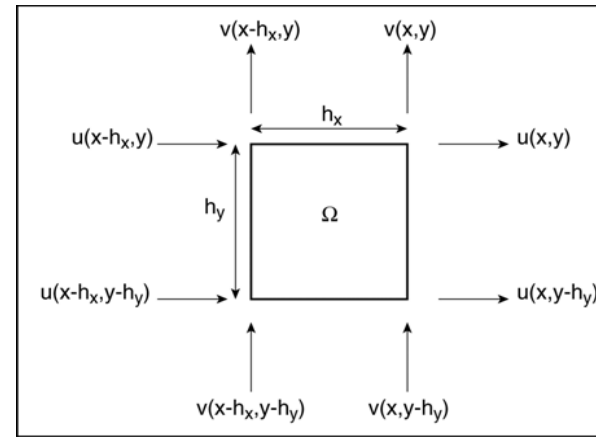
In recent years, a lot of effort has been put into looking for ways to stabilise the governing equations, many of which involve artificially adding terms to the equations to balance the convection, for example, the artificial diffusion method [7].

A new stabilisation method, known as finite increment calculus, has recently been developed [5,6]. By considering the balance of flux over a finite sized domain, higher order terms naturally appear in the governing equations, which supply the necessary stability for a classical Galerkin finite element discretisation to be used with equal order velocity and pressure interpolations.

## Basic Theory

### *Finite Calculus (FIC) Formulation*

To demonstrate this method, consider the flux problem associated with the incompressible conservation of mass in a 2D source less finite sized domain defined by four nodes, as shown in Figure 1.



**Figure 1 Mass balance in 2 dimensions**

Consider the flux problem (mass balance) through the domain, taking the average of the nodal velocities for each surface.

$$\frac{h_y}{2} \left( u(x-h_x, y) + u(x-h_x, y-h_y) - u(x, y) + u(x, y-h_y) \right) + \frac{h_x}{2} \left( u(x, y-h_y) + u(x-h_x, y-h_y) - u(x-h_x, y) + u(x, y) \right) = 0$$

**Eq. 6 Mass balance in a rectangular domain**

Now expand these velocities using a Taylor Expansion, retaining up to second order terms. Write  $u(x, y) = u$ :

$$u(x-h_x, y) = u - h_x \frac{\partial u}{\partial x} + \frac{h_x^2}{2} \frac{\partial^2 u}{\partial x^2} - O(h_x^3)$$

and similarly for the other two components of the velocity.

Substituting this back into the original flux balance equation gives, after simplification:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} - \frac{h_x}{2} \left( \frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 v}{\partial x \partial y} \right) - \frac{h_y}{2} \left( \frac{\partial^2 u}{\partial x \partial y} - \frac{\partial^2 v}{\partial y^2} \right) = 0$$

**Eq. 7 Stabilised 2D mass balance equation**

Note that as the domain size tends to zero, i.e.  $h_x, h_y \rightarrow 0$ , the standard form of the incompressible mass conservation equation in 2D is recovered. The underlined terms in the equation provide the necessary stabilisation to allow a standard Galerkin finite element discretisation to be applied. They come from admitting that the standard form of the equations is an unreachable limit in the finite element discretisation, i.e. by admitting that the element size cannot tend to zero, which is the basis for the finite element method. It also allows equal order interpolations of velocity and pressure to be used.

**Stabilised Navier Stokes Equations**

The Finite increment Calculus methodology presented above is used to formulate stabilised forms of the momentum balance and mass balance equations and the Neumann boundary conditions.

The velocity and pressure fields of a incompressible fluid moving in a domain  $\Omega \subset R^d (d=2,3)$  can be described by the incompressible Navier Stokes equations

$$\rho \frac{\partial u_i}{\partial t} + \rho \frac{\partial}{\partial x_j} (u_i u_j) + \frac{\partial p}{\partial x_i} - \frac{\partial s_{ij}}{\partial x_j} = \rho f_i$$

$$\frac{\partial u_i}{\partial x_i} = 0 \quad i, j = 1 \div N$$

**Eq. 8 Incompressible Navier Stokes equations**

## Tdyn Theoretical Background

Where  $1 \leq i, j \leq d$ ,  $\rho$  is the fluid density field,  $u_i$  is the  $i$ th component of the velocity field  $\mathbf{u}$  in the global reference system  $x_i$ ,  $p$  is the pressure field and  $s_{ij}$  is the viscous stress tensor defined by:

$$s_{ij} = 2\nu \left( \varepsilon_{ij} - \frac{1}{3} \frac{\partial u_k}{\partial x_k} \delta_{ij} \right) \quad \varepsilon_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right)$$

### ***Eq. 9 Viscous stress tensor***

The stabilized FIC form of the governing differential equations Eq. 8 can be written as

$$r_{m_i} - \frac{1}{2} h_{ij}^m \frac{\partial r_{m_i}}{\partial x_j} = 0 \quad \text{in } \Omega$$

### ***Eq. 10 Momentum equation***

$$r_d - \frac{1}{2} h_j^d \frac{\partial r_d}{\partial x_j} = 0 \quad \text{in } \Omega$$

### ***Eq. 11 Mass balance equation***

Summation convention for repeated indices in products and derivatives is used unless otherwise specified. In above equations, terms  $r_{m_i}$  and  $r_d$  denote the residual of Eq. 8, and  $h_{ij}^m$ ,  $h_j^d$  are the *characteristic length* distances, representing the dimensions of the finite domain where balance of mass and momentum is enforced. Details on obtaining the FIC stabilized equations and recommendation for the calculation of the stabilization terms can be find in Oñate 1998.

Let  $\mathbf{n}$  be the unit outward normal to the boundary  $\partial\Omega$ , split in two sets of disjoint components  $\Gamma_t$ ,  $\Gamma_u$  where the Neumann and Dirichlet boundary

conditions for the velocity are prescribed. The boundary conditions for the stabilized problem to be considered are (Oñate 2004):

$$n_j \sigma_{ij} - t_i + \frac{1}{2} h_{ij}^m n_j r_{m_i} = 0 \quad \text{on } \Gamma_t$$

$$u_j = u_j^p \quad \text{on } \Gamma_u$$

**Eq. 12 Boundary conditions for the stabilized problem**

Where  $t_i, u_j^p$  are the prescribed surface tractions and  $\sigma_{ij}$  are the total stresses, defined as

$$\sigma_{ij} = s_{ij} - p \delta_{ij}$$

Eq. 10 to Eq. 12 are the starting point for deriving stabilized FEM for solving the incompressible Navier-Stokes equations. An interesting feature of the FIC formulation is that it allows to use equal order interpolation for the velocity and pressure variables (García Espinosa 2003, Oñate 2004).

**Stabilized integral forms**

Residual form of the mass balance equation

From the momentum equations, it can be obtained (Oñate 2004)

$$\frac{\partial r_d}{\partial x_i} = \frac{h_{ii}^m}{2a_i} \frac{\partial r_{m_i}}{\partial x_j} \quad \text{no sum in } i$$

$$a_i = \frac{2\mu}{3} + \frac{u_i h_i^d}{2} \quad \text{no sum in } i$$

Substituting above equations into Eq. 11 and retaining only the terms involving the derivatives of  $r_{m_i}$  with respect to  $x_i$ , leads to the following alternative expression for the stabilized mass balance equation

$$r_d - \sum_{i=1}^{n_d} \tau_i \frac{\partial r_{m_i}}{\partial x_i} = 0, \quad \tau_i = \left( \frac{8\mu}{3h_{ij}^m h_j^d} + \frac{2\rho u_i}{h_{ii}^m} \right)^{-1} \quad \text{no sum in } i$$

**Eq. 13 Alternative mass balance equation**

The  $\tau_i$ 's in Eq. 13 when multiplied by the density are equivalent to the *intrinsic time parameters*, seen extensively in the stabilization literature. The interest of Eq. 13 is that it introduces the first space derivatives of the momentum equations into the mass balance equation. These terms have intrinsic good stability properties as explained next.

The weighted residual form of the momentum and mass balance equations is written as

$$\int_{\Omega} v_i \left[ r_{m_i} - \frac{1}{2} b_j \frac{\partial r_{m_i}}{\partial x_j} \right] d\Omega + \int_{\Gamma_t} v_i \left( n_j \sigma_{ij} - t_i + \frac{1}{2} b_j n_j r_{m_i} \right) d\Gamma = 0$$

$$\int_{\Omega} q \left[ r_d - \sum_{i=1}^N \tau_i \frac{\partial r_{m_i}}{\partial x_i} \right] d\Omega = 0$$

Where  $v_i, q$  are generic weighting functions. Integrating by parts the residual terms in above equations leads to

$$\int_{\Omega} v_i r_{m_i} d\Omega + \int_{\Gamma_t} v_i (n_j \sigma_{ij} - t_i) d\Gamma + \int_{\Omega} \frac{1}{2} h_j \frac{\partial v_i}{\partial x_j} r_{m_i} d\Omega = 0$$

$$\int_{\Omega} q r_d d\Omega + \int_{\Omega} \left[ \sum_{i=1}^N \tau_i \frac{\partial q}{\partial x_i} r_{m_i} \right] d\Omega - \int_{\Gamma} \left[ \sum_{i=1}^N q \tau_i n_i r_{m_i} \right] d\Gamma = 0$$

**Eq. 14 Weighted residual form of the momentum and mass balance equations**

We will neglect hereonwards the third integral in Eq. 14 by assuming that  $r_{m_i}$  is negligible on the boundaries. The deviatoric stresses and the pressure terms in the second integral of Eq. 14 are integrated by parts in the usual manner. The resulting momentum and mass balance equations are

$$\int_{\Omega} v_i \rho \left( \frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} \right) + \frac{\partial v_i}{\partial x_j} \left( \mu \frac{\partial u_i}{\partial x_j} - \delta_{ij} p \right) d\Omega - \int_{\Omega} v_i \rho f_i d\Omega - \int_{\Gamma_t} v_i t_i d\Gamma + \int_{\Omega} \frac{h_j}{2} \frac{\partial v_i}{\partial x_j} r_{m_i} d\Omega = 0$$

**Eq. 15 Resulting momentum equation**

and

$$\int_{\Omega} q \frac{\partial u_i}{\partial x_i} d\Omega + \int_{\Omega} \left[ \sum_{i=1}^N \tau_i \frac{\partial q}{\partial x_i} r_{m_i} \right] d\Omega = 0$$

**Eq. 16 Resulting mass balance equation**

In the derivation of the viscous term in Eq. 15 we have used the following identity holding for incompressible fluids (prior to the integration by parts)

$$\frac{\partial s_{ij}}{\partial x_j} = 2\mu \frac{\partial \varepsilon_{ij}}{\partial x_j} = \mu \frac{\partial^2 u_i}{\partial x_j \partial x_j}$$

**Convective and pressure gradient projections**

The computation of the residual terms are simplified if we introduce the convective and pressure gradient projections  $c_i$  and  $\pi_i$ , respectively defined as

## Tdyn Theoretical Background

$$c_i = r_{m_i} - u_j \frac{\partial u_i}{\partial x_j}$$

$$\pi_i = r_{m_i} - \frac{\partial p}{\partial x_i}$$

We can express the residual terms in Eq. 15 and Eq. 16 in terms of  $c_i$  and  $\pi_i$ , respectively which then become additional variables. The system of integral equations is now augmented in the necessary number by imposing that the residuals vanish (in average sense). This gives the final system of governing equations as:

$$\int_{\Omega} v_i \rho \left( \frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} \right) + \frac{\partial v_i}{\partial x_j} \left( \mu \frac{\partial u_i}{\partial x_j} - \delta_{ij} p \right) d\Omega - \int_{\Omega} v_i \rho f_i d\Omega -$$

$$- \int_{\Gamma_t} v_i t_i d\Gamma + \int_{\Omega} \frac{h_k}{2} \rho \frac{\partial v_i}{\partial x_k} \left( u_j \frac{\partial u_i}{\partial x_j} + c_i \right) d\Omega = 0$$

$$\int_{\Omega} q \frac{\partial u_i}{\partial x_i} d\Omega + \int_{\Omega} \left[ \sum_{i=1}^N \tau_i \frac{\partial q}{\partial x_i} \left( \frac{\partial p}{\partial x_i} + \pi_i \right) \right] d\Omega = 0$$

$$\int_{\Omega} b_i \left( u_j \frac{\partial u_i}{\partial x_j} + c_i \right) d\Omega = 0 \quad \text{no sum in } i$$

$$\int_{\Omega} w_i \left( \frac{\partial p}{\partial x_i} + \pi_i \right) d\Omega = 0 \quad \text{no sum in } i$$

### **Eq. 17 Final system of governing equations**

Being  $i, j, k = 1 \div N$  and  $b_i, w_i$  the appropriate weighting functions.

### **Monolithic time integration scheme**

In this section an implicit monolithic time integration scheme, based on a predictor corrector scheme for the integration of Eq. 17 is presented.

Let us first discretize in time the stabilized momentum Eq. 10, using the trapezoidal rule (or  $\theta$  method) as (see Zienkiewicz 1995)

$$\rho \left( \frac{u_i^{n+1} - u_i^n}{\delta t} + \frac{\partial}{\partial x_i} (u_i u_j)^{n+\theta} \right) + \frac{\partial p^{n+1}}{\partial x_i} - \frac{\partial s_{ij}^{n+\theta}}{\partial x_j} - \rho f_i^{n+\theta} - \frac{1}{2} \rho h_{mj} \frac{\partial}{\partial x_j} \left( u_j^{n+\theta} \frac{\partial u_i^{n+\theta}}{\partial x_j} + c_i^{n+\theta} \right) = 0$$

**Eq. 18 Discretization in time of the stabilized momentum equation**

where superscripts  $n$  and  $\theta$  refer to the time step and to the trapezoidal rule discretization parameter, respectively. For  $\theta = 1$  the standard backward Euler scheme is obtained, which has a temporal error of  $O(\delta t)$ . The value  $\theta = 0.5$  gives a standard Crank Nicholson scheme, which is second order accurate in time  $O(\delta t^2)$ .

An implicit fractional step method can be simply derived by splitting Eq. 18 as follows:

$$\rho \left( \frac{u_i^{*,n+1} - u_i^n}{\delta t} + \frac{\partial}{\partial x_i} (u_i u_j)^{n+\theta} \right) + \frac{\partial p^{n+1}}{\partial x_i} - \frac{\partial s_{ij}^{n+\theta}}{\partial x_j} - \rho f_i^{n+\theta} - \frac{1}{2} \rho h_{mj} \frac{\partial}{\partial x_j} \left( u_j^{n+\theta} \frac{\partial u_i^{n+\theta}}{\partial x_j} + c_i^{n+\theta} \right) = 0$$

$$u_i^{n+1} = u_i^{*,n+1} - \frac{\delta t}{\rho} \left( \frac{\partial p^{n+1}}{\partial x_i} - \frac{\partial p^n}{\partial x_i} \right)$$

**Eq. 19 Implicit fractional momentum equation**

On the other hand, substituting the last equation above into Eq. 11 and after some algebra leads to the following alternate mass balance equation

$$\rho \frac{\partial u_i^*}{\partial x_i} - \delta t \frac{\partial^2}{\partial x_i \partial x_i} (p^{n+1} - p^n) + \sum_{i=1}^N \tau_i \frac{\partial}{\partial x_i} \left( \frac{\partial p^{n+1}}{\partial x_i} + \pi_i^{n+1} \right) = 0$$

**Eq. 20 Implicit fractional mass balance equation**

The weighted residual form of above equations can be written as follows

$$\begin{aligned} & \int_{\Omega} v_i \rho \left( \frac{u_i^{*,n+1} - u_i^n}{\Delta t} + u_j^{*,n+\theta} \frac{\partial}{\partial x_j} u_i^{*,n+\theta} \right) d\Omega + \int_{\Omega} \rho \frac{\partial v_i}{\partial x_j} \left( \mu \frac{\partial u_i^{*,n+\theta}}{\partial x_j} - \delta_{ij} p^n \right) d\Omega - \\ & - \int_{\Omega} v_i \rho f_i^{n+\theta} d\Omega - \int_{\Gamma_t} v_i t_i^{n+\theta} d\Gamma + \int_{\Omega} \frac{b_k}{2} \rho \frac{\partial v_i}{\partial x_k} \left( u_j^{*,n+\theta} \frac{\partial u_i^{*,n+\theta}}{\partial x_j} + c_i^n \right) d\Omega = 0 \\ & \int_{\Omega} q \left( \rho \frac{\partial u_i^*}{\partial x_i} - \Delta t \frac{\partial^2}{\partial x_i \partial x_i} (p^{n+1} - p^n) \right) d\Omega + \int_{\Omega} \tau_i \frac{\partial q}{\partial x_i} \left( \frac{\partial p^{n+1}}{\partial x_i} + \pi_i^{n+1} \right) d\Omega = 0 \\ & \int_{\Omega} v_i \left[ u_i^{n+1} - u_i^* + \frac{\Delta t}{\rho} \left( \frac{\partial p^{n+1}}{\partial x_i} - \frac{\partial p^n}{\partial x_i} \right) \right] d\Omega = 0 \quad \text{no sum in } i \end{aligned}$$

**Eq. 21 Weighting residual form of the implicit fractional step scheme**

At this point, it is important to introduce the associated matrix structure corresponding to the variational discrete FEM form of Eq. 21:

## Tdyn Theoretical Background

$$\begin{aligned}
M \frac{1}{\delta t} \bar{U}^{n+1} - M \frac{1}{\delta t} U^n + K(\bar{U}^{n+\theta}) \bar{U}^{n+\theta} + \\
+S_1 \bar{U}^{n+\theta} + S_2 C - GP^n + M_{\Gamma_i} T = F \\
(\delta t L + L^{\tau_2}) P^{n+1} + D^{\tau_2} \Pi + G^T \bar{U}^{n+1} = \delta t L P^n \\
M \frac{1}{\delta t} U^{n+1} - M \frac{1}{\delta t} \bar{U}^{n+1} - GP^{n+1} - GP^n = 0 \\
NU^{n+1} + MC = 0 \\
GP^{n+1} + M\Pi = 0
\end{aligned}$$

Where  $U$ ,  $P$  are the vectors of the nodal velocity and pressure fields,  $T$  is the vector of prescribed tractions and  $\Pi$ ,  $C$  the vectors of convective and pressure gradient projections. Terms denoted by over-bar identify the intermediate velocity obtained from the fractional momentum equation.

The system of equations above includes an error due to the splitting of the momentum equation. This error can be eliminated by considering the analogous system of equations

$$\begin{aligned}
M \frac{1}{\delta t} U_{i+1}^{n+1} - M \frac{1}{\delta t} U_i^n + K(U_i^{n+\theta}) U_{i+1}^{n+\theta} + \\
+S_1 U_{i+1}^{n+\theta} + S_2 C_i - GP_i^{n+1} + M_{\Gamma_i} T = F \\
\delta t L [P_{i+1}^{n+1} - P_i^{n+1}] + L^{\tau_2} P_{i+1}^{n+1} + D^{\tau_2} \Pi + G^T U_{i+1}^{n+1} = 0 \\
NU_{i+1}^{n+1} + MC_{i+1} = 0 \\
GP_{i+1}^{n+1} + M\Pi_{i+1} = 0
\end{aligned}$$

### ***Eq. 22 Monolithic time integration scheme***

Where  $i$  is the iteration counter of the monolithic scheme. Basically, in this final formulation, the convergence of the resulting monolithic uncoupled scheme is enforced by the first term of the second equation of Eq. 22 (see Soto 2001).

**Compressible flows**

In the previous sections the basic of the incompressible flow solver implemented in Tdyn has been introduced. The extension of that algorithm to compressible flows is straightforward.

Eq. 20 can be easily modified to take into account the compressibility of the flow. The resulting equation is (see M. Vázquez, 1999)

$$\rho \frac{\partial u_i^*}{\partial x_i} + \frac{\alpha}{\delta t} (p^{n+1} - p^n) - \delta t \frac{\partial^2}{\partial x_i \partial x_i} (p^{n+1} - p^n) + \sum_{i=1}^N \tau_i \frac{\partial}{\partial x_i} \left( \frac{\partial p^{n+1}}{\partial x_i} + \pi_i^{n+1} \right) = f$$

**Eq. 23 Implicit fractional mass balance equation (compressible)**

In above equation the terms  $\alpha, f$  depends on the compressibility law used.

**Stabilised Heat Transfer Equations**

The Finite Increment Calculus theory presented above is also valid for the heat transfer balance equations Eq. 3. The stabilised form of the governing differential equations in 3 dimensions are written as follows:

$$r_F - \frac{h_{Fj}}{2} \frac{\partial r_F}{\partial x_j} = 0 \quad \text{on } \Omega_F, j = 1, 2, 3.$$

$$r_S - \frac{h_{Sj}}{2} \frac{\partial r_S}{\partial x_j} = 0 \quad \text{on } \Omega_S, j = 1, 2, 3.$$

**Eq. 24 Stabilised heat transfer balance**

where

$$r_F = \rho C_P \left( \frac{\partial \theta}{\partial t} + (\mathbf{u} \cdot \nabla) \theta \right) - k_F \Delta \theta - \rho q_F$$

$$r_S = \rho_S C \frac{\partial \theta}{\partial t} - \nabla \cdot (k_S \nabla \theta) - \rho_S q_S$$

and the new boundary conditions are:

$$\theta = \theta_c \quad \text{in} \quad \Gamma_\theta \times (0, t)$$

$$nk_F \nabla \theta + n \frac{h_{Fj}}{2} r_F = f_F \quad \text{in} \quad \Gamma_F \times (0, T)$$

$$nk_S \nabla \theta + n \frac{h_{Sj}}{2} r_S = f_S \quad \text{in} \quad \Gamma_S \times (0, T)$$

$$\theta(\mathbf{x}, 0) = \theta_0(\mathbf{x}) \quad \text{in} \quad \Omega \times \{0\}$$

***Eq. 25 Stabilised Heat transfer boundary conditions***

The stabilised Heat transfer balance equations are discretised in time in the standard way, to give the following equations:

$$\theta^{j+1, n+1} = \theta^n - \frac{\Delta t}{\rho C_P} [\rho C_P (\mathbf{u} \cdot \nabla) \theta - \nabla \cdot (k_F \nabla \theta) - \rho q_F]^{j, n+\theta}$$

$$\theta^{j+1, n+1} = \theta^n - \frac{\Delta t}{\rho C} [\nabla \cdot (k_S \nabla \theta) - \rho q_S]^{j, n+\theta}$$

***Eq. 26 Discretised Heat transfer equations***

## Turbulence Solvers

At high Reynolds numbers the flow will certainly be turbulent, and the resulting fluctuations in velocities need to be taken account of in the calculations. A process known as Reynolds averaging [3] is applied to the governing equations whereby the velocities,  $u_i$  are split into mean and a fluctuating component, where the fluctuating component,  $u_i'$ , is defined by:

## Tdyn Theoretical Background

$$u_i = \bar{u}_i + u_i'$$

Where

$$\bar{u}_i(x, t) = \frac{1}{T} \int_{-T/2}^{T/2} u_i(x, t + \tau) d\tau$$

This leads to extra terms in the governing equations that can be written in as a function of ‘Reynolds stresses’ tensor, defined in Cartesian coordinates as:

$$\tau_{ij}^R = -\overline{\rho u_i' u_j'}$$

Since the process of Reynolds averaging has introduced extra terms into the governing equations, we need extra information to solve the system of equations. To relate this terms to the other flow variables, a model of the turbulence is required.

A large number of turbulence models exist of varying complexity, from the simple algebraic models to those based on two partial differential equations. The more complex models have increased accuracy at the expense of longer computational time. It is therefore important to use the simplest model that gives satisfactory results.

Several turbulence models as Smagorinsky, k, k- $\epsilon$ , k- $\omega$ , k-kt and Spalart Allmarax have been implemented in Tdyn. The final form of the so-called Reynolds Averaged Navier Stokes Equations (RANSE) using these models are:

$$r_{mi} - \frac{h_{mj}}{2} \frac{\partial r_{mi}}{\partial x_j} = 0 \quad \text{on } \Omega, i, j = 1, 2, 3. \text{ No sum in } i$$

**Eq. 27 Stabilised Momentum balance**

$$r_d - \frac{h_{dj}}{2} \frac{\partial r_d}{\partial x_j} = 0 \quad \text{on } \Omega, j = 1,2,3$$

**Eq. 28 Stabilised Mass balance**

Where:

$$r_{mi} = \rho \left[ \frac{\partial u_i}{\partial t} + \frac{\partial(u_i u_j)}{\partial x_j} \right] + \frac{\partial p}{\partial x_i} - \frac{\partial}{\partial x_j} \left[ (\mu + \mu_T) \frac{\partial u_i}{\partial x_j} \right] - \rho f_i$$

$$r_d = \rho \frac{\partial u_i}{\partial x_i}$$

Being  $\mu_T$  the so-called eddy viscosity.

The theory behind the above mentioned models may be found in the references, but a basic outline of the three most representative models is given below.

**Zero equation (algebraic) model: The Smagorinsky model**

Zero equation models are the most basic turbulence models. They assume that the turbulence is generated and dissipated in the same place, and so neglect the diffusion and convection of the turbulence. They are based on the idea of turbulence viscosity, and prescribe a turbulence viscosity,  $\nu_t$ , either by means of empirical equations or experiment.

The Smagorinsky turbulence model is based on the idea of large eddy simulations (LES) in which the coherent large-scale structures are modelled directly within the computational mesh, whilst the small scales are modelled with the concept of eddy viscosity [4]. The Reynolds stress is written in terms of the turbulence kinetic energy,  $k$ , and eddy viscosity as:

$$\overline{u_i' u_j'} = \frac{2}{3} k \delta_{ij} - 2\nu_t \varepsilon_{ij}$$

where

## Tdyn Theoretical Background

$$\varepsilon = \frac{1}{2} \left( \nabla \cdot \mathbf{u}' + (\nabla \cdot \mathbf{u}')^T \right)$$

Smagorinsky proposed that the eddy viscosity depends on the mesh density and velocity gradients, and suggested the following expression:

$$\nu_t = \frac{\mu_t}{\rho} = Ch^e \sqrt{\varepsilon_{ij} \varepsilon_{ij}}$$

where  $\nu_t$  is the kinematic eddy viscosity,  $C$  is a constant of the order of 0.001, and  $h^e$  is the element size.

### **One-equation models: The kinetic energy model (k-model)**

One-equation models attempt to model turbulent transport, by developing a differential equation for the transport of one of turbulent quantities. In the kinetic energy model, the velocity scale of the turbulence is taken as the square root of the turbulence kinetic energy,  $k$ . Kolmogorov and Prandtl independently suggested the following relationship between the eddy viscosity, this velocity scale,  $\sqrt{k}$ , and length scale,  $L$ :

$$\nu_t = C_D \sqrt{k} L$$

where

$$k = \frac{1}{2} \overline{\mathbf{u}'^2} = \frac{1}{2} \overline{u'^2 + v'^2 + w'^2}$$

and  $C_D$  is an empirical constant, usually taken as 1.0.

Manipulation of the Reynolds and momentum equations leads to a differential equation for  $k$  [1]. The only gap in this analysis is that left by the length scale,  $L$ . This has to be specified either from experiment, or empirical equation, and in the analysis used here, it has been specified as 1% of the smallest mesh size. Clearly this is the largest downfall of the analysis, and many consider it insufficiently accurate [3]. However, it is more accurate than the zero-equation models and less computationally expensive than the more accurate 2-

equation models, and so it can be a good compromise between computational expense and accuracy, depending on the problem.

***Two-equation models: The k-ε model***

In order to increase the accuracy of our turbulence modelling, it is therefore necessary to develop a second differential transport equation to provide a complete system of closed equations without the need for empirical relationships. The k-ε turbulence model develops two such differential transport equations: one for the turbulence kinetic energy, k, and a second for the turbulent dissipation, ε. As for the k-model, the k-ε model relies on the Prandtl-Kolmogorov expression for the eddy viscosity above. From dimensional arguments, the turbulent dissipation, can be written in terms of the turbulence kinetic energy and the turbulence length scale, and the eddy viscosity written in terms of k and ε:

$$\varepsilon = C_\varepsilon \frac{k^{3/2}}{L}$$
$$\nu_t = C_\nu \frac{k^2}{\varepsilon}$$

Where  $C_\varepsilon$  and  $C_\nu$  are empirical constants, both are taken to have a value of 0.09.

In a similar way to that of the k-model, differential transport equations can be formulated for k and ε [3], thus closing the system of equations, without having to empirically define any of the turbulent quantities.

Since this method directly models the transport of all of the turbulent quantities, it is the most accurate. However, it involves solving 2 differential equations and is so the most computationally expensive. It is therefore necessary to evaluate the importance of turbulence in the problem before a model is selected. In many cases the easiest way to do this may be by trying several methods and using the simplest which gives the most accurate results. If the turbulence is a relatively small feature of the flow then it is not necessary to waste computational expense by modelling it with a 2-equation

model. At the same time, a zero-equation model would inadequately model a problem with large areas of turbulence, and in many cases this will lead to instabilities in the simulation.

## Implicit LES turbulence model

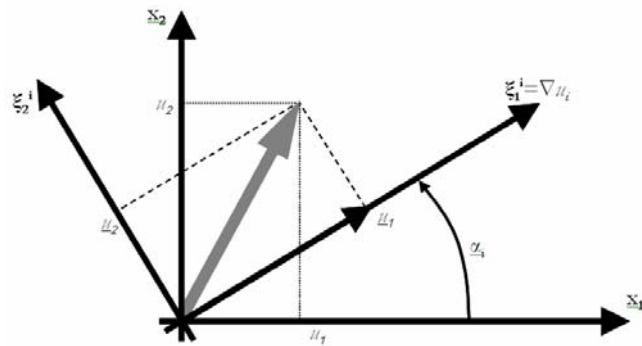
In recent years a significant progress has been carried out in the development of new turbulence models based on the fact that not the entire range of scales of the flow is interesting for the majority of engineering applications. In this type of applications information contained in "the large scales" of the flow is enough to analyse magnitudes of interest as velocity, temperature,...

Therefore, the idea that the global flow behaviour can be correctly approximated without the necessity to approximate the smaller scales correctly, is seen by many authors as a possible great advance in the modelling of turbulence. This fact has originated the design of turbulence models that describe the interaction of small scales with large scales. These models are commonly known as Large Eddy Simulation models (LES). Numerous applications has shown that an extension of the FIC method allow to model low and high Reynolds number flows. In the standard large eddy simulation, a filtering process is applied to the Navier Stokes (Eq. 8). After the filtering a new set of equations are obtained, variables of this new set of equations are the filtered velocities or also called large scale velocities. As a consequence of the filtering process a new term called the subgrid scale tensor appears into the momentum equation. The subgrid scale tensor is defined in function of the large scale velocities and the subscale velocities too. Then it is necessary to model this term in function only of the large scale velocities. Several models exist for the subgrid scale tensor, but basically all of them propose an explicit description of the subgrid scale, an analytic expression in term of large scale velocities. In conclusion, all of these models define the subgrid scale tensor as a new nonlinear viscous term.

Is our understanding that the stabilized FIC formulation, with an adequate evaluation of the characteristic lengths introduces the same effects of a LES model without given an explicit expression of the subgrid scale tensor. All the effects related to turbulence modelling are included into the non-linear stabilization parameters.

**Computation of the characteristic lengths**

The computation of the stabilization parameters is a crucial issue as they affect both the stability and accuracy of the numerical solution. The different procedures to compute the stabilization parameters are typically based on the study of simplified forms of the stabilized equations. Contributions to this topic are reported in (Oñate 1998 and García Espinosa 2005). Despite the relevance of the problem there still lacks a general method to compute the stabilization parameters for all the range of flow situations.



**Figure 2 Decomposition of the velocity for characteristic length evaluation.**

The application of the FIC/FEM formulation to convection-diffusion problems with sharp arbitrary gradients has shown that the stabilizing FIC terms can accurately capture the high gradient zones in the vicinity of the domain edges (boundary layers) as well as the sharp gradients appearing randomly in the interior of the domain (Oñate 1998). The FIC/FEM thus reproduces the best features of both the so called transverse (cross-wind) dissipation or shock capturing methods.

The approach proposed in this work is based on a standard decomposition of the characteristic lengths for every component of the momentum equation as

$$h_i = \alpha \frac{\mathbf{u}}{|\mathbf{u}|} + \beta \frac{\nabla u_i}{|\nabla u_i|}$$

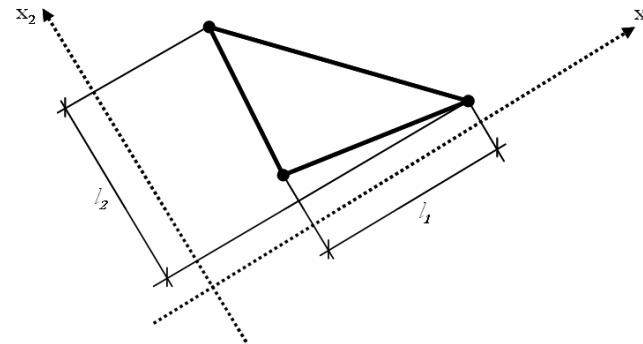
**Eq. 29 Standard decomposition of the characteristic lengths**

Being  $\alpha$  the projection of the characteristic length component in the direction of the velocity and  $\beta$  the projection in the direction of the gradient of every velocity component. The decomposition defined by Eq. 29 can be demonstrated to be unique if the characteristic lengths are understood as tensors written in a system of coordinates aligned with the principal curvature directions of the solution. Obviously, Eq. 29 is a simplification of that approach. However, it can be easily demonstrated that the variational discrete FEM form of the resulting momentum equations are equivalent using one or the other approach.

Then, the characteristic lengths are calculated as follows

$$\alpha = \left( \coth \gamma^u - \frac{1}{\gamma^u} \right) u_j l_j, \quad \gamma^u = \frac{u_j l_j}{2\mu}$$

Where  $l_j$  are the maximum projections of the element on every coordinate axis (see Figure 3).



**Figure 3 Calculation of  $l_j$  in a triangular element.**

and

$$\beta = \left( \coth \gamma_{ij} - \frac{1}{\bar{\gamma}_{ij}} \right) l^{\nabla u_i}, \quad \gamma_{ij} = u_i \frac{\nabla u_i}{|\nabla u_i|} \frac{l^{\nabla u_i}}{2\mu}$$

$$l^{\nabla u_i} = \frac{\nabla u_i}{|\nabla u_i|} l_j, \quad j = 1, 2$$

As for the length parameters  $h_i^d$  in the mass conservation equation, the simplest assumption  $h_i^d = h^d$  has been taken.

The overall stabilization terms introduced by the FIC formulation above presented have the intrinsic capacity to ensure physically sound numerical solutions for a wide spectrum of Reynolds numbers without the need of introducing additional turbulence modelling terms.

## Overlapping domain decomposition level set

This section introduces one of the algorithm implemented in Tdyn for the analysis of free surface flows. The main innovation of this method is the application of domain decomposition concept in the statement of the problem, in order to increase accuracy in the capture of free surface as well as in the resolution of governing equations in the interface between the two fluids. Free surface capturing is based on the solution of a level set equation, while Navier Stokes equations are solved using the iterative monolithic predictor-corrector algorithm presented above, where the correction step is based on the imposition of the divergence free condition in the velocity field by means of the solution of a scalar equation for the pressure.

### Statement of the problem

The velocity and pressure fields of two incompressible and immiscible fluids moving in the domain  $\Omega \subset R^d (d=2,3)$  can be described by the incompressible Navier Stokes equations for multiphase flows, also known as non-homogeneous incompressible Navier Stokes equations:

$$\begin{aligned}\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_j}(\rho u_j) &= 0 \\ \frac{\partial \rho u_i}{\partial t} + \frac{\partial}{\partial x_j}(\rho u_i u_j) + \frac{\partial p}{\partial x_i} - \frac{\partial \tau_{ij}}{\partial x_j} &= \rho f_i \\ \frac{\partial u_i}{\partial x_i} &= 0\end{aligned}$$

**Eq. 30 Non-homogeneous incompressible Navier Stokes equations**

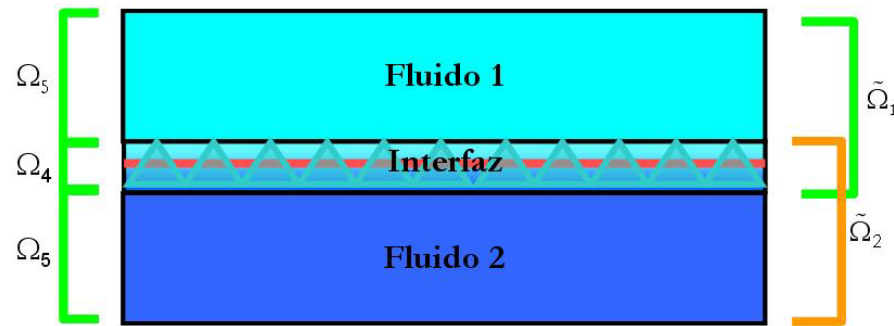
Where  $1 \leq i, j \leq d$ ,  $\rho$  is the fluid density field,  $u_i$  is the  $i$ th component of the velocity field  $u$  in the global reference system  $x_i$ ,  $p$  is the pressure field and  $\tau$  is the viscous stress tensor defined by:

$$\tau_{ij} = \mu(\partial_i u_j + \partial_j u_i)$$

Where  $\mu$  is the dynamic viscosity.

Let  $\Omega_1 = \{x \in \Omega | x \in \text{Fluid1}\}$  be the part of the domain  $\Omega$  occupied by the fluid number 1 and let  $\Omega_2 = \{x \in \Omega | x \in \text{Fluid2}\}$  be the part of the domain  $\Omega$  occupied by fluid number 2. Therefore  $\Omega_1, \Omega_2$  are two disjoint subdomains of  $\Omega$ . Therefore (see Figure 4)

$$\Omega = \text{int}(\overline{\Omega_1 \cap \Omega_2})$$



**Figure 4 Domain decomposition**

The system of Eq. 30 must be completed with the necessary initial and boundary conditions, as shown below.

It is usual in the literature to consider that the first equation of the Eq. 30 is equivalent to impose a divergence free velocity field, since the density is taken as a constant. However, in the case of multiphase incompressible flows, density can not be consider constant in  $\Omega \times (0, T)$ . Actually, it is possible to define  $\rho, \mu$  fields as follows:

$$\rho, \mu = \begin{cases} \rho_1, \mu_1 & x \in \Omega_1 \\ \rho_2, \mu_2 & x \in \Omega_2 \end{cases}$$

Let  $\psi : \Omega \times (0, T) \rightarrow R$  be a function, in below named Level Set function, defined as follows:

$$\psi(x, t) = \begin{cases} d(x, t) & x \in \Omega_1 \\ 0 & x \in \Gamma \\ -d(x, t) & x \in \Omega_2 \end{cases}$$

## Tdyn Theoretical Background

Where  $d(x,t)$  is the distance to the interface between the two fluids, denoted by  $\Gamma$ , of the point  $x$  in the time instant  $t$ . From above definition it is trivially obtained that:

$$\Gamma = \{x \in \Omega \mid \psi(x, \cdot) = 0\}$$

Since the level set 0 identify the free surface between the two fluids, the following relations can be obtained:

$$n(x,t) = \nabla \psi|_{(x,t)} ; \kappa(x,t) = \nabla \cdot (n(x,t))$$

Where  $n$  is the normal vector to the interface  $\Gamma$ , oriented from fluid 1 to fluid 2 and  $\kappa$  is the curvature of the free surface. In order to obtain above relations it has been assumed that function  $\psi$  accomplish:

$$|\nabla \psi| = 1 \quad \forall (x,t) \in \Omega \times (0,T)$$

Therefore, it is possible to redefine density and viscosity as follows:

$$\rho, \mu = \begin{cases} \rho_1, \mu_1 & \psi > 0 \\ \rho_2, \mu_2 & \psi < 0 \end{cases}$$

Let us write the density fields in terms of the level set function  $\psi$  as

$$\rho(x,t) = \rho(\psi(x,t)) \quad \forall (x,t) \in \Omega \times (0,T)$$

Then, density derivatives can be written as

$$\frac{\partial \rho}{\partial t} = \frac{\partial \rho}{\partial \psi} \frac{\partial \psi}{\partial t}, \quad \frac{\partial \rho}{\partial x_i} = \frac{\partial \rho}{\partial \psi} \frac{\partial \psi}{\partial x_i}$$

Inserting above relation in the first line of Eq. 30 gives

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_i} (\rho u_i) \Big|_{\frac{\partial u_i}{\partial x_i} = 0} = \frac{\partial \rho}{\partial t} + u_i \frac{\partial \rho}{\partial x_i} = \frac{\partial \rho}{\partial \psi} \frac{\partial \psi}{\partial t} + \frac{\partial \rho}{\partial \psi} u_i \frac{\partial \psi}{\partial x_i} = \frac{\partial \rho}{\partial \psi} \left[ \frac{\partial \psi}{\partial t} + u_i \frac{\partial \psi}{\partial x_i} \right] = 0$$

What gives as a result that the multiphase Navier Stokes problem are equivalent to solve the following system of equations:

$$\begin{aligned} \frac{\partial \rho u_i}{\partial t} + \frac{\partial}{\partial x_j} (\rho u_i u_j) + \frac{\partial p}{\partial x_i} - \frac{\partial \tau_{ij}}{\partial x_j} &= \rho f_i \\ \frac{\partial u_i}{\partial x_i} &= 0 \end{aligned}$$

**Eq. 31 Non-homogeneous Navier Stokes equations**

Coupled with the equation

$$\frac{\partial \psi}{\partial t} + u_i \frac{\partial \psi}{\partial x_i} = 0$$

**Eq. 32 Level set equation**

Eq. 32 defines the transport of the level set function due to the velocity field obtained by solving Eq. 31.

As a conclusion, the free surface capturing problem can be described by above equations. In this formulation, the interface between the two fluids is defined by the level 0 of  $\psi$ .

Denoting by over-bar the prescribed values, the boundary conditions of the above presented problem, to be considered are

$$\begin{aligned}
 & u = \bar{u} \quad \text{on } \Gamma_u \\
 & p = \bar{p}, \quad n_j \tau_{ij} = \bar{t}_i \quad \text{on } \Gamma_p \\
 & \left. \begin{aligned}
 & u_j n_j = \bar{u}_n, \quad n_j \tau_{ij} g_i = \bar{t}_1 \\
 & n_j \tau_{ij} s_i = \bar{t}_2
 \end{aligned} \right\} \quad \text{on } \Gamma_\tau
 \end{aligned}$$

**Eq. 33 Boundary conditions of the ODD level set problem**

Where the boundary  $\partial\Omega$  of the domain  $\Omega$  has been split in three disjoint sets:  $\Gamma_u, \Gamma_p$  where the Dirichlet and Neumann boundary conditions are imposed and  $\Gamma_\tau$  where the Robin conditions for the velocity are set. In above vectors  $g, s$  span the space tangent to  $\Gamma_\tau$ . In a similar way, the boundary conditions for Eq. 32 are defined

$$\psi = \bar{\psi} \quad \text{on } \Gamma_u$$

Finally, the initial conditions of the problem are

$$u = u_0 \quad \text{on } \Omega, \quad \psi = \psi_0 \quad \text{on } \Omega$$

Where  $\Gamma_0 = \{x \in \Omega \mid \psi_0(x) = 0\}$  defines the initial position of the free surface between the two fluids.

## References

- [1] R. Codina "A discontinuity capturing crosswind dissipation for the finite element solution of the convection diffusion equation", *Comp. Meth. Appl. Mech. Eng.*, pages 110:325-342, 1993.
- [2] J. García, "A Finite Element Method for the Hydrodynamic Analysis of Naval Structures" (in Spanish), Ph.D. Thesis, Univ. Politècnica de Catalunya, Barcelona, December 1999.
- [3] C. Hirsch "Numerical computation of internal and external flows", John Wiley and Sons Ltd., Chichester, UK, 1991.
- [4] C. Kleinstreuer "Engineering Fluid Dynamics, An Interdisciplinary Systems Approach", Cambridge University Press, Cambridge, UK, 1997.
- [5] E. Oñate "A stabilised finite element method for incompressible viscous flows using a finite increment calculus formulation. Research report No. 150", CIMNE, Barcelona, January 1999.
- [6] García, J., Oñate, E., Sierra, H., Sacco, C. and Idelsohn, S. "A Stabilised Numerical Method for Analysis of Ship Hydrodynamics". Proceedings Eccomas Conference on CFD, 7-11 September 1998, Athens, John Wiley, 1998.
- [7] E. Oñate, M. Manzan, "Stabilization techniques for finite element analysis of convection-diffusion problems", Publication CIMNE No. 183, Barcelona, February 2000.
- [8] O.C. Zienkiewicz, T.L. Taylor "The finite element method". McGrawHill Book Company, Volume 1, Maidenhead, UK, 1989.
- [9] E. Oñate, J. García and S. Idelsohn, "On the stabilisation of numerical solutions of advective-diffusive transport and fluid flow problems". *Comp.Meth. Appl. Mech. Engng*, 5, 233-267, 1998.

## Tdyn Theoretical Background

[10] J.E. Bardina, P.G. Huang and T.J. Coakley, "Turbulence Modelling Validation, Testing, and Development". NASA technical Memorandum 110446, Moffett Field, California, April 1997.

[11] D.C. Wilcox "Turbulence Modelling for CFD". DCW industries, Inc. 5354 Palm Drive, La Cañada, California, 1993.

[12] M. Vázquez, R. Codina and O.C. Zienkiewicz "Numerical modelling of Compressible Laminar and Turbulent Flow: The CBS Algorithm". Monograph CIMNE N° 50, Barcelona, April 1999.

[13] J. García-Espinosa and E. Oñate. "An Unstructured Finite Element Solver for Ship Hydrodynamics". *Jnl. Appl. Mech.* Vol. 70: 18-26. (2003).

[14] E. Oñate, J. García-Espinosa y S. Idelsohn. "Ship Hydrodynamics". *Encyclopedia of Computational Mechanics*. Eds. E. Stein, R. De Borst, T.J.R. Hughes. John Wiley & Sons (2004).

[15] O.C. Zienkiewicz and Codina, R., "A general algorithm for compressible and incompressible flow. Part I, The Split, Characteristic Based Scheme", *Int. Journ. Num. Meth. Fluids*, 20, 869-885 (1995).

[16] O. Soto, R. Lohner, J Cebral. An Implicit Monolithic Time Accurate Finite Element Scheme for Incompressible Flow Problems. 15th AIAA Computational Fluid Dynamics Conference. Anaheim (USA) 2001.

[17] J. García-Espinosa, E. Oñate and J. Bloch Helmers. "Advances in the Finite Element Formulation for Naval Hydrodynamics Problems". *International Conference on Computational Methods in Marine Engineering (Marine 2005)*. June 2005. Oslo (Norway).