

**POLITECNICO DI MILANO**

Facoltà di Ingegneria Industriale e dell'Informazione

Dipartimento di Chimica, Materiali e Ingegneria Chimica

“Giulio Natta”

Dipartimento di Energia



**GENERAZIONE AUTOMATICA DI GRIGLIE DI  
CALCOLO PER LETTI IMPACCATI DI SFERE  
CATALITICHE**

Relatori: Prof. Alberto CUOCI

Prof. Matteo MAESTRI

Correlatore: Ing. Stefano REBUGHINI

Tesi di Laurea Magistrale in Ingegneria Chimica di:

Salvatore RAFFA 801931

Anno Accademico 2014-2015



# Sommario

I reattori chimici si presentano in diverse tipologie per fronteggiare situazioni operative estremamente diversificate, sia per la natura delle specie chimiche coinvolte (reagenti e prodotti della reazione), sia per le condizioni fisiche alle quali operano. Un esempio ci viene offerto dai reattori gas-solido, nei quali vengono condotte le reazioni catalitiche eterogenee.

Oggi, circa il 90% dei processi industriali coinvolgono reazioni catalitiche condotte in reattori eterogenei a letto fisso. La classe più comune di reattori eterogenei a letto fisso è quella dei reattori tubolari impaccati di sfere catalitiche. Tali reattori consistono in sfere catalitiche impaccate all'interno di un vessel (o tubo). Nel momento in cui il fluido reagente attraversa il letto catalitico, avvengono fenomeni complessi di natura chimica e fisica a diverse scale spaziali. Alla scala più piccola (micro-scala), quella intraparticellare, i reagenti diffondono all'interno del granulo, si adsorbono e reagiscono sui siti attivi presenti nel catalizzatore, poi desorbono ed infine diffondono per lasciare i pori del catalizzatore. Alla scala del reattore invece, (macro-scala) la convezione è il meccanismo dominante di trasporto di materia e calore ed è estremamente complessa da studiare, a causa della presenza dell'impaccamento casuale di sfere.

Un potente mezzo in grado di descrivere in modo dettagliato tali fenomeni è la fluidodinamica computazionale o CFD. La fluidodinamica computazionale, risolve numericamente le equazioni di trasporto di quantità di moto, energia e di materia delle specie chimiche, su una geometria arbitrariamente complessa attraverso delle opportune tecniche di discretizzazione spaziale e temporale. E' per questo uno strumento estremamente utile e versatile qualora si vogliano studiare sistemi complessi come i reattori a letto fisso con cinetiche dettagliate, sia omogenee che eterogenee. La CFD, sebbene ancora non sia in grado di sostituire completamente l'attività sperimentale, garantisce un sensibile risparmio di tempo ed anche economico rispetto alla conduzione in laboratorio dello studio da affrontare. Infatti, ogni qualvolta si voglia condurre lo studio in laboratorio su una nuova geometria,

---

occorre costruirne materialmente l'apparecchiatura, utilizzare gli opportuni strumenti di misura e quindi sottoporlo al test. Attraverso la CFD è sufficiente condurre un numero di studi inferiore rispetto all'attività sperimentale, al fine di ottenere informazioni ad un livello di dettaglio arbitrario dello stesso caso da studiare.

Attraverso la CFD la discretizzazione del dominio geometrico del sistema, ottenendo un dominio computazionale costituito da celle chiamato griglia di calcolo (o *mesh*). Rispetto all'indagine sperimentale, la generazione, per quanto complessa, richiederà tempi di gran lunga inferiori. Sebbene la CFD offra anche un grado arbitrario di complessità per la descrizione dei fenomeni di trasporto che avvengono nei reattori catalitici, d'altro canto, pone dei limiti per lo studio dei reattori di interesse industriale a causa delle elevate risorse computazionali richieste. Uno dei maggiori limiti della CFD è la generazione della griglia di calcolo per reattori impaccati di sfere catalitiche. Le principali difficoltà sono soprattutto associate alla presenza dei punti di contatto sfera/sfera e sfera/parete. Infatti, dal punto di vista computazionale, avere un punto di contatto significa avere area nulla. Questo effetto si ripercuote sulla difficile generazione della griglia di calcolo, generando celle distorte vicino i punti di contatto. La distorsione di una cella, viene valutata attraverso la *skewness*, che indica la deviazione degli angoli della cella tetraedrica (o esaedrica) rispetto ad una cella ideale di un tetraedro equilatero (o cubo). Valori elevati della *skewness* introducono errori di discretizzazione molto significativi nel processo di risoluzione delle equazioni differenziali. Questo, impatta sull'accuratezza della soluzione numerica della simulazione CFD e problemi di convergenza soprattutto per alti valori del numero di Reynolds.

Il presente lavoro di Tesi ha come obiettivo l'implementazione di metodologie che consentano la generazione automatica di griglie di calcolo per letti impaccati di sfere catalitiche volti a ridurre elevati la *skewness* vicino i punti di contatto. Sono stati implementati e confrontati 4 diversi metodi di modifica dei punti di contatto: due metodi globali, chiamati *gap* e *overlap*, e due locali, chiamati *bridge* e *cap*. Nel più recente studio sull'argomento [1] è stato valutato l'effetto delle modifiche sulla griglia di calcolo nel punto di contatto. Tale studio è stato però effettuato senza tenere in considerazione i fenomeni di scambio di materia e di calore associato a

reazioni catalitiche eterogenee, ragione per cui l'analisi proposta nel seguente lavoro di Tesi costituisce il primo obiettivo affrontato nel presente lavoro.

Le geometrie e la griglie di calcolo sono state generate attraverso il software open-source SALOME. Grazie alle caratteristiche di SALOME è stato possibile modificare automaticamente la geometria dei punti di contatto e generare così la griglia di calcolo desiderata attraverso degli *script*.

Inizialmente, lo studio si è concentrato su un semplice caso sfera/sfera utilizzando i risolutori `catalyticFOAM` [2,3] e `chtMultiRegionFOAM` [4]. Il primo è un codice capace di accoppiare la fluidodinamica computazionale alla micro-cinetica dettagliata. Le reazioni eterogenee interessano solo la superficie del catalizzatore, trascurando la modellazione della fase solida. Il secondo è un solver nativo di `OpenFOAM` ed è in grado di risolvere problemi di scambio termico. Questa prima analisi è stata fondamentale per comprendere l'effetto delle modifiche dei punti di contatto sui meccanismi di scambio di materia e di calore tra due sfere catalitiche. Successivamente, lo studio è stato rivolto all'analisi di parti più o meno estese di letti impaccati. In questa seconda configurazione geometrica proposta, i punti di contatto tra le sfere si trovano nelle zone stagnanti del reattore, in modo da indagare i meccanismi di scambio di materia e di calore locali tra le sfere. Il confronto tra i risultati ottenuti dai metodi di modifica dei punti di contatto e i risultati della simulazione fluidodinamica per i casi di riferimento, ha permesso di valutare l'accuratezza dei diversi metodi usati per lo studio dei fenomeni di scambio di materia e di calore, dimostrando che la reattività chimica è influenzata dalla variazione della superficie catalitica delle sfere, mentre lo scambio termico dipende dalle conducibilità termiche utilizzate per modellare i punti di contatto.

Nella terza parte della Tesi, lo studio si è focalizzato sul confronto dei metodi di modifica locale *bridge* e *cap* per il caso sfera/sfera, dove i fenomeni di scambio termico sono accoppiati alla reazione eterogenea superficiale. Questo studio è stato fondamentale per comprendere quale dei due approcci fosse il più appropriato per modellare efficacemente i punti di contatti in reattori eterogenei a letto fisso, dove coesistono reazioni catalitiche eterogenee e fenomeni di scambio termico per effetto del calore di reazione.

Infine, per verificare i risultati ottenuti anche per sistemi più complessi, come reattori impaccati di sfere catalitiche, è stato effettuato uno studio sistematico volto ad indagare l'effetto di modifica dei punti di contatto sulle proprietà geometriche dei letti impaccati di sfere. Per affrontare questo studio è stato pertanto necessario creare degli algoritmi che fossero in grado di generare automaticamente la griglia di calcolo per letti impaccati di sfere utilizzando i quattro metodi descritti.

In conclusione si è dunque dimostrata la possibilità di generare automaticamente delle griglie di calcolo per letti impaccati di sfere, caratterizzate da un sufficiente livello di accuratezza, tale da consentire il loro utilizzo per analisi fluidodinamiche accoppiate a reazioni chimiche eterogenee.

# Abstract

Chemical reactors come in several types to address operational situations and are extremely diversified, both by the nature of the chemical species involved (reactants and products of the reaction), as for the physical conditions in which they operate. An example is offered by the gas-solid reactors, which are carried out in heterogeneous catalytic reactions.

Today, approximately 90% of the industrial processes involving catalytic reactions performed in heterogeneous catalytic fixed-bed reactors. The most common class of tubular heterogeneous fixed bed reactors is the catalytic packed reactor of spheres. These reactors consist of catalytic spheres packed inside a vessel (or tube). When the reactant fluid passes through the catalytic bed, complex chemical and physical phenomena occur at different spatial scales. On the smaller scale (micro-scale), the intraparticle, reagents diffuse inside the granule, adsorb and react on the active sites in the catalyst, then exude and finally diffuse back to leave the pores of the catalyst. On the large scale of the reactor, (macro-scale) convection is the dominant mechanism of transport of matter and heat and is extremely complex to study, due to the random presence of the packing of spheres.

A powerful tool able to describe in detail these phenomena is computational fluid dynamics or CFD. Computational fluid dynamics, solves numerically transport equations of momentum, energy and the chemical species, on an arbitrarily complex geometry through the appropriate spatial and temporal discretization techniques. This is an extremely useful and versatile tool if you want to study complex systems such as fixed bed reactors with detailed kinetics, both homogeneous and heterogeneous. The CFD, although still not able to completely replace the experimental activity, guarantees a significant saving of time compared to practical experiment in the laboratory of the study. In fact, whenever you want to carry out the study in the laboratory of a new geometry the model must be physically built first, the appropriate measuring instruments must be put in place and then submit it to the test.

Using CFD is enough to conduct a number of studies lower than the experimental activity, in order to obtain information at an arbitrary level of detail of the same case in study.

The numerical investigation requires the discretization of the system's geometric domain, resulting in a computational domain which consists of cells called computational grid (or *mesh*). Compared to the experimental investigation, the grid generation, however complex, requires time less time. Although the CFD provides an arbitrary degree of complexity for the description of transport phenomena that occur in catalytic reactors, on the other hand, it sets limits for the study of the reactors of industrial interest due to the high computational resources required. One of the major limitations of CFD is the generation of the computational grid for catalytic packed bed reactors of spheres. The main difficulties are mainly associated with the presence of the contact points sphere/sphere and wall/sphere. In fact, from the computational point of view, to have a contact point means having an area equal to zero. This effect is reflected on the difficult generation of the computational grid, generating distorted cells near the contact points. The distortion of a cell, is evaluated through the *skewness*, which indicates the deviation of the angles of the tetrahedron cell (or quadrangle) with respect to an ideal cell of an equilateral tetrahedron (or cube). Therefore, high levels of *skewness* introduce very significant errors of discretization in the process of solving differential equations. This affects the accuracy of the numerical solution of the CFD simulation and convergence problems especially for high values of the Reynolds number.

The goal of this Thesis is the implementation of methods which enable the automatic generation of computational grids for catalytic packed bed reactors of spheres aimed at reducing elevated values of *skewness* near the contact points.

Four different methods of changing contact points have been implemented and compared. These methods are two global methods, called *gap* and *overlap*, and two local, called *bridge* and *cap*. In the most recent study on the subject [1] the effect of modifications on grid at the contact point was evaluated. The study was carried out, however, without taking into account the phenomena of chemical reactivity, which is why this analysis is the first objective addressed in this work.



Geometries and grids were generated through the open-source software SALOME. Thanks to the characteristics of SALOME it was possible to automatically change the geometry of the contact points and generate the desired computational grid by scripting.

Initially, the study focused on a simple case sphere/sphere using `catalyticFOAM` [2,3] and `chtMultiRegionFOAM` [4] solvers. The first is a code capable of coupling computational fluid dynamics and the micro-detailed kinetic. The heterogeneous reactions take place only on the surface of the catalyst, neglecting the modeling of the solid phase. The second is a native of `OpenFOAM` solver and is able to solve problems of heat transfer. This first analysis was critical to understand the effect of changes the contact points on the heat and mass transfer phenomena between two catalytic spheres. Subsequently, the study has been devoted to the analysis of parts more or less extensive of packed beds. In this second geometric configuration, the contact points between the spheres are found in stagnant zones of the reactor, in order to investigate the local mechanisms of mass and heat transfer between spheres. The comparison between the results obtained from the methods of modification of the contact points, with the results of fluid dynamics simulation for the reference cases, have enabled us to assess the accuracy of the different methods used to study the mass and heat transfer phenomena, demonstrating that the chemical reactivity is influenced by the variation of the catalytic surface of the spheres, while the heat transfer depends on the thermal conductivity used to model the contact points.

In the third part of this work, the study focused on comparing the local methods *bridge* and *cap* for the sphere/sphere case, where the heat transfer phenomena are coupled to the heterogeneous reaction surface. This study was essential to understand which of the two approaches is the most appropriate to model effectively the contact points in catalytic fixed-bed reactors, where heterogeneous catalytic reactions and heat transfer phenomena coexist due to the heat of the reaction.

Finally, to check the results for more complex systems, such as catalytic packed bed reactors of spheres, a systematic study aimed at investigating the effect of changing the contact points on the geometric properties of the packed bed of spheres was

carried out. To address this study, it was therefore necessary to create *script* that would be able to automatically generate the computational grid for packed beds of spheres using the four methods.

In conclusion, the possibility to generate automatically the computational grids for packed beds of spheres has been demonstrated, characterized by a sufficient level of accuracy, such as to allow their use for fluid flow analysis coupled to heterogeneous chemical reactions.

- [1] Dixon A.G., Nijemeisland M., Stitt E.H., “*Systematic meshdevelopment for 3D CFD simulation of fixed beds: Contact points study*”, Computers and Chemical Engineering, 48(10), pp. 135-153, (2013).
- [2] Maestri M., Cuoci A., “*Coupling CFD with detailed microkinetic modeling in heterogeneous catalysis*”, Chemical Engineering Science, 96, pp. 106-117, (2013).
- [3] Maestri M., Cuoci A., [www.catalyticfoam.polimi.it](http://www.catalyticfoam.polimi.it).
- [4] <http://www.openfoam.com/>

# INDICE

<b>Sommario</b> .....	<b>III</b>
<b>Abstract</b> .....	<b>VII</b>
<b>Indice delle Figure</b> .....	<b>XV</b>
<b>Indice delle Tabelle</b> .....	<b>XXIV</b>
<b>Nomenclatura</b> .....	<b>XXVIII</b>
<b>1 Modellazione CFD di reattori a letto fisso</b> .....	<b>1</b>
1.1 Reattori catalitici a letto fisso .....	1
1.2 Problematiche connesse alla modellazione CFD di reattori a letto fisso .....	3
1.2.1 Tecniche per modellare letti impaccati: stato dell'arte .....	5
1.3 Scopo del lavoro e risultati principali.....	10
Bibliografia.....	12
<b>2 Generazione della mesh</b> .....	<b>16</b>
2.1 Introduzione.....	16
2.2 Tipi di mesh.....	16
2.2.1 Mesh strutturate .....	18
2.2.2 Mesh non strutturate .....	23
2.2.3 Mesh ibride .....	25
2.3 Qualità della mesh .....	26
2.3.1 Ortogonalità della mesh .....	27
2.3.2 Mesh <i>skewness</i> .....	28
2.3.3 Mesh Aspect ratio .....	29
2.3.4 Smoothness .....	30
2.4 Presentazione dei generatori di mesh .....	30
2.4.1 SnappyMeshHex .....	31
2.4.2 Salome .....	32
2.5 Scelta del generatore di mesh.....	34
Bibliografia .....	<b>35</b>

---

<b>3</b>	<b>Semplice caso studio sfera/sfera.....</b>	<b>37</b>
3.1	Introduzione.....	37
3.2	Studio della reattività chimica.....	39
3.2.1	Generazione della geometria e della mesh.....	39
3.2.1.1	Metodo di modifica locale “ <i>bridge</i> ”.....	41
3.2.1.3	Metodo di modifica globale “ <i>gap</i> ”.....	45
3.2.1.4	Metodo di modifica globale “ <i>overlap</i> ”.....	47
3.2.2	Analisi della convergenza della mesh.....	51
3.2.3	Analisi dei risultati ottenuti.....	57
3.3	Studio dello scambio termico.....	69
3.3.1	Generazione della geometria e della mesh.....	69
3.3.2	Analisi della convergenza della mesh.....	76
3.3.3	Analisi dei risultati ottenuti.....	77
3.4	Conclusioni.....	83
	Bibliografia.....	85
<b>4</b>	<b>Studio dei fenomeni di scambio di materia e di calore per un sistema costituito da 4 sfere.....</b>	<b>87</b>
4.1	Introduzione.....	87
4.2	Studio della reattività chimica.....	89
4.2.1	Analisi della convergenza della mesh.....	92
4.2.2	Risultati e discussioni.....	95
4.3	Studio dello scambio termico.....	104
4.4	Conclusioni.....	112
	Bibliografia.....	115
<b>5</b>	<b>Confronto dei metodi di modifica locale su un caso sfera/sfera in presenza di una reazione chimica superficiale eterogenea esotermica.....</b>	<b>116</b>
5.1	Introduzione.....	116
5.2	Creazione del caso: dominio geometrico e generazione della mesh.....	118
5.3	Risultati e discussioni.....	120
5.4	Conclusioni.....	128
	Bibliografia.....	130

---

---

<b>6</b>	<b>Analisi sistematica di reattori impaccati di sfere .....</b>	<b>131</b>
6.1	Introduzione.....	131
6.2	Proprietà geometriche dei letti impaccati di sfere .....	132
6.2.1	Aspect Ratio.....	133
6.2.2	Grado di vuoto .....	134
6.2.3	Area catalitica per unità di volume .....	134
6.3	Approccio DEM-SALOME per la generazione di letti impaccati .....	136
6.3.1	Implementazione dei metodi di modifica dei punti di contatto in letti impaccati di sfere .....	142
6.3.2	Generazione della mesh .....	148
6.4	Valutazione delle proprietà geometriche dei letti impaccati .....	150
6.5	Simulazione di un caso studio .....	156
6.5.1	Simulazione <i>DEM</i> e generazione della mesh .....	156
6.5.2	Risultati della simulazione.....	159
6.6	Conclusioni.....	163
	Bibliografia.....	164
<b>7</b>	<b>Conclusioni.....</b>	<b>166</b>
	Bibliografia.....	169
	<b>APPENDICE - Script per le generazione delle mesh in SALOME .....</b>	<b>170</b>

# Indice delle Figure

Figura 1.1 Particolare di un reattore catalitico a letto fisso di sfere [2].....	1
Figura 1.2 Letto impaccato random di 200 sfere.....	4
Figura 1.3 Metodi di modifica: globali (a) e (b) e locali (c) e (d) dei punti di contatto sfera/sfera [25].....	6
Figura 2.1 Cella triangolare e cella a quadrilatero.....	15
Figura 2.2 Celle tridimensionali.....	16
Figura 2.3 Generazione di una griglia cartesiana sulla superficie di un semi cilindro [2].....	17
Figura 2.4 Esempio di mesh curvilinea ortogonale su un profilo alare [2].....	19
Figura 2.5 Mesh di tipo “ <i>body-fitted</i> ” attorno ad un semi cilindro [2].....	19
Fig. 2.6 Mapping process di un dominio fisico in un dominio computazionale: (a) dominio fisico in coordinate cartesiane; (b) dominio computazionale ricavato attraverso un “ <i>mapping process</i> ” del dominio geometrico [2].....	20
Fig. 2.7 Mesh non-ortogonale strutturata di un cilindro di un motore a 4 tempi [2].....	21
Figura 2.8 Esempio di block structured mesh per una sfera.....	22
Figura 2.9 Esempio di <i>mesh</i> non strutturata per una sfera.....	23
Figura 2.10 <i>Mesh</i> ibrida per un semplice caso studio sfera/sfera.....	25
Figura 2.11 Ortogonalità tra due celle adiacenti.....	26
Figura 2.12 Indice di <i>skewness</i> tra due celle adiacenti.....	27
Figura 2.14 Smoothness in una cella.....	29
Figura 2.15 Esempio di generazione della <i>mesh</i> con SnappyHexMesh su una porzione	

---

di sfera.....	30
Figura 3.1 Dimensioni del reattore e delle semisfere utilizzate.....	38
Figura 3.2 Particolare della mesh in una sezione trasversale del reattore.....	39
Figura 3.3 Analisi geometrica di un cilindro (in azzurro il <i>bridge</i> ) interposto tra due sfere.....	40
Figura 3.4 Sfere con diverse dimensioni del bridge: (a) <i>bridge 1</i> , (b) <i>bridge 2</i> , (c) <i>bridge 3</i> , (d) <i>bridge 4</i> .....	41
Figura 3.5 Creazione del piano tangente al punto di contatto tra le sfere.....	42
Figura 3.6 Creazione di un cap nella sfera attraverso l'operazione booleana "cut".....	43
Figura 3.7 Creazione di un <i>cap</i> in entrambe le sfere.....	43
Figura 3.8 Sfere con diverse dimensioni del <i>cap</i> :(a) <i>cap 1</i> , (b) <i>cap 2</i> , (c) <i>cap 3</i> , (d) <i>cap 4</i> .....	44
Figura 3.9 Sfere con diverse dimensioni del <i>gap</i> :(a) <i>gap 1</i> , (b) <i>gap 2</i> , (c) <i>gap 3</i> , (d) <i>gap 4</i> .....	45
Figura 3.10 Sfere con diverse sovrapposizioni:(a) <i>overlap 1</i> , (b) <i>overlap 2</i> , (c) <i>overlap 3</i> , (d) <i>overlap 4</i> .....	46
Figura 3.11 Schematizzazione della condizione a gradiente nullo per la temperatura, lungo la coordinata x di una generica superficie [10].....	49
Figura 3.12 Sezione longitudinale di tre mesh ibride a numero di celle via via maggiore.....	51
Figura 3.13 Particolare dell'ipotesi utilizzata in SALOME per generare una mesh tetraedrica con dimensione delle celle assegnata nel punto di contatto.....	52
Figura 3.14 Confronto tra area catalitica esatta e quella generata dalla mesh per il caso di riferimento.....	54
Figura 3.15 Sviluppo del campo di moto all'interno del reattore per il caso di	

---



---

riferimento.....	56
Figura 3.16 Frazione massiva di O <sub>2</sub> all'interno del reattore per il <i>bridge method</i> .....	56
Figura 3.17 Frazione massiva di O <sub>2</sub> all'interno del reattore per il <i>cap method</i> .....	57
Figura 3.18 Frazione massiva di O <sub>2</sub> all'interno del reattore per il <i>gap method</i> .....	57
Figura 3.19 Frazione massiva di O <sub>2</sub> all'interno del reattore per <i>l'overlap method</i> .....	57
Figura 3.20 Conversione di O <sub>2</sub> per (a) <i>bridge</i> , (b) <i>cap</i> per Da=0.015.....	58
Figura 3.21 Conversione di O <sub>2</sub> per (a) <i>gap</i> , (b) <i>overlap</i> per Da=0.015.....	58
Figura 3.22 Conversione di O <sub>2</sub> per (a) <i>bridge</i> , (b) <i>cap</i> per Da=1.5.....	59
Figura 3.23 Conversione di O <sub>2</sub> per (a) <i>gap</i> , (b) <i>overlap</i> per Da=1.5.....	59
Figura 3.24 Conversione di O <sub>2</sub> per (a) <i>bridge</i> , (b) <i>cap</i> per Da=150.....	59
Figura 3.25 Conversione di O <sub>2</sub> per (a) <i>gap</i> , (b) <i>overlap</i> per Da=150.....	60
Figura 3.26 Campo di moto attorno al punto di contatto per il <i>bridge method</i> .....	62
Figura 3.27 Campo di moto attorno al punto di contatto (a) <i>cap</i> , (b) <i>gap</i> .....	63
Figura 3.28 Campo di moto attorno al punto di contatto per <i>l'overlap method</i> .....	64
Figura 3.29 Particolare del campo di moto attorno al punto di contatto.....	65
Figura 3.30 Frazione massiva di O <sub>2</sub> vicino il punto di contatto per il <i>bridge method</i> .....	66
Figura 3.32 Frazione massiva di O <sub>2</sub> vicino il punto di contatto per il <i>cap method</i> .....	66
Figura 3.31 Frazione massiva di O <sub>2</sub> vicino il punto di contatto per il <i>gap method</i> .....	67
Figura 3.33 Frazione massiva di O <sub>2</sub> vicino il punto di contatto per <i>l'overlap method</i> .....	67

---

---

Figura 3.34 Schematizzazione del dominio geometrico e dei metodi implementati.....	68
Figura 3.35 Particolare del punto di contatto utilizzato come nodo della meshtetraedrica per il caso di riferimento.....	69
Figura 3.36 Scheda di lavoro in ambiente SALOME per creare un caso multiregione.....	70
Figura 3.37 Modellazione del caso studio.....	71
Figura 3.38 Modellazione numerica del <i>bridge</i> secondo il modello Dixon <i>et al.</i> [2].....	73
Figura 3.39 Mesh di sfere a numero di celle crescente.....	75
Figura 3.40 Valori del flusso termico rispetto al N° di celle per il caso di riferimento....	76
Figura 3.41 Distribuzione della temperatura lungo la coordinata x passante per i centri delle semisfere.....	76
Figura 3.42 Profilo di temperatura secondo il modello Dixon <i>et al.</i> [4].....	77
Figura 3.43 Profilo di temperatura secondo il modello <i>bridge</i> semplificato.....	77
Figura 3.44 Profilo di temperatura secondo il modello <i>cap</i> .....	78
Figura 3.45 Profilo di temperatura secondo il modello <i>gap</i> .....	78
Figura 3.46 Profilo di temperatura secondo il modello <i>overlap</i> .....	79
Figura 3.47 Confronto dei flussi termici tra il riferimento e i modelli dei <i>bridges</i> .....	80
Figura 3.48 Confronto dei flussi termici tra il riferimento e il modello <i>cap</i> .....	80
Figura 3.49 Confronto dei flussi termici tra il riferimento e il modello <i>gap</i> .....	81
Figura 3.50 Confronto dei flussi termici tra il riferimento e il modello <i>overlap</i> .....	81
Figura 4.1 Modifiche locali dei punti di contatto sfera/sfera e sfera/parete. Implementazione del <i>bridge method</i> per i casi (a) e (c), <i>cap method</i> per i casi (b) e (d).....	89
Figura 4.2 Valori delle conversione di O <sub>2</sub> rispetto al numero di celle per i casi <i>bridge</i> 1 e	

---

---

<i>bridge 2</i> .....	92
Figura 4.3 Valori delle conversione di O <sub>2</sub> rispetto al numero di celle per i casi <i>bridge 3</i> e <i>bridge 4</i> .....	92
Figura 4.4.Valori delle conversione di O <sub>2</sub> rispetto al numero di celle per i casi <i>cap 1</i> e <i>cap 2</i> .....	93
Figura 4.5.Valori delle conversione di O <sub>2</sub> rispetto al numero di celle per i casi <i>cap 3</i> e <i>cap 4</i> .....	93
Figura 4.6 Valori delle conversione di O <sub>2</sub> rispetto al numero di celle per il caso di riferimento.....	93
Figura 4.7 Valori di conversione di O <sub>2</sub> in funzione del numero di Reynolds per i 4 casi con <i>bridges</i> confrontati con il caso di riferimento.....	94
Figura 4.8 Valori di conversione di O <sub>2</sub> in funzione del numero di Reynolds per i 4 casi con <i>caps</i> confrontati con il caso di riferimento.....	96
Figura 4.9 Campo di moto e frazioni massive di idrogeno, ossigeno e acqua a Re=142.24 lungo la sezione longitudinale del reattore.....	97
Figura 4.10 Frazione massiva di ossigeno per i casi con <i>bridge 4</i> e <i>cap 4</i> , a Re=142.24 lungo una sezione trasversale al reattore.....	98
Figura 4.11 Profili di velocità lungo la linea <i>y</i> in corrispondenza della modifica implementata per i casi con <i>caps</i> e <i>bridges</i> a numero di Reynolds pari a 142.24.....	98
Figura 4.12 Frazione massiva di ossigeno per <i>bridge 4</i> e <i>cap 4</i> , a Re=142.24 lungo una sezione trasversale al reattore.....	99
Figura 4.13 Confronto dei profili delle frazioni massive di ossigeno per <i>caps</i> e <i>bridges 4</i> in corrispondenza del punto di contatto, a Re=142.24.....	99
Figura 4.14 Frazioni massive di ossigeno per <i>cap 4</i> e <i>bridge 4</i> , a Re=142.24.....	99

---

Figura 4.15 Confronto dei profili delle frazioni massive di acqua per <i>caps</i> e <i>bridges</i> 4 in corrispondenza del punto di contatto, a $Re=142.24$ .....	100
Figura 4.16 Frazioni massive di acqua per i casi con <i>bridge</i> 4 e <i>cap</i> 4, a $Re=1422.4$ .....	100
Figura 4.17 Profili di velocità lungo la linea $y$ in corrispondenza della modifica implementata per i casi con <i>bridges</i> e <i>caps</i> a numero di Reynolds pari a 1422.4.....	101
Figura 4.18 Frazione massiva di ossigeno per <i>bridge</i> 4 e <i>cap</i> 4 a $Re=1422.4$ lungo una sezione trasversale al reattore.....	101
Figura 4.19 Frazioni massive di acqua per <i>bridge</i> 4 e <i>cap</i> 4 a $Re=1422.4$ lungo una sezione trasversale al reattore.....	102
Figura 4.20 Confronto dei profili delle frazioni massive di ossigeno per <i>caps</i> e <i>bridges</i> in corrispondenza del punto di contatto a $Re=1422.4$ .....	102
Figura 4.21 Confronto dei profili delle frazioni massive di acqua per <i>caps</i> e <i>bridges</i> in corrispondenza del punto di contatto a $Re=1422.4$ .....	102
Figura 4.22 Schematizzazione della geometria considerata per lo studio dello scambio termico.....	103
Figura 4.23 Distribuzione della temperatura all'interno delle sfere per il caso <i>bridge</i> ....	103
Figura 4.24 Modello per descrivere il meccanismo di scambio termico per un sistema costituito da 4 sfere in serie applicando le 4 modifiche dei punti di contatto sfera/sfera e sfera/parete.....	104
Figura 4.25 Modelli utilizzati per modellare i punti di contatto sfera/sfera e sfera/parete per lo studio dello scambio termico.....	106
Figura 4.26 Discretizzazione del fenomeno di scambio termico lungo la coordinata $z$ del tubo.....	107
Figura 4.27 Profili delle temperature lungo l'asse $z$ del tubo passante per i centri delle sfere.....	110

---

Figura 4.28. Particolare dei salti termici a distanza pari a $z=25.4$ mm dalla parete mantenuta a $T_1$ .....	111
Figura 4.29 Valori di temperatura raggiunti per i 4 metodi sulla superficie della sfera 1 che si trova vicino la parete mantenuta a $T_1$ .....	111
Figura 4.30 Valori di temperatura raggiunti per i 4 metodi sulla superficie della sfera 4 che si trova vicino la parete mantenuta a $T_2$ .....	112
Figura 5.1 Sezione del letto impaccato di sfere lungo la lunghezza del reattore [2].....	117
Figura 5.2. Mesh ibrida 3D.....	118
Figura 5.3. Modifica dei punti di contatto sfera/sfera e sfera/parete con <i>bridges</i> e <i>caps</i> rispettivamente.....	119
Figura 5.4 Frazioni massive delle specie chimiche lungo il reattore per il <i>bridge method</i> .....	121
Figura 5.5 Frazioni massive delle specie chimiche lungo il reattore per il <i>cap method</i> .....	122
Figura 5.6 Campo di moto del fluido lungo il piano yz passante per il centro del <i>bridge</i> sfera/sfera.....	123
Figura 5.7 Campo di moto del fluido lungo il piano xz passante per il centro del <i>bridge</i> sfera/sfera.....	123
Figura 5.8 Distribuzione della temperatura all'interno delle regioni solide per il <i>cap</i> ....	124
Figura 5.9 Distribuzione della temperatura all'interno delle regioni solide per il <i>bridge</i> .....	124
Figura 5.10 Consumo di ossigeno in prossimità dei <i>bridges</i> .....	125
Figura 5.11 Campo di moto in prossimità della modifica effettuata tra le sfere per il <i>cap</i> (a) ed il <i>bridge</i> (b).....	126
Figura 5.12 Profilo di temperatura all'interno delle sfere per il <i>cap method</i> .....	127

---

---

Figura 5.13 Profilo di temperatura all'interno delle sfere per il <i>bridge method</i> .....	127
Figura 6.1 reattore con $N=2.2$ (a); reattore con $N=7.87$ (b).....	133
Figura 6.2 Diagramma di lavoro utilizzato per generare la <i>mesh</i> ibrida di letti impaccati.....	136
Figura 6.3 Generazione di un letto impaccato attraverso una simulazione <i>DEM</i> .....	139
Figura 6.4 Vista dall'alto di un letto impaccati dopo aver applicato il <i>gap method</i> .....	142
Figura 6.5 Vista dall'alto di un letto impaccati dopo aver applicato l' <i>overlapmethod</i> ....	143
Figura 6.6 Vista dall'alto di letti impaccati modificati. <i>Bridge method (a)</i> ; <i>cap method (b)</i> .....	146
Figura 6.7 Vista laterale di letti impaccati di 200 sfere. <i>bridge method (a)</i> ; a <i>cap method (b)</i> .....	147
Figura 6.8 <i>Mesh</i> ibrida 3D ottenuta per un letto impaccato di 20 sfere.....	148
Figura 6.9 Modifica dei punti di contatto attraverso il <i>bridge method</i> per un letto impaccato di 20 sfere.....	149
Figura 6.10 Modifica dei punti di contatto attraverso il <i>cap method</i> per un letto impaccato di 20 sfere.....	149
Figura 6.11 Impaccamenti utilizzati a diversi valori di $N$ . $N=2.2$ ( a); $N=3.4$ (b).....	150
Figura 6.12 Impaccamenti utilizzati a diversi valori di $N$ . $N=5.96$ (a); $N=7.99$ (b).....	151
Figura 6.13 Distanza $h$ tra le sfere ottenuta attraverso i metodi di modifica locale.....	152
Figura 6.14 Confronto dell'errore % di area catalitica per i metodi di modifica locale.....	152
Figura 6.15 Variazione % di $a_v$ rispetto al riferimento per i letti impaccati a diversi	

---

---

valori di $N$ , implementando il <i>gap</i> e l' <i>overlap method</i> .....	154
Figura 6.16 Variazione % dell'area catalitica rispetto al riferimento per i letti impaccati a diversi valori di $N$ , implementando il <i>gap</i> e l' <i>overlap method</i> .....	154
Figura 6.17 Variazione % dell'area catalitica rispetto al riferimento per i letti impaccati a diversi valori di $N$ , implementando il <i>gap</i> e l' <i>overlap method</i> .....	155
Figura 6.18 Impaccamento attraverso una simulazione <i>DEM</i> di un letto impaccato di 20 sfere e successiva modifica dei punti di contatto.....	157
Figura 6.19 <i>Mesh</i> ibrida 3D di un letto impaccato di 20 sfere con le dimensioni sopra riportate.....	158
Figura 6.20 Campo di moto che si sviluppa all'interno del letto impaccato.....	159
Figura 6.21 Frazione massiva di $H_2O$ che si sviluppa sulla superficie delle sfere e nei <i>bridges</i> .....	160
Figura 6.22 Frazione massiva di $O_2$ che si sviluppa nei <i>bridges</i> (a); campo di moto locale attorno ai <i>bridges</i> (b).....	161
Figura 6.23 Distribuzione della temperatura all'interno le sfere.....	162

# Indice delle Tabelle

Tabella 1.1 Processi catalitici a letto fisso [4].....	2
Tabella 2.1 Qualità delle celle a diversi valori di <i>skewness</i> [17].....	28
Tabella 3.1 Valori del raggio e della lunghezza del bridge per diversi rapporti $r_b/d_s$ .....	41
Tabella 3.2 Valori h del cap utilizzati pari alla metà della lunghezza dei <i>bridges</i> $l_b$ .....	44
Tabella 3.3 Diversi valori dei diametri per il metodo globale “ <i>gap method</i> ”.....	45
Tabella 3.4 Valori dei diametri per i 4 casi <i>dell’overlap</i> .....	46
Tabella 3.5 Valori delle costanti cinetiche a tre diversi numeri di Damköhler.....	47
Tabella 3.6 Frazioni massive delle specie chimiche in ingresso e condizioni operative utilizzate.....	47
Tabella 3.7 Condizioni al contorno assegnate a ciascuna patch del reattore.....	48
Tabella 3.8 Differenti schemi numerici utilizzati per il caso di riferimento.....	49
Tabella 3.9 Confronto della mesh tra il riferimento ed i 4 casi del <i>bridge method</i> .....	54
Tabella 3.10 Confronto della mesh tra il riferimento ed i 4 casi del <i>cap method</i> .....	55
Tabella 3.11 Confronto della mesh tra il riferimento ed i 4 casi del <i>gap method</i> .....	55
Tabella 3.12 Confronto della mesh tra il riferimento ed i 4 casi dell’ <i>overlap method</i> ...	55
Tabella 3.13 Confronto degli errori percentuali tra i metodi di modifica locale.....	60
Tabella 3.14 Variazione percentuale di area catalitica per il <i>bridge method</i> .....	60
Tabella 3.15. Variazione percentuale di area catalitica per il <i>cap method</i> .....	61
Tabella 3.16 Variazione percentuale di area catalitica per il <i>gap method</i> .....	61



---

Tabella 3.17 Variazione percentuale di area catalitica per <i>l'overlap method</i> .....	61
Tabella 3.18 Dimensioni dei <i>bridges</i> utilizzati.....	70
Tabella 3.19 Dimensioni dei <i>caps</i> utilizzati.....	70
Tabella 3.20 Dimensioni delle sfere per creare i 4 casi del <i>gap</i> .....	70
Tabella 3.21 Dimensioni delle sfere per creare i 4 casi di <i>overlap</i> .....	70
Tabella 3.22 Condizioni al contorno assegnate a ciascuna patch nel box e nelle diverse regioni.....	72
Tabella 3.23 Proprietà fisiche del fluido.....	74
Tabella 3.24 Proprietà fisiche delle sfere.....	74
Tabella 3.25 Proprietà fisiche dei <i>bridges</i> utilizzati.....	74
Tabella 3.26 Valori delle conducibilità termiche effettive dei <i>bridges</i> .....	75
Tabella 4.1 Valori delle dimensioni del <i>bridge</i> a diversi rapporti dimensione <i>bridge</i> /dimensione sfera per le due modifiche sfera/sfera e sfera/parete.....	89
Tabella 4.2 Valori delle dimensioni del <i>cap</i> a diversi rapporti dimensione del <i>cap</i> /dimensione sfera per le due modifiche sfera/sfera e sfera/parete.....	90
Tabella 4.3 Condizioni operative utilizzate per lo studio della reattività chimica.....	90
Tabella 4.4 Valori del numero di Reynolds a diverse velocità del fluido in ingresso al reattore.....	90
Tabella 4.5 Confronto tra area geometrica esatta e superficie catalitica discretizzata a diverso numero di celle.....	91
Tabella 4.6 Variazione percentuale di conversione di O <sub>2</sub> per i casi con <i>bridges</i> rispetto al caso di riferimento.....	95
Tabella 4.7 Valori di variazione percentuale della conversione di O <sub>2</sub> per i casi con il <i>cap</i> rispetto al caso di riferimento.....	96

---

Tabella 4.8 Dimensioni delle modifiche locali e globali utilizzate.....	103
Tabella 4.9 Proprietà termiche e termodinamiche utilizzate per il fluido.....	104
Tabella 4.10 Proprietà termiche e termodinamiche utilizzate per le regioni solide sfere e <i>bridges</i> .....	104
Tabella 4.11 Valori delle distanze sfera/sfera e sfera/parete utilizzate dopo la modifica dei punti di contatto.....	108
Tabella 4.12 Valori delle resistenze totali, resistenza sfera/parete e sfera/sfera calcolate per il sistema studiato. ....	108
Tabella 4.12 Valori delle resistenze totali, resistenza sfera/parete e sfera/sfera calcolate per il sistema studiato.....	109
Tabella 5.1 Valori utilizzati per modificare i punti di contatto sfera/sfera e sfera/parete con imetodi <i>bridge</i> e <i>cap</i> .....	119
Tabella 5.2 Valori caratteristici delle mesh ottenute per il <i>cap</i> ed il <i>bridge</i> .....	119
Tabella 5.3 Condizioni operative utilizzate per il caso studio.....	120
Tabella 5.4 Proprietà fisiche del fluido utilizzate.....	120
Tabella 5.5 Proprietà fisiche delle regioni solide utilizzate.....	120
Tabella 5.6 Valori di conversione percentuale di O <sub>2</sub> in funzione dell'area catalitica offerta dai due metodi di modifica locale.....	120
Tabella 6.1 Valutazione delle proprietà geometriche di un letto impaccato random di 200 sfere .....	140
Tabella 6.2 Valutazione delle proprietà geometriche del letto impaccato dopo aver applicato il <i>gap method</i> .....	143
Tabella 6.3 Valutazione delle proprietà geometriche del letto impaccato dopo aver applicato <i>l'overlap method</i> .....	144
Tabella 6.4 Valutazione delle proprietà geometriche del letto impaccato dopo aver	

---

applicato il <i>bridge method</i> .....	145
Tabella 6.5 Valutazione delle proprietà geometriche del letto impaccato dopo aver applicato il <i>cap method</i> .....	146
Tabella 6.6 Valori delle dimensioni dei letti impaccati e numero di sfere utilizzate per ottenere impaccamenti a 4 diversi valori di $N$ .....	150
Tabella 6.7 Proprietà ottenute per i 4 letti impaccati a diversi valori di $N$ .....	151
Tabella 6.8 Valori di Errore % di area catalitica per i due metodi di modifica locale a due valori distinti di $N$ .....	153
Tabella 6.9 Errori di $a_v$ e del grado di vuoto ottenuti per i metodi di modifica locale....	155
Tabella 6.10 Caratteristiche geometriche del letto impaccato random di 20 sfere.....	156
Tabella 6.11 Caratteristiche della <i>mesh</i> ibrida utilizzata.....	158
Tabella 6.12 Caratteristiche geometriche del letto impaccato e risultati della simulazione.....	159

# Nomenclatura

$N$	=	aspect ratio di letti impaccati [-]
$k_f$	=	conducibilità termica del fluido [W/(m·K)]
$k_s$	=	conducibilità termica delle sfere [W/(m·K)]
$k_{eff}$	=	conducibilità termica effettiva dei bridges [W/(m·K)]
$k_{fr}$	=	conducibilità termica ridotta nel gap di fluido tra le sfere [W/(m·K)]
$\mathbf{t}_{ij}$	=	vettore coppia agente sulla sfera $i$ per effetto della sfera $j$ [N·m]
$k_{cin}$	=	costante cinetica [kmol, m, s]
$\mathbf{g}$	=	vettore accelerazione di gravità [m/s <sup>2</sup> ]
$d_s$	=	diametro della sfera [m]
$d_{attuale}$	=	diametro della sfera dopo la modifica globale [m]
$d_{nominale}$	=	diametro nominale della sfera [m]
$Q_w$	=	calore trasferito [W]
$q$	=	flusso termico specifico [W/m <sup>2</sup> ]
$\mathbf{f}_{ij}$	=	vettore forza di interazione tra la sfera $i$ e la sfera $j$ [N]
$h_b$	=	lunghezza del bridge [m]
$L_{tubo}$	=	lunghezza del reattore [m]
$m_i$	=	massa della sfera $i$ [kg]
$I_i$	=	momento di inerzia della sfera $i$ [kg/m <sup>2</sup> ]
$Da_i$	=	numero di Damköhler della specie $i$ [-]
Re	=	numero di Reynolds [-]
$PM_{mix}$	=	peso molecolare medio della miscela [kg/kmol]
$P$	=	pressione [Pa]
$r_b$	=	raggio del bridge [m]
R	=	raggio della sfera [m]
$a_v$	=	rapporto superficie catalitica sfere sul volume del reattore [m <sup>2</sup> /m <sup>3</sup> ]

$\hat{R}_{tot}$	=	resistenza termica totale [K·m <sup>2</sup> /W]
$T$	=	temperatura [K]
$v$	=	modulo della velocità convettiva del fluido [m/s]
$\mathbf{v}_i$	=	vettore velocità della sfera $i$ [m/s]

*Lettere greche*

$\gamma$	=	accomodation coefficient [-]
$\rho_f$	=	densità del fluido [kg/m <sup>3</sup> ]
$\rho_s$	=	densità del solido [kg/m <sup>3</sup> ]
$\rho_{mix}$	=	densità media della miscela [kg/m <sup>3</sup> ]
$\beta$	=	fattore di forma [-]
$\varepsilon$	=	grado di vuoto del letto impaccato [-]
$\Lambda$	=	libero cammino medio delle molecole del gas [m]
$\boldsymbol{\omega}_i$	=	vettore momento angolare della sfera $i$ [kg/(m <sup>2</sup> ·s)]

*Acronimi*

AR	=	aspect ratio mesh
CAD	=	computer aided design
CAE	=	computer aided engineering
CFD	=	computational fluid dynamics
DEM	=	discrete element method
PDE	=	partial differential equation
STL	=	standard triangle language

# 1 Modellazione CFD di reattori a letto fisso

## 1.1 Reattori catalitici a letto fisso

Un reattore catalitico a letto fisso è un insieme di particelle catalitiche con dimensioni solitamente uniformi (sfere, cilindri, anelli, ecc.), disposte in modo casuale e tenute saldamente le une con le altre all'interno di un vessel (o tubo). Il reattore così descritto risulta essere un impaccamento di particelle catalitiche. I reagenti sono forniti al reattore attraverso il fluido che attraversa il letto impaccato. Appena i reagenti entrano in contatto con le particelle cataliticamente attive subiscono trasformazioni chimiche, che sono di solito accompagnate da rilascio o consumo di calore [2]. Se necessario, il calore viene rimosso o fornito attraverso la parete del tubo. Un parametro importante di un reattore a letto fisso è  $N$ , definito come il rapporto tra il diametro del tubo ed il diametro delle particelle. In particolare per reazioni fortemente esotermiche (o fortemente endotermiche) si lavora con valori compresi tra  $2 < N < 8$ .

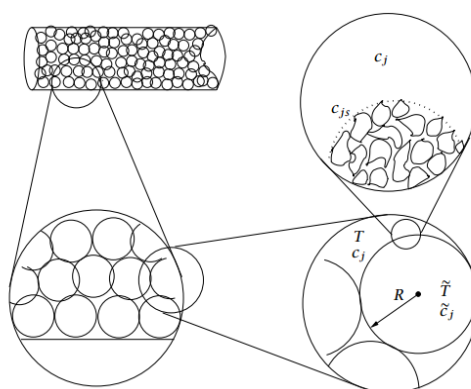


Figura 1.1 Particolare di un reattore catalitico a letto fisso di sfere [2].

La prima applicazione commerciale di un reattore catalitico a letto fisso risale al 1831 quando Peregrine Philips, un venditore di aceto, ha brevettato un processo per

sintetizzare l'anidride solforica facendo passare aria e anidride solforosa su un letto caldo con una spugna di platino come materiale catalitico [3]. Visto che il catalizzatore non era stato consumato nella reazione, ha portato il vantaggio di poter essere riutilizzato ripetutamente senza la necessità di separare e rigenerare il catalizzatore. Da quel momento in poi i reattori catalitici a letto fisso sono diventati quelli più utilizzati per reazioni gas-solido e liquido-solido.

Nonostante l'esistenza dei reattori a letto fluidizzato, i reattori a letto fisso sono ampiamente utilizzati per processi di larga scala industriale, come nell'industria petrolifera (es. reforming catalitico e hydro-treating) e nella chimica di base (es. sintesi ammoniacca e acido solforico).

Nella Tabella 1.1 sono riportati i più importanti processi catalitici a letto fisso.

<u>Industria chimica di base</u>	<u>Industria petrolchimica</u>	<u>Raffinazione del petrolio</u>
Steam reforming primario	Ossido di etilene	Reforming catalitico
Steam reforming secondario	Dicloruro di etilene	Isomerizzazione
Conversione di monossido di carbonio	Vinilacetato	Polimerizzazione
Metanazione di monossido di carbonio	Butadiene	Idrodesolforazione
Sintesi di ammoniacca	Anidride maleica	Idrocracking
Sintesi di acido solforico	Anidride ftalica	
Sintesi di metanolo	Cicloesano	
Oxosintesi	Stirene	
	Idrodealchilazione	

**Tabella 1.1 Processi catalitici a letto fisso [4].**

Una delle caratteristiche principali del reattore a letto fisso è la sua semplicità con le conseguenze che ne derivano di bassi costi per la costruzione, il funzionamento e la manutenzione. Un altro notevole vantaggio è la facilità nella separazione del catalizzatore dalla corrente dei reagenti e prodotti effluente dal reattore.

## 1.2 Problematiche connesse alla modellazione CFD di reattori a letto fisso

L'uso della fluidodinamica computazionale (o anche CFD) sta assumendo grande importanza, come strumento per ottenere informazioni dettagliate sui fenomeni di trasporto che avvengono nei letti impaccati [5]. Attraverso l'approccio CFD le equazioni di Navier Stokes, le equazioni di bilancio di materia e di energia vengono risolte numericamente in un dominio geometrico discretizzato attraverso una griglia di calcolo (o *mesh*).

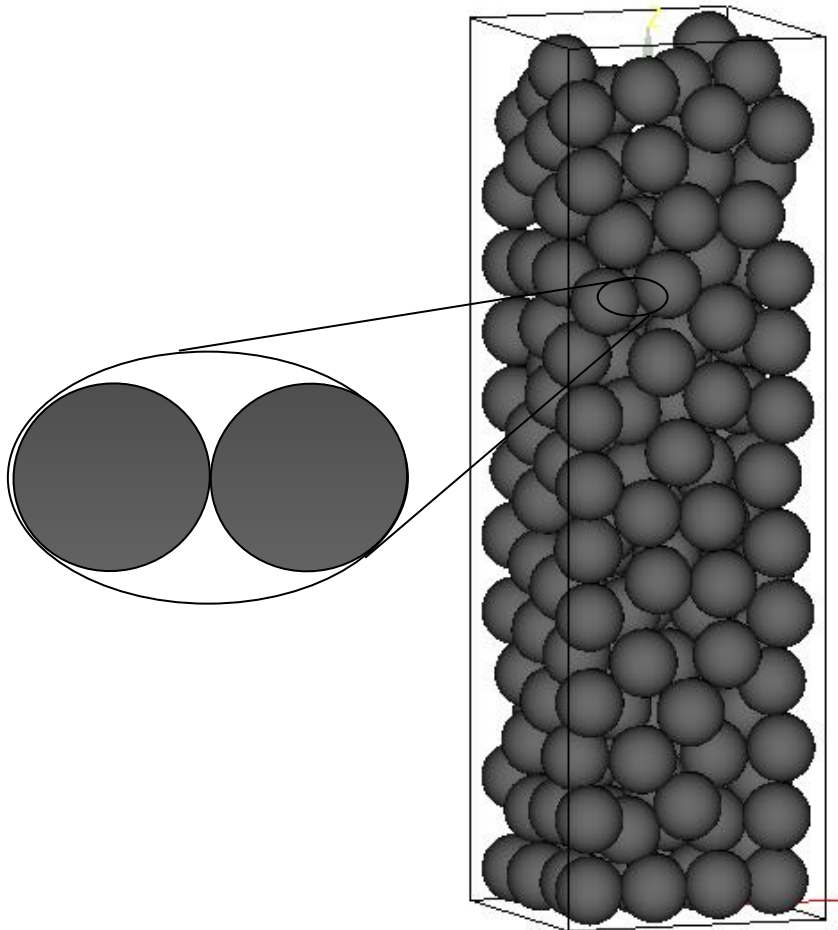
Il primo studio CFD per indagare il campo di moto in reattori a letto fisso è stato fatto da Dalman e Merkin [6]. Lo studio è stato eseguito su due sfere evidenziando come si sviluppa il campo di moto attorno a due sfere in un reattore tubolare. Lo studio ha dimostrato come all'aumentare del numero di Reynolds lo scambio termico migliora a causa dei vortici che si creano tra le sfere. Il modello che è stato implementato era un modello bidimensionale. Lloyd e Boehm [7] hanno studiato lo stesso problema bidimensionale, ma con 8 sfere anziché 2.

In questo lavoro è stato dimostrato come l'effetto della distanza tra le sfere si riflette sui valori dei "drag coefficients". In seguito, sono stati sviluppati modelli tridimensionali, il primo da Dixon *et al.* [8] nei quali lo studio è stato focalizzato sulla determinazione dei numeri di Nusselt in un letto con 3 sfere. Studi successivi sono stati svolti da Logtenberg *et al.* [9] su letto con 8 sfere. L'impaccamento era formato da due strati, ognuno dei quali contenente 4 sfere. Le sfere sono state poste a sufficiente distanza tra loro, al fine di garantire l'inesistenza dei punti di contatto sfera/sfera e sfera/parete. Lo scopo del lavoro è stato quindi quello di confrontare i risultati ricavati dalla simulazione CFD, con un modello teorico sullo scambio termico formulato da Dixon e Cresswell [10]. Nello studio condotto dagli autori Logtenberg *et al.* [11] su un letto impaccato con 10 sfere è stato mostrato lo sviluppo del campo di moto e dei fenomeni di scambio termico. La generazione della *mesh* è stata fatta includendo i punti di contatto sfera/sfera e sfera/parete e sono state riscontrate delle problematiche durante la generazione della *mesh* vicino i punti di contatto. Per ovviare a questo problema sono state create delle linee circolari in



prossimità dei punti di contatto, operazione fatta manualmente per rendere la *mesh* più fitta nelle zone dove è difficile generarla. Recentemente sono stati svolti studi da Calis e Nijenhuis [12] e Nijemeisland e Dixon [13] su letti impaccati con 40-50 sfere con bassi valori di  $N$ . Sono presenti, inoltre, altri studi su letti impaccati con un numero di sfere tra 100 e 1000 condotti in condizioni laminari, con la geometria modificata, in alcune zone del letto impaccato, per poter elaborare una *mesh* di buona qualità [14-23].

Per simulazioni in regime turbolento spesso sono stati riscontrati problemi di natura numerica come il non raggiungimento della convergenza.



**Figura 1.2** Letto impaccato random di 200 sfere.

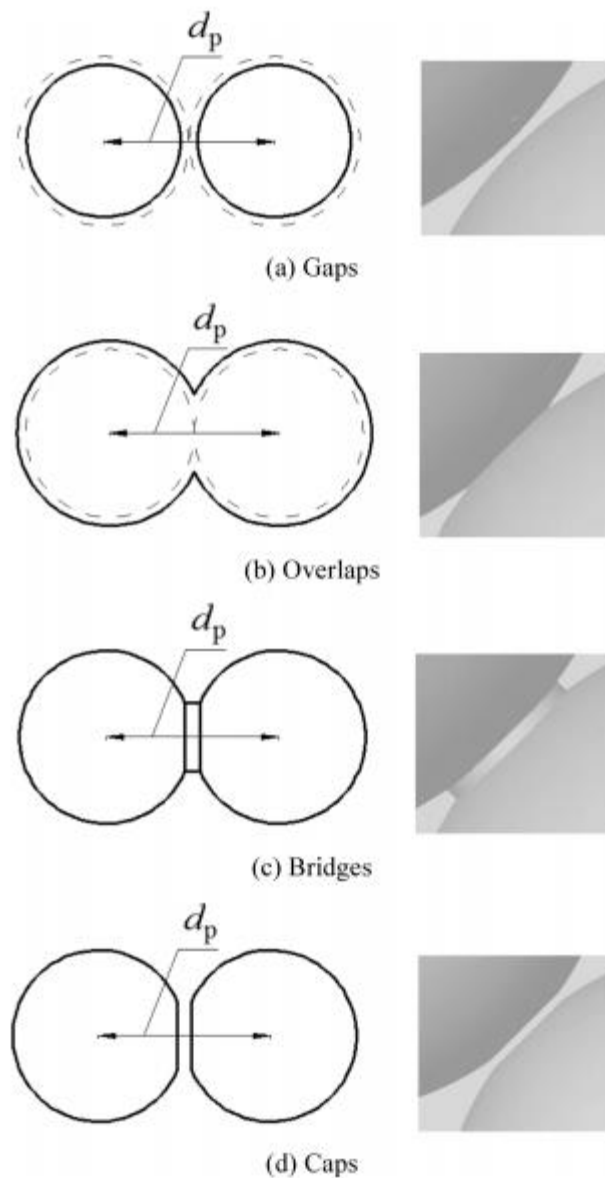
Gli autori sostengono che i punti di contatto nei letti impaccati presentino dei problemi nella CFD. La causa di tali problemi deriva dalla geometria intrinseca di un

letto impaccato, come la presenza di punti di contatto sfera/sfera e sfera/parete (Figura 1.2). Infatti dal punto di vista computazionale avere un punto di contatto vuol dire area nulla. Questo dà come risultato problemi nella generazione della *mesh* nelle zone vicino il punto di contatto, con la conseguente presenza di valori molto alti della *skewness* e dell'aspect ratio in quelle zone, in particolare alcune delle facce della stessa cella risultano molto più grandi rispetto alle altre. Inoltre, risulta poco fattibile modificare manualmente la *mesh* in prossimità dei punti di contatto, considerato l'elevato numero di sfere in un letto impaccato. Il problema potrebbe essere risolto imponendo una dimensione lineare delle celle talmente piccola da garantire bassi valori della *skewness*, però questo comporterebbe *mesh* con elevato numero di celle e quindi elevati tempi di calcolo. Alcuni autori hanno proposto diverse soluzioni a questo problema attraverso l'implementazione di metodi per la modifica dei punti di contatto. Altri, invece, in particolari casi sono riusciti a generare una *mesh* incorporando i punti di contatto [21]. Questi aspetti verranno discussi in dettaglio nel prossimo paragrafo attraverso uno studio approfondito dello stato dell'arte.

### **1.2.1 Tecniche per modellare letti impaccati: stato dell'arte**

Partendo dal lavoro di Magnico [21] è stato evidenziato come i punti di contatto, siano stati inclusi, utilizzando una *mesh* di tipo strutturata molto definita vicino ai punti di contatto. Con tale metodica sono stati ottenuti buoni risultati per simulazioni DNS a bassi numeri di  $Re$ . Infatti, è stata utilizzata una *mesh* molto fine con lati della cella pari a  $d_s/40$  e un numero di celle di circa 9.6-17.7 milioni per letti impaccati rispettivamente con 326 e 620 sfere. Nel 1999 Lund e Nguyen [24] hanno sviluppato una *mesh* triangolare non strutturata senza punti di contatto in un modello di scambio termico 2D senza flusso convettivo. Per evitare basse qualità della griglia di calcolo nel lavoro svolto da Baker e Tabor [17] è stata applicata una correzione dell'indice di *skewness*. Le simulazioni sono state condotte con  $700 < Re < 5000$  utilizzando una *mesh* non strutturata con dimensione delle celle di circa  $d_s/12$ , originando una *mesh* di 1.0168 milioni di celle per un letto impaccato con 160 sfere. Facendo un'attenta analisi in ordine cronologico possono essere identificati 4 diversi approcci di modifica del punto di contatto.

Tali metodologie sono riportate nella Figura 1.3.



**Figura 1.3 Metodi di modifica: globali (a) e (b) e locali (c) e (d) dei punti di contatto sfera/sfera [25].**

La Figura 1.3 rappresenta i metodi implementati per la modifica dei punti di contatto. I primi due sono dei metodi di modifica globale (l'intera superficie sferica è interessata alla modifica), gli altri sono dei metodi di modifica locale (interessano solo la zona del punto di contatto). Ciascun metodo è stato sviluppato in maniera indipendente da ciascun autore.

La più semplice modifica effettuata per migliorare le prestazioni sopra citate è quella di diminuire il diametro delle sfere (*gap method*) eliminando così il contatto tra le sfere. Negli studi svolti da Calis e Nijenhuis [12], Romkes e Dautzenberg [26] e Dixon *et al.* [27] sono state utilizzate delle sfere con diametro pari al 99% di quello. Anche nel lavoro di Bai *et al.* [16] è stata applicata una riduzione del 99.5 % dei diametri delle sfere per un letto impaccato con sfere da 10 mm di diametro con valori del numero di Reynolds compresi tra 2000 e 20000 e valori di  $N$  tra 2.14 e 3.23. Soleymani e Turunen [23] invece, hanno implementato il *gap* tra le sfere con una *mesh* con basso numero di celle per bassi valori di  $Re$ , riportando alti valori del grado di vuoto nel range 0.54-0.67. Gunjal e Ranade [28] hanno impostato un *gap* di 1 mm tra le sfere con una riduzione del 98% dei diametri delle sfere rispetto al diametro nominale, a bassi valori del numero di Reynolds ( $Re < 100$ ), al fine di poter creare 5 strati di celle prismatiche attorno le superfici delle sfere. Lopes e Quinta-Ferreira [20] nelle loro simulazioni hanno ridotto il diametro del 97%, conducendo la simulazione per un letto tickle bed.

Nel 2007 Lee *et al.* [29] hanno scelto di utilizzare un *gap* di 1 mm tra le sfere con diametro nominale di 57 mm, mentre Reddy e Joshi [22] hanno ridotto le sfere del 99%, per ottenere piccoli valori della *skewness* vicino i punti di contatto, implementando una *mesh* non strutturata con piccoli angoli di triangolazioni secondo il metodo Delaunay.

Un diverso approccio è stato seguito da Guardo e Coussirat [30] dove le sfere sono state espanse dell'1% rispetto al diametro nominale. Questo è l'unico lavoro nel quale è stata implementata tale modifica integrale per l'intero letto impaccato. Di contro, Laguerre e Ben Amara [31] hanno implementato l'*overlap* di 1 mm per contatto sfera/sfera, mentre per il contatto sfera/parete è stato utilizzato il *gap* di 1 mm. Tale caso era relativo ad una convezione naturale per un impaccamento di sfere con diametro di 75 mm. Il metodo appena descritto consente di evitare il punto di contatto tra le sfere ottenendo così una superficie condivisa sfera/sfera.

Un metodo simile è stato utilizzato da Xu e Jiang [32] dove è stata implementata una piccolissima sovrapposizione delle sfere (*overlap method*), ricavata dalla porosità desiderata del letto impaccato, per eliminare il problema della generazione della

---

mesh per i punti di contatto ed inoltre, per non avere inoltre grosse variazioni del grado di vuoto del letto. I due approcci di *gap* e *overlap* citati, comportano una variazione delle dimensioni delle intere sfere in un letto impaccato. Nel lavoro di Baiet *al.* [16] è stato applicato il *gap* e l'*overlap* alle sfere di un letto impaccato con grosse variazioni del grado di vuoto e quindi anche delle perdite di carico. Una regola abbastanza generale sostiene che un incremento dell'errore dell'1% sul grado di vuoto comporti un errore del 3% circa sulle perdite di carico Dixon *et al.* [1]. I due metodi globali *gap* e *overlap* danno errori sul grado di vuoto di circa il 4% con un *gap* del 99% e un *overlap* del 101%. Questi errori si riflettono sulle perdite di carico con errori del 12-15% in accordo con lo studio di Bai *et al.* [16].

Per evitare queste modifiche drastiche sul grado di vuoto, alcuni ricercatori hanno considerato modifiche locali della sfera vicino al punto di contatto, dove la modifica questa volta interessa solo una piccola parte della superficie sferica. Nel 1999 con il lavoro di Logtenberg *et al.* [11] sono state create delle piccole sfere con centro in corrispondenza del punto di contatto, ma si è evidenziato il problema della localizzazione dei punti di contatto e dell'opportuna valutazione della conducibilità termica delle piccole sfere. Un metodo simile è stato utilizzato nel lavoro di Tobis [33] dove si è simulato un flusso attorno a sfere di 38 mm di diametro connesse tra loro tramite un *neck* di raggio pari a 4 mm per riprodurre i risultati sperimentali ottenuti nei confronti delle perdite di carico.

Uno studio molto più mirato è stato sviluppato da Kuroki *et al.* [34] per simulazioni su scambio termico e campi di moto. Il loro metodo si è fondato sulla sostituzione di un *bridge* (*bridge method*) con il punto di contatto, eliminando così il problema della generazione della *mesh*. Con tale metodo non sono state riscontrate grosse variazioni sul grado di vuoto e quindi sulle perdite di carico. Inoltre, in questi lavori è stato implementato il *bridge method* su sfere di diametro di 3 cm a diversi valori di  $N$  pari a 4, 5, 6, 7, 8.

La generazione della mesh inizia dal *bridge* (con  $r_b = d_s/8$ ), per poi estendersi sulle sfere e poi sull'intero volume. Per la dimensione della mesh, Kuroki *et al.* [34] hanno utilizzato dimensioni in un intervallo tra  $d_s/6$  e  $d_s/15$ , dimostrando come i *bridges* non influenzino le grandezze macroscopiche per  $Re$  compreso tra 1 e 1000. È stato

inoltre simulato uno scambio termico implementando il *bridge method* come modifica locale dei punti di contatto, ma i risultati ottenuti in termini di numero di Nusselt sono risultati notevolmente diversi rispetto a quelli forniti dalla letteratura. Negli anni successivi è stato ideato un altro metodo per eliminare il punto di contatto tra le sfere, creando un flat (*cap method*) in corrispondenza del punto di contatto come mostrato in Figura 1.4 (d) [18]. La variazione del grado di vuoto dovuta a questa implementazione è stata la più piccola in confronto agli altri metodi di modifica già descritti. Infatti è stato evidenziato come per raggi del *bridge* pari a  $d_s/8$  si sono avuti errori sul grado di vuoto dello 0.3% che vuol dire 1% di errore per le perdite di carico. Con il *cap method*, invece, stando alle dimensioni suggerite da Eppinger e Seidler [18], si avrebbe una variazione del grado di vuoto dello 0.05% , con conseguente errore trascurabile sulle perdite di carico. Questi metodi sono stati analizzati separatamente e pochi sono i lavori che evidenzino il confronto tra i 4 metodi proposti per il trattamento dei punti di contatto sfera/sfera e sfera/parete. Nello studio di Augier *et al.* [15] sono stati analizzati fluidi liquidi in regime laminare ( $Re < 100$ ). Qui è stato studiato l'effetto della variazione del diametro delle sfere in un range tra -20% e + 20% nei confronti dello scambio termico. Non è stato considerato lo scambio termico sfera/sfera, ma lo scambio termico sfera/fluido. Inoltre è stata riscontrata una variazione del numero di Nusselt dello  $\pm 5\%$  rispetto alle correlazioni di letteratura. Il più recente lavoro sui punti di contatto è stato affrontato da Dixon *et al.* [1] nel cui lavoro sono stati sistematicamente implementati tutti e quattro i metodi di modifica geometrica della sfera. In particolare lo studio si è rivolto alla determinazione del coefficiente di drag e allo studio dei fenomeni di scambio termico in regime turbolento. Su base puramente teorica è stata ricavata una conducibilità termica fittizia del *bridge* che tiene conto delle proprietà termiche sia del fluido che della sfera. Gli autori hanno dimostrato che i metodi di modifica locali sono i migliori rispetto a quelli globali per il calcolo del coefficiente di drag e delle perdite di carico. Nonostante ciò solo il *bridge method* ha offerto la capacità di un approccio necessario ad un buon calcolo dello scambio termico sfera/sfera e sfera/parete a costo di risorse computazionali notevoli.

Sebbene in letteratura ci siano numerosi studi sull'analisi dei 4 approcci sopra citati, l'analisi dei punti di contatto in presenza di reazioni chimiche sembra non aver avuto un'analisi sistematica.

### **1.3 Scopo del lavoro e risultati principali**

Lo scopo di questo lavoro è quello indagare quale tra i metodi locali *bridge* e *cap* e globali *gap* e *overlap* di modifica dei punti di contatto è il più appropriato per descrivere correttamente i fenomeni di scambio di materia e termico in reattori catalitici eterogenei. Dopo aver scelto il tipo *mesh* più appropriata per modellare i letti impaccati di sfere, si è passati all'implementazione ed il confronto dei metodi partendo da casi semplici studio fino a reattori impaccati di sfere.

Nel Capitolo 3 sono stati studiati i fenomeni di reattività chimica e scambio termico in maniera indipendente per un caso sfera/sfera implementando i 4 metodi di modifica del punto di contatto. Tali approcci sono stati poi confrontati con un caso di riferimento.

La reattività chimica è stata studiata a diversi numeri di Damköhler per approfondire lo studio dal *regime chimico* fino al regime *diffusivo esterno*. Lo scambio termico è stato studiato con "fluido stazionario", (cioè imponendo la condizione di adesione per la velocità del fluido sulle pareti delle sfere) implementando il modello elaborato da Dixon *et al.* [1] per la determinazione della conducibilità termica del *bridge*. Lo stesso studio è stato fatto implementando un secondo modello più semplice assumendo la conducibilità del *bridge* e delle sfere uguali.

Nel Capitolo 4 è stata studiata una configurazione geometrica un po' più complessa: un sistema costituito da 4 sfere poste in serie in un reattore. Per verificare l'effetto delle modifiche locali dei punti di contatto anche a diversi regimi fluidodinamici, sono state fatte delle prove a diversi valori del numero di Reynolds per indagare l'effetto delle modifiche locali sul campo di moto in presenza di una reazione chimica catalitica superficiale per una diversa disposizione delle sfere nel reattore. Mentre per lo studio dello scambio termico le 4 modifiche dei punti di contatto sono state confrontate per evidenziare i diversi profili di temperatura che si instaurano all'interno delle sfere.

Nel Capitolo 5 sono stati studiati sia la reattività chimica che lo scambio termico confrontando i due metodi di modifica locale (*cap* e *bridge*) per un caso sfera/sfera. Grazie al calore generato dalla reazione chimica superficiale, si sono analizzati i profili di temperatura generati all'interno delle sfere ed indagati i campi di moto locali nella zona fluida in prossimità dei punti di contatto. Lo scopo di tale studio è stato quello di mostrare le differenze tra i due metodi locali rispetto ai fenomeni di scambio di materia assieme e quelli di scambio termico.

Infine nel Capitolo 6 sono stati analizzati in dettaglio l'effetto delle modifiche dei punti di contatto sulle grandezze macroscopiche che caratterizzano i letti impaccati di sfere. L'analisi si è focalizzata sull'effetto dell'area catalitica ed il grado di vuoto per diversi rapporti diametro del tubo e della sfera. Dopo aver tratto delle considerazioni su entrambi i fenomeni di trasporto, è stato studiato un "caso studio" di letto impaccato random di 20 sfere implementando il metodo di modifica locale *bridge*. Lo studio è stato eseguito in condizioni laminari in presenza di una reazione chimica catalitica esotermica. Sono stati indagati i fenomeni di scambio di materia, di calore ed i campi di moto locali tra le sfere. Infine sono state fornite delle importanti indicazioni per modellare in maniera efficace i reattori catalitici eterogenei per garantire lo studio dei fenomeni di scambio termico e di natura fluidodinamica che si instaurano.



---

## Bibliografia

- [1] Dixon A.G., Nijemeisland M., Stitt E.H., “*Systematic meshdevelopment for 3D CFD simulation of fixed beds: Contact points study*”, Computers and Chemical Engineering, 48, pp. 135-153, (2013).
- [2] <http://jbrwww.che.wisc.edu/home/jbraw/chemreacfun/ch7/slides-masswrxn.pdf>
- [3] Iordanidis A.A., “*Mathematical modeling of catalytic fixed bed reactors*”, Ph.D. Thesis, University of Twente, (2002).
- [4] Froment G.F. e Bischoff K.B., “*Chemical Reactor Analysis and Design*”, 2nd Edition, John Wiley & Sons, New York, (1990).
- [5] Kuipers J.A.M., Van Swaaij W.P.M., “*Computational fluid dynamics applied to chemical reaction engineering*”, Advances in Chemical Engineering, 24, pp. 227–328, (1998).
- [6] Dalman M.T., Merkin J.H., McGreavy C., “*Fluid flow and heat transfer past two spheres in a cylindrical tube*”, Computers & Fluids, 14, pp. 267-281,(1986).
- [7] Lloyd B. e Boehm R., “*Flow and heat transfer around a linear array of spheres*”, Numerical Heat Transfer Part A, 26, pp. 237-252, (1994).
- [8] Derkx O.R. e Dixon A.G., “*Determination of the fixed bed wall heat transfer coefficient using computational fluid dynamics*”, Numerical Heat Transfer Part A, 29, pp. 777-794, (1996).
- [9] Logtenberg S.A. e Dixon A.G., “*Computational fluid dynamics studies of fixed bed heat transfer*”, Chemical Engineering and Processing, 37, pp. 7-21, (1998)
- [10] Dixon A.G. e Cresswell D.L., “*Theoretical prediction of effective heat transfer parameters in packed beds*”, AIChE Journal, 25, pp. 663-676, (1979).
- [11] Logtenberg S.A., Nijemeisland M., Dixon A.G., “*Computational fluid dynamics simulations of fluid flow and heat transfer at the wall-particle contact points in a fixed bed reactor*”, Chemical Engineering Science, 54, pp. 2433-2439, (1999).

- 
- [12] Calis H.P.A., Nijenhuis J., Paikert B.C., Dautzenberg F M., van den Bleek C. M., “*CFD modeling and experimental validation of pressure drop and flow profile in a novel structured catalytic reactor packing*”, *Chemical Engineering Science*, 56(4), pp. 1713-1720, (2001).
- [13] Nijemeisland M. e Dixon, A.G., “*Comparison of CFD simulations to experiment for convective heat transfer in a gas-solid fixed bed*”, *Chemical Engineering Journal*, 82(1-3), pp. 231-246, (2001).
- [14] Atmakidis T. e Kenig E.Y., “*CFD-based analysis of the wall effect on the pressure drop in packed beds with moderate tube/particle diameter ratios in the laminar flow regime*”, *Chemical Engineering Journal*, 155(1-2), pp. 404-410,(2009).
- [15] Augier F., Idoux F., Delenne, J.Y., ”*Numerical simulations of transfer and transport properties inside packed beds of spherical particles*”, *Chemical Engineering Science*, 65(3), pp. 1055-1064,(2010).
- [16] Bai H., Theuerkauf J., Gillis P.A., Witt P.M., “*A coupled DEM and CFD simulation of flow field and pressure drop in fixed bed reactor with randomly packed catalyst particles*”, *Industrial and Engineering Chemistry Research*, 48(8),pp. 4060-4074, (2009).
- [17] Baker M.J. e Tabor G.R., “*Computational analysis of transitional air flow through packed columns of spheres using the finite volume technique*”, *Computers and Chemical Engineering*, 34(6), pp. 878-885,(2010).
- [18] Eppinger T., Seidler K., Kraume M., “*DEM-CFD simulations of fixed bed reactors with small tube to particle diameter ratios*”, *Chemical Engineering Journal*, 166(1), pp. 324-331, (2011).
- [19] Jafari A., Zamankhan P., Mousavi S.M., e Pietarinen K., “*Modeling and CFD simulation of flow behavior and dispersivity through randomly packed bed reactors*”, *Chemical Engineering Journal*, 144(3), pp. 476-482,(2008).
- [20] Lopes R.J.G. e Quinta-Ferreira, R.M., “*Numerical simulation of trickle-bed reactor hydrodynamics with RANS-based models using a volume of fluid technique*”, *Industrial and Engineering Chemistry*, 48(4), pp. 1740-1748, (2009).
-

- [21] Magnico P., “*Hydrodynamic and transport properties of packed beds in small tube-to-sphere diameter ratio: Pore scale simulation using an Eulerian and a Lagrangian approach*”, *Chemical Engineering Science*, 58(22), pp. 5005-5024, (2003).
- [22] Reddy R.K., & Joshi, J. B., “*CFD modeling of pressure drop and drag coefficient in fixed and expanded beds*”, *Chemical Engineering Research and Design*, 86(5), pp. 444-453, (2008).
- [23] Soleymani A., Turunen I., Yousefi H., Bastani D., “*Numerical investigations of fluid flow and lateral fluid dispersion in bounded granular beds in a cylindrical coordinates system*”, *Chemical Engineering Technology*, 30(10), pp. 1369-1375, (2007).
- [24] Lund K.O., Nguyen H., Lord S.M., Thompson C., “*Numerical correlation for thermal conduction in packed beds*”, *Canadian Journal of Chemical Engineering*, 77, pp. 769, (1999).
- [25] Bu S.S., Yang J., Zhou M., Li S.Y., Wang Q.W., Guo Z.X., “*On contact point modifications for forced convective heat transfer analysis in a structured packed bed of spheres*”, *Nuclear Engineering and Design*, 270, pp. 21–33, (2014).
- [26] Romkes S.J.P., Dautzenberg F.M., van den Bleek C.M., Calis H.P. A., “*CFD modeling and experimental validation of particle-to-fluid mass and heat transfer in a packed bed at very low channel to particle diameter ratio*”, *Chemical Engineering Journal*, 96(1-3), pp. 3-13, (2003).
- [27] Dixon A.G., Nijemeisland M., Stitt E.H., “*CFD simulation of reaction and heat transfer near the wall of a fixed bed*”, *International Journal of Chemical Reactor Engineering*, 1(1), (2003).
- [28] Gunjal P.R., Ranade V.V., Chaudhari R.V., “*Computational study of a single phase flow in packed beds of spheres*”, *American Institute of Chemical Engineers Journal*, 51(2), pp. 365-378, (2005).
- [29] Lee J., Yoon S., Park, G., Lee, W., “*Turbulence-induced heat transfer in PBMR core using LES and RANS*”, *Journal of Nuclear Science and Technology*, 44(7), pp.985-996, (2007).

- [30] Guardo A., Coussirat, M., Larrayoz, M. A., Recasens, F., & Egusquiza, E., "CFD flow and heat transfer in nonregular packings for fixed bed equipment design". *Industrial & Engineering Chemistry Research*, 43(22), pp.7049-7056, (2004).
- [31] Laguerre O., Ben Amara S., Alvarez G., Flick, D., "Transient heat transfer by free convection in a packed bed of spheres: Comparison between two modelling approaches and experimental results", *Applied Thermal Engineering*, 28(1), pp. 14-24, (2008).
- [32] Xu R.-N. e Jiang P.X., "Numerical simulation of fluid flow in microporous media", *International Journal of Heat and Fluid Flow*, 29(5), pp.1447-1455, (2008).
- [33] Tobis J., "Influence of bed geometry on its frictional resistance under turbulent flow conditions", *Chemical Engineering Science*, 55(22), pp.5359-5366, (2000).
- [34] Kuroki M., Ookawara S., Street D., Ogawa K., "High-fidelity CFD modeling of particle-to-fluid heat transfer in packed bed reactors, *European Congress of Chemical Engineering*, 6, (2007).

## 2 Generazione della mesh

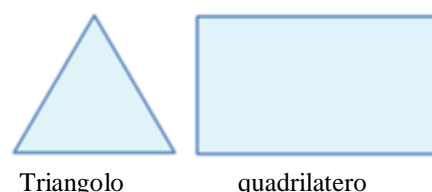
### 2.1 Introduzione

Le equazioni della fluidodinamica non sono in generale risolvibili analiticamente in forma chiusa nel loro dominio di definizione. Pertanto, per l'analisi fluidodinamica di un dato processo ci si avvale di tecniche di soluzione numerica (volumi finiti, elementi finiti, differenze finite, ecc.). L'idea che sta alla base di queste tecniche consiste nel suddividere il dominio iniziale in sottodomini più piccoli e di geometria più semplice, di risolvere le equazioni all'interno di ciascun sottodominio e di raccordare le soluzioni ai contorni in maniera tale da rispettare le condizioni iniziali sul dominio originale. Con il termine *mesh* o griglia si indica l'insieme di sottodomini o celle in cui è stato suddiviso il dominio iniziale [1].

In questo Capitolo si analizzeranno le diverse *mesh* usate nella CFD, le loro differenze e si discuterà della qualità di una *mesh*. Il Capitolo conclude focalizzando lo studio sui diversi generatori di *mesh* open-source disponibili, motivando infine la scelta di SALOME come appropriato generatore di *mesh* per il seguente lavoro di Tesi

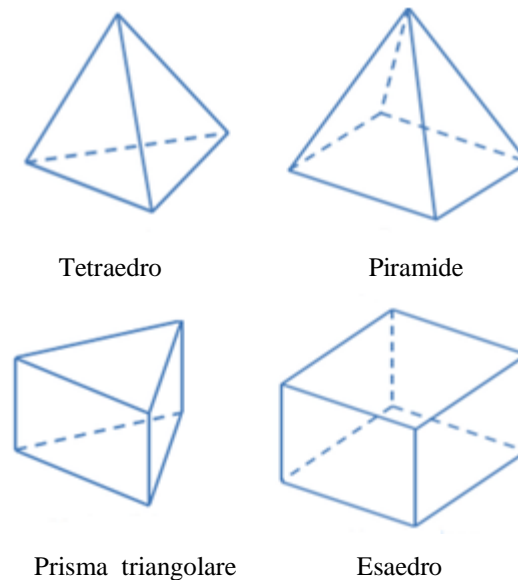
### 2.2 Tipi di mesh

La suddivisione del dominio iniziale in sottodomini più piccoli comporta l'introduzione di piccoli elementi geometrici chiamati celle. A seconda se il dominio iniziale è bidimensionale o tridimensionale gli elementi che comporranno la mesh saranno differenti. Per domini geometrici bidimensionali esistono due tipi di celle: i triangoli e i quadrilateri mostrati in Figura 2.1.



**Figura 2.1** Cella triangolare e cella a quadrilatero.

La superficie di una mesh elaborata con celle triangolari è sempre la più facile e veloce da creare: è la cella più utilizzata per mesh di tipo non strutturato. Le celle con quattro lati sono quelle comunemente utilizzate in griglie strutturate. Per domini tridimensionali le celle comunemente utilizzate sono mostrate in Figura 2.2. Ci sono tetraedri, piramidi quadrangolari, prismi triangolari ed esaedri.



**Figura 2.2** Celle tridimensionali.

Celle prismatiche triangolari o piramidali possono essere considerate come degenerazione delle celle esaedriche, dove alcuni lati sono stati ridotti a zero.

La scomposizione della geometria originaria può avvenire secondo vari criteri che dipendono dal particolare caso che si vuole studiare. Di seguito si elencano le principali tipologie di mesh [2]:

- ✓ mesh strutturate (cartesiane, curvilinee e block-structured grid)
- ✓ mesh non strutturate
- ✓ mesh ibride

In generale le mesh strutturate offrono una migliore accuratezza della soluzione, specialmente se le celle sono allineate come la direzione del flusso, mentre quelle non strutturate offrono una migliore adattabilità alla geometria, quindi vengono

particolarmente utilizzate per geometrie complesse. *Mesh ibride* di alta qualità sono in grado di combinare i vantaggi di entrambe le tipologie di *mesh*.

### 2.2.1 Mesh strutturate

Distinguiamo tre tipi di mesh strutturata: cartesiana, *body-fitted grid*, *block-structured grid*.

In generale si può affermare che:

- ✓ I punti della griglia sono posizionati nei punti di intersezione delle linee della mesh
- ✓ I vertici interni della griglia hanno un numero ben preciso di punti circostanti
- ✓ I vertici di una mesh possono essere “tradotti” in una matrice, la loro posizione nella griglia e nella matrice viene data dagli indici  $i, j$  per mesh 2D e  $i, j, k$  per mesh 3D.

Le prime *mesh* adottate furono quelle Cartesiane [3], che consistono in una suddivisione in celle quadrangolari (rettangoli o quadrati) del dominio di calcolo. Nelle tre dimensioni si fa uso di cubi o parallelepipedi. Nonostante queste *mesh* siano le più semplici da generare, il loro uso in caso di superfici non planari costringe ad approssimare le superfici con elementi piccolissimi, oppure a ricorrere ad algoritmi interpolatori che di norma sono funzione della geometria da descrivere. La Figura 2.3 rappresenta un esempio di *mesh* cartesiana generata sulla superficie di un semi cilindro.

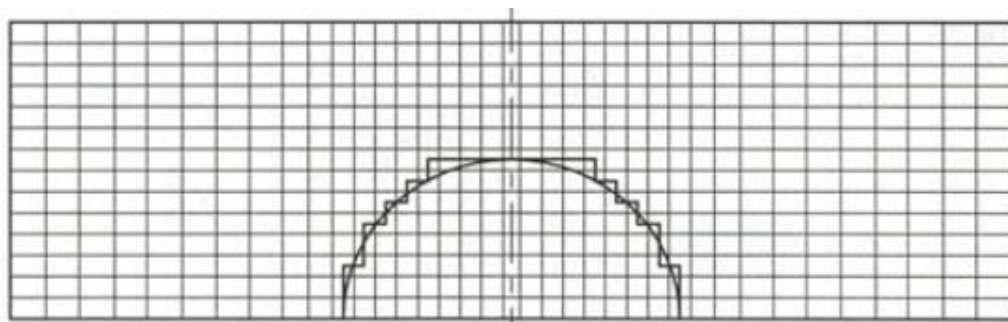


Figura 2.3 Generazione di una griglia cartesiana sulla superficie di un semi cilindro [2].

L'unico modo per rappresentare la superficie curva del cilindro è quella di utilizzare un'approssimazione a gradino o *stepwise* chiamata anche "*stepwise approximation*", ma questo metodo è abbastanza tedioso e richiede elevato tempo di calcolo. Inoltre le celle che si trovano dentro la parte solida del cilindro non prendono parte al calcolo quindi risulta necessario bloccare quelle piccole porzioni solide interessate, comportando maggiore accumulo di memoria e spreco di risorse computazionali. Questo mostra come i metodi CFD basati su sistemi di coordinate cartesiane o cilindriche abbiano delle limitazioni per geometrie irregolari e quindi anche molto complesse. Le *mesh* cartesiane offrono indubbiamente una grande facilità di programmazione, ma di contro un'estrema difficoltà (se non impossibilità) nel descrivere superfici che non siano planari. Per casi studio interessati da geometrie molto complesse sarebbe pertanto necessario lavorare con *mesh* che possano adattarsi meglio alla geometria da studiare.

Le *mesh* di tipo *body conforming* o *body fitted* rappresentano l'evoluzione delle *mesh* Cartesiane. In tale tipo di *mesh* si definiscono delle coordinate curvilinee che seguono l'andamento del corpo con il risultato di adattarsi perfettamente alla geometria. Diversamente da quanto accade con le *mesh* di tipo Cartesiano, in questo caso la PDE (*partial differential equation*) non può essere risolta nello spazio fisico, ma è necessario ricorrere ad uno spazio computazionale [5]. L'esistenza di due spazi costringe all'uso di trasformazioni di spazi vettoriali, ed impone anche una trasformazione delle condizioni al contorno [6].

Esistono due diversi tipi di *mesh body fitted*:

- ✓ Mesh curvilinee ortogonali
- ✓ Mesh non ortogonali.

Nelle *mesh* di tipo ortogonale le linee della griglia si intersecano in maniera perpendicolare tra loro (Figura 2.4).



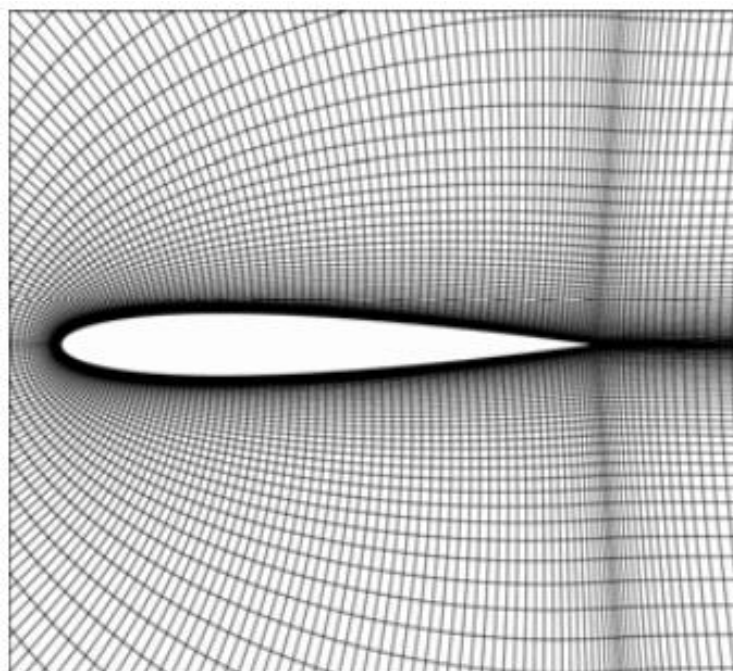


Figura 2.4 Esempio di mesh curvilinea ortogonale su un profilo alare [2].

Nelle mesh non-ortogonali invece le linee della griglia di calcolo non si intersecano tra loro formando un angolo di  $90^\circ$  (Figura 2.5). In entrambe le *mesh* la geometria coincide esattamente con le con la griglia di calcolo, quindi anche dettagli complessi della geometria possono essere incorporati accuratamente senza ricorrere all'approssimazione a gradino.

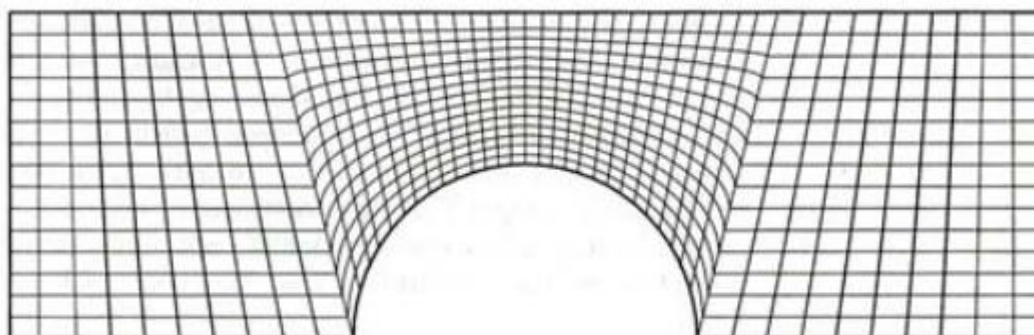
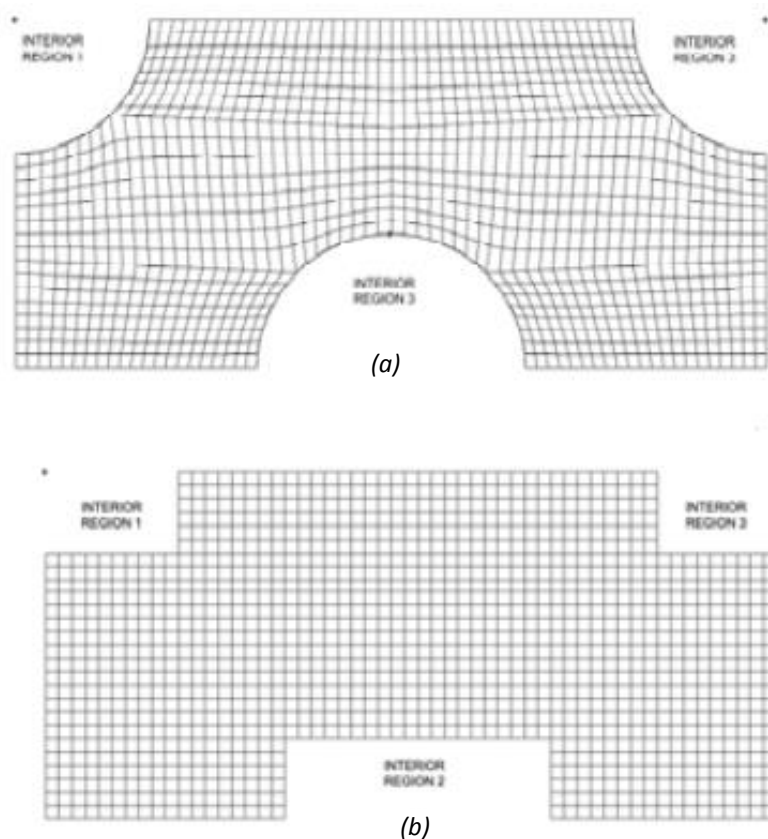


Figura 2.5 Mesh di tipo “body-fitted” attorno ad un semi cilindro [2].

Le *mesh* di tipo *body-fitted* hanno significativi vantaggi rispetto alle cartesiane, ma di contro danno dei problemi per la complessità geometrica. Infatti, le equazioni da risolvere in coordinate curvilinee risultano più complesse. La principale differenza

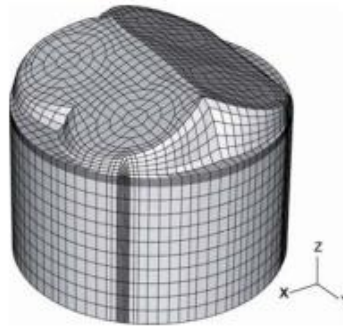
tra le diverse formulazioni sta nella mesh e nella scelta delle variabili dipendenti delle equazioni da risolvere. Per avere una *mesh* ben definita, soprattutto nei contorni della geometria, sarà necessario un elevato numero di celle, che di contro rappresenta una grande risorsa computazionale per la risoluzione del sistema lineare di equazioni. L'uso di mesh ortogonali e non garantisce l'adattamento alla geometria, ma possono verificarsi anche delle problematiche legate alla creazione della *mesh*. Infatti, per generare una *mesh* che includa tutti i dettagli della geometria sarà necessario mappare il dominio geometrico in un dominio computazionale. Un esempio di “*mappig process*” di una geometria è mostrato sotto in Figura 2.6.



**Figura 2.6** Mapping process di un dominio fisico in un dominio computazionale: (a) dominio fisico in coordinate cartesiane; (b) dominio computazionale ricavato attraverso un “*mapping process*” del dominio geometrico [2].

Per questa semplice geometria la mappatura risulta immediata, ma quando il dominio geometrico diventa più complesso o/e coinvolge un largo numero di dettagli interni,

tale operazione può risultare abbastanza lunga e tediosa. Le figure sottostanti mostrano la difficoltà che si riscontra per la generazione della mesh di un cilindro per un motore a 4 tempi (Figura 2.7). Le celle utilizzate sono di tipo esaedrico e quadrangolare. Inoltre sono state applicate diverse tecniche di smussamento per poter descrivere bene i dettagli geometrici.



**Figura 2.7 Mesh non-ortogonale strutturata di un cilindro di un motore a 4 tempi [2].**

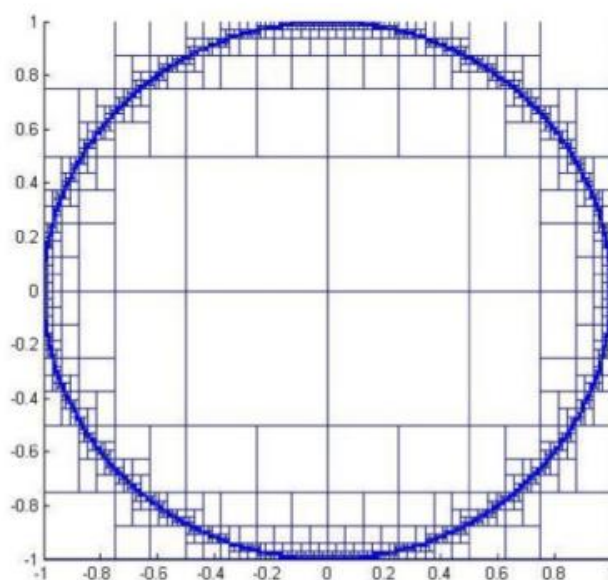
Il risultato è una mesh che contiene celle con angoli molto acuti e con elevato aspect ratio anche dopo lo smoothing. Le quattro regioni che hanno un'elevata densità della mesh sono caratterizzate da alti valori di indice di skewness (questi concetti saranno approfonditi nel Paragrafo 2.4) che possono dare problemi di stabilità ai solver CFD. Nonostante le mesh di tipo *body-fitted* hanno dei vantaggi rispetto alle *mesh* strutturate cartesiane, si incontrano ancora problemi con griglie ortogonali. A tal proposito si evidenziano le seguenti problematiche:

- ✓ elevato tempo di generazione della *mesh*
- ✓ se il dominio non può essere "mappato" interamente in un dominio rettangolare 2D o tridimensionale (parallelepipedo) 3D, questo si traduce in celle con elevato indice di *skewness*
- ✓ nelle zone in cui la mappatura è delicata, neanche le tecniche di smussamento consentono di aumentare la qualità della *mesh*
- ✓ se la geometria è molto complessa la mappatura risulta anche impossibile da effettuare.

Per superare i problemi riscontrati con la generazione della mesh con i metodi strutturati è stato sviluppato un altro metodo, chiamato block structured grid. Con

questo approccio il dominio è suddiviso in regioni ed ogni regione ha una *mesh* strutturata [7,8]. La *mesh* in ciascuna regione può essere diversa ed è inoltre possibile utilizzare diversi sistemi di coordinate, tale metodo risulta pertanto più flessibile del precedente [6].

La novità sta nel fatto che l'approccio consente l'uso di una *mesh* molto fitta, in zone dove questa è richiesta. Le *mesh* di tipo *blockstructured* con regioni sovrapposte sono chiamate griglie composite o chimera grids. La Figura 2.8 mostra un esempio di *mesh blockstructured*.



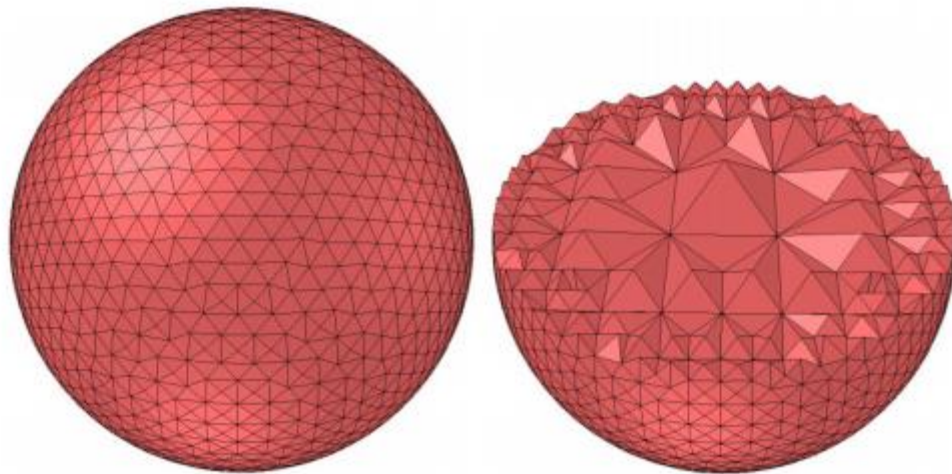
**Figura 2.8** Esempio di *block structured mesh* per una sfera.

Questo approccio combina i vantaggi di una griglia cartesiana che è facile da generare con equazioni semplici da discretizzare [6]. Di contro essa non è una *mesh* di tipo *body conforming*.

### 2.2.2 Mesh non strutturate

Le *mesh* di tipo *non strutturato* sono costituite da triangoli in due dimensioni e da piramidi, prismi o tetraedri in tre dimensioni. Le *mesh* non strutturate sono relativamente recenti (primi anni novanta) [9]. Tale *mesh* può essere pensata come

un caso limite delle *multi-block grid*, dove ciascuna cella è trattata come un blocco. Il vantaggio è che l'arrangiamento delle celle non è imposto, come per il caso delle mesh strutturate, può essere localizzata una mesh più fitta nelle zone più complesse senza spreco di risorse computazionali e adattandosi bene alla geometria da studiare [10]. Le celle possono avere qualsiasi forma, in CFD le celle triangolari o quadrangolari sono le più usate per problemi 2D, mentre per problemi 3D sono utilizzate celle esaedriche, prismatiche triangolari e tetraedriche. La Figura 2.9 sottostante mostra un esempio di mesh non strutturata.



**Figura 2.9** Esempio di *mesh* non strutturata per una sfera.

Mesh di questo tipo consentono di descrivere geometrie complesse [11], con costi computazionali e operativi (in termini di tempo speso per la generazione della *mesh*) drasticamente ridotti, rispetto a quanto necessario per sviluppare una mesh strutturata. Si accenna al fatto che i concetti base della generazione di mesh non strutturate sono stati esposti da Dirichlet in una sua pubblicazione del 1850 [12]. Il principale vantaggio di una mesh non strutturata è che è sempre possibile suddividere un dominio in elementi triangolari, nel caso di problemi bidimensionali, o tetraedrici, nel caso di problemi 3D [13]. È inoltre possibile variare la densità della griglia conformemente al gradiente della soluzione, la mesh viene infittita solo dove serve, per esempio laddove avvengono i fenomeni fisici di maggior interesse [14]. Ciò non

sempre è possibile quando si adottano *mesh* strutturate a causa delle difficoltà insite nella descrizione di geometrie complesse.

### **2.2.3 Mesh ibride**

Da un punto di vista di generazione delle *mesh*, la soluzione di utilizzare una *mesh* non strutturata è certamente quella più semplice da adottare sia in termini di tempo computazionale di generazione della *mesh* sia in termini di “ore” necessarie per impostare la *mesh*. È stato visto che la generazione di *mesh* strutturate, comporta una serie di problemi laddove le superfici dei corpi da descrivere non siano particolarmente semplici.

Si è pensato, quindi, di ideare delle *mesh* di tipo ibrido, ossia strutturate e non strutturate, per usufruire dei vantaggi di entrambe le tipologie [15].

Da un lato, le *mesh* strutturate garantiscono una maggiore accuratezza del risultato ottenuto dalla simulazione, dall’altro, quelle non strutturate facilitano il processo di generazione della *mesh* per geometrie complesse. In questa maniera si ottengono risultati soddisfacenti in termini di tempi di calcolo e precisione. In applicazioni di tipo fluidodinamico è spesso conveniente adottare *mesh* ibride, ovvero di tipo non strutturato nelle zone in cui si hanno notevoli dettagli geometrici e di tipo strutturato nel resto del volume di controllo dove la geometria è più semplice da discretizzare. Nella Figura 2.10 si riporta un esempio di *mesh* ibrida generata per un caso studio di reattività chimica.

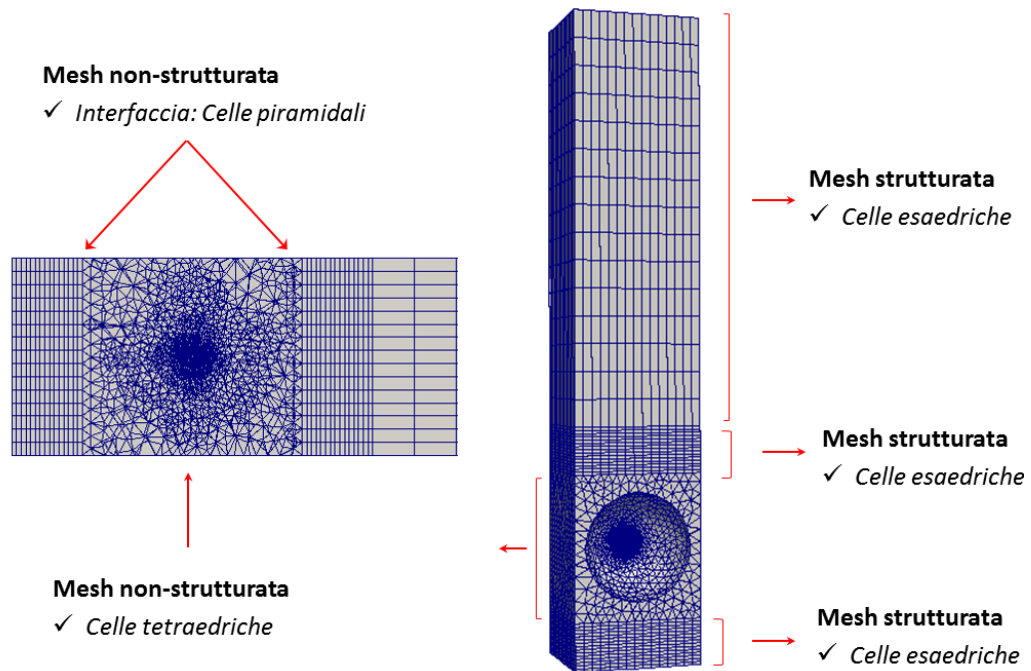


Figura 2.10 Mesh ibrida per un semplice caso studio sfera/sfera.

## 2.3 Qualità della mesh

Costruire una *mesh* non significa semplicemente trovare un qualsiasi ricoprimento del dominio di interesse, ma trovare una mesh di buona qualità. Chiaramente è difficile dare una definizione univoca di qualità di una *mesh*. Una *mesh* ottimale deve essere considerata in relazione al motivo per cui si è dovuta costruirla ed al suo successivo utilizzo. Pertanto non esiste una teoria generale che consenta di discriminare la qualità della *mesh*. Una regola abbastanza generale è quella di prestare molta attenzione alla forma delle celle, poiché la forma delle celle ha un effetto molto pronunciato sui metodi numerici. Oltre alla forma delle celle si possono verificare alcune caratteristiche della *mesh*.

Di seguito si presentano i più comuni strumenti per verificare le caratteristiche di una *mesh*.

### 2.3.1 Ortogonalità della mesh

L'ortogonalità di una *mesh* rappresenta la deviazione angolare del vettore  $\mathbf{S}$  (localizzato sul centro della faccia  $f$ ) dal vettore  $\mathbf{d}$  che connette i centri  $\mathbf{P}$  ed  $\mathbf{N}$  delle due celle. La non ortogonalità è un contributo significativo di errore in OpenFOAM per il modo in cui vengono calcolati i gradienti tra i centri delle facce delle celle [16]. OpenFOAM calcola i gradienti tra le celle considerando la differenza dei valori tra i centri di celle adiacenti e dividendola per la distanza. Il calcolo richiede un prodotto vettoriale dei gradienti che può introdurre un errore significativo a causa del non allineamento tra il vettore passante tra centri delle celle e quello normale alla faccia della cella come mostrato in Figura 2.11.

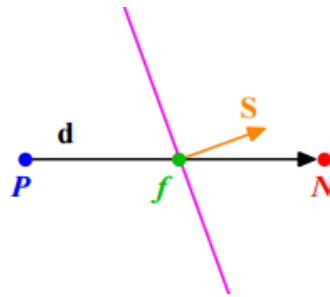


Figura 2.11 Ortogonalità tra due celle adiacenti.

Questo termine è particolarmente critico per i termini diffusivi, poiché la discretizzazione del Laplaciano porta a:

$$\mu \nabla^2 \Phi \rightarrow \iiint \nabla \cdot (\mu \nabla \Phi) dV \rightarrow \sum_f \mu_f \mathbf{S}_f \cdot (\nabla \Phi)_f \quad (2.1)$$

dove  $\Phi$  è la grandezza risolta,  $\mu$  è la diffusività (ad esempio, diffusività materiale, conducibilità termica, viscosità, ecc.), mentre  $\mathbf{S}_f$  rappresenta la linea direzionale di ciascuna faccia  $f$  della cella. In OpenFOAM è possibile correggere questo effetto aumentando il `nNonOrthogonalCorrectors` che si trova nel



fvSolutiondictionary. Il contributo significativo di errore in generale si ha per valori di angolo compresi tra 70 e 80 gradi.

### 2.3.2 Mesh skewness

L'indice di *skewness* (Figura 2.12) è la deviazione del vettore  $\mathbf{d}$  che connette le due celle  $P$  ed  $N$ , dal centro della faccia  $f$ . Il vettore deviazione è rappresentato con  $\Delta$  e  $f_i$  è il punto dove il vettore  $\mathbf{d}$  interseca la faccia  $f$ .

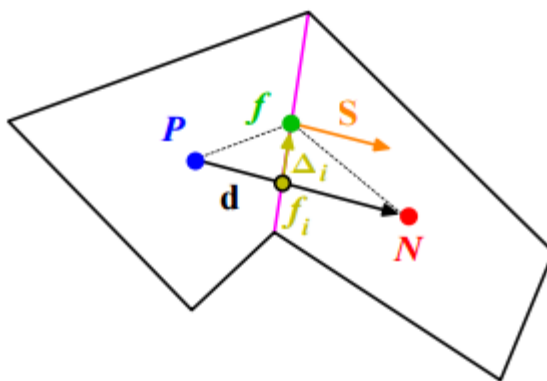


Figura 2.12 Indice di *skewness* tra due celle adiacenti.

Valori elevati dell'indice di *skewness* possono introdurre una significativa diffusione numerica alla soluzione a causa del modo in cui vengono calcolati i flussi tra celle adiacenti [16]. Il metodo per correggere la *skewness* chiamato skewCorrected method può aiutare a minimizzare questo effetto, ma alcune prove con simulazioni hanno dimostrato che l'accuratezza della soluzione per effetto della modifica mesh non sempre viene raggiunta. La *skewness* è importante nei termini convettivi presenti nelle equazioni da discretizzare, ad esempio:

$$(\mathbf{v} \cdot \nabla) \Phi \rightarrow \iiint \nabla \cdot (\mathbf{v} \Phi) dV \rightarrow \sum_f \mathbf{S}_f \cdot (\mathbf{v}_f \Phi_f) \quad (2.2)$$

dove  $\mathbf{v}$  è la velocità del fluido,  $\Phi$  è la grandezza risolta, mentre  $\mathbf{S}_f$  rappresenta la linea direzionale di ciascuna faccia  $f$  della cella. Alti valori dell'indice di *skewness* comportano una errata interpolazione della variabile fisica di studio (ad esempio una

frazione massiva, oppure una componente della velocità, ecc.) compromettendo i risultati della simulazione. Inoltre elevati valori dell'indice di *skewness* influenzano i termini convettivi delle equazioni da risolvere e aggiunge diffusività numerica alla soluzione. Valori ideali dell'indice di *skewness* sono quelli prossimi alla zero, ma in generale valori dell'ordine di 0.3–0.80 sono accettabili. In Tabella 2.1 si riporta la classificazione della qualità di una cella sulla base della *skewness*.

Valore dell'indice di <i>skewness</i>	0-0.25	0.25-0.5	0.5-0.8	0.8-0,95	0.95-0.99	0.99-1.00
qualità della cella	eccellente	buona	accettabile	povera	scarsa	pessima

Tabella 2.1 Qualità delle celle a diversi valori di *skewness* [17].

### 2.3.3 Mesh Aspect ratio

L'aspect ratio (Figura 2.13) di una cella è il rapporto tra la massima e minima lunghezza di una cella.

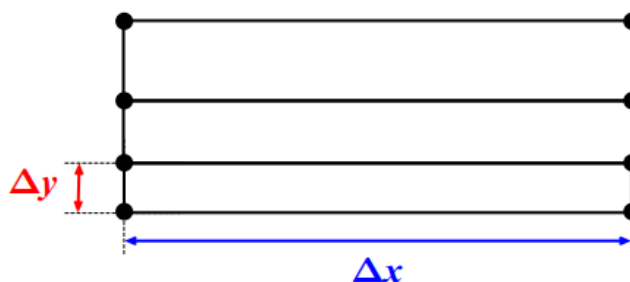


Figura 2.13 Lunghezze dei lati di una cella.

In generale è preferibile avere valori di aspect ratio (AR) prossimi ad uno nelle zone in cui si hanno forti gradienti della quantità in esame [18]. Questo viene generalmente ben rispettato per le mesh di tipo strutturato. In termini pratici, alti valori dell'aspect ratio portano ad avere delle matrici mal condizionate, peggiorando la velocità e l'accuratezza con cui vengono risolte le equazioni lineari, mentre bassi rapporti aumentano la velocità di calcolo. Alti valori di aspect ratio, inoltre, si ripercuotono su una sbagliata interpolazione della grandezza tra celle adiacenti traducendosi in una bassa accuratezza della soluzione. L'aspect ratio ideale per mesh

triangolari è la radice quadrata del rapporto tra il più grande ed il più piccolo degli autovalori dell'Hessiano. Per mesh triangolari, non fa tanta differenza se le celle sono molto allungate con angoli molto grandi o molto piccoli, l'importante è che l'aspect ratio non superi il valore ideale suggerito, poiché si rischierebbe di avere delle celle di bassa qualità [18].

### 2.3.4 Smoothness

Conosciuta anche come velocità di crescita, o uniformità che definisce la transizione tra due celle adiacenti (Figura 2.14).

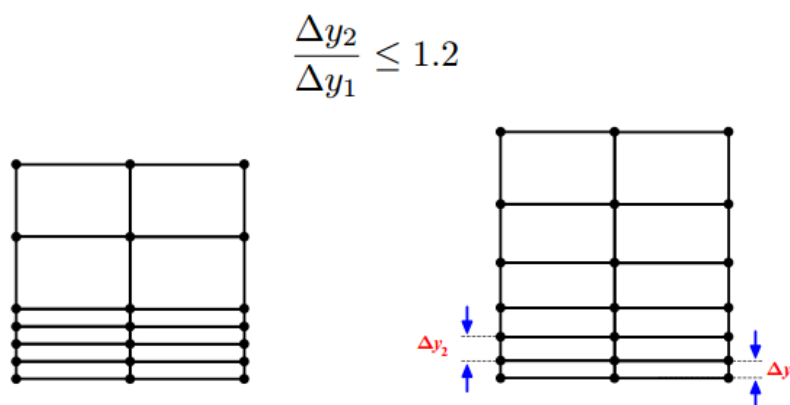


Figura 2.14 Smoothness in una cella.

Alti valori di questo rapporto aggiungono diffusività numerica alla soluzione, quindi rapporti ideali si aggirano intorno a 1.2 che equivale a variazioni del 20% circa [19].

## 2.4 Presentazione dei generatori di mesh

Creare una *mesh* per una simulazione CFD è un processo abbastanza tedioso, specialmente per geometrie complesse con piccoli angoli, superfici sferiche oppure piccole distanze tra le superfici di corpi solidi. La generazione di una *mesh* pur essendo solo un requisito per l'esecuzione di calcoli, comunque ha un forte impatto sulla efficienza e la precisione di calcolo; l'elemento o la forma e le dimensioni delle celle influenzano sia la velocità, che la precisione numerica dei risultati.

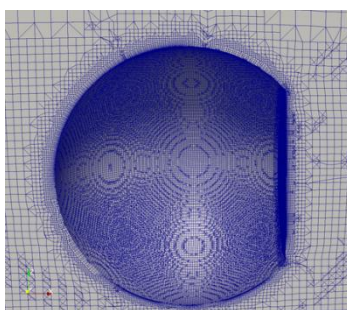
È importante sottolineare che i risultati nella CFD sono molto sensibili rispetto al tipo di *mesh*. Esistono diversi generatori di mesh open-source come Engrid [20],

---

GMSH [21], NETGEN [22], Tetgen [23], SALOME [24] ed il generatore di mesh nativo di OpenFOAM: SnappyHexMesh [25]. In questo lavoro di Tesi saranno presentati due generatori di mesh SnappyHexMesh e SALOME. Questi software sono open-source, il che significa che il codice di tali software è liberamente disponibile e modificabile da chiunque. Verranno descritti in maniera separata e oggettiva ed infine si giustificheranno le motivazioni della scelta di SALOME come generatore di *mesh* per la generazione automatica di *mesh* ibride.

### 2.4.1 SnappyMeshHex

Il metodo utilizzato da SnappyHexMesh per generare una *mesh* è abbastanza diverso rispetto ai tradizionali metodi. Questo metodo utilizza una procedura automatica per creare *mesh* esaedriche ortogonali sia attorno che all'interno di una assegnata superficie geometrica. La superficie su cui SnappyHexMesh crea la *mesh* è in formato STL (*Standard Triangle Language*) e può essere generata con un software pre-processing come SALOME o gmsh. Una *mesh* in formato STL è sostanzialmente una *mesh* triangolare che approssima la superficie considerata attraverso triangoli, pertanto il file contiene solo le coordinate degli angoli dei triangoli costituenti la *mesh*. Questo è un grande vantaggio per rappresentare una geometria ma allo stesso momento si perdono informazioni circa la forma matematica originale della superficie (es. sfere o cilindri).



**Figura 2.15** Esempio di generazione della *mesh* con SnappyHexMesh su una porzione di sfera.

La *mesh* di partenza viene generata con blockMesh che ricopre tutto il dominio geometrico. Questa *mesh* ha l'impostazione di “*refinement level pari a 0*”. Il livello

di definizione della mesh o anche *refinement level* sulla superficie STL viene gestita dal file `SnappyHexMeshDict` in cui vengono definiti i valori desiderati per avere la mesh voluta. Ad esempio i layers che vengono generati sulla superficie e la massima non-ortogonalità consentita della mesh possono essere definiti a valori desiderati. L'intero processo di generazione della *mesh* è iterativo e sequenziale. La qualità della *mesh* può essere migliorata in alcuni casi utilizzando la funzione “*mesh layer*”, questo consente di creare dei layers sulla superficie che si adattano meglio sulla superficie geometrica.

### **2.4.2Salome**

SALOME è una piattaforma CAE (Computer Aided Engineering) che integra soluzioni open-source per la simulazione, meshing, il calcolo numerico sul modello (solving) e la visualizzazione ed analisi dei risultati (post-processing o visualization). SALOME ha molteplici funzionalità per la creazione e manipolazione della geometria solida. Si possono creare nuovi solidi su superfici già esistenti aggiungendo semplicemente nuove superfici e generare così dei volumi chiusi. Esistono delle funzioni che permettono di riparare geometrie che presentano dei difetti, rimuovere piccoli dettagli e combinare più superfici tra loro.

Inoltre grazie alla vasta gamma di algoritmi che SALOME offre, si possono generare mesh tetraedriche, esaedriche e prismatiche. Per la generazione di *mesh* tetraedriche ci sono diversi parametri che bisogna definire al fine di ottenere la mesh desiderata. Per generare mesh di tipo esaedrico si può modificare la dimensione lineare delle celle in particolari zone, ad esempio infittire attraverso un grading la mesh nella zona di interesse. Oltre alle *mesh* cartesiane si possono creare anche delle mesh strutturate di tipo “*body fitted*”. Le *mesh* prismatiche vengono generate per estrusione di una “*surface mesh*” esistente in volume. La *mesh* può essere composta da *mesh* di ordine inferiore chiamate “*sub-meshes*” semplificando così il dispendio computazionale complessivo per la generazione della *mesh*. Questo perché gli algoritmi della generazione e ottimizzazione della mesh non gestiscono l'intera geometria, ma solo porzioni di geometria alla volta. Se la mesh generata contiene delle anomalie, l'utente può modificare la *mesh* anche manualmente attraverso delle funzioni

disponibili come “riparazione della *mesh*”. Potenzialmente si può anche modificare la posizione dei singoli nodi. Una volta generata la *mesh*, si può passare ad una verifica attraverso diversi “controls tools” quali:

- ✓ *length*: dà informazioni sulla lunghezza della cella
- ✓ *free edges*: mostra i lati di una cella che appartengono ad una faccia di una cella
- ✓ *length 2D*: verifica della lunghezza dei lati una cella
- ✓ *borders at multi-connection 2D*: mostra il numero di facce che si trovano sul lato di una cella selezionata
- ✓ *area*: verifica dell’area di una cella
- ✓ *taper*: verifica dell’ortogonalità di una faccia di una cella
- ✓ *aspect ratio*: verifica il valore dell’aspect ratio e lo confronta con il valore ideale
- ✓ *minimum angle*: verifica dell’angolo minimo tra due facce di due celle adiacenti
- ✓ *warping angle*: verifica la planarità di una superficie quadrata o rettangolare costruita da due triangoli
- ✓ *skew*: verifica l’indice di *skewness* di un cella
- ✓ *aspect ratio 3D*: verifica l’aspect ratio per celle 3D
  
- ✓ *volume*: da informazioni circa il volume delle celle della mesh.

La generazione automatica di una *mesh* non strutturata che utilizza tetraedri è relativamente veloce con entrambi i metodi (2D *surface meshing* con gli entrambi algoritmi Mefisto e Netgen). Per una mesh 3D si utilizza in genere l’algoritmo NETGEN-1D2D3D oppure una combinazione di Mefisto, NETGEN e discretizzazione 1D. La caratteristica più importante di SALOME è quella di consentire l’utente di generare in maniera sequenziale la geometria e la *mesh* con le caratteristiche desiderate attraverso codici di calcolo scritti in linguaggio python.

## 2.5 Scelta del generatore di mesh

Esistono diversi generatori di *mesh* open-source ed ognuno ha i propri vantaggi e svantaggi pertinenti al caso da studiare. Inizialmente nel lavoro di Tesi è stato utilizzato il generatore di *mesh* nativo di OpenFOAMSnappyHexMesh, ma considerando che le geometrie da studiare sono più o meno complesse si è preferito utilizzare un software che crei una *mesh* che sia non strutturata in corrispondenza dei dettagli geometrici, tale da garantire una *mesh* ben definita nelle zone laddove SnappyHexMesh non riesce bene a creare la *mesh*, per la presenza di zone in cui la geometria risulta abbastanza complessa. Il grande vantaggio di SALOME è quello di creare sia la geometria che la *mesh* in maniera sequenziale attraverso codici di calcolo in python. A differenza di SnappyHexMesh, SALOME non necessita file STL della superficie geometrica del caso da studiare. Questo è un grande vantaggio soprattutto in termini di risorse computazionali, infatti una volta compilato lo script definendo opportunamente i parametri geometrici del caso da creare ed il tipo di mesh da generare viene lanciato da terminale (*batch mode*) utilizzando le librerie di SALOME. Un altro vantaggio è quello di garantire la generazione della mesh ibride anche per geometrie molto complicate, come ad esempio letti impaccati con sfere a costo di parallelizzazione del codice [26] per evitare elevati tempi di calcolo.

In Appendice si illustreranno i codici di calcolo scritti per generare le mesh per i diversi casi studio e si mostrerà la versatilità di SALOME nella generazione delle mesh indipendentemente dal tipo di approccio di modifica dei punti di contatto implementato.

---

## Bibliografia

- [1] Fletcher C.A.J., “*Computational Techniques for Fluid Dynamics*”, I and II, Springer-Verlag, Berlin, (1991).
- [2] Versteeg H.K.a.W.M., “*An introduction to Computational Fluid Dynamics*”, LongmanScientific & Technical, (1995).
- [3] [http://ta.twi.tudelft.nl/users/vuik/numanal/alagon\\_scriptie.pdf](http://ta.twi.tudelft.nl/users/vuik/numanal/alagon_scriptie.pdf)
- [4] Concus P., Golub G.H. O’Leary, D.P., “*A generalized Conjugate Gradient Method for the Numerical Solution of Elliptic Partial Differential Equations*”, Sparse Matrix Computations, pp.309-332, New York, (1986).
- [5] [http://cfm.mace.manchester.ac.uk/twiki/pub/Main/TimCraftNotes\\_All\\_Access/ms4-bodyfg.pdf](http://cfm.mace.manchester.ac.uk/twiki/pub/Main/TimCraftNotes_All_Access/ms4-bodyfg.pdf)
- [6] Briggs W.L.A, “*Multigrid Tutorial*”, 2<sup>nd</sup> edn, SIAM Publications, (1987).
- [7] Hestenes M.R. e Stiefel E.L., “*Methods of Conjugate Gradient for Solving Linear Systems*”, 49 pp. 409-436, (1952).
- [8] Hutchinson B.R. e Raithby G.D., “*A Multigrid Method Based on the Additive Correction Strategy*”, Numerical Heat Transfer, 9(5) pp.511-537, (1986).
- [9] Joe F. Thompson, “*A Reflection on Grid generation in the 90s: Trends Needs and Influences*”, 5th International Conference on Numerical Grid Generation in Computational Field Simulations, Mississippi State University, pp.1029-1110, (1996).
- [10] Staten M.L. e N.L. Jones, “*Local Refinement of Three-Dimensional Finite Element Meshes*”, Engineering with Computers, 13, pp.165-174, (1997).
- [11] Chen H. e Bishop J., “*Delaunay Triangulation for Curved Surfaces*”, Proceedings, 6th International Meshing Roundtable, pp.115-127, (1997).
- [12] Dirichlet G.L., “*Über die Reduction der positiven quadratischen Formen mit drei unbestimmten ganzen Zahlen*”, Journal für die reine und angewandte Mathematik, 40, pp.209-227, (1850).
- [13] Timothy J. Baker, “*Automatic Mesh Generation for Complex Three-Dimensional Regions Using a Constrained Delaunay Triangulation*”,



- Engineering with Computers, 5, pp.161-175,(1989).
- [14] George P.L. e Borouchaki H., Delaunay, "*Triangulation and Meshing: Application to Finite Elements*", Hermes, France, pp.413,(1998).
- [15] Guskov I., Khodakovsky A., "*Hybrid Meshes: Multiresolution using regular and irregular refinement*", SoCG , pp. 264-272, (2002).
- [16] Rhoads J., "*Effects of grid quality on solution accuracy*", OpenFOAM workshop, (2014).
- [17] <http://www.bakker.org/dartmouth06/engs150/07-mesh.pdf>
- [18] Miller G.L. Talmor D. Teng S.H., "*Optimal Good-Aspect-Ratio Coarsening for Unstructured Meshes*", Proceedings of the Eighth Annual ACM-SIAM Symposium on Discrete Algorithms , New Orleans, LA., (1997).
- [19] Field D.A., "*Laplacian smoothing and Delaunay triangulations*", Communications in Applied Numerical Methods, 4, pp. 709-712, (1988).
- [20] Engrid, web site: <http://engits.eu/en/engrid>
- [21] GMSH,web site: <http://geuz.org/gmsh/>
- [22] NETGEN, web site: <http://www.hpfem.jku.at/netgen/>
- [23] Tetgen, web site: <http://wias-berlin.de/software/tetgen/>
- [24] SALOME, web site: <http://www.salome-platform.org/>
- [25] SnappyHexMesh, web site:  
<http://cfdirect.openfoam/userguide/snappyhexmesh/>
- [26] <http://www.opencascade.org/about/news/issue181/>

## 3 Semplice caso studio sfera/sfera

### 3.1 Introduzione

Da un'attenta analisi di letteratura, è stato visto che sono stati effettuati pochi studi CFD per sistemi costituiti da due sfere poste a contatto Kim *et al.* [1], Yoon e Yang [2], Schouveiler e Brydon [3] hanno investigato il campo di moto a valle di due sfere poste ad una certa distanza tra loro, ma non hanno proposto dei metodi risolutivi alle problematiche riscontrate sulla generazione della *mesh* descritti nel Capitolo 1. Dixon *et al.* [4] hanno svolto degli studi sullo scambio termico per un sistema costituito da due sfere. Gli studi da loro svolti si sono basati sull'effetto delle modifiche sia locali che globali dei punti di contatto rispetto ai fenomeni di scambio termico. Il modello da loro proposto ha consentito di descrivere meglio lo scambio termico tra due sfere. Lund *et al.* [5] hanno elaborato un modello termico numerico dettagliato per due semisfere poste a contatto considerando la rugosità delle sfere e la coesione non perfetta vicino il punto di contatto a causa del piccolo gap di fluido. Gli autori hanno mostrato come la resistenza termica sia fortemente influenzata dalla dimensione del gap tra le sfere e soprattutto dalle diverse conducibilità termiche del fluido e dei solidi. Nonostante gli studi citati sullo scambio termico siano stati affrontati in maniera sistematica e dettagliata, fornendo utili informazioni per la modellazione numerica dei punti di contatto tra le sfere, sembrano non dare informazioni in presenza di una reazione chimica catalitica superficiale.

Lo scopo di questo Capitolo è quello di coprire questa parte mancante e fornire così una metodologia sistematica per la generazione di geometrie e *mesh* in grado di modificare in maniera efficace i punti di contatto tra due sfere e discutere soprattutto sull'influenza delle modifiche effettuate sulla reattività chimica e scambio termico. In particolare, i punti di contatto tra le due sfere sono stati modificati con i 4 approcci descritti nel Capitolo 1.

Per confrontare i 4 metodi di modifica implementati è stato necessario generare un caso senza modifica del punto di contatto. Pertanto è stata richiesta una sistematica metodologia per la creazione delle 4 geometrie ed il caso di riferimento al fine di garantire un efficace confronto tra i metodi utilizzati e criticare così i risultati ottenuti. L'effetto delle modifiche dei punti di contatto sulla reattività chimica è stato studiato utilizzando il solver `catalyticSimpleFOAM` [6,7]. Inoltre per indagare tale effetto sul processo chimico superficiale rispetto a quello diffusivo esterno, le analisi sono state condotte a tre diversi numeri di Damköhler ( $Da_1 = 0.015$ ,  $Da_1 = 1.5$ ,  $Da_1 = 150$ ). Queste analisi consentono così di criticare i 4 metodi utilizzati a diversi regimi: dal regime chimico fino al regime di mass transfer. Infine i risultati ottenuti sono stati analizzati rispetto alla variazione di area catalitica offerta dai diversi approcci. Lo studio sullo scambio termico è stato condotto con il solver nativo di OpenFOAM: `chtMultiRegionSimpleFoam` [8]. La particolarità di questo studio a differenza del lavoro di Dixon *et al.* [4] è che le prove sono state effettuate imponendo la condizione di no slip per la velocità sulle pareti del reattore e della superficie delle sfere ed il volume della regione fluida è stato considerevolmente ridotto per diminuire il dispendio computazionale per effettuare le simulazioni. Questo studio base è stato importante per comprendere il meccanismo di scambio termico tra le sfere per ogni approccio di modifica implementato. Infatti, la presenza di un flusso convettivo, che incide direttamente sulle sfere, potrebbe modificare i meccanismi di scambio termico a causa delle modifiche effettuate, come descritto dallo studio riportato da Dixon *et al.* [4].

La generazione delle geometrie e *mesh* per entrambi gli studi sono state fatte per via automatica attraverso gli *script* in linguaggio `python` [9] riportati in Appendice. Una volta creati gli *script* sono stati lanciati dal terminale utilizzando la libreria di SALOME e generando in maniera sequenziale la geometria e la *mesh* con le caratteristiche desiderate.

## 3.2 Studio della reattività chimica

Per studiare l'effetto della singola modifica sulla grandezza macroscopica di interesse, l'analisi è stata condotta in maniera sistematica studiando altri 4 sotto casi per ciascun approccio implementato, pertanto i casi da studiare in totale sono risultati 16 più il caso di riferimento. I 4 metodi sono stati tra loro confrontati valutando la conversione della specie chimica limitante coinvolta nella reazione chimica superficiale.

### 3.2.1 Generazione della geometria e della mesh

Il dominio geometrico del caso sfera/sfera è mostrato nella Figura 3.1. Per evitare grossi dispendi computazionali sono state considerate solo due semisfere posizionate all'interno di un reattore con sezione rettangolare con dimensioni assegnate.

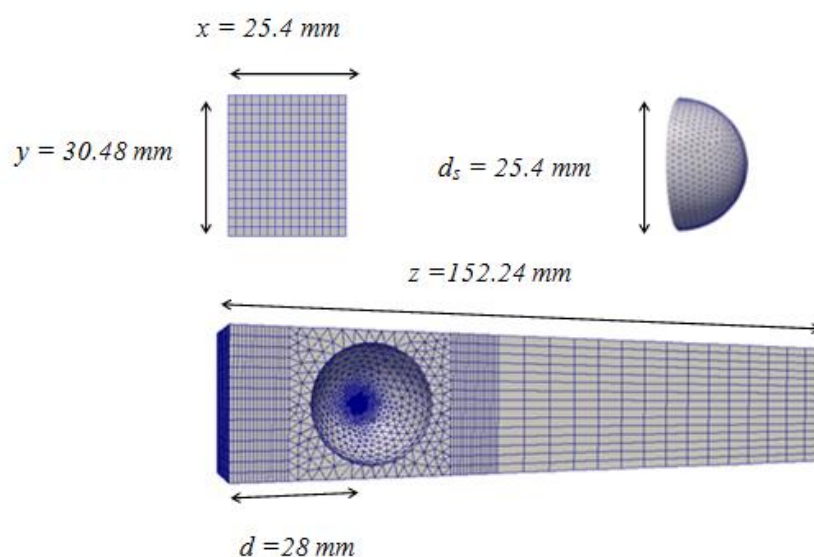
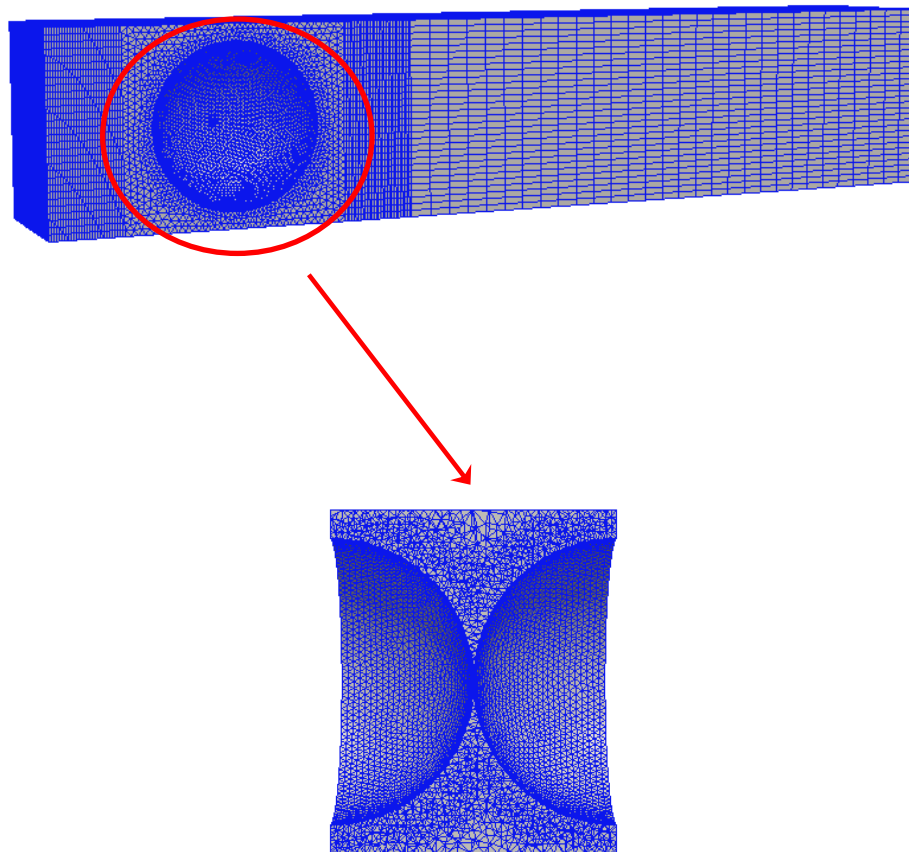


Figura 3.1 Dimensioni del reattore e delle semisfere utilizzate.

L'outlet del reattore è stato posto a distanza sufficientemente lontana dalle superfici catalitiche delle sfere per consentire lo sviluppo del campo di moto. Scelte opportunamente le dimensioni del reattore e delle sfere, si è passati alla creazione delle modifiche locali e globali delle sfere per eliminare i punti di contatto. Il caso di riferimento è stato creato senza nessuna modifica alle sfere. Il vertice che divide

le superfici delle due sfere è stato utilizzato come un vertice per la generazione della *mesh*, pertanto il punto di contatto è stato considerato come un vero e proprio nodo della *mesh* non strutturata. Per evitare elevati valori dell'indice di *skewness* e dell'aspect ratio in quella zona è stata imposta una dimensione lineare delle celle molto bassa compresa tra  $d_p/230$  e  $d_p/250$ , ottenendo delle *mesh* con elevato numero di celle e con valori accettabili dell'indice di *skewness* intorno a 0.65–0.80 e con aspect ratio tra 2–7, per contro però i tempi di calcolo sono risultati notevolmente più elevati rispetto ai 4 approcci.



**Figura 3.2** Particolare della *mesh* in una sezione trasversale del reattore.

### 3.2.1.1 Metodo di modifica locale “bridge”

Uno dei metodi proposti per la modifica del punto di contatto sfera/sfera è quello del bridge. Questo metodo proposto da Ookawara *et al.* [10] consiste nel sostituire il punto di contatto con un cilindro. Il cilindro interposto tra le due sfere garantisce così l'assenza di una zona difficile per la generazione della *mesh*. L'approccio sembra essere abbastanza interessante, ma per verificare l'influenza del *bridge* sulla reattività chimica è necessario fare delle prove su dei casi con un *bridge* di dimensione assegnata. Dalle considerazioni di tipo geometrico, (Figura 3.3) è stato possibile calcolare il raggio del *bridge*  $r_b$ , la lunghezza  $h_b$  del bridge conoscendo il diametro della sfera ed imponendo un dato valore di rapporto raggio *bridge* su diametro sfera ( $r_b/d_s$ ).

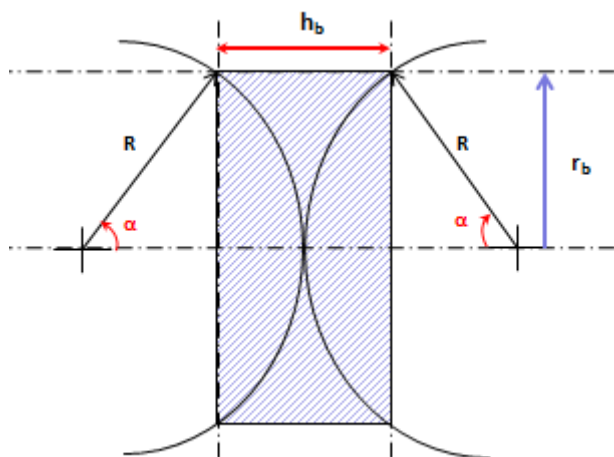


Figura 3.3 Analisi geometrica di un cilindro (in azzurro il *bridge*) interposto tra due sfere.

$$r_b = R \cdot \sin \alpha \quad (3.1)$$

$$l_b = 2 \cdot R - 2 \cdot R \cdot \cos \alpha \quad (3.2)$$

$$l_b = 2 \cdot R \left( 1 - \cos \left[ \sin^{-1} \left( \frac{r_b}{R} \right) \right] \right) \quad (3.3)$$

Nella Tabella 3.1 sono riportati i valori del raggio e lunghezza del *bridge* per diversi rapporti  $r_b/d_s$ . I valori  $r_b/d_s$  sono gli stessi utilizzati da Dixon *et al* [4] nel loro studio.

	$r_b/d_s$	$r_b$ [mm]	$h_b$ [mm]
<i>bridge 1</i>	0.05	1.27	0.127
<i>bridge 2</i>	0.10	2.54	0.513
<i>bridge 3</i>	0.15	3.81	1.171
<i>bridge 4</i>	0.20	5.08	2.121

Tabella 3.1 Valori del raggio e della lunghezza del *bridge* per diversi rapporti  $r_b/d_s$ .

Sia la geometria che la mesh sono state generate per via automatica attraverso lo script in python chiamato `bridge_method-cata_2sfere.py` riportato in Appendice. Lo script implementato è risultato abbastanza semplice, infatti una volta creati il reattore e le semisfere di dimensioni assegnate, è stato generato un cilindro tra le sfere di dimensioni assegnate, come mostrato in Figura 3.4.

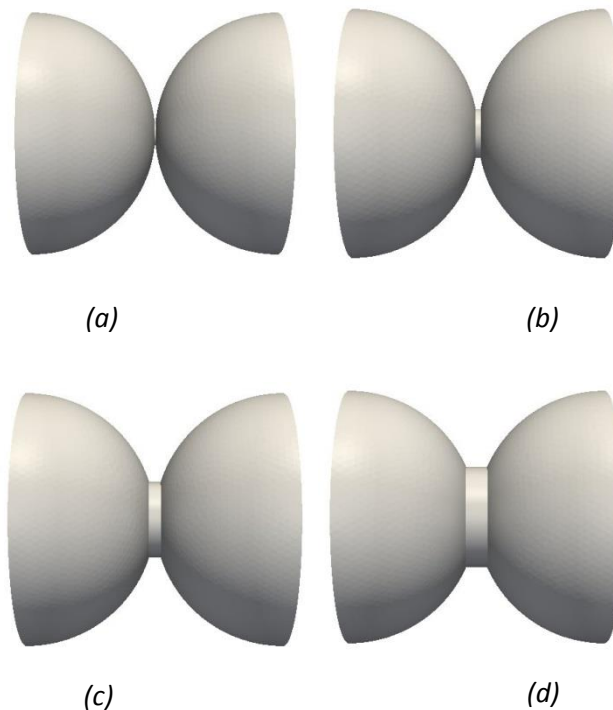


Figura 3.4 Sfere con diverse dimensioni del *bridge*: (a) *bridge 1*, (b) *bridge 2*, (c) *bridge 3*, (d) *bridge 4*.

Le superfici così modificate sono state nominate come `reactingWall`, le quali rappresentano le superfici catalitiche effettive mostrate in Figura 3.4.

### 3.2.1.2 Metodo di modifica locale “cap”

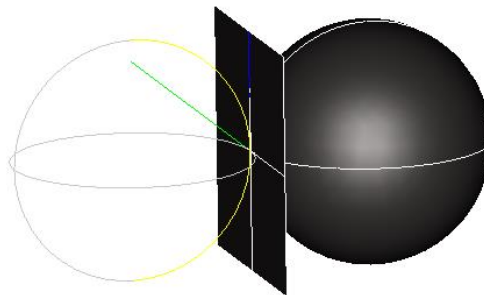
L'altro metodo di modifica locale proposto da Eppinger e Seidler [11] è il *cap*. Questo metodo consente di eliminare il punto di contatto sfera/sfera rimuovendo una porzione di volume della sfera in corrispondenza del punto di contatto, le sfere così avranno una certa distanza tra loro maggiore di zero. Il metodo può essere implementato in due modi in SALOME.

Il primo metodo consiste nella creazione di un *bridge* tra le sfere e successivamente attraverso la funzione booleana tra solidi “cut operation” si rimuove una porzione di volume dalle sfere pari esattamente a quello del *bridge* interposto.

Il secondo metodo invece è un po' più complesso. Considerando due sfere (sfera 1 e sfera 2) poste a contatto, si procede come segue:

1. identificazione della posizione del punto di contatto.

Ottenuto il punto di contatto, si è proceduto con la creazione di un vettore normale alla superficie con origine nel punto di contatto, creando così un piano tangente alla superficie diretto come il vettore ad essa normale (Figura 3.5).



**Figura 3.5 Creazione del piano tangente al punto di contatto tra le sfere.**

2. Creato il piano perpendicolare alla linea congiungente i due centri della sfere, si è proceduto con l'estrusione del piano verso l'interno della sfera di un valore desiderato che il valore del *cap* per ciascuna sfera. Il processo così ha generato un parallelepipedo. Infine attraverso la funzione booleana “cut” la sfera è stata tagliata dal parallelepipedo generato, il solido risultante è stato una sfera con un *cap* in corrispondenza del punto di contatto (Figura 3.6).



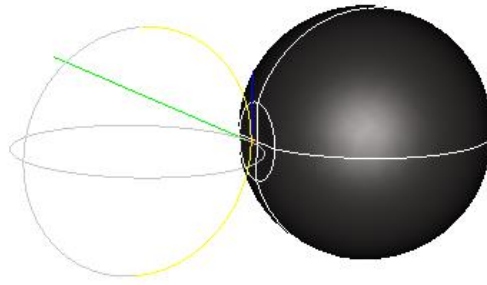


Figura 3.6 Creazione di un cap nella sfera attraverso l'operazione booleana "cut".

3. Infine gli stessi step sono stati ripetuti anche per la sfera 2 (Figura 3.7).

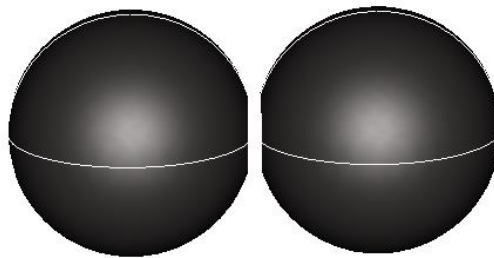


Figura 3.7 Creazione di un *cap* in entrambe le sfere.

Entrambi i metodi sono corretti ed è utile utilizzarli per casi semplici come quello sopra citato. Sebbene il primo approccio sia più veloce rispetto al secondo, non riesce a garantire la stessa precisione e qualità della geometria finale per sistemi più complessi, come ad esempio un letto impaccato. Questi aspetti verranno discussi in dettaglio nel Capitolo 6.

Tale procedura è stata resa automatica in uno script chiamato `cap_method-cata_2sfere.py` per generare i 4 sotto casi con caps a dimensione assegnata (Figura 3.8). I valori del *cap* per ciascuna sfera sono stati definiti  $h$  e corrispondono esattamente a metà lunghezza dei *bridges* calcolati precedentemente. In Tabella 3.2 sono riportati i 4 valori del *cap*.

	$h_b$ [mm]	$h$ [mm]
<i>cap 1</i>	0.127	0.0635
<i>cap 2</i>	0.513	0.2565
<i>cap 3</i>	1.17	0.585
<i>cap 4</i>	2.21	1.605

Tabella 3.2 Valori  $h$  del caputilizzati pari alla metà della lunghezza dei *bridges*  $h_b$ .

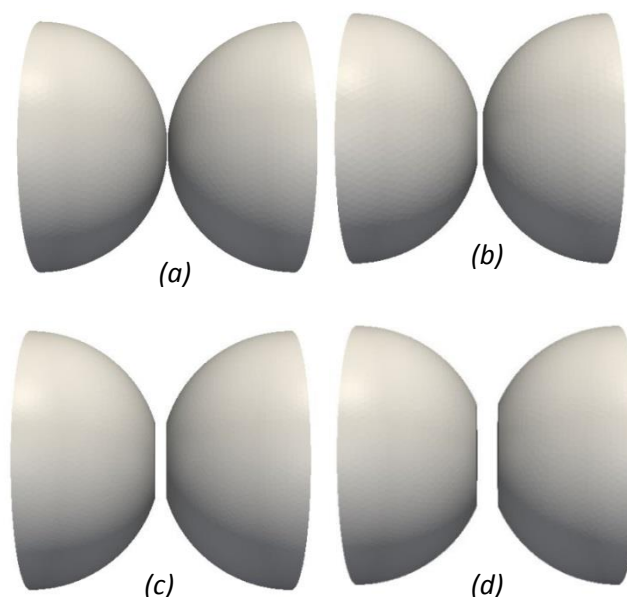


Figura 3.8 Sfere con diverse dimensioni del *cap*: (a) *cap 1*, (b) *cap 2*, (c) *cap 3*, (d) *cap 4*.

### 3.2.1.3 Metodo di modifica globale “*gap*”

Il metodo globale *gap* è un metodo di modifica globale, pertanto la modifica effettuata interessa tutta la superficie della sfera, a differenza dei metodi di modifica locale. Tale metodo consiste nella riduzione del diametro delle sfere evitando così che si tocchino [12]. Risulta fondamentale definire in maniera ragionevole la minima distanza tra le sfere per creare il *gap*. L'ipotesi di base è quella di utilizzare un valore di *gap* minimo tale per cui si garantisca un dispendio computazionale minore rispetto al riferimento, in termini di numero di celle e di tempo speso per generare la *mesh*. Effettuando delle prove è stato visto come applicando una riduzione dei diametri

delle sfere pari allo 0.25 % si ottengono mesh con valori di skewness compresi tra 0.61 e 0.725 (Tabella 3.3 e Figura 3.9). Con valori percentuali inferiori è necessario aumentare considerevolmente il numero di celle per ottenere una buona qualità della *mesh*. L'implementazione di questo approccio, nello script `gap_method-cata_2sfere.py` per generare i 4 sotto casi, consiste nel definire la variazione percentuale dei diametri delle sfere desiderata e successivamente generare le semisfere con *gap* assegnato.

	diametro gap [mm]	Variazione % diametro
<i>gap 1</i>	12.668	0.25
<i>gap 2</i>	12.637	0.5
<i>gap 3</i>	12.605	0.75
<i>gap 4</i>	12.573	1

Tabella 3.3 Diversi valori dei diametri per il metodo globale “*gap method*”.

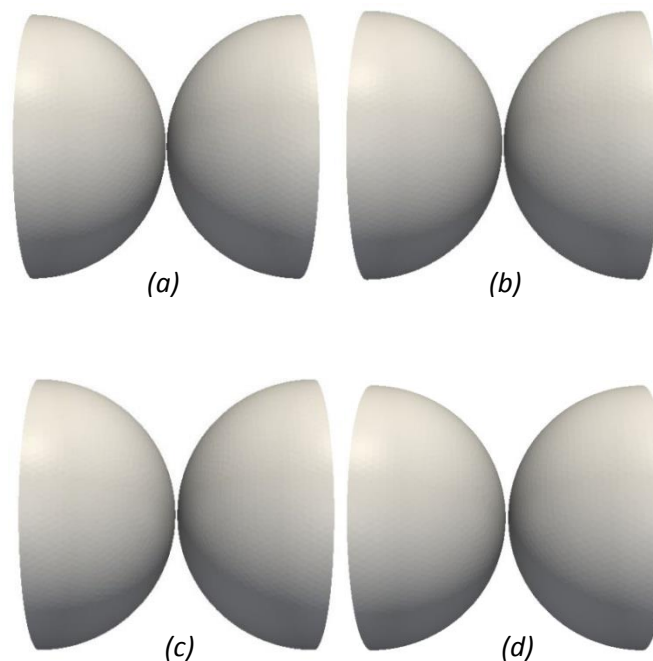


Figura 3.9 Sfere con diverse dimensioni del *gap*:(a)*gap 1*, (b) *gap 2*, (c) *gap 3*, (d) *gap 4*.

### 3.2.1.4 Metodo di modifica globale “*overlap*”

L'ultimo metodo di modifica globale del punto di contatto è l'*overlap* o semplicemente sovrapposizione. Date due sfere che condividono il punto di contatto l'idea di base di questo metodo è stata quella di aumentare il diametro delle sfere eliminando così il punto di contatto. Si è creata quindi una superficie di contatto tanto più grande quanto maggiore è stata la sovrapposizione [13].

Di seguito si riportano i valori dei diametri utilizzati per creare i 4 sotto casi con sovrapposizioni sempre maggiori (Tabella 3.4 e Figura 3.10).

	diametro overlap [mm]	Variazione % diametro
<i>Overlap 1</i>	12.900	1.550
<i>Overlap 2</i>	13.527	6.114
<i>Overlap 3</i>	14.153	10.266
<i>Overlap 4</i>	14.780	14.073

Tabella 3.4 Valori dei diametri per i 4 casi dell'*overlap*.

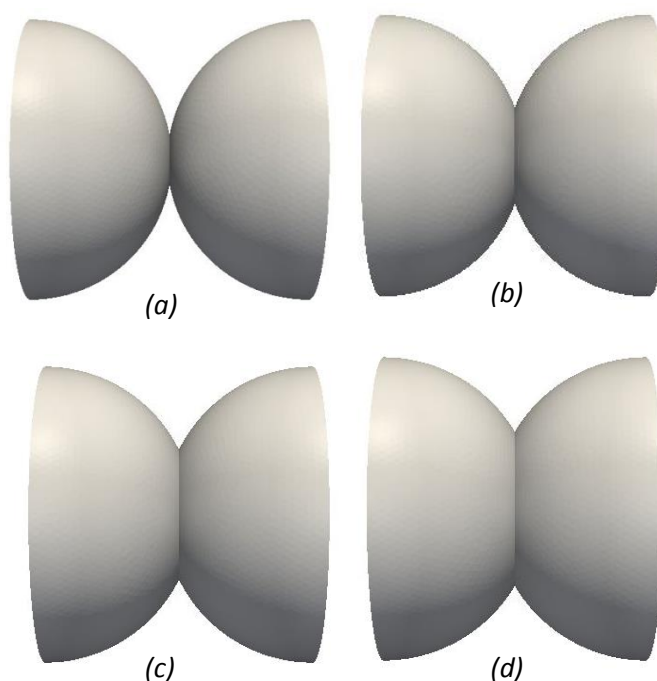


Figura 3.10 Sfere con diverse sovrapposizioni: (a) *overlap 1*, (b) *overlap 2*, (c) *overlap 3*, (d) *overlap 4*.

In generale questo metodo elimina considerevolmente la generazione di celle di bassa qualità. Questo è possibile da valori di sovrapposizione da circa l'1 % in su. Per valori inferiori è necessario localizzare delle celle di dimensioni molto piccole nelle zona di congiunzione delle due superfici. Pertanto sono state utilizzate geometrie con sovrapposizioni maggiori dell'1 % per evitare dispendio di elevate risorse computazionali.

I 4 sotto casi sono stati generati attraverso lo script `overlap_method-cata_2sfere.py` riportato in Appendice. Per tutti i casi è stata utilizzata una cinetica semplice:



a diversi numeri di Damköhler per indagare il la conversione nei diversi regimi, da regime chimico fino a regime di mass transfer. In Tabella 3.5 sono riportati i valori delle costanti cinetiche per i diversi numeri di Damköhler.

Da [-]	$k_{cin}$ [kmol/(m <sup>2</sup> · s)]
0,015	$53,59 \cdot 10^{-6}$
1,5	$53,59 \cdot 10^{-4}$
150	$53,59 \cdot 10^{-2}$

**Tabella 3.5 Valori delle costanti cinetiche a tre diversi numeri di Damköhler.**

La frazione massiva del fluido in ingresso e le condizioni operative sono riportate in Tabella 3.6 come anche i valori delle temperature, pressione, velocità.

Specie	frazione massiva [-]	condizioni operative
H <sub>2</sub>	$29.85 \cdot 10^{-4}$	$v = 0,2 \text{ m/s}$
O <sub>2</sub>	$11.845 \cdot 10^{-3}$	$P = 101325 \text{ Pa}$
N <sub>2</sub>	$98.517 \cdot 10^{-2}$	$T = 373,15 \text{ K}$

**Tabella 3.6 Frazioni massive delle specie chimiche in ingresso e condizioni operative utilizzate.**

L'azoto è la specie inerte mentre l'ossigeno è la specie limitante. Il numero di Reynolds vale:

$$Re = \frac{v \cdot d_s \cdot \rho}{\mu} \quad (3.5)$$

mentre il numero di Damköhler rispetto alla specie limitante si può calcolare come:

$$Da_{O_2} = \frac{PM_{mix} \cdot R_{O_2} \cdot d_s}{\rho_{mix} \cdot \mathfrak{D}_{O_2}} \quad (3.6)$$

dove  $PM_{mix}$  e  $\rho_{mix}$  sono rispettivamente il peso molecolare e la densità della miscela,  $R_{O_2}$  è la velocità di reazione riferita alla specie chimica limitante,  $d_s$  il diametro della sfera e  $\mathfrak{D}_{O_2}$  la diffusività dell'ossigeno.

Le condizioni al contorno (*boundary conditions*) [14] sono state fissate come segue:

	<b>T [K]</b>	<b>U [m/s]</b>	<b>P [Pa]</b>	<b>frazione massiva [-]</b>
<b>inlet</b>	fixedValue	fixedValue	zeroGradient	fixedValue
<b>outlet</b>	zeroGradient	zeroGradient	fixedValue	zeroGradient
<b>reacting Wall</b>	zeroGradient	fixedValue	zeroGradient	zeroGradient
<b>fixedWall</b>	fixedValue	fixedValue	zeroGradient	zeroGradient

**Tabella 3.7** Condizioni al contorno assegnate a ciascuna patch del reattore.

Il `fixedValue` è una condizione al contorno che impone un dato valore assegnato della grandezza considerata. Tale condizione prende il nome di *condizione di Dirichlet*. Ad esempio è stato utilizzato per imporre valori di frazione massiva entranti al reattore, il valore della velocità del fluido in ingresso e la sua temperatura. L'altra condizione al contorno utilizzata è stata il gradiente nullo della grandezza sulla patch considerata. Questa condizione si impone utilizzando il termine `zeroGradient` ed è stata denominata *condizione di Neumann* [15].

Ad esempio avere un gradiente nullo di temperatura all'uscita del reattore vuol dire imporre la condizione adiabatica di superficie perfettamente isolata.

La Figura 3.11 mostra il significato matematico di questa condizione al contorno.



**Figura 3.11** Schematizzazione della condizione a gradiente nullo per la temperatura, lungo la coordinata  $x$  di una generica superficie [10].

Gli schemi numerici adottati sono stati quelli di default [14-19]. Solo per il caso di riferimento alcuni schemi numerici sono stati modificati e riportati in Tabella 3.8.

	<i>Default</i>	<i>In questo lavoro</i>
<b>interpolationSchemes</b>	linear	linear
<b>gradSchemes</b>	Gauss linear	Gauss linear
<b>snGradSchemes</b>	orthogonal	limited 0.5
<b>laplacianSchemes</b>	Gauss linear orthogonal	Gauss linear limited 0.5

**Tabella 3.8** Differenti schemi numerici utilizzati per il caso di riferimento.

- Lo schema `interpolationSchemes` contiene i termini interpolati tipicamente dai centri della cella ai centri delle facce della cella. Lo schema scelto è quello lineare.
- Il `gradSchemes` consente di definire come viene calcolato il gradiente della grandezza fisica considerata normale alla superficie della cella. Pertanto è la componente normale alla superficie del gradiente dei valori tra i centri di due celle che condividono la stessa faccia. Lo schema utilizzato in questo lavoro è `limited  $\psi$` . Il simbolo  $\psi$  si chiama fattore di blending e varia tra 0 e 1. Con

valori di  $\psi$  pari a 1 si ha uno schema `corrected` valido per mesh di buona qualità, con valori invece pari a 0 lo schema diventa `uncorrected` utilizzato quando si hanno delle elevate non ortogonalità e quindi per mesh di scarsa qualità. Considerando che la mesh, per il caso di riferimento, presenta delle facce non ortogonali tra loro il valore  $\psi$  è stato settato pari a 0.5 in accordo con i suggerimenti forniti da Guerrero [18].

Anche il `nNonOrthogonalCorrectors` è stato impostato ad un valore pari a 3.

- Il `gradSchemes` contiene i termini con l'operatore gradiente. La prima parola specifica il tipo di discretizzazione a volumi finiti, che è appunto l'integrazione di Gauss, mentre la parola `linear` indica che l'interpolazione dei valori dai centri della cella ai centri della faccia è lineare.
- Il `laplacianSchemes` contiene i termini con l'operatore laplaciano. Il tipo di discretizzazione per il calcolo del laplaciano è quello di Gauss. Tale schema richiede anche di definire sia il tipo di interpolazione che il modo in cui viene calcolato il gradiente normale alla superficie della cella. Lo schema scelto in questo lavoro per l'interpolazione è stato di tipo lineare, mentre il calcolo del gradiente normale alla superficie è stato impostato a `limited 0.5` anziché `orthogonal` come di default.

### 3.2.2 Analisi della convergenza della mesh

Per avere dei risultati accurati, occorre avere una soluzione che non dipenda dalla discretizzazione del dominio. Per verificare ciò è necessario fare delle prove di sensitività della soluzione dalla *mesh* in modo da avere un numero tale di celle che garantiscano una soluzione indipendente dal tipo di discretizzazione utilizzata e in modo da ottimizzare i tempi di calcolo. Questo numero di celle deve essere il minimo possibile. La *mesh* utilizzata è stata una *mesh* ibrida (Figura 3.12) pertanto ci sono due tipi di celle: celle strutturate (esaedri) e celle non strutturate (tetraedri e piramidi).



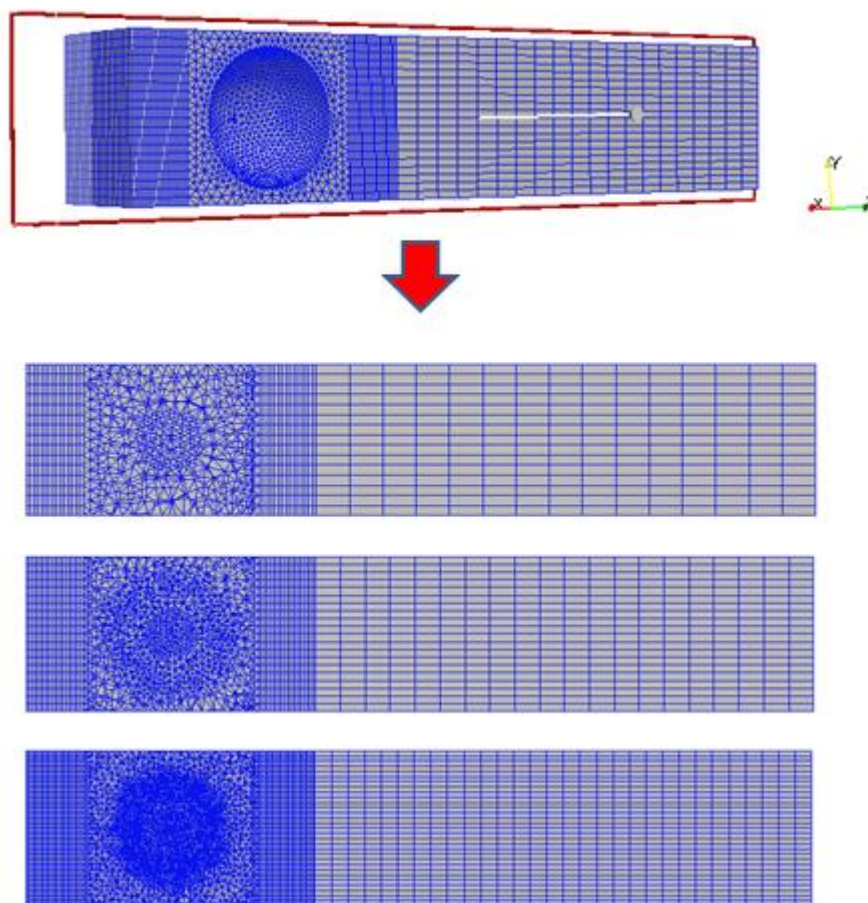
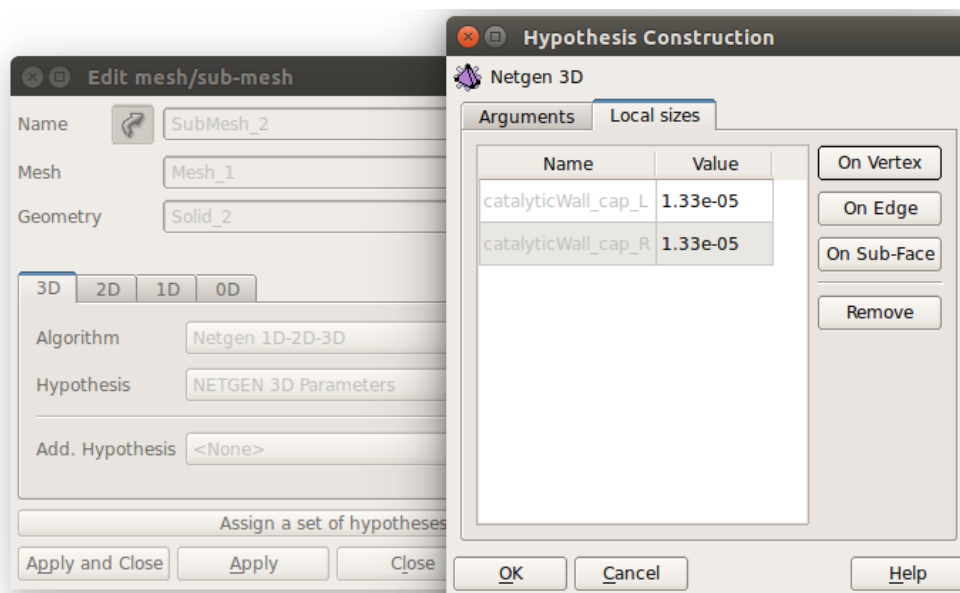


Figura 3.12 Sezione longitudinale di tre *mesh* ibride a numero di celle via via maggiore.

Per infittire la *mesh* si sono gradualmente aumentate contemporaneamente sia le celle non strutturate che quelle strutturate. La generazione della *mesh* strutturata è stata fatta attraverso l'algoritmo di default per *mesh* esaedriche imponendo un numero di celle fissato sul lato del reattore lungo la direzione  $z$ . Mentre per la *mesh* non strutturata è stato utilizzato l'algoritmo NETGEN-1D2D3D.

All'interfaccia tra le due diverse *mesh* ci sono delle celle piramidali che garantiscono la continuità della *mesh*. I formati disponibili in SALOME per importare le *mesh* direttamente in ambiente OpenFOAM non consentono di esportare *mesh* con celle piramidali. Questo problema è stato risolto utilizzando uno script in python chiamato `salomeToOpenFOAM` [20] che consente di importare qualsiasi tipo di *mesh* in ambiente OpenFOAM.

In generale il caso di riferimento è stato quello che ha richiesto più risorse computazionali. L'alto numero di celle per il caso di riferimento si spiega perché sono stati imposti dei bassi valori della dimensione lineare delle celle tetraedriche nel punto di contatto tra le sfere per ridurre al minimo celle con facce non ortogonali tra loro. È stato imposto un valore compreso tra  $d_s/400$  e  $d_s/500$ , con un growth rate compreso tra 0.04 e 0.06 garantendo così una crescita molto lenta delle celle nella zona circostante il punto di contatto. Un valore di 0.04 genera celle con una variazione del lato pari al 4 % rispetto al lato della cella adiacente.



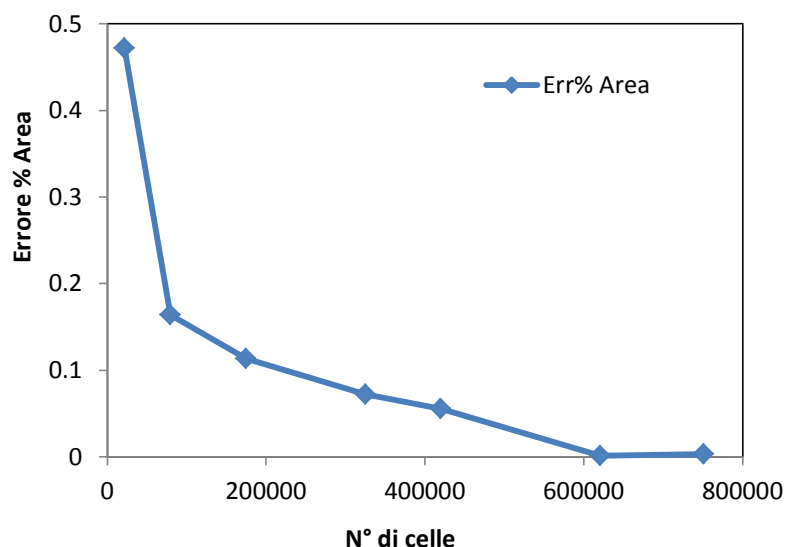
**Figura 3.13** Particolare dell'ipotesi utilizzata in SALOME per generare una *mesh* tetraedrica con dimensione delle celle assegnata nel punto di contatto.

Dal punto di contatto si genera così una *mesh* fitta che si sviluppa nella regione fluida circostante. Mantenendo bassi valori del growth rate e della dimensione della cella si riescono a eliminare quasi del tutto celle con elevate non ortogonalità.

Lo stesso principio è stato applicato anche per i metodi di modifica locale. Applicando il *cap* oppure il *bridge* con dimensioni molto piccole si potrebbe ottenere *mesh* con elevate non ortogonalità e per ridurle al minimo sarebbe necessario imporre valori molto bassi delle dimensioni delle celle per garantire *mesh* di buone

qualità. Però così non si risolve il problema di fondo poiché il prezzo da pagare è quello di utilizzare risorse computazionali paragonabili a quelle richieste per generare il caso di riferimento descritto. Sono stati utilizzati valori del *cap* e dimensione del bridge riportati nelle Tabelle 3.1–3.4 che garantiscono *mesh* con ottimi valori dell’aspect ratio e della *skewness* [19] con dispendi computazionali inferiori rispetto al caso di riferimento. Per il *cap method* è stata imposta una dimensione lineare delle celle sulle superfici del cap generate dopo la modifica. Per infittire la *mesh* nel filetto di fluido tra le sfere si sono imposti bassi valori del growth rate delle celle tetraedriche. Infine per il *bridge method* la dimensione lineare delle celle è stata impostata sulla superficie laterale del bridge a valori compresi tra  $d_s/300$  e  $d_s/400$ . I metodi di modifica globale hanno richiesto un minor numero di celle rispetto agli altri metodi. Le simulazioni sono state eseguite con mesh a numero di celle via via crescente. Dopo di che sono stati riportati in un grafico i valori della conversione della specie limitante ( $O_2$ ) calcolati per ogni simulazione effettuata rispetto al numero di celle utilizzate. Si è proceduto in questo modo finché non si è notato un plateau, cioè fino a che non si è ottenuto un valore di conversione indipendente dal numero di celle. Il valore di plateau della conversione di  $O_2$  ottenuta è quindi quello di cui si terrà conto poiché indipendente dal numero di discretizzazioni del dominio e quindi più accurato. Per eseguire simulazioni con numero di celle via via crescente, le soluzioni sono state così rimappate attraverso l’utility `mapFields`. Questo comando consente di interpolare i risultati ottenuti su una mesh ad un numero fissato di celle a mesh con più elevato numero di celle. I risultati di conversione di  $O_2$  ottenuti sono stati riportati in un grafico. Considerando che lo scopo di questo studio è quello di confrontare dei valori di conversione ottenuti dai vari approcci con il caso di riferimento, l’indipendenza dei risultati rispetto alla discretizzazione del dominio è fondamentale. Pertanto visto che la superficie delle sfere è una superficie curva si è ritenuto necessario monitorare l’area catalitica generata dalla mesh. In pratica si sono confrontati i valori di area catalitica direttamente calcolata da SALOME che è l’area geometrica esatta del caso con l’area catalitica generata dalla mesh calcolata con il postProcessore [22]. Nella Figura 3.14 si riporta la variazione di area catalitica per il caso di riferimento da un numero di celle pari 22512 fino ad una *mesh* con 751569

celle. Quest'ultima mesh ha un errore pari allo 0.0085 % rispetto alla superficie catalitica teorica.



**Figura 3.14** Confronto tra area catalitica esatta e quella generata dalla mesh per il caso di riferimento.

Lo studio della convergenza della *mesh* è stato fatto a  $Da = 0.015$ . Per numeri di Damköhler superiori sono state fatte delle prove con la mesh precedente rispetto a quella di convergenza per  $Da = 0.015$ , per verificare l'indipendenza della discretizzazione del dominio anche a numeri di Damköhler più alti. Solo il caso di riferimento è stato portato da 620651 a 751569 celle per  $Da = 150$ . In basso si riportano le Tabelle 3.9–3.12 per ciascun caso riportando le caratteristiche delle mesh utilizzate.

	N° di celle totali	N° di celle catalitiche	Errore % Area catalitica	MAX valore di skewness	MAX non ortogonalità
<b>riferimento</b>	751569	45221	0.0085	0.791	82.55
<b>bridge 1</b>	619896	44985	0.0091	0.558	62.01
<b>bridge 2</b>	511956	43025	0.0075	0.621	66.67
<b>bridge 3</b>	521569	41198	0.0098	0.615	54.58
<b>bridge 4</b>	428927	39954	0.0101	0.587	60.11

**Tabella 3.9** Confronto della *mesh* tra il riferimento ed i 4 casi del *bridge method*.

	N° di celle totali	N° di celle catalitiche	Errore % Area catalitica	MAX valore di skewness	MAX non ortogonalità
<b>riferimento</b>	751569	45221	0.0085	0.791	82.55
<b>cap 1</b>	702356	46254	0.0115	0.618	71.33
<b>cap 2</b>	766563	44254	0.0078	0.629	66.45
<b>cap 3</b>	805246	41325	0.0066	0.584	54.21
<b>cap 4</b>	953626	42596	0.0072	0.591	56.54

Tabella 3.10 Confronto della *mesh* tra il riferimento ed i 4 casi del *cap method*.

	N° di celle totali	N° di celle catalitiche	Errore % Area catalitica	MAX valore di skewness	MAX non ortogonalità
<b>riferimento</b>	751569	45221	0.0085	0.791	82.55
<b>gap 1</b>	766981	40159	0.0095	0.742	74.21
<b>gap 2</b>	799856	39885	0.00714	0.720	70.98
<b>gap 3</b>	752212	36895	0.00841	0.611	64.54
<b>gap 4</b>	925246	37011	0.0125	0.622	64.12

Tabella 3.11 Confronto della *mesh* tra il riferimento ed i 4 casi del *gap method*.

	N° di celle totali	N° di celle catalitiche	Errore % Area catalitica	MAX valore di skewness	MAX non ortogonalità
<b>riferimento</b>	751569	45221	0.00850	0.791	82.55
<b>overlap 1</b>	666895	42232	0.0055	0.664	75.33
<b>overlap 2</b>	611252	41145	0.00978	0.619	68.03
<b>overlap 3</b>	556985	40025	0.01261	0.548	54.16
<b>overlap 4</b>	601256	39652	0.00992	0.601	58.88

Tabella 3.12 Confronto della *mesh* tra il riferimento ed i 4 casi dell' *overlap method*.

L'analisi delle caratteristiche delle *mesh* e dei tempi di calcolo richiesti per la loro generazione è importante per valutare il dispendio computazionale globale richiesto per il caso considerato.

### 3.2.3 Analisi dei risultati ottenuti

Le Figure dalla 3.15 alla 3.19 mostrano come le frazioni massive ed il campo di moto si sviluppano all'interno del reattore. Per semplicità è stato riportato un quarto di reattore per evidenziare i fenomeni coinvolti nel punto di contatto.

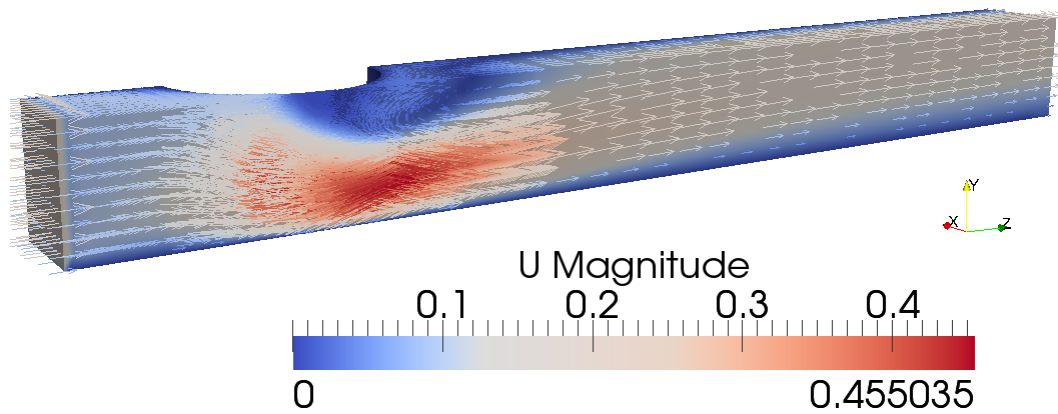


Figura 3.15 Sviluppo del campo di moto all'interno del reattore per il caso di riferimento.

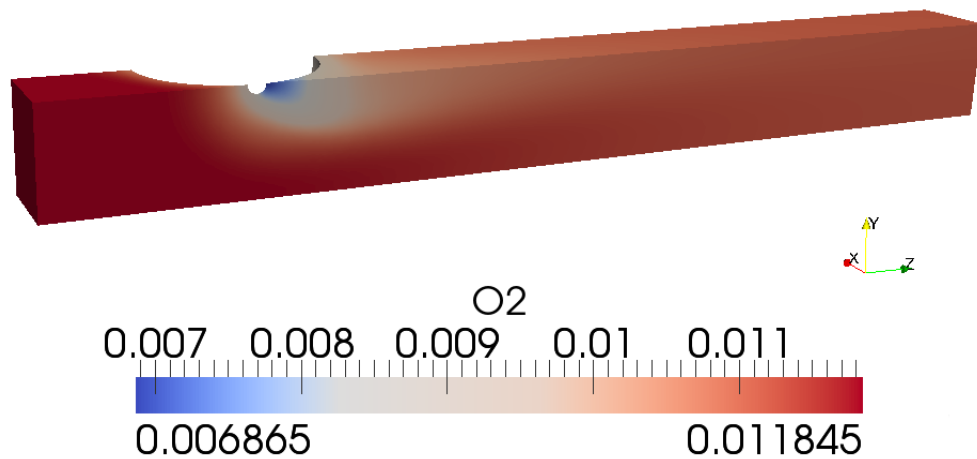


Figura 3.16 Frazione massiva di O<sub>2</sub> all'interno del reattore per il bridge method.

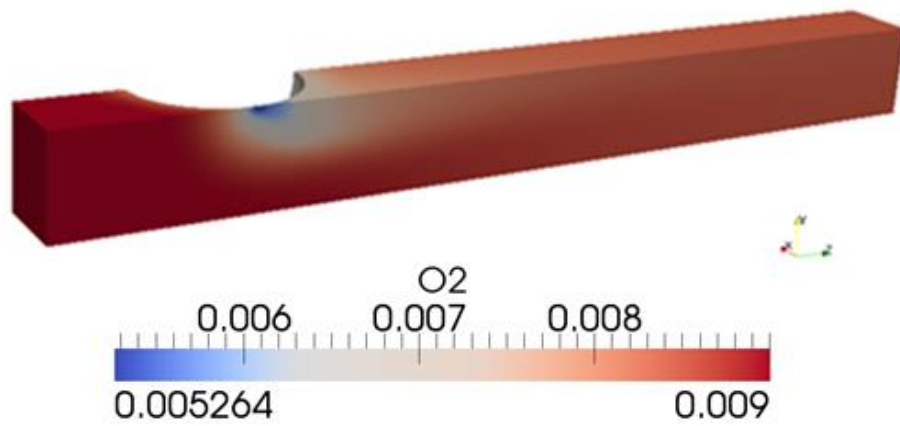


Figura 3.17 Frazione massiva di  $O_2$  all'interno del reattore per il cap method.

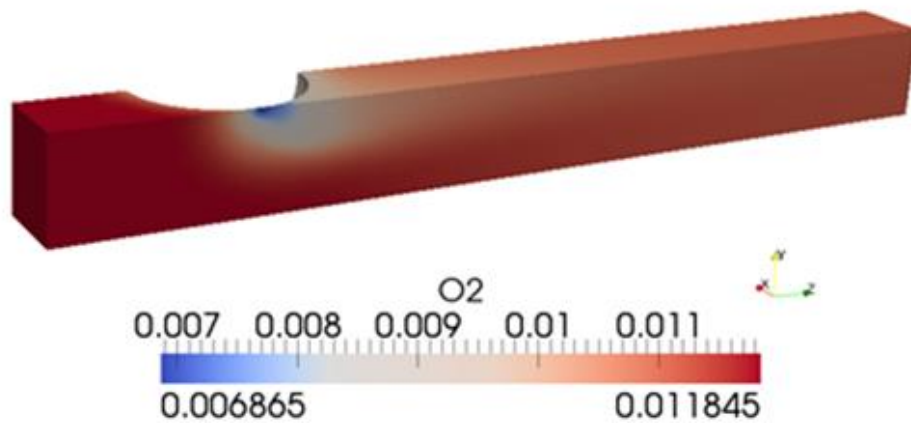


Figura 3.18 Frazione massiva di  $O_2$  all'interno del reattore per il gap method.

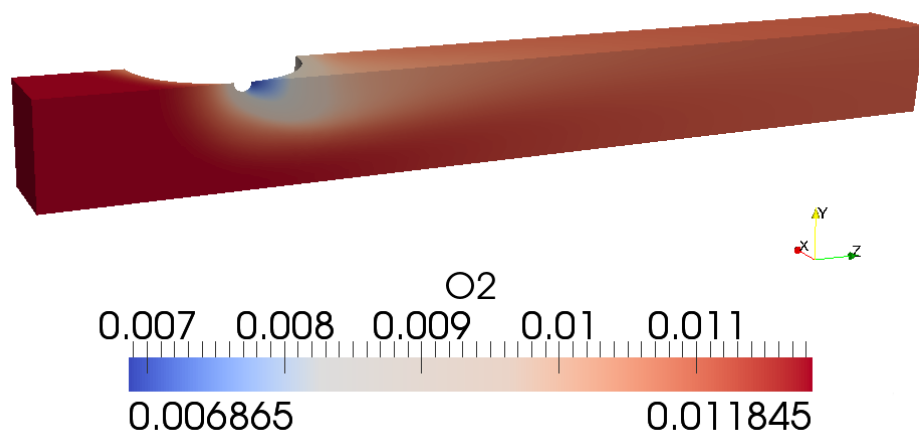


Figura 3.19 Frazione massiva di  $O_2$  all'interno del reattore per l'overlap method.

Una volta verificata l'indipendenza del dominio di calcolo dal numero di celle si è passati all'analisi vera e propria dei risultati ottenuti e al confronto dei 4 approcci utilizzati per modificare i punti di contatto.

Le Figure dalla 3.20 alla 3.25 riportano i valori di conversione della specie chimica limitante a tre diversi numeri di Damköhler. Il valore di conversione riportato sull'asse delle ordinate è quello del caso di riferimento. Infatti per i metodi di modifica locale sia il raggio del *bridge* che la dimensione del *cap* applicati sono nulli. Nei metodi di modifica globale invece il rapporto tra area modificata e quella teorica delle sfere è unitario. Confrontando i 4 metodi implementati con il caso di riferimento si nota che i metodi di modifica locale forniscono dei risultati più accurati rispetto a quelli globali.

➤ Valori di conversione di  $O_2$  per  $Da = 0.015$

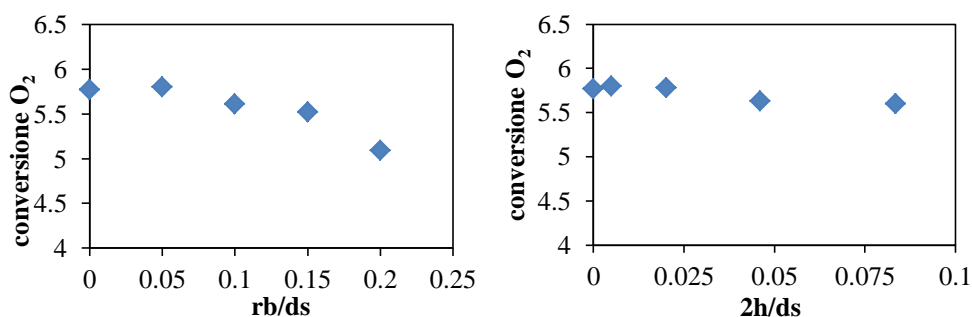


Figura 3.20 Conversione di  $O_2$  per (a) *bridge*, (b) *cap* per  $Da=0.015$ .

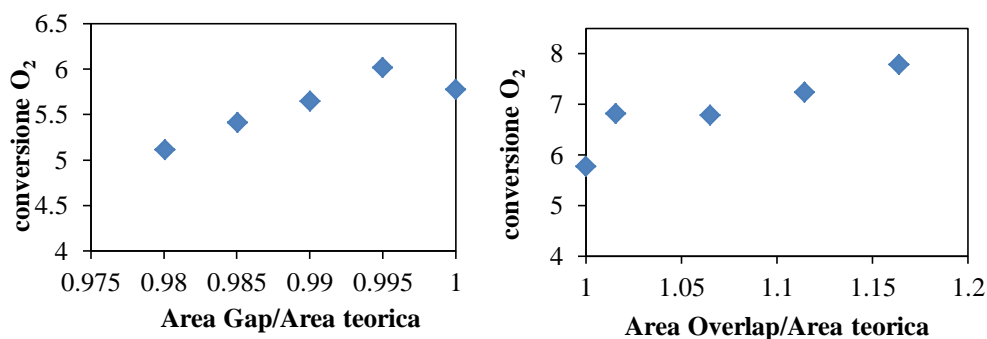


Figura 3.21 Conversione di  $O_2$  per (a) *gap*, (b) *overlap* per  $Da=0.015$ .



➤ Valori di conversione di  $O_2$  per  $Da = 1.5$

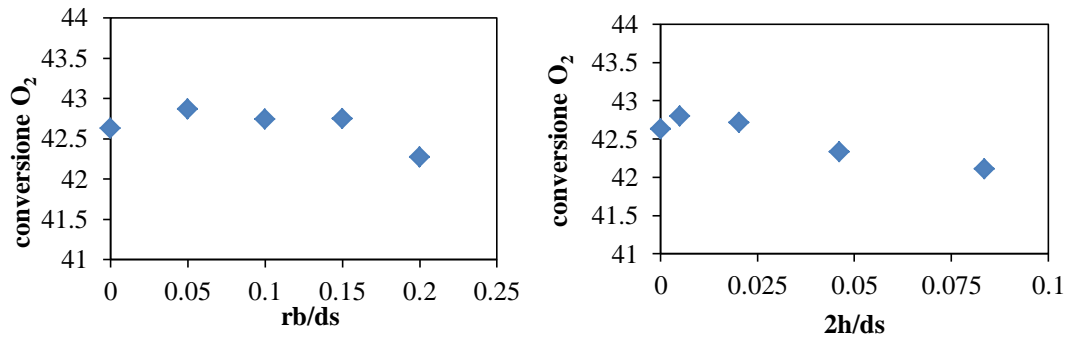


Figura 3.22 Conversione di  $O_2$  per (a) *bridge*, (b) *cap* per  $Da=1.5$ .

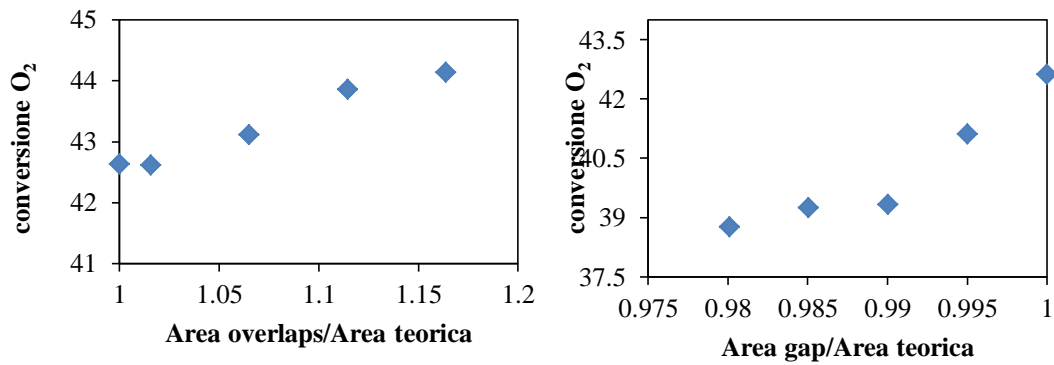


Figura 3.23 Conversione di  $O_2$  per (a) *gap*, (b) *overlap* per  $Da=1.5$ .

➤ Valori di conversione di  $O_2$  per  $Da = 150$

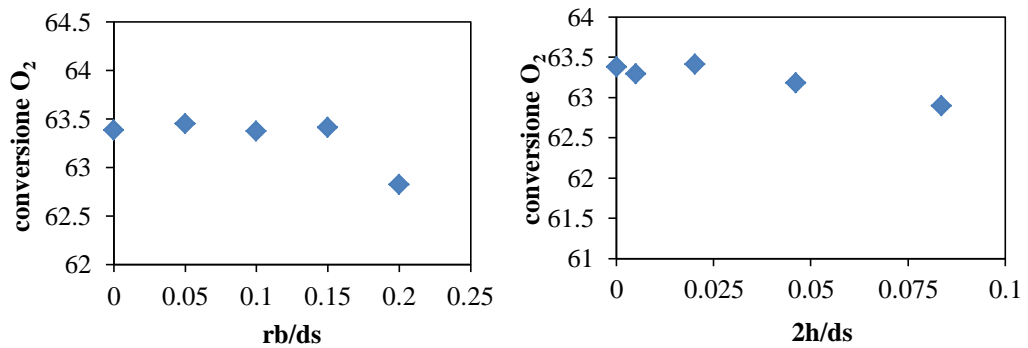


Figura 3.24 Conversione di  $O_2$  per (a) *bridge*, (b) *capper*  $Da=150$ .

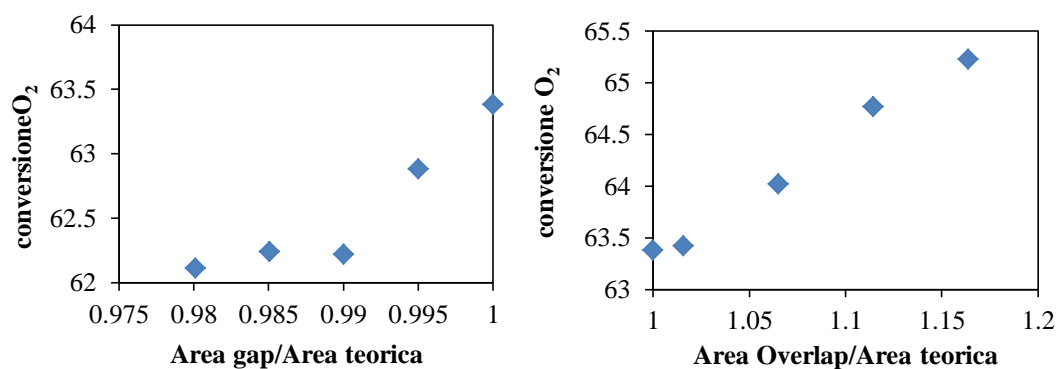


Figura 3.25 Conversione di O<sub>2</sub> per (a) *gap*, (b) *overlap* per Da=150.

Per numeri di Damköhler pari 0.015 i metodi di modifica locale hanno errori percentuali rispetto al caso di riferimento pari a quelli riportati in Tabella 3.13:

	Err % ( <i>bridge</i> ) conversione O <sub>2</sub>	Err % ( <i>cap</i> ) conversione O <sub>2</sub>	
<i>bridge 1</i>	0.519	0.519	<i>cap 1</i>
<i>bridge 2</i>	2.773	0.173	<i>cap 2</i>
<i>bridge 3</i>	4.333	2.426	<i>cap 3</i>

Tabella 3.13 Confronto degli errori percentuali tra i metodi di modifica locale.

Questa differenza tra i diversi metodi è dovuta alla differente modifica apportata alla superficie delle sfere e quindi alla superficie catalitica coinvolta nella reazione chimica superficiale. I metodi di modifica globale modificano maggiormente la superficie delle sfere rispetto ai metodi locali. Nelle Tabelle 3.14–3.17 si riportano le variazioni percentuali di area catalitica per i 4 approcci rispetto al caso di riferimento, pertanto più piccole sono le variazioni di area catalitica e più soddisfacenti sono i risultati.

	superficie catalitica [cm <sup>2</sup> ]	variazione % Area catalitica
<b>riferimento</b>	20.268	0
<b>bridge 1</b>	20.177	0.451
<b>bridge 2</b>	19.941	1.616
<b>bridge 3</b>	19.615	3.224
<b>bridge 4</b>	19.253	5.009

Tabella 3.14 Variazione percentuale di area catalitica per il *bridge method*.

	superficie catalitica [cm <sup>2</sup> ]	variazione % Area catalitica
<b>referimento</b>	20.268	0
<b>cap 1</b>	20.268	0.000986
<b>cap 2</b>	20.264	0.020228
<b>cap 3</b>	20.247	0.106078
<b>cap 4</b>	20.198	0.348330

Tabella 3.15. Variazione percentuale di area catalitica per il *cap method*.

	superficie catalitica [cm <sup>2</sup> ]	variazione % Area catalitica
<b>referimento</b>	20.268	0
<b>gap 1</b>	20.167	0.498
<b>gap 2</b>	20.066	0.996
<b>gap 3</b>	19.965	1.493
<b>gap 4</b>	19.865	1.989

Tabella 3.16 Variazione percentuale di area catalitica per il *gap method*.

	superficie catalitica [cm <sup>2</sup> ]	variazione % Area catalitica
<b>referimento</b>	20.268	0
<b>overlap 1</b>	20.587	1.574
<b>overlap 2</b>	21.587	6.509
<b>overlap 3</b>	22.587	11.443
<b>overlap 4</b>	23.587	16.376

Tabella 3.17 Variazione percentuale di area catalitica per l'*overlap method*.

La minima variazione di superficie delle sfere è data dal *cap*, seguita dal *bridge* ed infine i metodi di modifica globale. In particolare tutti i metodi comportano una diminuzione di superficie catalitica tranne l'*overlap* che, a causa dell'espansione delle sfere, si ha un aumento di superficie catalitica. Questo trend viene seguito anche dalla conversione, infatti tutti i metodi di modifica dei punti di contatto comportano una diminuzione di conversione, tranne per l'*overlap* che all'aumentare del diametro delle sfere l'area catalitica aumenta e quindi anche la conversione di ossigeno. Per interpretare meglio i risultati, è importante indagare anche il campo di

moto vicino il punto di contatto per comprendere al meglio i fenomeni di scambio di materia che si instaurano. Infatti ciascun metodo potrebbe offrire situazioni fluidodinamiche differenti.

Dalle Figure 3.26–3.28 si confrontano i campi di moto locali per i 4 metodi utilizzati.

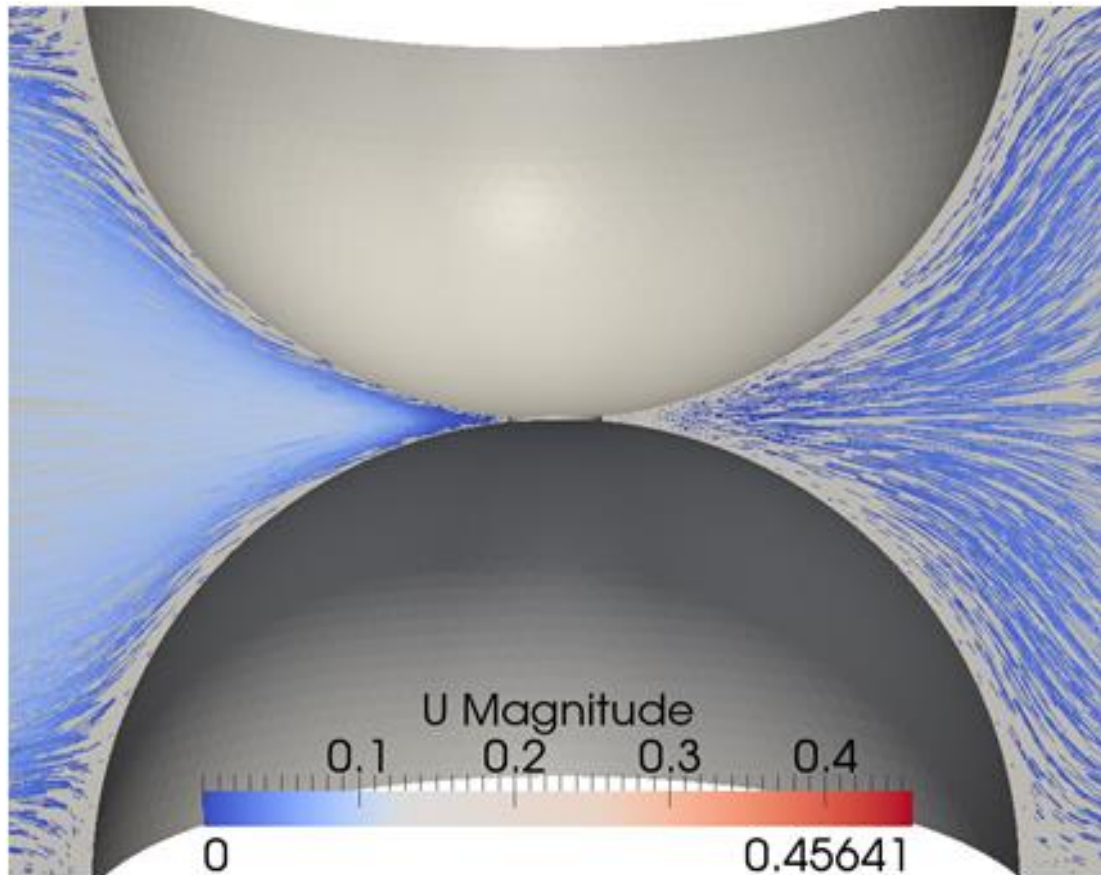


Figura 3.26 Campo di moto attorno al punto di contatto per il *bridge method*.

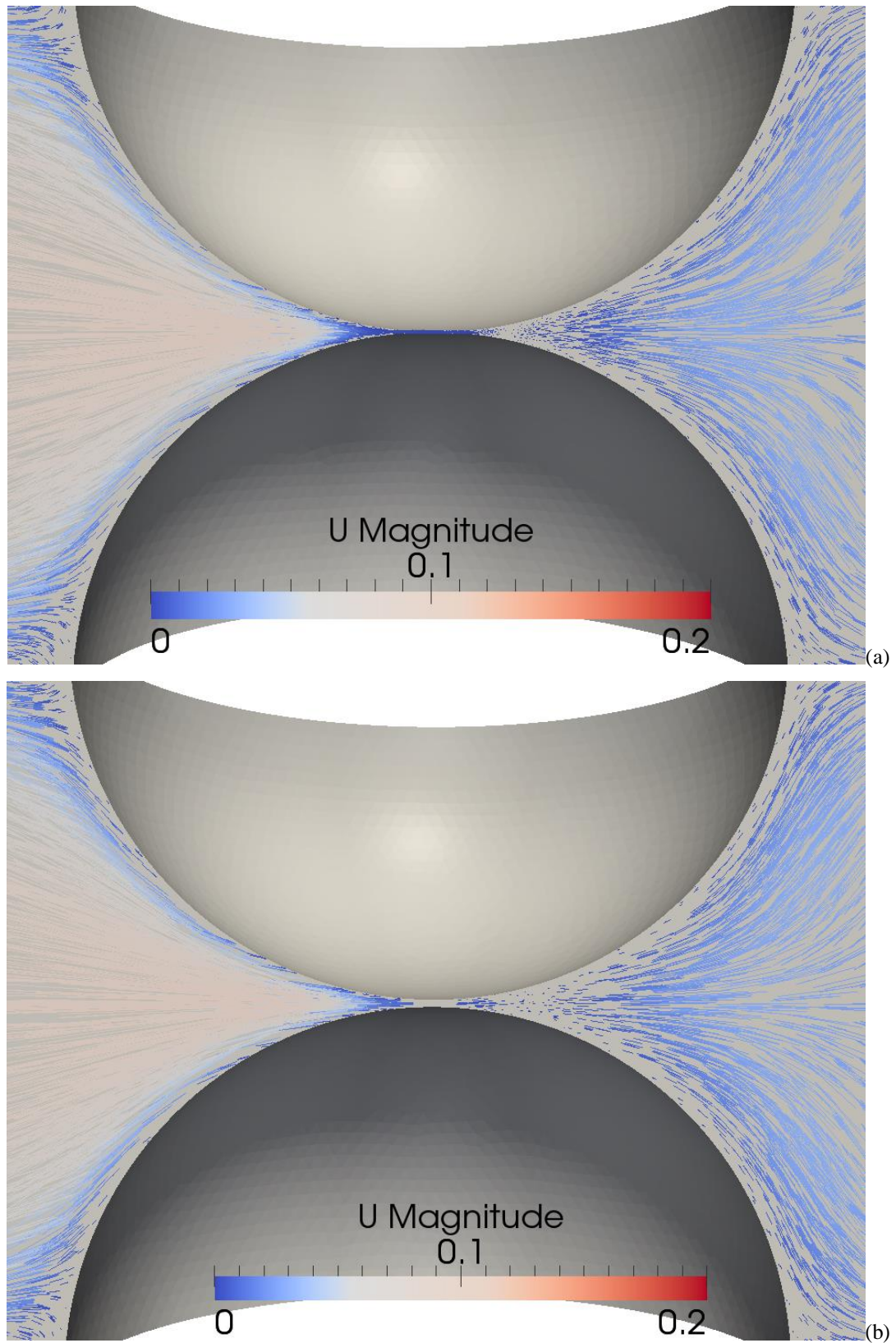
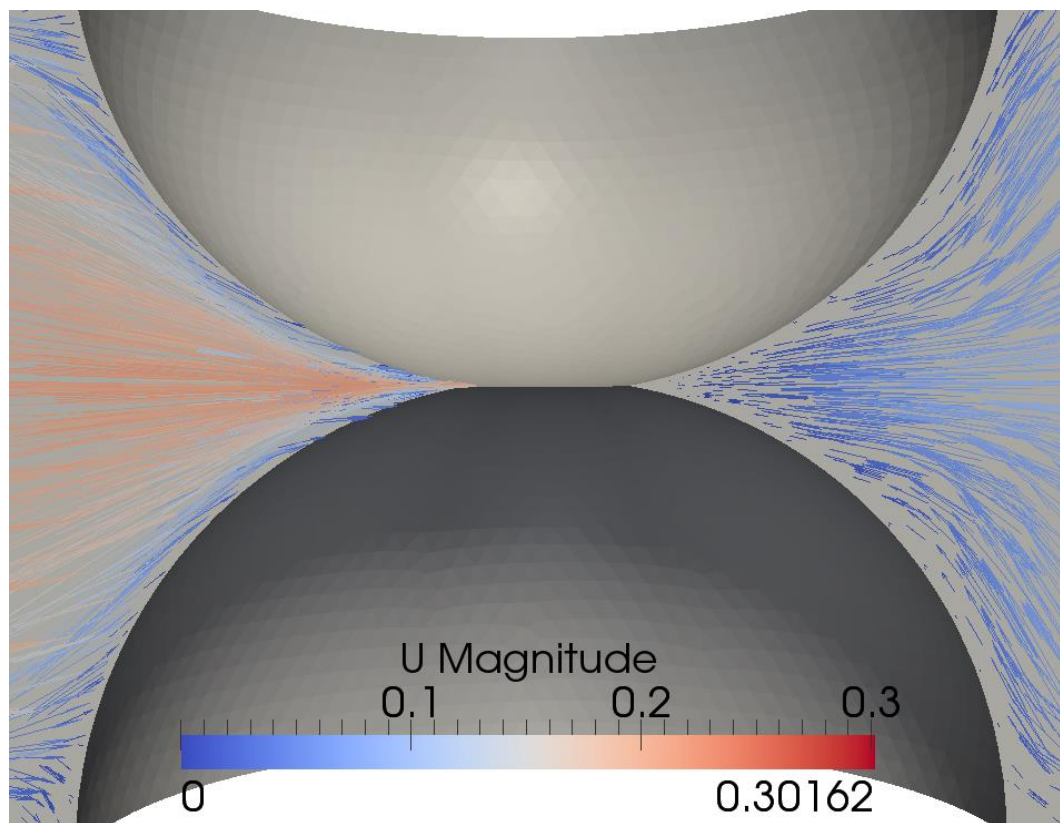


Figura 3.27 Campo di moto attorno al punto di contatto *cap* (a), *gap* (b).



**Figura 3.28** Campo di moto attorno al punto di contatto per *l'overlap method*.

Il punto di contatto è coinvolto dai fenomeni di scambio di materia. Nella Figura 3.29 si evidenzia la presenza dei vortici laminari che consente un rimescolamento delle specie chimiche e quindi un elevato tempo di contatto delle specie in quelle zone. La ragione a questo fenomeno è dovuta sostanzialmente alla configurazione geometrica proposta, infatti, il flusso convettivo diretto sulle sfere consente la generazione di zone di ristagno a valle le sfere garantendo così il trasferimento delle specie chimiche dalla fase fluida sulla superficie catalitica. Il fattore che gioca un ruolo importante in questo processo è l'area catalitica disponibile per la reazione chimica catalitica superficiale.

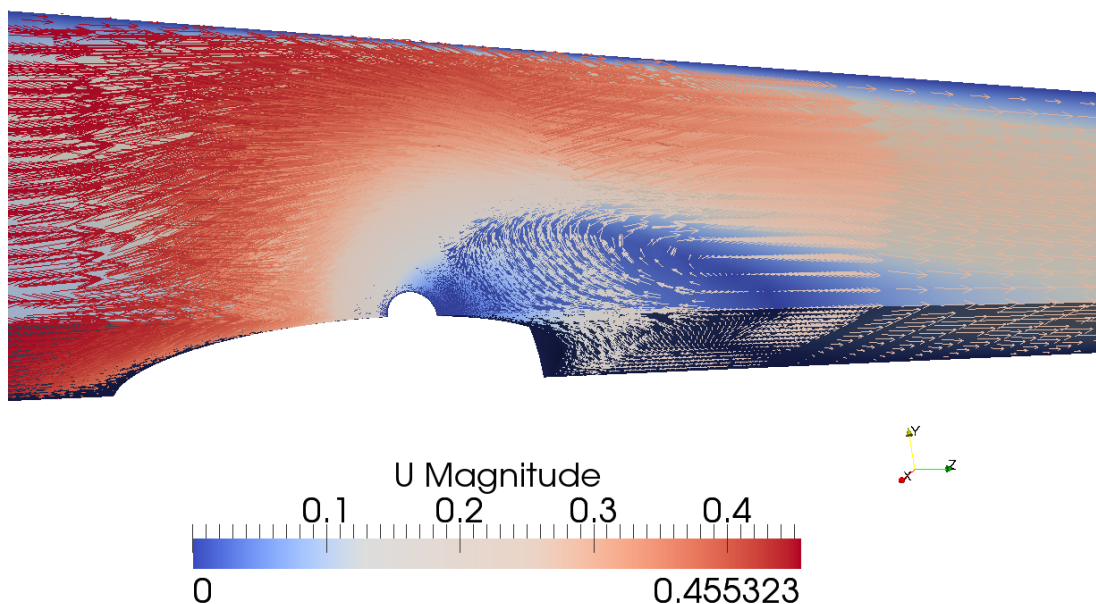


Figura 3.29 Particolare del campo di moto attorno al punto di contatto.

Dalle Figure è possibile osservare come il punto di contatto sia pienamente coinvolto dal flusso convettivo che investe le due sfere, poiché il flusso è diretto perpendicolarmente ai due diametri delle semisfere passanti per il punto di contatto. Elghobashi e Sirignano [1] hanno investigato il campo di moto a valle di due sfere poste ad una certa distanza tra loro, dimostrando che, quando la distanza tra le sfere è molto piccola, a limite pari a zero, le scie che si sviluppano a valle delle sfere sono simmetriche rispetto al piano passante per il punto di contatto. Le sfere possono essere considerate così come un unico corpo solido attraversato dal fluido in ingresso al reattore. Dietro le sfere si sviluppano inoltre delle zone di ristagno con la presenza di vortici laminari che consentono il rimescolamento delle specie chimiche e quindi garantiscono maggiori tempi di contatto rispetto alla zona a monte le sfere. Quest'effetto è ben visibile nelle Figure 3.26–3.28 prima descritte, dove si riportano i campi di moto locale per i diversi approcci implementati. Le Figure 3.30–3.33 mostrano le frazioni massive delle specie chimiche che si sviluppano sulla superficie catalitica delle sfere. Valutando la variazione percentuale di area catalitica si nota che i metodi di modifica locale offrono una minima variazione di area catalitica rispetto a quelli globali. Questo perché sia il *cap* che il *bridge* comportano una modifica solo nel punto di contatto. Confrontando inoltre il *cap* con il *bridge* si nota che la

sostituzione di un bridge con il punto di contatto comporta una maggiore perdita di area catalitica rispetto all'applicazione del *cap* tra le sfere.

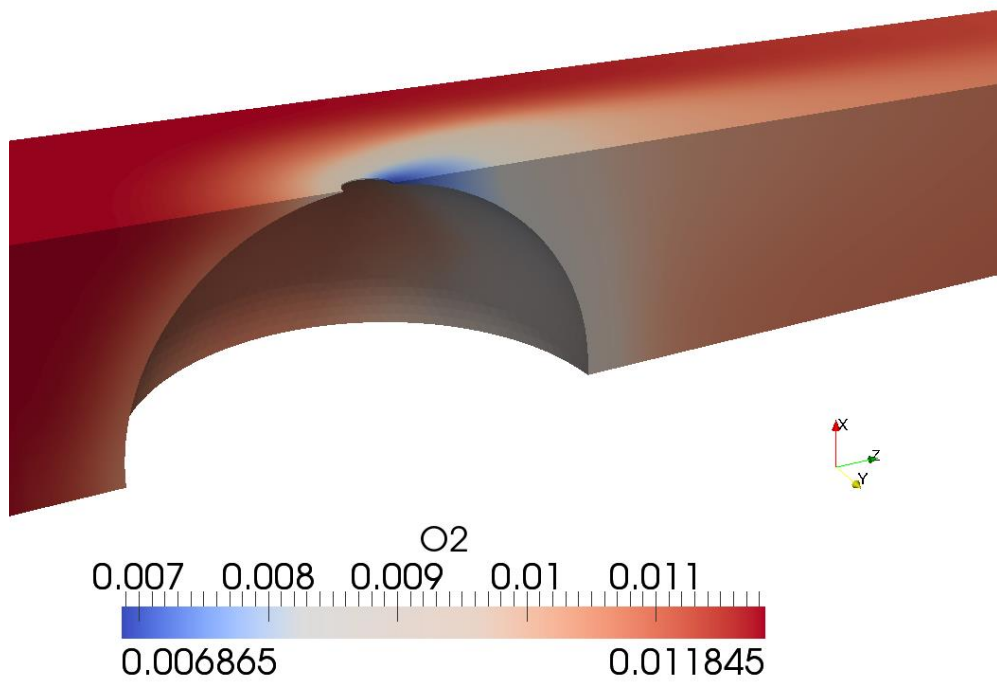


Figura 3.30 Frazione massiva di O<sub>2</sub> vicino il punto di contatto per il *bridge method*.

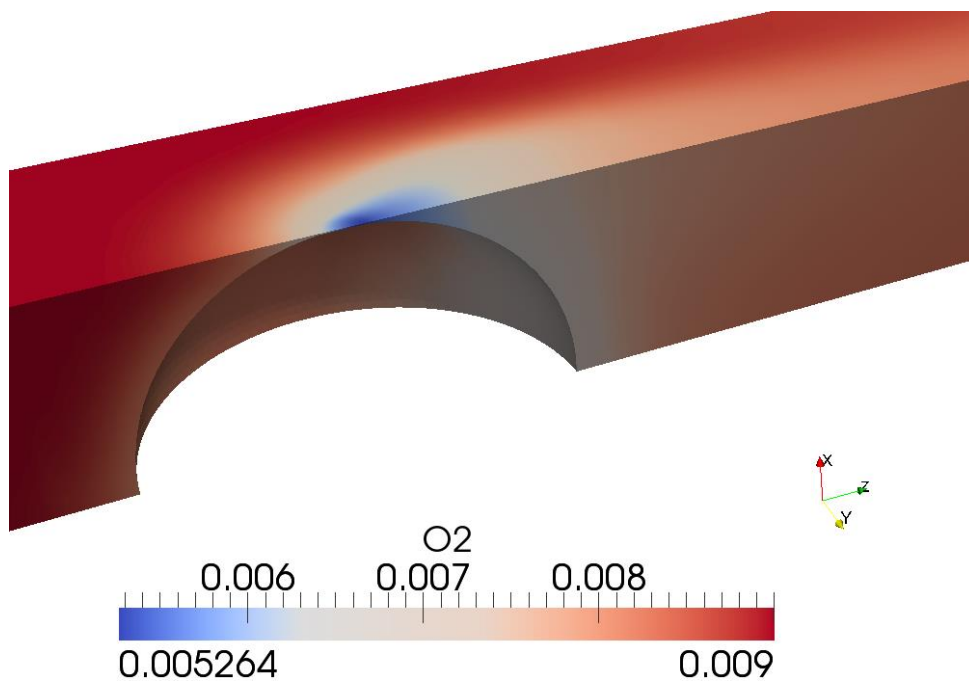


Figura 3.32 Frazione massiva di O<sub>2</sub> vicino il punto di contatto per il *cap method*.



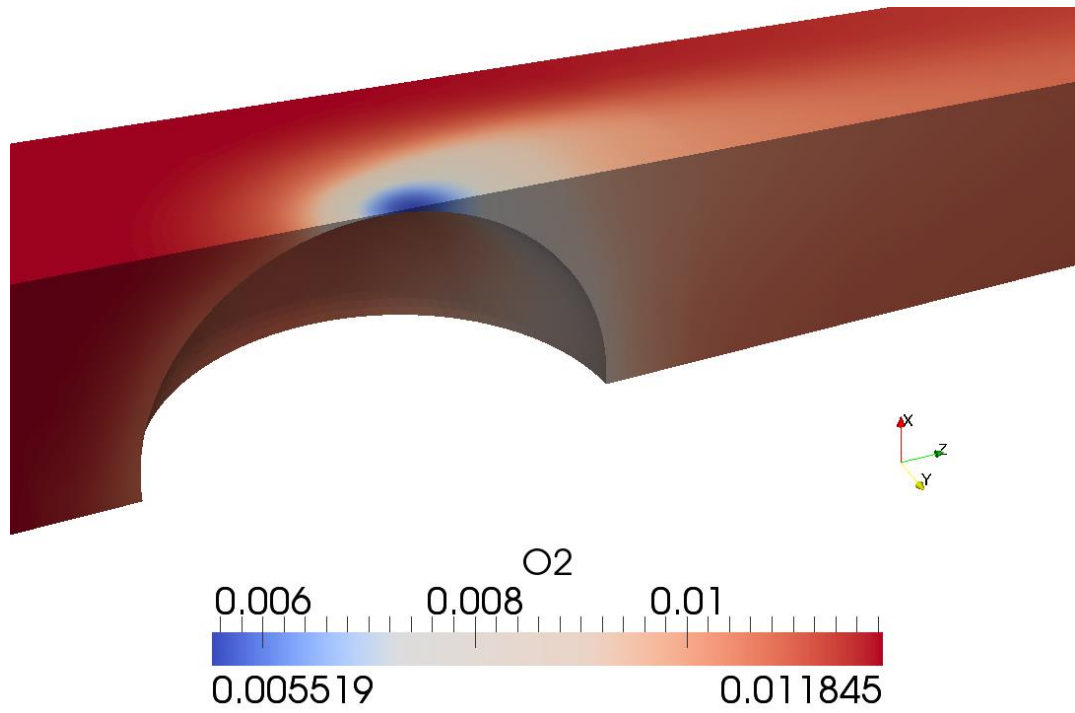


Figura 3.31 Frazione massiva di O<sub>2</sub> vicino il punto di contatto per il *gap method*.

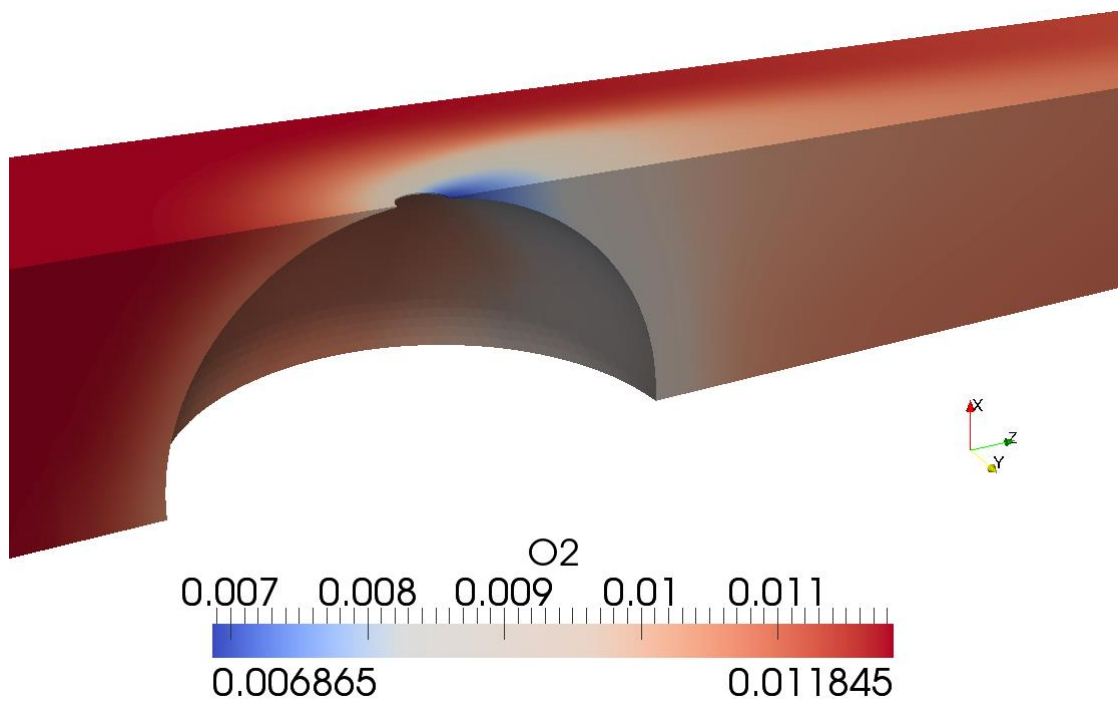


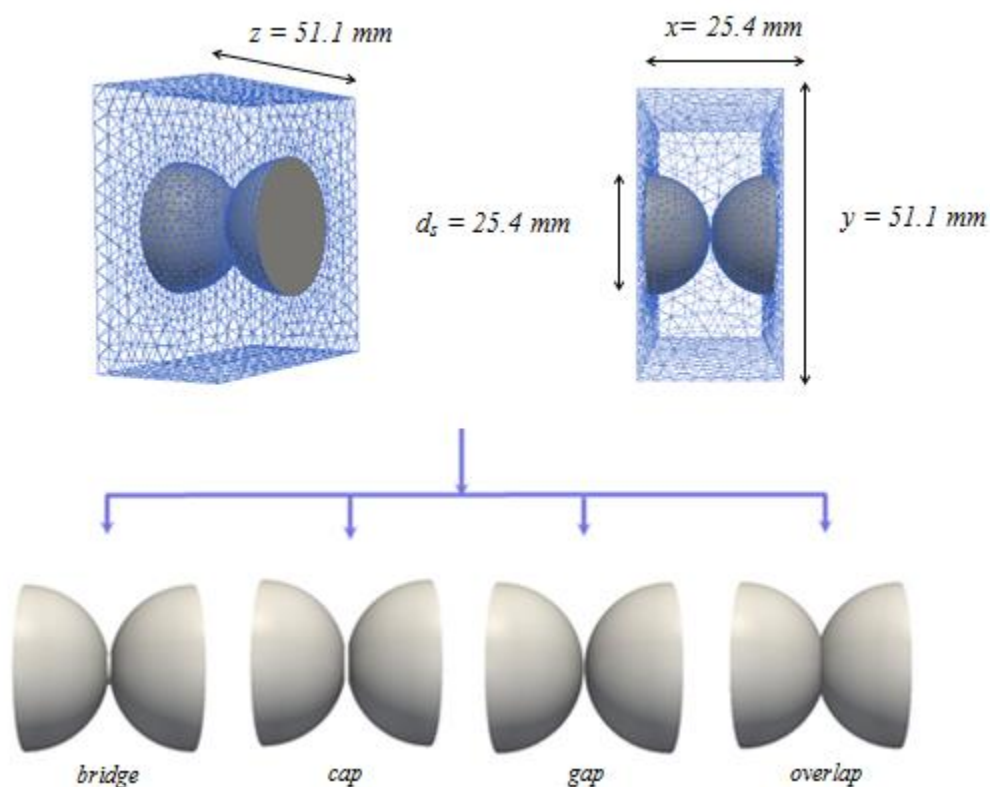
Figura 3.33 Frazione massiva di O<sub>2</sub> vicino il punto di contatto per l'*overlap method*.

### 3.3 Studio dello scambio termico

In questo paragrafo sono stati descritti gli effetti delle modifiche dei punti di contatto tra due sfere sui fenomeni di scambio termico. Per studiare l'effetto della singola modifica sulle grandezze di interesse, l'analisi è stata condotta in maniera sistematica studiando altri 4 sotto casi per ciascun approccio implementato. I 4 metodi sono stati tra loro confrontati valutando il flusso termico uscente dalla regione solida che si trova ad una temperatura più alta e analizzando i profili di temperatura che si sviluppano tra le due sfere.

#### 3.3.1 Generazione della geometria e della mesh

Le dimensioni del box che contiene le due semisfere sono riportate in Figura 3.34.

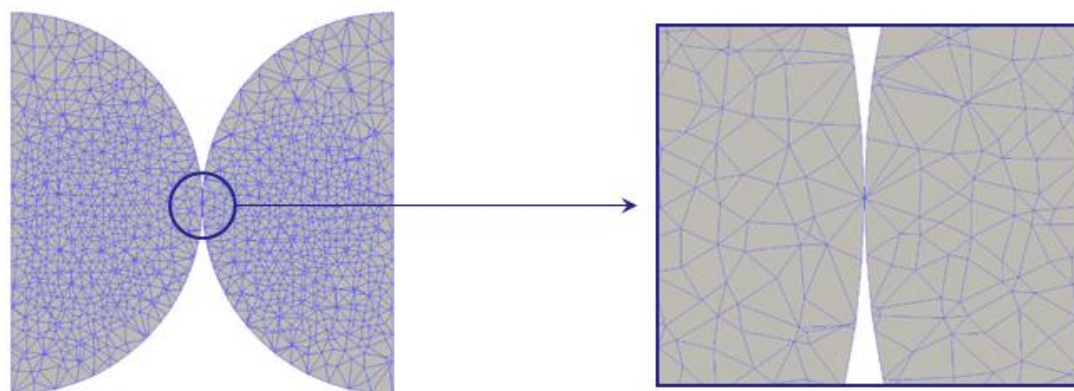


**Figura 3.34** Schematizzazione del dominio geometrico e dei metodi implementati.

Il caso di riferimento ha la particolarità di introdurre il punto di contatto come uno dei nodi della mesh tetraedrica. Come si può vedere nella Figura 3.35, il nodo nel punto di contatto viene condiviso da entrambe le semisfere. La procedura per creare

le 4 modifiche dei punti di contatto è identica a quella vista per lo studio della reattività chimica (Paragrafo 3.2.1).

Per distinguere le diverse regioni create (sfere, bridge e fluido) l'intera geometria è stata ripartita in solidi, successivamente con l'opzione "*gruppi di Volume*" le diverse regioni solide di interesse sono state nominate come mostrato in Figura 3.36. Considerata la ridotta regione fluida si è ritenuto necessario creare la *mesh* totalmente non strutturata utilizzando l'algoritmo NETGEN-1D2D3D di SALOME. Una volta generata la mesh, questa è stata esportata in formato UNV e poi convertita in ambiente OpenFOAM.



**Figura 3.35** Particolare del punto di contatto utilizzato come nodo della mesh tetraedrica per il caso di riferimento.

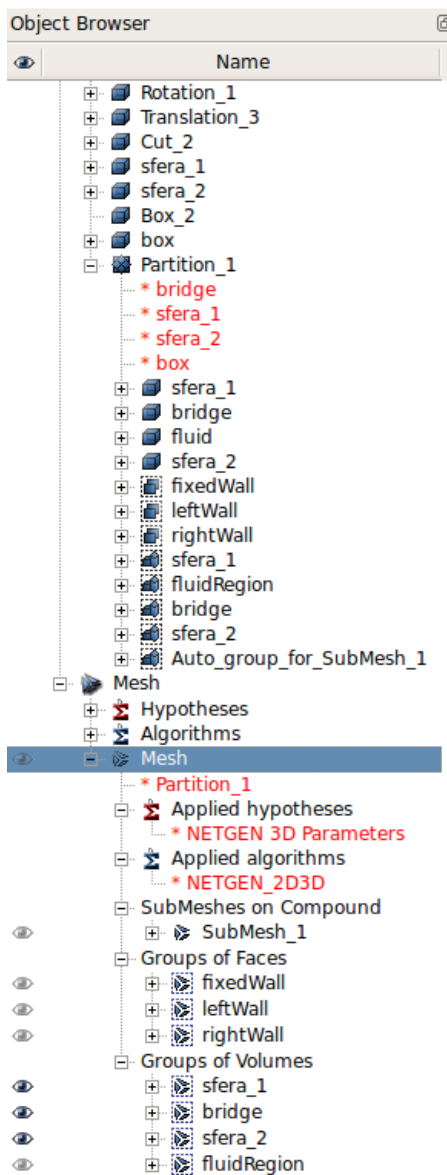


Figura 3.36 Scheda di lavoro in ambiente SALOME per creare un caso multiregione.

Le modifiche locali e globali effettuate alle sfere sono riportate nelle Tabelle 3.18–3.21.

	$r_b/d_s$	$r_b$ [mm]	$h_b$ [mm]
<i>bridge 1</i>	0.05	1.27	0.127
<i>bridge 2</i>	0.10	2.54	0.513
<i>bridge 3</i>	0.15	3.81	1.171
<i>bridge 4</i>	0.20	5.08	2.121

Tabella 3.18 Dimensioni dei *bridges* utilizzati.

	$l_b$ [mm]	$h$ [mm]
<i>cap 1</i>	0,127	0,0635
<i>cap 2</i>	0,513	0,2565
<i>cap 3</i>	1.17	0,585
<i>cap 4</i>	2.121	1.0605

Tabella 3.19 Dimensioni dei caps utilizzati.

	diametro gap [mm]
<i>gap 1</i>	12.668
<i>gap 2</i>	12.636
<i>gap 3</i>	12.604
<i>gap 4</i>	12.573

Tabella 3.20 Dimensioni delle sfere per creare i 4 casi del gap.

	diametro overlap [mm]
<i>Overlap 1</i>	12.900
<i>Overlap 2</i>	13.527
<i>Overlap 3</i>	14.153
<i>Overlap 4</i>	14.780

Tabella 3.21 Dimensioni delle sfere per creare i 4 casi di overlap.

Tutti i casi sono stati studiati nelle stesse condizioni operative. In Figura 3.37 si evidenzia come le due pareti esterne delle semisfere siano state fissate a due temperature differenti per garantire un flusso termico tra le sfere.

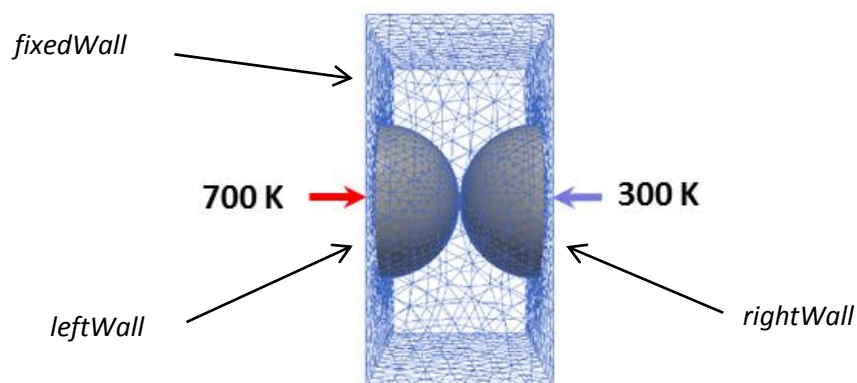


Figura 3.37 Modellazione del caso studio.

Le condizioni iniziali utilizzate sono state a valori di pressione di 101325 Pa, temperatura omogenea in tutto il sistema pari a 500 K e velocità del fluido posta pari a zero all'interno della regione fluida. Le condizioni al contorno (*boundary conditions*) sono state fissate come segue in Tabella 3.22 [23].

	Temperatura	Velocità	Pressione	Pressione statica
<b>leftWall</b>	fixedValue	fixedValue	zeroGradient	zeroGradient
<b>fixedWall</b>	zeroGradient	fixedValue	zeroGradient	zeroGradient
<b>rightWall</b>	fixedValue	fixedValue	zeroGradient	zeroGradient
<b>fluidRegion to solidRegion</b>	(*)	fixedValue	zeroGradient	fixedFluxPressure

**Tabella 3.22** Condizioni al contorno assegnate a ciascuna patch nel box e nelle diverse regioni.

Oltre alle grandezze fisiche viste nello studio della reattività chimica, il solver `chtMultiRegionSimpleFoam` considera anche un'altra grandezza che il termine `p_rgh`. Questo termine è la pressione dinamica, definita come:

$$p_{rgh} = p - g \cdot h \cdot \rho \quad (3.7)$$

che è il valore della pressione totale meno il contributo della pressione statica. La condizione al contorno fissata è quella di gradiente nullo (*condizione di Neumann*) sulle pareti solide, e di flusso costante (`fixedFluxPressure`) sulle regioni solide. Quest'ultima condizione è conosciuta perché garantisce una migliore convergenza della soluzione. Affinché il problema sia ben posto le condizioni all'interfaccia tra le diverse regioni devono essere tra loro congruenti. La condizione al contorno utilizzata per la temperatura è `compressible::turbulentTemperatureCoupledBaffleMixed` (\*) [21]. Questa condizione viene utilizzata quando esiste scambio termico tra due regioni accoppiate tra loro e non tiene conto del calore scambiato per irraggiamento, ma solo quello scambiato per conduzione e convezione. Si nota come per garantire

uno studio con “*fluido stazionario*” per la velocità è stata imposta la condizione di adesione sulle pareti del box e delle sfere.

Il *bridge method* prevede la sostituzione del punto di contatto sfera/sfera con un bridge e la valutazione della sua conducibilità termica è stata fatta considerando due modelli. Il primo modello è quello elaborato da Dixon *et al.* [4]. L’idea di base di questo modello è quella di determinare un conducibilità termica fittizia che tenga conto sia delle proprietà termiche del fluido che delle quelle del solido. Pertanto facendo un bilancio dei flussi termici su una generica porzione del *bridge* che considera flussi entranti dalla sfera ed uscenti dal *bridge* si ottiene:

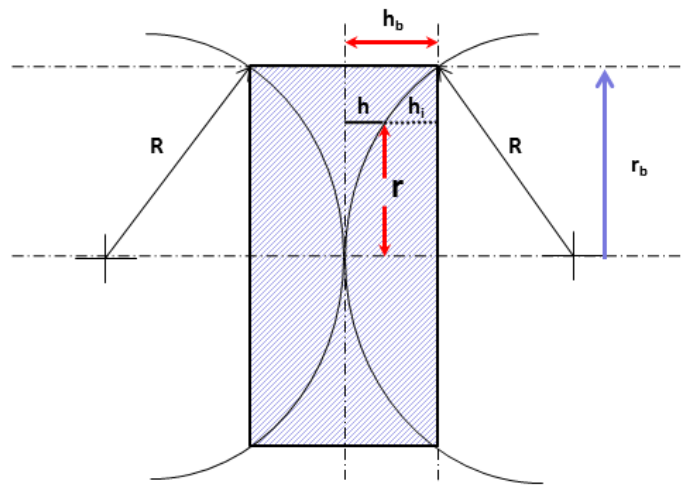


Figura 3.38 Modellazione numerica del *bridge* secondo il modello Dixon *et al.* [2].

$$h_b = h + h_i \quad (3.8)$$

$$k_{eff} = \frac{2h_b}{r_b^2} \int_0^{r_b} \frac{k_s k_{fr}}{h_i k_{fr} + h k_s} r dr \quad (3.9)$$

$$Q_w = - \frac{k_{eff}}{h_b} (T_1 - T_2) (\pi r_b^2) \quad (3.10)$$

dove  $h_b$  è la metà della lunghezza del *bridge*. La conducibilità termica del fluido viene calcolata con la seguente relazione, che prende il nome di conducibilità termica ridotta a causa del “filetto di fluido” tra le due sfere:

$$k_{fr} = \frac{k_f}{1 + (2\Lambda/h)(2 - \gamma)/(\gamma)} \quad (3.11)$$

Questa relazione tiene conto del fatto che al diminuire del “filetto di fluido” tra le due sfere il valore della conducibilità termica del fluido tende a zero, effetto noto come “effetto Smoluchowski” [25]; il coefficiente  $\gamma$  si chiama “accomodation coefficient” [26];  $\Lambda$  è il cammino libero medio delle molecole del gas. Se l’effetto di Smoluchowski si può ritenere trascurabile, allora la conducibilità termica del fluido si può ritenere costante e l’integrale per il calcolo della conducibilità effettiva può essere risolto analiticamente. In questo lavoro sono stati utilizzati dei valori di  $\Lambda = 0.07 \cdot 10^{-6}$  m,  $\gamma = 0.1$ , la conducibilità termica delle sfere è stata considerata pari 1 W/(m · K).

Proprietà del fluido			
$\rho_f$ [kg/m <sup>3</sup> ]	$C_{p_f}$ [J/(kg · K)]	$k_f$ [W/(m · K)]	$\mu$ [Pa · s]
1.225	1006.43	0.0242	$1.7894 \cdot 10^{-5}$

Tabella 3.23 Proprietà fisiche del fluido.

Proprietà delle sfere		
$\rho_s$ [kg/m <sup>3</sup> ]	$C_{p_s}$ [J/(kg · K)]	$k_s$ [W/(m · K)]
1947	1000	1

Tabella 3.24 Proprietà fisiche delle sfere.

Proprietà dei bridges		
$\rho_s$ [kg/m <sup>3</sup> ]	$C_{p_s}$ [J/(kg · K)]	$k$ [W/(m · K)]
1947	1000	$k_{eff}$

Tabella 3.25 Proprietà fisiche dei bridges utilizzati.

Nella Tabella 3.26 si riportano i valori di conducibilità effettiva dei *bridges* a diversi rapporti  $r_b/d_s$ .



$r_b/d_s$	$k_{\text{eff}}$ [W/(m · K)]
0.05	0.0685
0.1	0.0841
0.15	0.0892
0.2	0.0919

Tabella 3.26 Valori delle conducibilità termiche effettive dei *bridges*.

Oltre a questo modello è stato implementato un modello più semplice dove le conducibilità termiche dei *bridges* e delle sfere sono state assunte uguali cioè pari a 1 W/(K · m).

### 3.3.2 Analisi della convergenza della mesh

Per avere dei risultati accurati, occorre avere una soluzione che non dipenda dalla discretizzazione del dominio. Per verificare ciò è stato necessario fare delle prove di sensitività della soluzione dalla mesh in modo da avere un numero tale di celle che garantiscano una soluzione indipendente dal tipo di discretizzazione utilizzata e in modo da ottimizzare i tempi di calcolo.

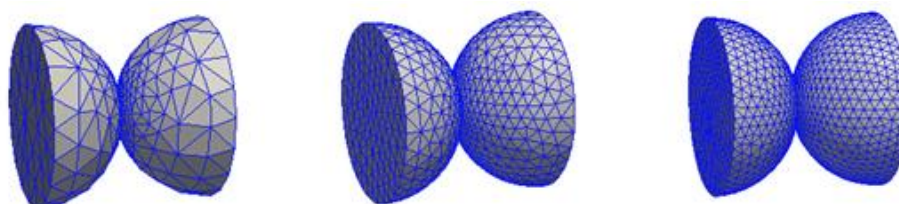


Figura 3.39 Mesh di sfere a numero di celle crescente.

La convergenza della mesh è stata fatta monitorando il flusso termico entrante (o uscente) dalle regioni solide. La convergenza è stata così raggiunta quando all'aumentare del numero di celle il flusso termico dei solidi raggiunge un plateau. Infine per interpolare i risultati ottenuti con una mesh con un dato numero di celle a mesh con numero di celle superiore è stato utilizzato il comando base di OpenFOAM `mapFields`.

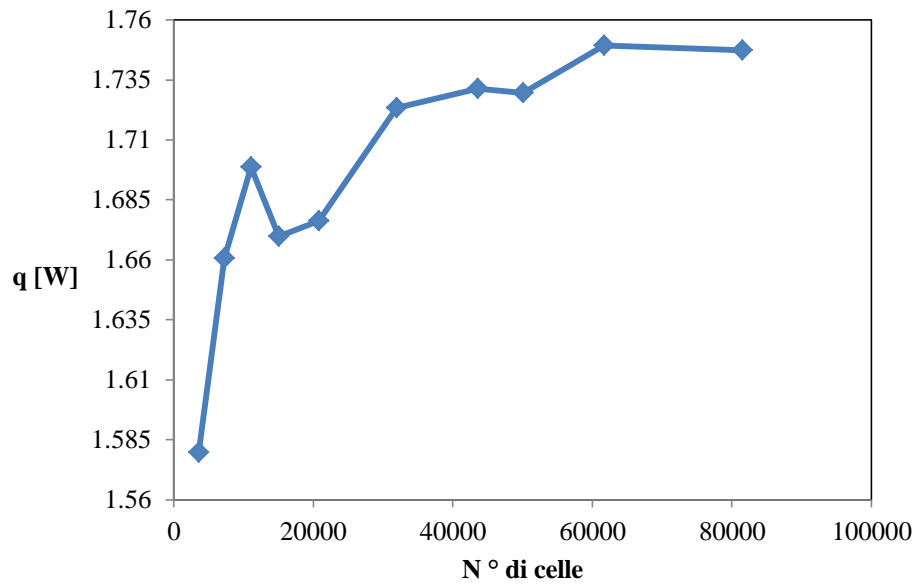


Tabella 3.40 Valori del flusso termico rispetto al N° di celle per il caso di riferimento.

### 3.3.3 Analisi dei risultati ottenuti

Per confrontare gli effetti delle 4 modifiche utilizzate sui fenomeni di scambio termico tra due sfere sono stati riportati i profili di temperatura sviluppatasi all'interno delle sfere, come mostrato in Figura 3.41 ed anche i flussi termici uscenti dalla sfera più calda ed entranti nella semisfera più fredda.

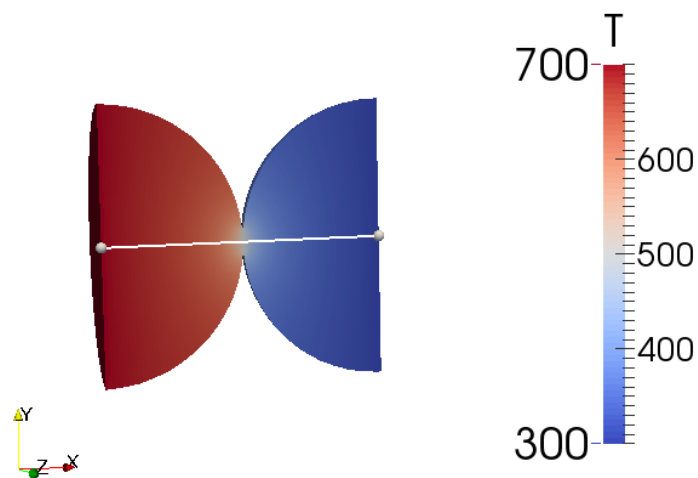


Figura 3.41 Distribuzione della temperatura lungo la coordinata x passante per i centri delle semisfere.

Nelle Figure 3.42–3.46 si riportano i profili di temperatura all'interno delle sfere per i 4 metodi.

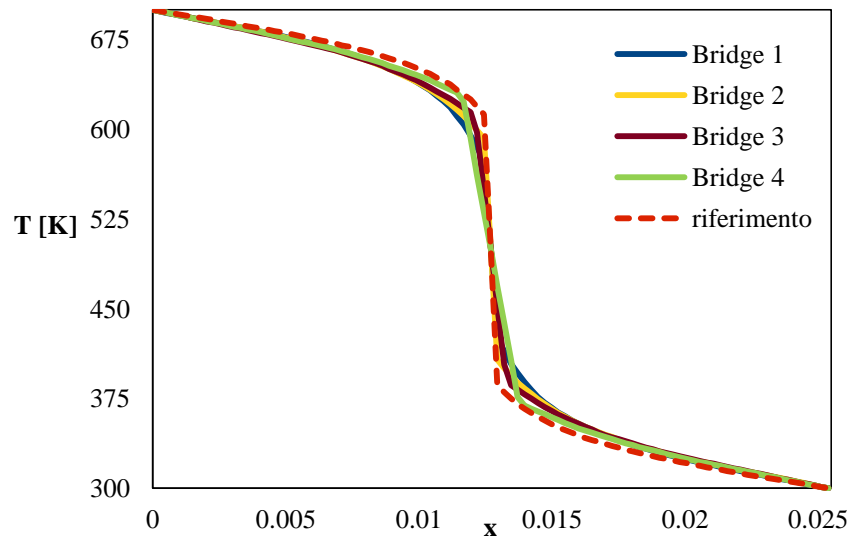


Figura 3.42 Profilo di temperatura secondo il modello Dixon *et al.* [4].

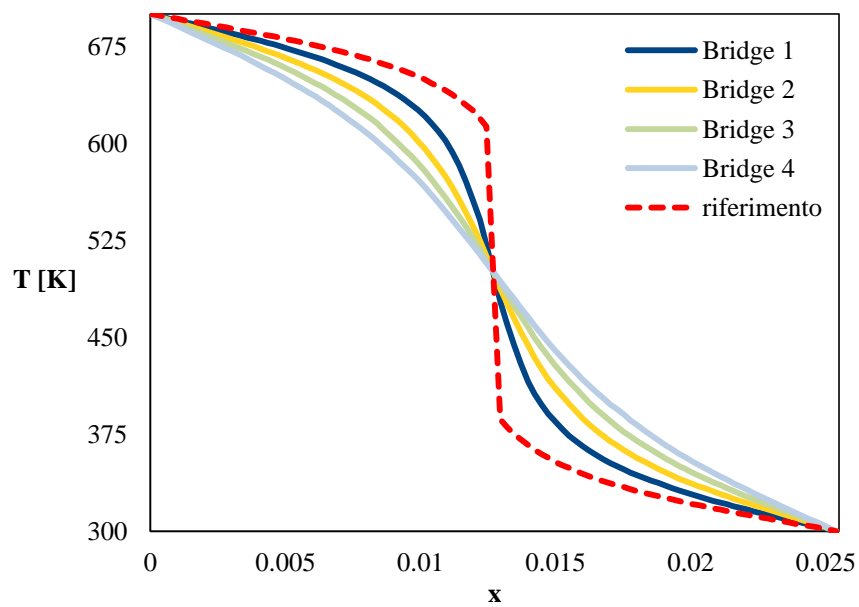


Figura 3.43 Profilo di temperatura secondo il modello *bridge* semplificato.

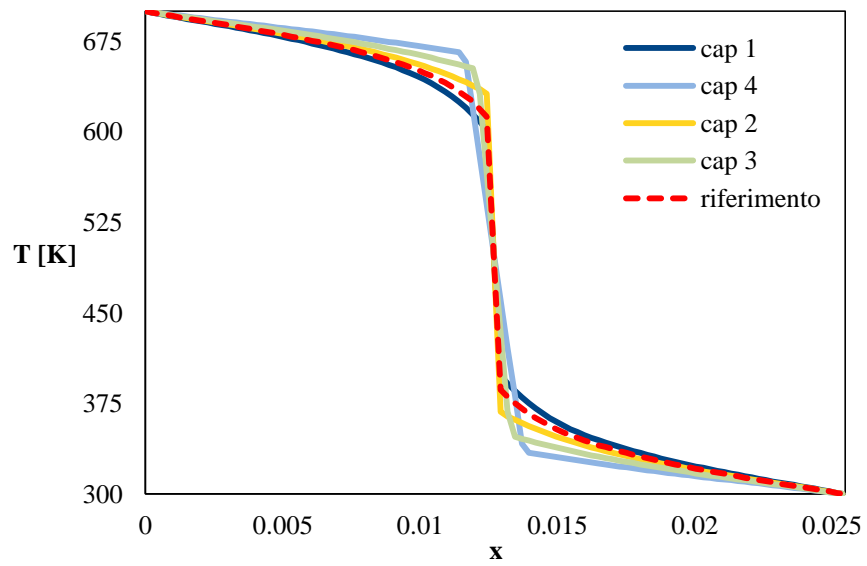


Figura 3.44 Profilo di temperatura secondo il modello *cap*.

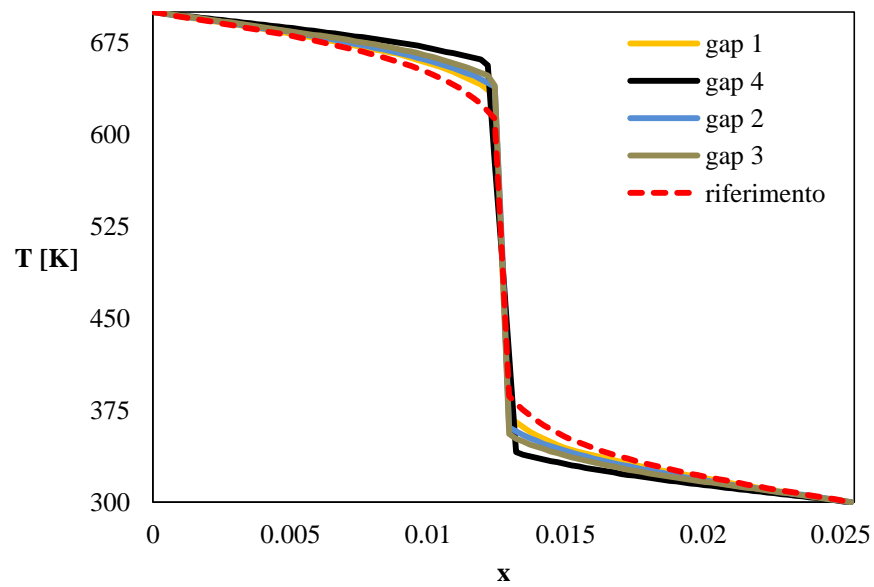
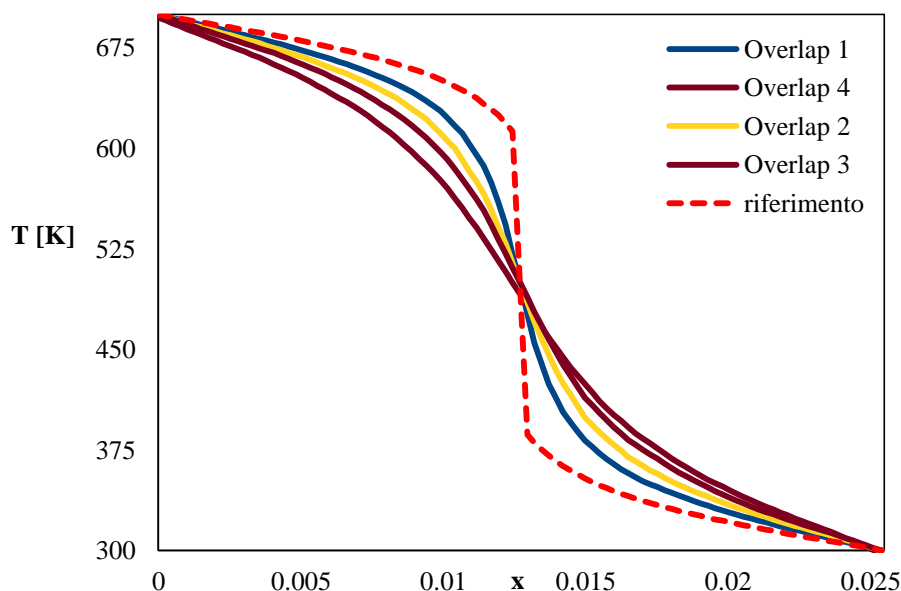


Tabella 3.45 Profilo di temperatura secondo il modello *gap*.



**Figura 3.46 Profilo di temperatura secondo il modello *overlap*.**

Tra gli approcci utilizzati, il modello Dixon *et al.* [2] descrive meglio il profilo di temperatura all'interno delle sfere. Infatti confrontando i salti termici vicino il punto di contatto si nota che sono molto simili al caso di riferimento.

Gli altri metodi forniscono dei profili molto diversi e poco realistici. Analizzando il *cap* ed il *gap method* è stato evidenziato che la presenza di fluido tra le sfere crea una rilevante discontinuità della conducibilità termica effettiva e quindi salti termici molto pronunciati. In particolare per il *cap 1* si è osservato che una distanza tra le superfici delle sfere molto piccola, pari a  $d_s/200$ , non fornisce grosse variazioni del profilo di temperatura. Valori del *cap* più elevati forniscono viceversa delle alte differenze di temperature tra le superfici delle sfere in corrispondenza del *cap* applicato. Il meccanismo di conduzione termica tra le superfici del *bridge* (con il modello *bridge* semplificato) e le superfici di contatto (per l'*overlap method*) garantiscono una continuità della regione solida con coefficiente di scambio termico costante. Il sistema complessivo può essere visto pertanto come un'unica regione solida che varia forma lungo la coordinata  $x$ . Anche se questi due modelli offrono il vantaggio di considerare un unico valore di conducibilità termica delle regioni solide (la ragione verrà fornita in dettaglio nel Capitolo 6), ma fornisce dei profili termici

poco realistici. Un'analisi interessante è stata quella di confrontare i flussi termici uscenti dalla sfera che si trovano a temperatura più alta rispetto al riferimento. Nella Figure 3.47–3.50 sono stati riportati i flussi termici sull'asse delle ordinate, mentre sulle ascisse si riportano i rapporti  $r_b/d_s$  e  $2h/d_s$  per il *bridge* ed il *cap* rispettivamente, mentre per i metodi globali si riporta il rapporto  $d_{attuale}/d_{nominale}$ .

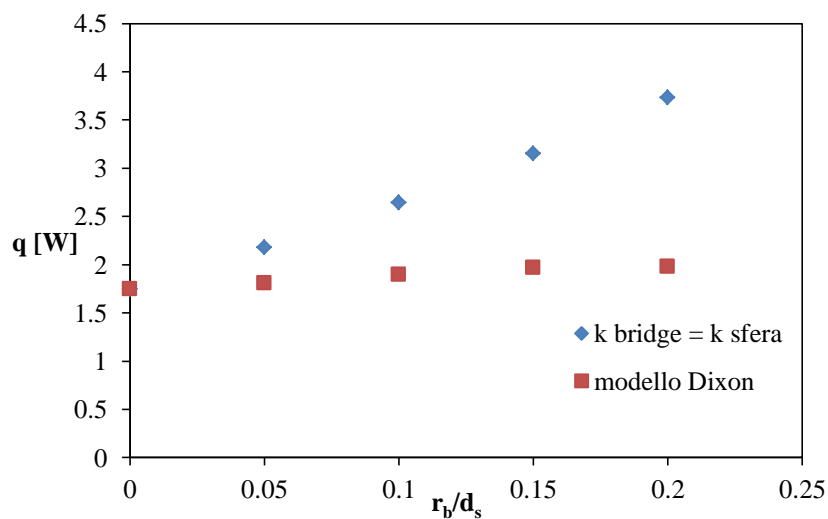


Figura 3.47 Confronto dei flussi termici tra il riferimento e i modelli dei *bridges*.

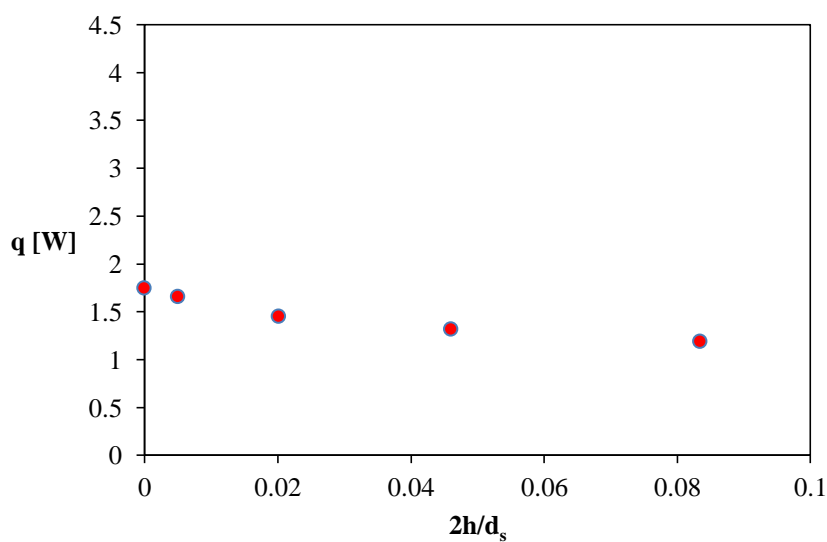


Figura 3.48 Confronto dei flussi termici tra il riferimento e il modello *cap*.

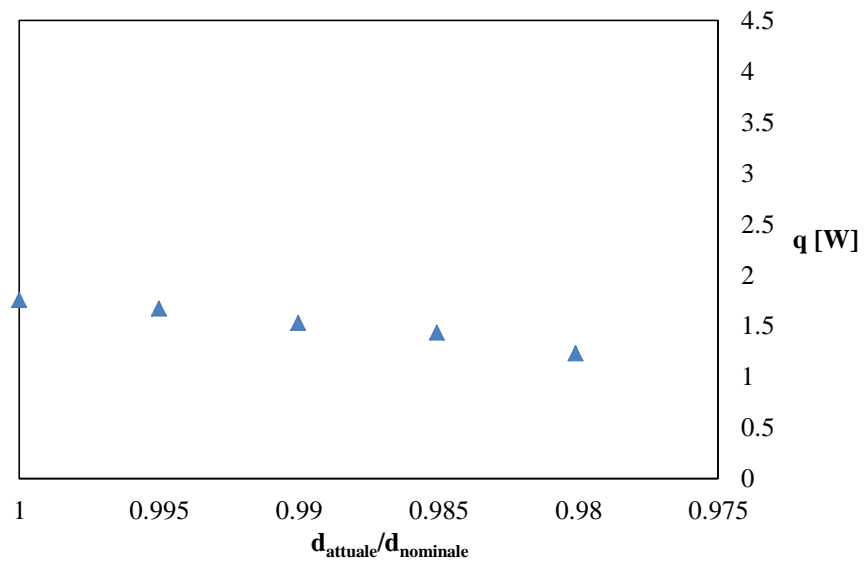


Figura 3.49 Confronto dei flussi termici tra il riferimento e il modello *gap*.

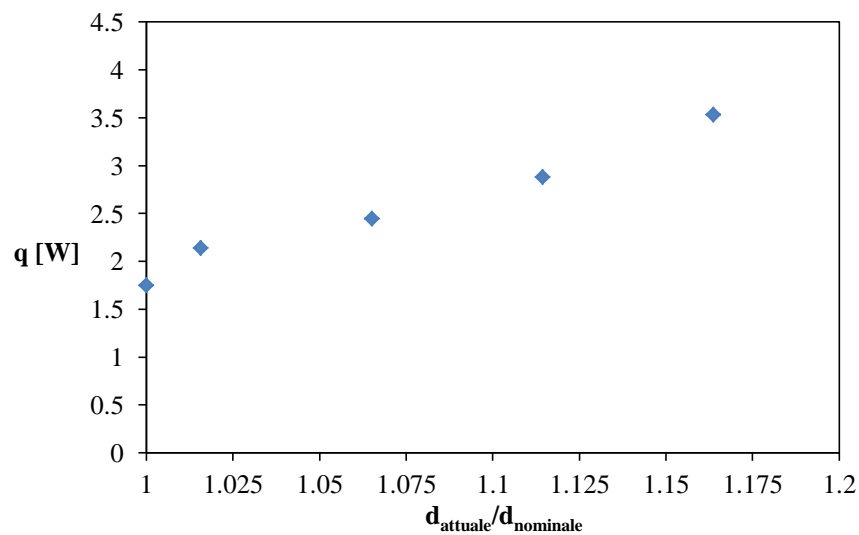


Figura 3.50 Confronto dei flussi termici tra il riferimento e il modello *overlap*.

La Figura 3.47 mostra chiaramente le deviazioni dei due modelli rispetto il riferimento. In particolare il modello Dixon *et al.* [4] sembra essere più accurato per descrivere correttamente il flusso termico trasferito tra le due sfere, anche aumentando il rapporto  $r_b/d_s$  la variazione del flusso termico rispetto al caso riferimento si può ritenere accettabile.

Come visto dall'analisi dei profili di temperatura, il cap e gap method hanno fornito dei valori dei flussi termici che diminuiscono all'aumentare della distanza tra le due sfere. Questo perché le modifiche oltre a provocare una diminuzione del volume di solido delle sfere esiste un filetto di fluido tra di esse con un coefficiente di scambio termico di circa 40 volte inferiore rispetto a quello delle sfere.

Aumentando il diametro delle sfere, si è ottenuta la sostituzione del punto di contatto con una superficie di contatto. Questa superficie è tanto maggiore quanto più grande sarà la sovrapposizione delle sfere. Si introduce pertanto un altro meccanismo di scambio termico che è la conduzione.

A differenza degli altri metodi visti, i risultati ottenuti per i flussi termici si discostano drasticamente dal caso di riferimento anche per piccoli incrementi del diametro delle sfere in ragione del contributo conduttivo per lo scambio termico. È evidente come *l'overlap* tra le sfere non sia un metodo efficiente per modellare correttamente il meccanismo di scambio termico tra due sfere che si trovano a contatto.

### 3.4 Conclusioni

Da un'attenta analisi dei risultati ottenuti sia per la reattività chimica che per lo scambio termico è stato visto che i metodi di modifica locale dei punti di contatto sono i più appropriati per modellare un punto di contatto sfera/sfera. L'area catalitica gioca un ruolo fondamentale nella valutazione dell'effetto delle modifiche sui valori di conversione dei reagenti coinvolti nella reazione chimica catalitica superficiale. La validità di questa ipotesi è stata confermata analizzando la variazione percentuale di area catalitica ed indagando i campi di moto locali vicino il punto di contatto. È stato visto che quelle zone fossero caratterizzate da elevati tempi di contatto e quindi elevata conversione dei reagenti in prodotti.

I metodi di modifica globale anche se richiedono dispendi computazionali inferiori rispetto ai metodi locali non sono dei buoni candidati per essere implementati, poiché comportano una variazione significativa dell'area catalitica



Tra i metodi locali il *cap* è quello che garantisce la minima variazione possibile di superficie catalitica. Questo si ripercuote sui valori di conversione ottenuti che sono stati molto più realistici rispetto agli altri metodi descritti.

Lo studio sui fenomeni di scambio termico ha dimostrato come confrontando i diversi metodi di modifica implementati, il modello proposto da Dixon *et al.* [4] fornisca dei risultati più accurati. L'*overlap* ed il modello *bridge* semplificato considerano un unico coefficiente di scambio termico e grazie alla conduzione si ottengono flussi termici molto elevati e profili termici poco realistici. Sia il *gap method* che il *cap method*, grazie alla presenza del filetto di fluido che si trova tra le due sfere, creano una discontinuità dei profili di temperatura la quale comporta una perdita di flusso termico nella regione fluida circostante.

Questi studi mostrano come i fenomeni di scambio di materia vengono descritti più appropriatamente dal *cap method*, mentre lo scambio termico viene meglio descritto dal *bridge method*, utilizzando il modello elaborato da Dixon *et al.* [4].

Un interessante studio potrebbe essere quello di indagare i fenomeni di scambio di materia accoppiati a quelli di scambio termico, ad esempio uno studio dell'effetto delle modifiche dei punti di contatto per un caso in presenza di una reazione chimica catalitica esotermica. Questo studio verrà trattato in dettaglio nel Capitolo 5 in cui si confrontano i due metodi di modifica locale per capire quale dei due approcci risulta il più appropriato. Lo studio sistematico descritto in questo Capitolo, oltre a dare i risultati descritti, fornisce delle importanti indicazioni sulla generazione automatica delle geometrie e mesh attraverso codici di calcolo. Questo è un grande vantaggio poiché i metodi di modifica dei punti di contatto possono essere implementati in codici di calcolo per generare geometrie e mesh di sistemi di sfere anche complessi ad esempio letti impaccati.

---

## Bibliografia

- [1] Kim I., Elghobashi S., Sirignano W.A., “*Three-dimensional flow over two spheres placed side by side*”, Journal of Fluid Mechanics, 246, pp. 465- 488, (1993).
- [2] Yoon D.H. e Yang K.S., “*Characterization of flow pattern past two spheres in proximity*”, Physics of Fluids, 21, (2009).
- [3] Schouveiler L., Brydon A., Leweke T., Thompson M.C., “*Interactions of the wakes of two spheres placed side by side*”, European Journal of Mechanics B/Fluids, 23(1), pp. 137–145,(2004).
- [4] Dixon A.G., Nijemeisland M., Stitt E.H., “*Systematic mesh development for 3D CFD simulation of fixed beds: Contact points study*”,Computers and Chemical Engineering,48, pp. 135-153, (2013).
- [5] Lund K.O., Nguyen H., Lord S.M., “*Numerical Correlation for Thermal Conduction in Packed Beds*”, The Canadian Journal of Chemical Engineering, 77(4), pp. 769–774, 1999.
- [6] Maestri M. e Cuoci A., *Coupling CFD with detailed microkinetic modeling in heterogeneous catalysis*, Chemical Engineering Science, 96, pp. 106-117, (2013).
- [7] Maestri M. e Cuoci A., [www.catalyticfoam.polimi.it](http://www.catalyticfoam.polimi.it).
- [8] <http://www.openfoam.com/>
- [9] <http://www.learnpython.org/>
- [10] Ookawara S., Kuroki M., Street D., Ogawa K., “*High-fidelity DEM-CFD modeling of packed bed reactors for process intensification*”, European Congress of Chemical Engineering, 6. (2007).
- [11] Eppinger T., Seidler K., Kraume M., “*DEM-CFD simulations of fixed bed reactors with small tube to particle diameter ratios*”, Chemical Engineering Journal, 166(1) pp. 324-331, (2011).
- [12] Calis H.P.A., Nijenhuis J., Paikert B.C., Dautzenberg F.M., van den Bleek, C. M.. *CFD modeling and experimental validation of pressure drop and flow*

- 
- profile in a novel structured catalytic reactor packing*”, Chemical Engineering Science, 56(4), pp. 1713-1720, (2001).
- [13] Guardo A., Coussirat M., Larrayoz M.A., Recasens F., Egusquiza E., “*CFD flow and heat transfer in nonregular packings for fixed bed equipment design*”. Industrial & Engineering Chemistry Research, 43(22), pp.7049-7056, (2004).
- [14] <http://cfdirect.openfoam/user-guide/boundaries/>
- [15] Incropera F.P., DeWitt D.P., Bergman T.L. Lavine A.S., “*Fundamentals of Heat and Mass Transfer*”, 7th Edition, John Wiley, (2011).
- [16] [www.bakker.org/dartmouth06/engs150/06-bound.ppt](http://www.bakker.org/dartmouth06/engs150/06-bound.ppt)
- [17] <http://cfdirect.openfoam/user-guide/fvschemes/>
- [18] <http://www.dicat.unige.it/guerrero/oftraining/9tipsandtricks.pdf>
- [19] [http://info.tuwien.ac.at/ViennaOpenFOAMUserGroup/9thb\\_OpenFOAM\\_Tutorial.pdf](http://info.tuwien.ac.at/ViennaOpenFOAMUserGroup/9thb_OpenFOAM_Tutorial.pdf)
- [20] <https://github.com/nicolasedh/salomeToOpenFOAM>
- [21] [http://www.dicat.unige.it/guerrero/oftraining/3\\_2mesh\\_quality\\_assessment.pdf](http://www.dicat.unige.it/guerrero/oftraining/3_2mesh_quality_assessment.pdf)
- [22] <https://github.com/srebughini/catalyticFOAMutilities>
- [23] <http://www.cfd-online.com/Forums/openfoam/99559-boundary-condition-chtmultiregionfoam.html>
- [24] <http://www.openfoam.org/version2.3.0/thermal.php>
- [25] Schlünder E.U., “*Wärmeübergang an bewegte Kugelschüttungen bei kurzfristigem Kontakt*”, Chemie Ingenieur Technik, 43(11), pp. 651–654, (1971).
- [26] Saxena S.C. e Joshi R.K. “*Thermal Accommodation and Adsorption Coefficients of Gases*”, Hemisphere Publishing Corporation, New York, (1989).

# 4 Studio dei fenomeni di scambio di materia e di calore per un sistema costituito da 4 sfere

## 4.1 Introduzione

Nel Capitolo 3 sono stati studiati i fenomeni di scambio di materia e termico per un caso semplice sfera/sfera. Il punto di contatto è stato modificato con i 4 metodi di modifica sia globali che locali. Da un'attenta analisi si è constatato che i metodi di modifica locale sembrano essere i più appropriati per modellare un punto di contatto sfera/sfera. In particolare, è stato osservato che per simulazioni CFD in presenza di una reazione chimica, il *cap method* garantisce dei risultati più accurati rispetto al *bridge* in ragione della minima variazione di area catalitica. Questo non è vero invece per lo scambio termico, in quanto la presenza di un filetto di fluido tra le sfere e la perdita di volume di solido delle sfere non garantisce l'integrale trasferimento del calore tra le due sfere, con conseguente dissipazione di energia termica nella regione fluida.

Lo scopo di questo Capitolo è quello di studiare i fenomeni di scambio di materia e di calore per configurazioni più vicine ai casi reali di reattori impaccati a letto fisso. Lo studio è stato eseguito su un sistema costituito da 4 sfere in serie all'interno di un reattore dove è stata considerata anche la presenza di punti di contatto sfera/parete.

Per lo studio della reattività chimica è stato utilizzato il solver `catalyticSimpleFoam` [1,2], mentre lo scambio termico è stato studiato attraverso il solver nativo di OpenFOAM: `chtMultiRegionSimpleFoam` [3].

Sono stati affrontati i problemi relativi alla generazione della mesh per creare il caso di riferimento e parallelamente i metodi di modifica dei punti di contatto sono stati sistematicamente studiati. Durante la generazione della *mesh* si è visto che un punto

di contatto sfera/parete è più complicato da gestire: infatti l'elevato numero di celle con facce non ortogonali costringe l'utente a generare delle mesh molto fitte in quelle zone per garantire bassi valori dell'indice di *skewness*. Questo problema potrebbe essere risolto implementando le modifiche sia globali che locali sulle sfere descritte per il caso sfera/sfera. È chiaro che in una modifica di un punto di contatto sfera/parete la superficie catalitica interessata alla modifica è quella di una sola sfera e non due come per il caso sfera/sfera. Pertanto la porzione di area catalitica e la porzione di volume della sfera interessata alla modifica locale (sia per il *bridge* che per il *cap*) è inferiore rispetto al caso sfera/sfera. Compatibilmente con la fluidodinamica imposta dalla modifica locale tra la sfera e la parete, questa considerazione permette di affermare che i risultati ottenuti sullo studio del trasferimento di materia nel Capitolo 3 possono essere estesi anche per un punto di contatto sfera/parete in quanto la modifica locale comporta la minima variazione possibile di area catalitica. Gli approcci di modifica globale non sono stati applicati per lo studio della reattività chimica. È stato osservato che per garantire piccole variazioni di area catalitica è necessario imporre una diminuzione (o aumento) percentuale del diametro delle sfere con valori compresi tra 0.05% e 0.2%. Si generano così delle mesh tetraedriche con un elevato numero di facce non-ortogonali tra loro e questo si ripercuote sul calcolo dei gradienti tra celle adiacenti. Si potrebbero aumentare considerevolmente il numero di celle nelle zone dove la distanza tra le sfere è minima, ma ciò non risolverebbe il problema. Un'altra soluzione è quella di diminuire (o aumentare) il diametro delle sfere a valori più bassi (o più alti), ma per un sistema costituito da 4 sfere la variazione di area catalitica rispetto al caso di riferimento è notevole. La variazione considerevole della geometria globale delle sfere potrebbe riflettersi sulla stima della conversione della specie limitante e potrebbe inoltre influenzare il campo di moto tra le sfere. Infatti dalle prove svolte nel Capitolo 3, facendo riferimento alle Tabelle 3.13–3.17, si è visto che una diminuzione di circa il 3% dell'area catalitica (*gap method*) comporta una diminuzione di conversione di circa il 4% rispetto al caso di riferimento per valori di Damköhler pari a 0.015. Nelle stesse condizioni operative è stato visto che per l'*overlap* l'espansione della superficie delle sfere provoca un aumento della

conversione dovuto all'incremento della superficie catalitica: un'espansione di circa il 3% della superficie catalitica comporta un incremento della conversione del 7% circa. In questo Capitolo lo studio della reattività chimica è stato eseguito a diversi numeri di Reynolds per indagare l'effetto delle due modifiche sul campo di moto locale ed i profili delle frazioni massive delle specie chimiche in prossimità dei punti di contatto.

Lo studio sullo scambio termico descritto nel Capitolo 3 ha dimostrato che il *bridge method* è il metodo più appropriato per modellare un punto di contatto sfera/sfera. Per simulare condizioni sufficientemente rappresentative di casi di interesse industriale (dove la parete esterna dei reattori a letto fisso è mantenuta ad una temperatura fissata, ad esempio per reazioni fortemente esotermiche dove fluido refrigerante mantiene la temperatura esterna dei tubi a valori più bassi rispetto alla temperatura di reazione), è stato imposto un gradiente termico tra le due estremità del tubo, investigando così i salti termici nelle zone in cui è stato modificato il punto di contatto sfera/sfera e sfera/parete. In questa sede sono stati confrontati in maniera qualitativa i risultati ottenuti per i 4 approcci evidenziando i profili termici sviluppatasi nelle sfere. Lo studio è stato eseguito in assenza di reazione chimica, imponendo per la velocità la condizione *no slip* sulle pareti delle sfere e del tubo.

## 4.2 Studio della reattività chimica

Il sistema costituito da 4 sfere poste in serie all'interno di un reattore a sezione rettangolare è mostrato in Figura 4.1.

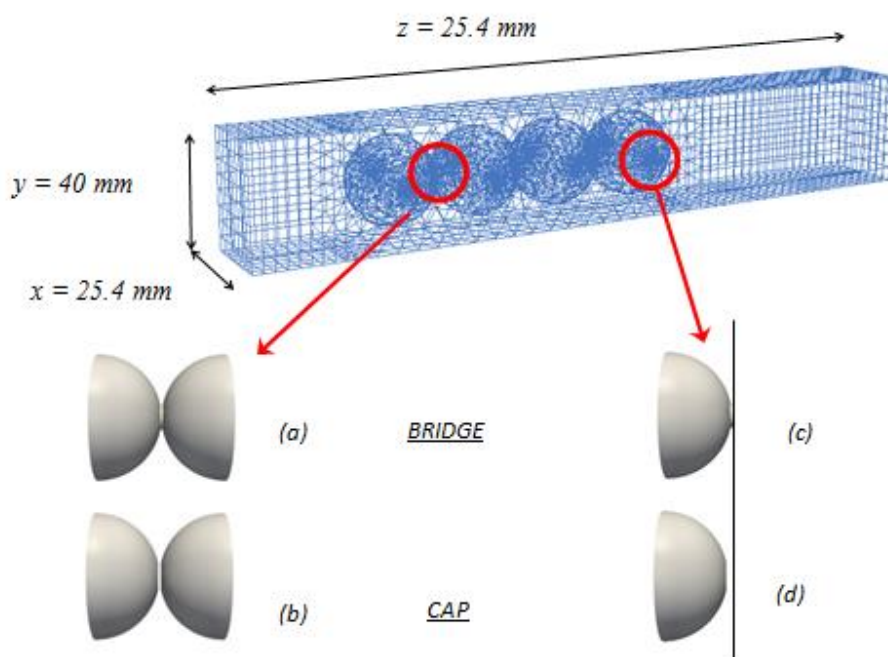


Figura 4.1 Modifiche locali dei punti di contatto sfera/sfera e sfera/parete. Implementazione del *bridge method* per i casi (a) e (c), *cap method* per i casi (b) e (d).

I valori utilizzati per le lunghezze dei *bridges* e del *cap* sono riportati in Tabella 4.1 e 4.2.

	rapporto dimensione bridge/sfera	raggio del bridge	modifica sfera/sfera	modifica sfera/parete
metodo	$r_b/d_s$	$r_b$ [mm]	$h_b$ [mm]	$h_b/2$ [mm]
<i>bridge 1</i>	0.05	1.270	0.127	0.0635
<i>bridge 2</i>	0.1	2.54	0.513	0.2565
<i>bridge 3</i>	0.15	3.81	1.170	0.585
<i>bridge 4</i>	0.2	5.08	2.121	1.061

Tabella 4.1 Valori delle dimensioni del bridge a diversi rapporti dimensione *bridge*/dimensione sfera per le due modifiche sfera/sfera e sfera/parete.

	<b>rapporto dimensione cap/sfera</b>	<b>modifica sfera/sfera</b>	<b>modifica sfera/parete</b>
<b>metodo</b>	<b>2h/d<sub>s</sub></b>	<b>h [mm]</b>	<b>h/2 [mm]</b>
<i>cap 1</i>	0.005000	0.127	0.0635
<i>cap 2</i>	0.020197	0.513	0.2565
<i>cap 3</i>	0.046063	1.170	0.5850
<i>cap 4</i>	0.083504	2.121	1.0605

**Tabella 4.2** Valori delle dimensioni del cap a diversi rapporti dimensione del cap/dimensione sfera per le due modifiche sfera/sfera e sfera/parete.

La reazione chimica è la stessa descritta nel Capitolo 3, le condizioni operative utilizzate sono riportate in Tabella 4.3.



<b>Specie</b>	<b>Frazione massiva</b>	<b>condizioni operative</b>
H <sub>2</sub>	29.85 · 10 <sup>-4</sup>	Da=0.015 k <sub>cin</sub> =53.59 · 10 <sup>-6</sup>
O <sub>2</sub>	11.845 · 10 <sup>-3</sup>	P=101325 Pa
N <sub>2</sub>	98.517 · 10 <sup>-2</sup>	T=373.15 K

**Tabella 4.3** Condizioni operative utilizzate per lo studio della reattività chimica.

I confronti sono stati eseguiti a diversi numeri di Reynolds:

<b>Velocità [m/s]</b>	<b>Numero di Reynolds</b>
0.2	142.24
0.5	355.6
1	711.2
2	1422.4

**Tabella 4.4** Valori del numero di Reynolds a diverse velocità del fluido in ingresso al reattore.



### 4.2.1 Analisi della convergenza della mesh

Il numero di celle richiesto per ottenere dei risultati che non dipendessero dalla discretizzazione del dominio di calcolo è stato considerevolmente elevato. Per entrambi i metodi sono state imposte delle dimensioni lineari delle celle tetraedriche nelle zone vicino al punto di contatto pari ad un quarto del valore del *cap 1* e della lunghezza del *bridge* rispettivamente applicato tra le sfere e tra sfera e parete. La generazione della mesh per il caso di riferimento è stata abbastanza complicata, in particolare per la presenza dei punti di contatto delle sfere con la parete. Allo scopo di evitare di ottenere delle celle non ortogonali tra loro con elevati valori dell'indice di skewness, sono stati generati dei vertici nei diversi punti di contatto e considerati come nodi della mesh non strutturata, come per il semplice caso sfera/sfera descritto nel Capitolo 3. Grazie alla funzione “*local size*” di SALOME, è stata imposta nei nodi della mesh una dimensione lineare di circa  $d_s/300$ . Per monitorare l'area catalitica discretizzata, si sono confrontati i valori della superficie catalitica generata dalla mesh con la superficie catalitica esatta della geometria fornita da SALOME. I valori sono stati riportati in Tabella 4.5.

	Numero di celle	Area teorica [cm <sup>2</sup> ]	Area discretizzata [cm <sup>2</sup> ]	Errore % Area
<i>bridge 1</i>	1720550	80.433	80.429	0.0042
<i>bridge 2</i>	1604753	78.780	78.776	0.0046
<i>bridge 3</i>	1498920	76.499	76.495	0.0046
<i>bridge 4</i>	1412707	73.969	73.965	0.0054
<i>riferimento</i>	2595874	81.073	81.072	0.0006
<i>cap 1</i>	1731105	81.070	81.062	0.0099
<i>cap 2</i>	1758875	81.040	81.035	0.0056
<i>cap 3</i>	1812529	80.920	80.915	0.0054
<i>cap 4</i>	1956875	80.580	80.572	0.0096

Tabella 4.5 Confronto tra area geometrica esatta e superficie catalitica discretizzata a diverso numero di celle.

Di seguito si riportano i grafici che riportano i valori della conversione rispetto al numero di celle utilizzate per i diversi casi.

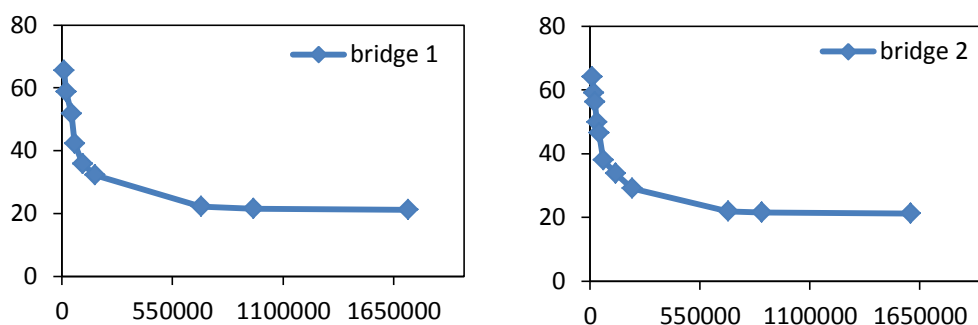


Figura 4.2 Valori delle conversione di  $O_2$  rispetto al numero di celle per i casi *bridge 1* e *bridge 2*.

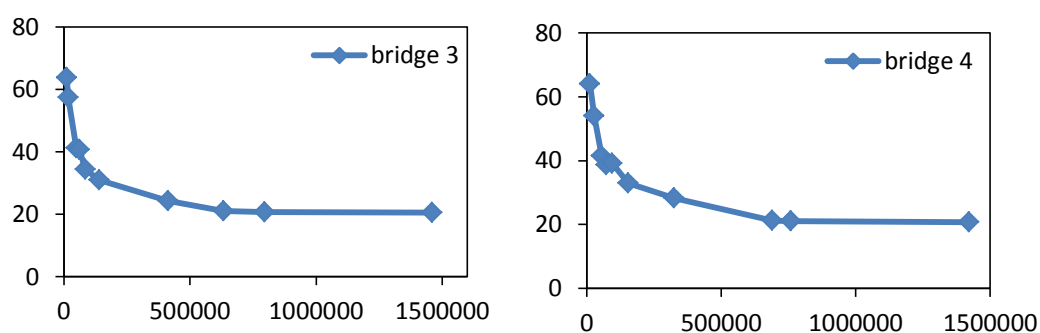


Figura 4.3 Valori delle conversione di  $O_2$  rispetto al numero di celle per i casi *bridge 3* e *bridge 4*.

Le Figure 4.2 e 4.3 mostrano l'effetto della discretizzazione del dominio geometrico sulla conversione della specie limitante. I grafici mostrano che la convergenza è stata raggiunta a valori molto alti del numero di celle. In particolare si nota che il *bridge 1* ha richiesto un numero più elevato di celle rispetto agli altri 3 casi in ragione delle piccole dimensioni del bridge. Infatti un aumento notevole delle celle tetraedriche è dovuto alla presenza di piccoli dettagli geometrici come la superficie laterale del bridge. La generazione della *mesh* non strutturata inizia dal luogo geometrico che presenta i più piccoli dettagli geometrici, che sono appunto le superfici laterali del bridge propagandosi poi nella regione fluida circostante. Imponendo delle dimensioni lineari delle celle pari a un valore assegnato si ha un aumento considerevole delle celle in quelle zone. Modulando poi opportunamente il *growth Rate* è stato possibile imporre una variazione lineare delle celle in maniera graduale. Ad esempio, per la generazione di queste *mesh* sono stati imposti valori di circa 0.04. Questo vuol dire che si ha una variazione del 4% dei lati di due celle adiacenti. Questo valore piccolo

garantisce una crescita piuttosto lenta delle celle da zone a piccoli dettagli (*bridge*) fino alla regione fluida circostante, garantendo bassi valori della *skewness* e di non ortogonalità della *mesh*.

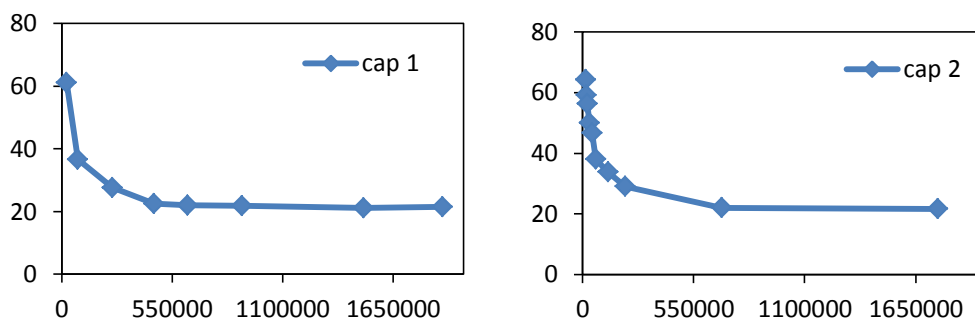


Figura 4.4. Valori delle conversione di O<sub>2</sub> rispetto al numero di celle per i casi *cap 1* e *cap 2*.

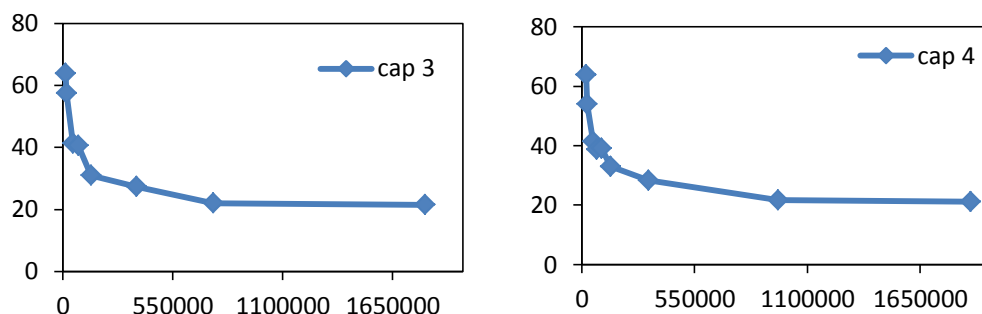


Figura 4.5. Valori delle conversione di O<sub>2</sub> rispetto al numero di celle per i casi *cap 3* e *cap 4*.

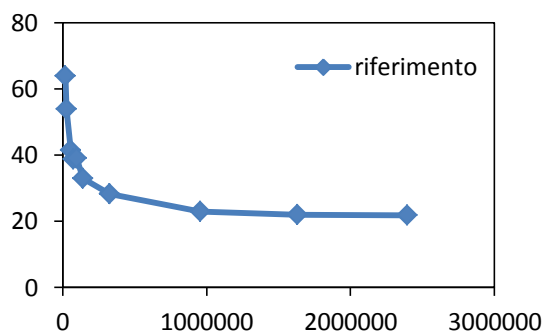


Figura 4.6 Valori delle conversione di O<sub>2</sub> rispetto al numero di celle per il casodi riferimento.

A differenza del metodo locale *bridge*, nel caso del *cap* in ragione del filetto di fluido che si trova tra le sfere e le sfere e la parete del reattore, il numero di celle richiesto è superiore rispetto ai 4 casi del *bridge*. È stato necessario infittire le zone dove la

modifica dei punti di contatto è stata applicata per indagare sia il campo di moto locale che i profili di frazioni massive delle specie chimiche. I valori di convergenza della mesh riportati sono stati ottenuti per valori del numero di Reynolds pari a 142.24. Questa convergenza raggiunta è stata rispettata anche per numeri di Reynolds superiori.

#### 4.2.2 Risultati e discussioni

Una volta verificato che le soluzioni delle simulazioni CFD sono indipendenti dalla discretizzazione del dominio di calcolo, è stato possibile confrontare i risultati ottenuti dalle simulazioni per i due metodi di modifica locale, con il caso di riferimento. Il confronto è stato fatto sulla conversione della specie limitante. In Figura 4.7 si mostra l'andamento della conversione di  $O_2$  per diverse dimensioni del *bridge* rispetto al numero di Reynolds. Come si può vedere, a bassi numeri di Reynolds si hanno delle conversioni dell'ordine del 20% circa, mentre all'aumentare di quest'ultimo la conversione della specie limitante diminuisce a causa della diminuzione del tempo di contatto delle specie chimiche nel reattore.

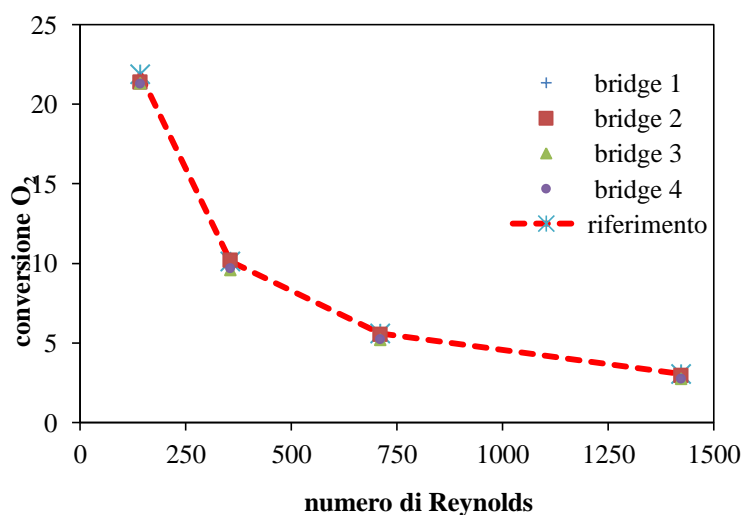


Figura 4.7 Valori di conversione di  $O_2$  in funzione del numero di Reynolds per i 4 casi con *bridges* confrontati con il caso di riferimento.

Quest'analisi dimostra chiaramente che piccole variazioni della superficie catalitica dovute all'implementazione dei *bridges* comportano variazioni comunque accettabili della conversione. Dai risultati ottenuti si evidenzia inoltre che la variazione percentuale di conversione dei 4 casi studiati (dal *bridge 1* fino al *bridge 4*) si mantiene entro errori del 10% circa. I valori sono riportati in Tabella 4.6.

	Variazione % conversione O <sub>2</sub>			
	Re 1	Re 2	Re 3	Re 4
<i>bridge 1</i>	1.005	-0.692	0.536	1.947
<i>bridge 2</i>	2.239	-0.593	1.055	2.938
<i>bridge 3</i>	2.513	5.237	7.512	9.541
<i>bridge 4</i>	2.742	5.650	6.964	9.610

**Tabella 4.6** Variazione percentuale di conversione di O<sub>2</sub> per i casi *conbridges* rispetto al caso di riferimento.

Dalla tabella si vede che il caso con un *bridge* di dimensioni più piccole rispetto agli altri fornisce variazioni sotto il 2% di conversione. Implementando *bridges* con dimensioni maggiori si ha maggiore errore sulla conversione e quindi meno accuratezza dei risultati. In generale, da questa analisi si dimostra che anche per valori elevati del numero di Reynolds gli errori percentuali di conversione rispetto al caso di riferimento sono contenuti entro il 10% circa.

Questi valori ottenuti sono stati poi confrontati con i risultati ottenuti applicando il *cap* come modifica locale. In Figura 4.7 si riportano i valori di conversione ottenuti per il *cap method* a diversi numeri di Reynolds.

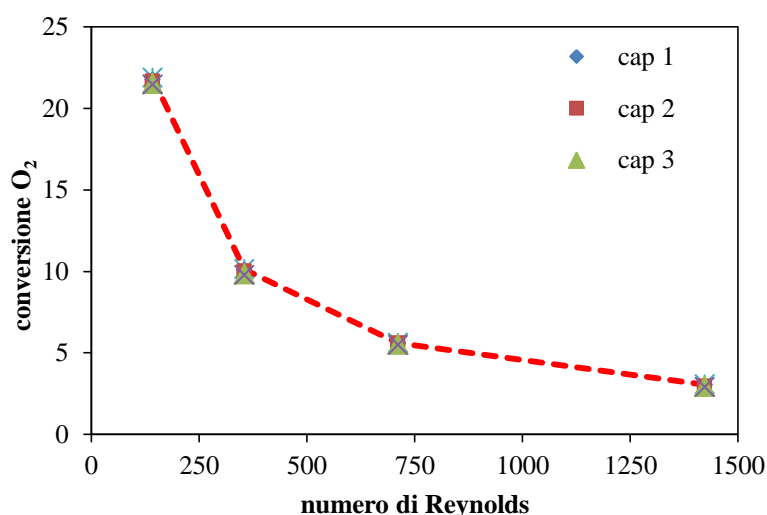


Figura 4.8 Valori di conversione di O<sub>2</sub> in funzione del numero di Reynolds per i 4 casi *concaps* confrontati con il caso di riferimento.

A differenza del metodo precedente, i valori di conversione ottenuti per i 4 casi a dimensione diversa del *cap* variano entro valori contenuti del 4% rispetto al caso di riferimento.

	Variazione % conversione O <sub>2</sub>			
	Re 1	Re 2	Re 3	Re 4
<i>cap 1</i>	0.411	0.296	0.357	0.627
<i>cap 2</i>	1.142	1.383	0.536	3.599
<i>cap 3</i>	1.645	2.867	1.607	4.259
<i>cap 4</i>	2.011	3.557	2.143	3.962

Tabella 4.7 Valori di variazione percentuale della conversione di O<sub>2</sub> per i casi con il *cap* rispetto al caso di riferimento.

In pratica per valori del *cap* molto piccoli di circa  $d_s/200$  (valore utilizzato per il *cap 1*) si hanno delle variazioni percentuale di conversione fino allo 0.7% circa. Un aumento del valore del *cap* tra le sfere e sfera parete comporta una diminuzione della conversione di ossigeno compatibilmente con la diminuzione di superficie catalitica. A parità di distanza imposta tra le sfere (pari a  $d_s/12$ ) con i due metodi di modifica locali si evidenzia che il *cap* ha una errore percentuale minore rispetto al *bridge*. Questo risultato si spiega analizzando i profili delle frazioni massive delle specie chimiche nelle zone vicino il punto di contatto. Anche se entrambi i metodi sono

delle modifiche locali, il *cap method* comunque garantisce una variazione inferiore di superficie catalitica rispetto al *bridge* e quindi maggiore conversione di reagenti in prodotti.

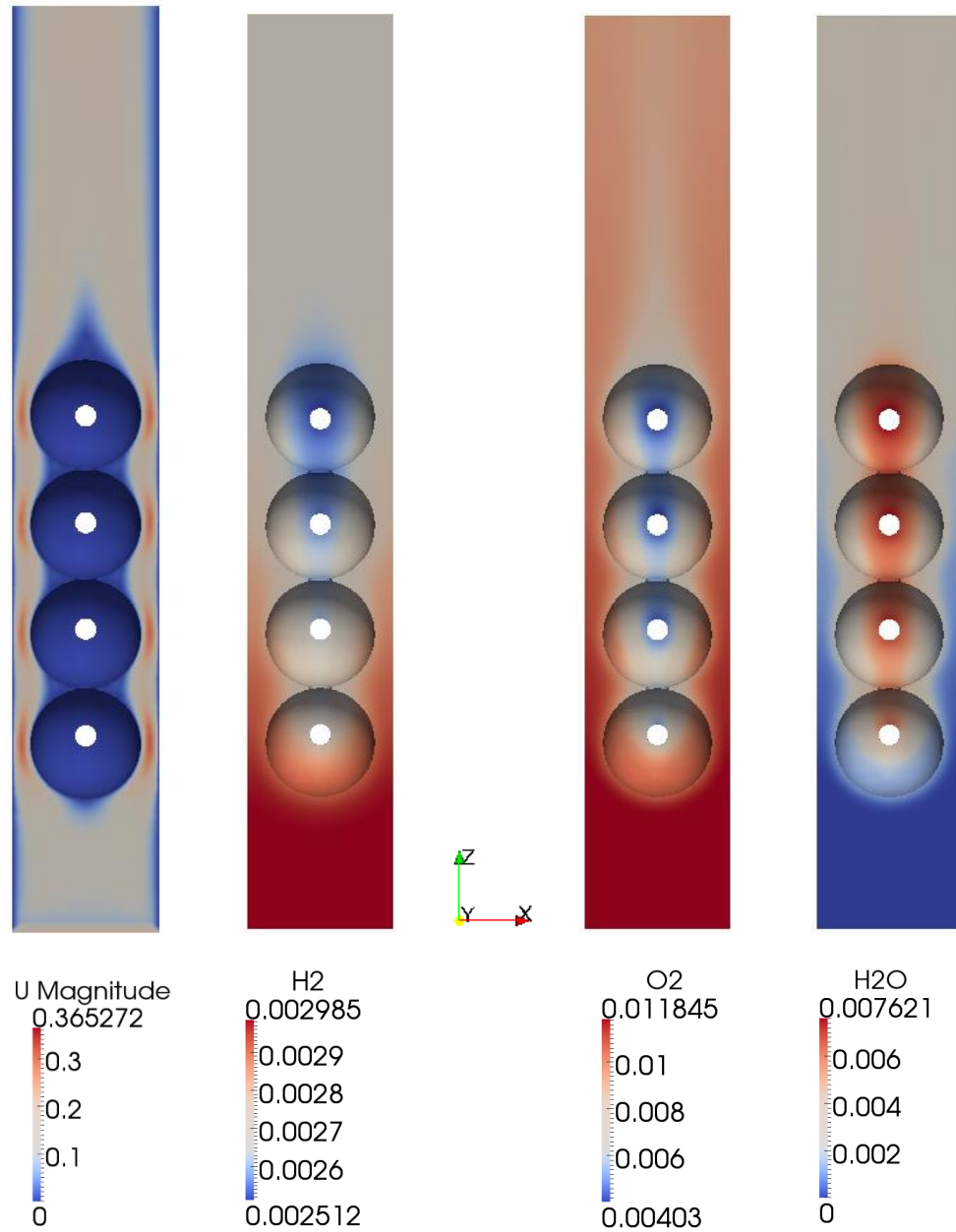


Figura 4.9 Campo di moto e frazioni massive di idrogeno, ossigeno e acqua a  $Re=142.24$  lungo la sezione longitudinale del reattore.

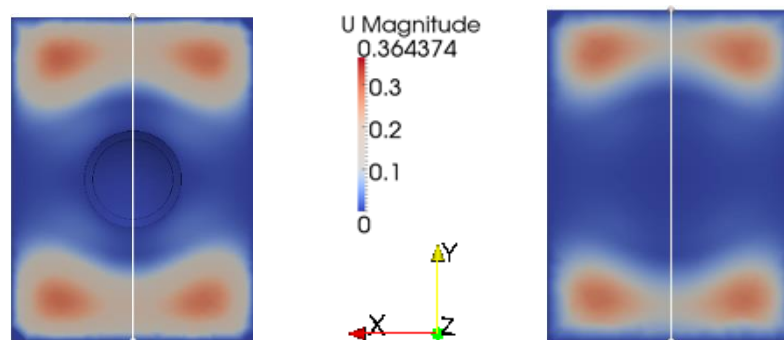


Figura 4.10 Frazione massiva di ossigeno per i casi con bridge 4 e cap 4, a  $Re=142.24$  lungo una sezione trasversale al reattore.

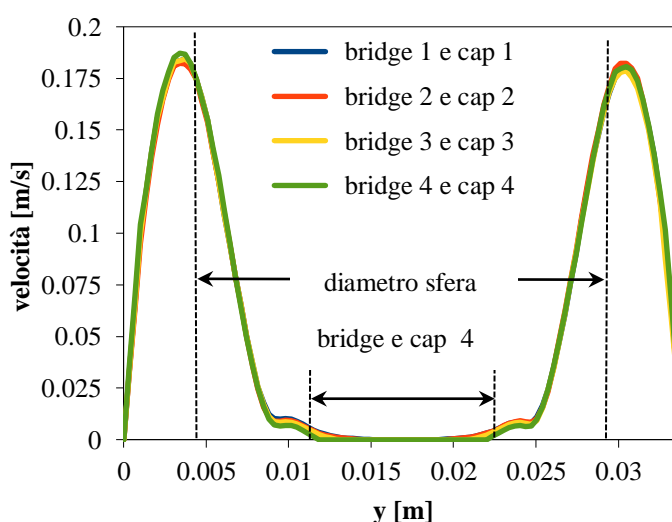


Figura 4.11 Profili di velocità lungo la linea  $y$  in corrispondenza della modifica implementata per i casi con caps e bridges a numero di Reynolds pari a 142.24.

Nella Figura 4.11 si riportano i profili di velocità sviluppatasi lungo una coordinata  $y$  del reattore passante per il punto di contatto sfera/sfera, come mostrato in Figura 4.10. Analizzando i profili del campo di moto, si nota che le massime componenti di velocità si trovano nelle zone lontane dai punti di contatto. Tale profilo diminuisce drasticamente all'avvicinarsi nelle zone dove la modifica è stata applicata fino ad arrivare a valori prossimi allo zero. Per valori di  $y$  compresi tra 11.85 mm e 22.01 mm si nota un appiattimento delle curve: la presenza dei *bridges* non permette il passaggio del fluido, mentre il *cap* applicato tra le sfere garantisce che il trasferimento di materia lungo il filetto di fluido avvenga solo attraverso flussi diffusivi molto piccoli rispetto ai flussi convettivi periferici.



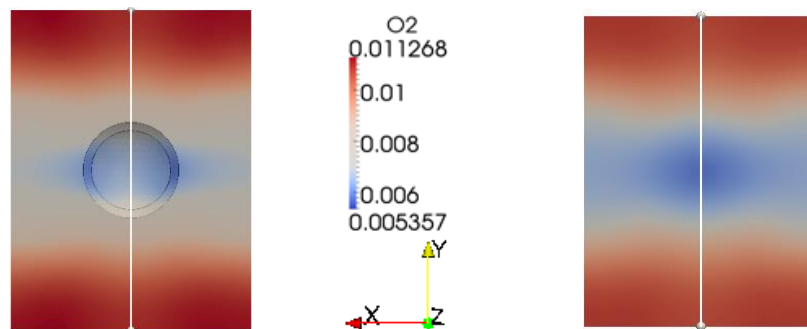


Figura 4.12 Frazione massiva di ossigeno per *bridge 4 cap 4*, a  $Re=142.24$  lungo una sezione trasversale al reattore.

Le Figure 4.13 e 4.15, che riportano i profili delle frazioni massive di ossigeno consumato e acqua prodotta, dimostrano che il trasporto che governa il trasferimento di materia delle specie chimiche dalla fase fluida alla superficie catalitica vicino al punto di contatto avviene grazie ai flussi diffusivi.

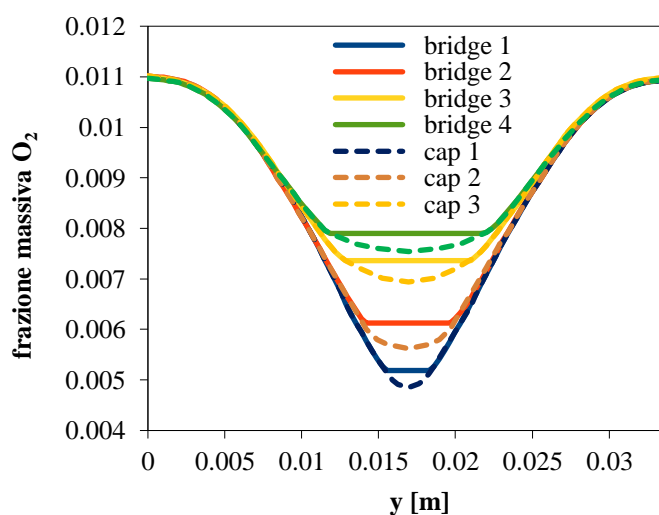


Figura 4.13 Confronto dei profili delle frazioni massive di ossigeno per *caps* e *bridges 4* in corrispondenza del punto di contatto, a  $Re=142.24$ .

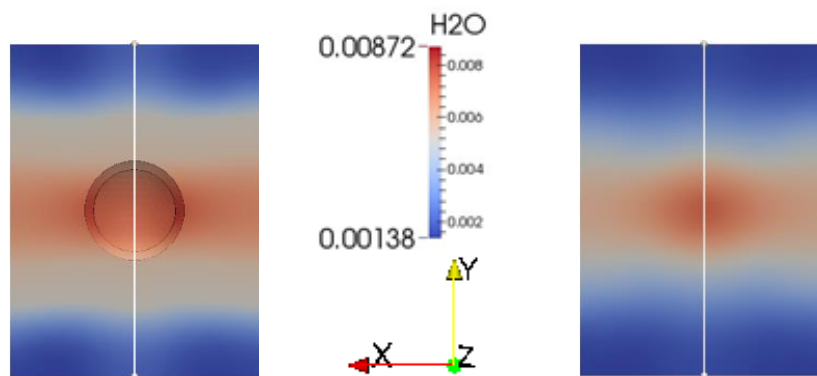


Figura 4.14 Frazioni massive di ossigeno per *cap 4* e *bridge 4*, a  $Re=142.24$ .

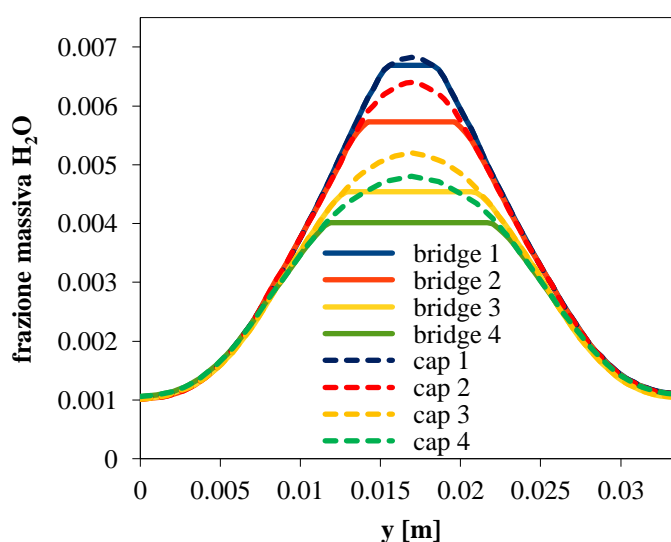


Figura 4.15 Confronto dei profili delle frazioni massive di acqua per *cap* e *bridge 4* in corrispondenza del punto di contatto, a  $Re=142.24$ .

A parità di distanza sfera/sfera per i metodi di modifica locale il *cap method* consente un maggior consumo di reagenti e quindi maggior produzione di acqua. Inoltre si nota che per distanze sfera/sfera via via più piccole (cioè minore variazione di area catalitica) si hanno dei picchi più alti (o più bassi) della frazione massiva del prodotto (o del reagente) di reazione. Questi due fenomeni sono spiegabili con l'aumento di area catalitica che provoca una maggiore conversione dei reagenti in prodotti. Pertanto sebbene i punti di contatto siano regioni stagnanti rispetto alle zone pienamente coinvolte dal flusso convettivo, la superficie catalitica interessata in tali zone viene coinvolta comunque nella reazione catalitica superficiale.

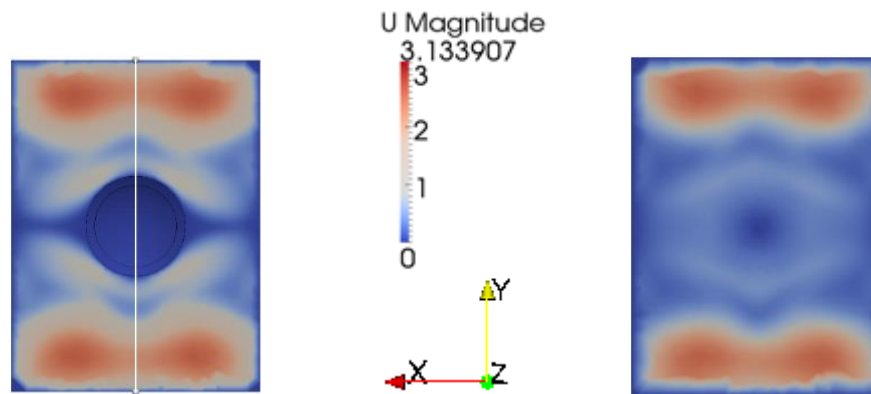


Figura 4.16 Frazioni massive di acqua per i casi con *bridge 4* e *cap 4*, a  $Re=1422.4$ .

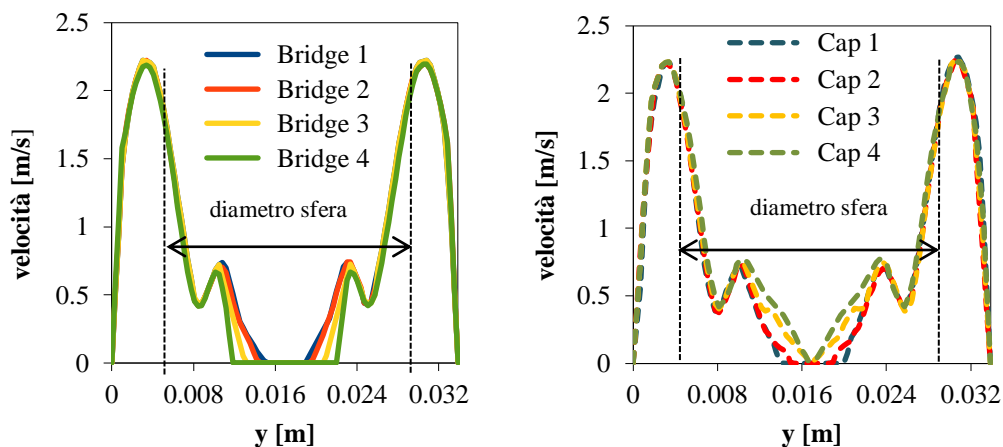


Figura 4.17 Profili di velocità lungo la linea  $y$  in corrispondenza della modifica implementata per i casi con *bridges* e *caps* a numero di Reynolds pari a 1422.4.

All'aumentare del numero di Reynolds si ha un contributo considerevole dei flussi convettivi anche nelle zone vicine alla modifica del punto di contatto sfera/sfera. A differenza della Figura 4.9, nella Figura 4.17 si nota che l'appiattimento delle curve avviene nelle zone solo dove è stato applicato il *bridge*. All'aumentare del *cap* tra le sfere si ha un maggiore passaggio del fluido che diminuisce al diminuire della distanza tra le sfere. In queste condizioni fluidodinamiche i flussi diffusivi vengono meno ed il trasporto di materia è principalmente governato dai flussi convettivi.

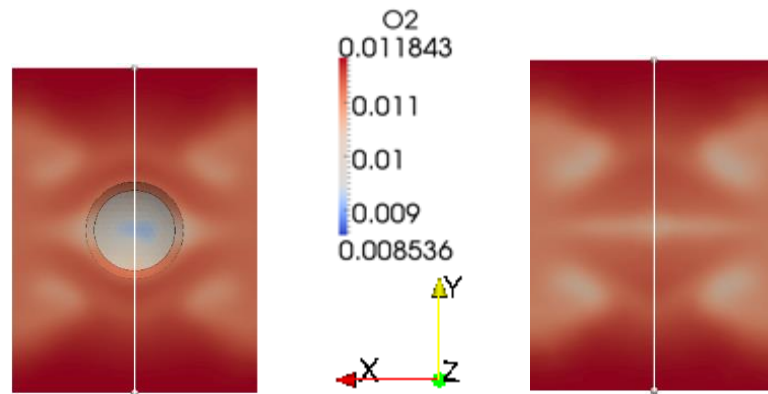


Figura 4.18 Frazione massiva di ossigeno per *bridge 4* e *cap 4* a  $Re=1422.4$  lungo una sezione trasversale al reattore.

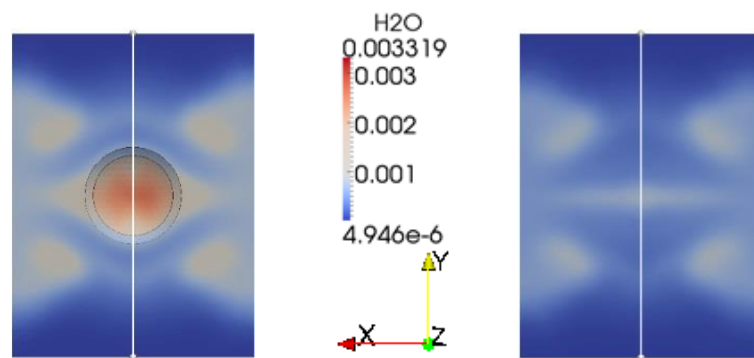


Figura 4.19 Frazioni massive di acqua per *bridge 4* e *cap 4* a  $Re=1422.4$  lungo una sezione trasversale al reattore.

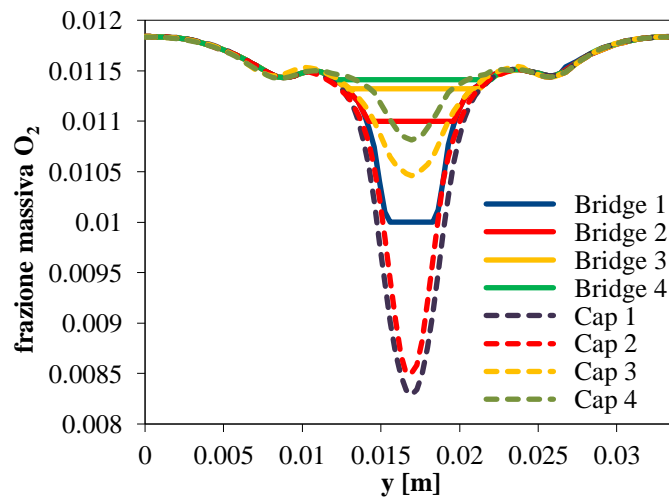
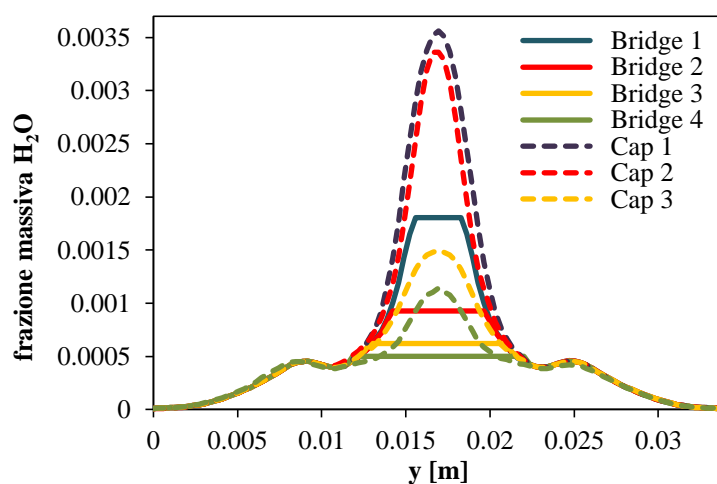


Figura 4.20 Confronto dei profili delle frazioni massive di ossigeno per *caps* e *bridges* in corrispondenza del punto di contatto a  $Re=1422.4$ .



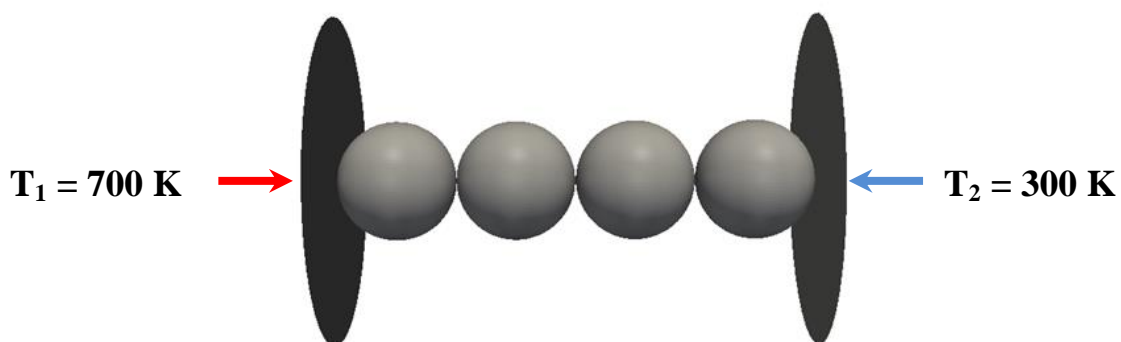
**Figura 4.21** Confronto dei profili delle frazioni massive di acqua percaps e bridges in corrispondenza del punto di contatto a  $Re=1422.4$ .

A numeri di Reynolds più elevati, come si può vedere dalla Figura 4.22, grazie alle condizioni fluidodinamiche più spinte, le zone interessate da modifiche locali sono pienamente coinvolte da fenomeni di reazione chimica superficiale.

### 4.3 Studio dello scambio termico

Dallo studio svolto nel Capitolo 3 si è dimostrato che il modello elaborato da Dixon *et al.* [4] è il più appropriato per modellare i punti di contatto sfera/sfera.

Lo scopo di questo studio è quello di verificare se l'approccio Dixon può essere esteso ad un caso più complesso costituito da 4 sfere. La geometria considerata consiste di 4 sfere poste in serie all'interno di un tubo a sezione circolare.



**Figura 4.22** Schematizzazione della geometria considerata per lo studio dello scambio termico.

I valori delle modifiche effettuate sulle sfere sono riportate nella Tabella 4.8:

<i>Bridge</i>	<i>Cap</i>	<i>Gap</i>	<i>Overlap</i>
$rb/d_s$	$2h/d_s$	diminuzione % diametro	aumento % diametro
0.1	0.0202	0.5	6.5

Tabella 4.8 Dimensioni delle modifiche locali e globali utilizzate.

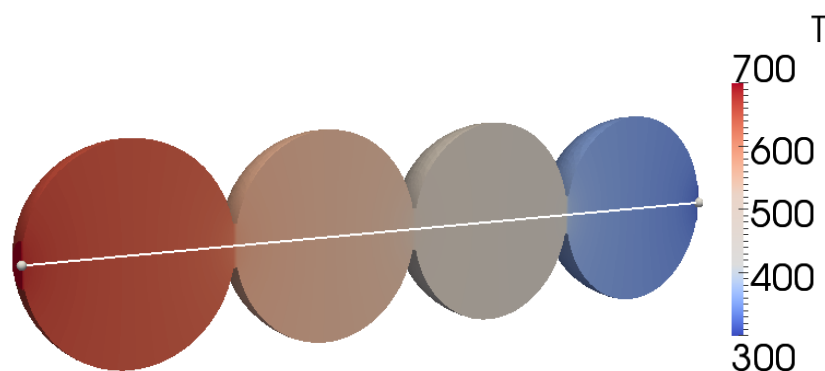
Implementando sia i metodi globali che locali la distanza tra le sfere e la sfera e la parete è stata modificata. Applicando il *cap* ed il *gap* con dimensioni riportate in tabella 4.8 è stata creata una zona fluida tra le sfere e sfera/parete. Nell'overlap si ha ancora il contatto tra le sfere per mezzo delle superfici che si creano per effetto della sovrapposizione, mentre nel *bridge method* i punti di contatto sono stati sostituiti con un *bridge* di dimensioni assegnate riportate in Tabella 4.8. La sua conducibilità termica è stata calcolata secondo il modello Dixon *et al.* [1]. Nella Tabella 4.9 si riportano le proprietà del fluido e delle regioni solide utilizzate.

<i>Proprietà del fluido</i>			
$\rho_f$ [kg/m <sup>3</sup> ]	$C_{p_f}$ [J/(kg · K)]	$\mu$ [Pa · s]	$k_f$ [W/(m · K)]
1.225	1006.43	$1.789 \cdot 10^{-5}$	0.0242

Tabella 4.9 Proprietà termiche e termodinamiche utilizzate per il fluido.

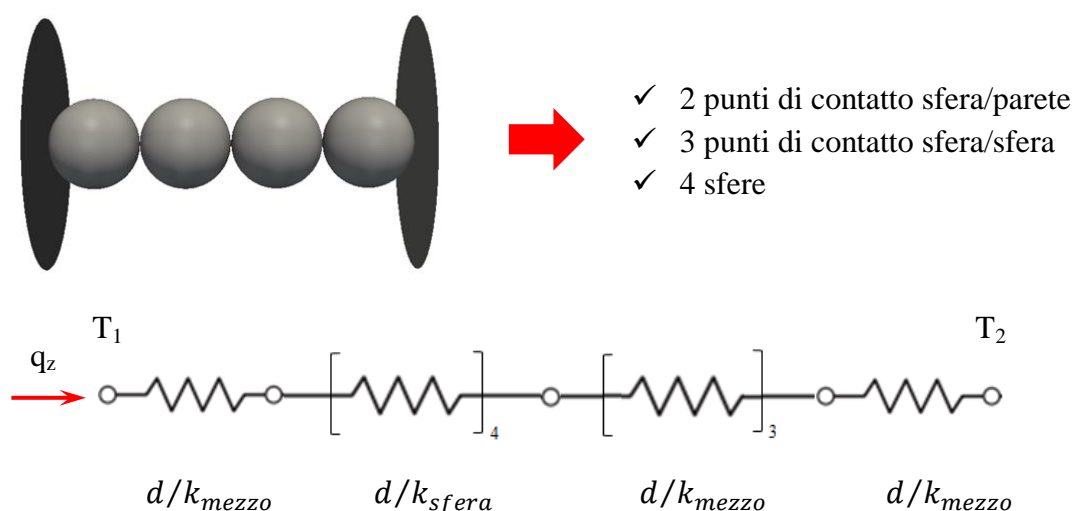
	<i>sfera</i>	<i>bridge sfera/sfera</i>	<i>bridge sfera/parete</i>
$\rho_s$ [kg/m <sup>3</sup> ]	1947	1947	1947
$C_{p_s}$ [J/(kg · K)]	1000	1000	1000
$k_s$ [W/(m · K)]	1	0.0841	0.04205

Tabella 4.10 Proprietà termiche e termodinamiche utilizzate per le regioni solide sfere e bridges.



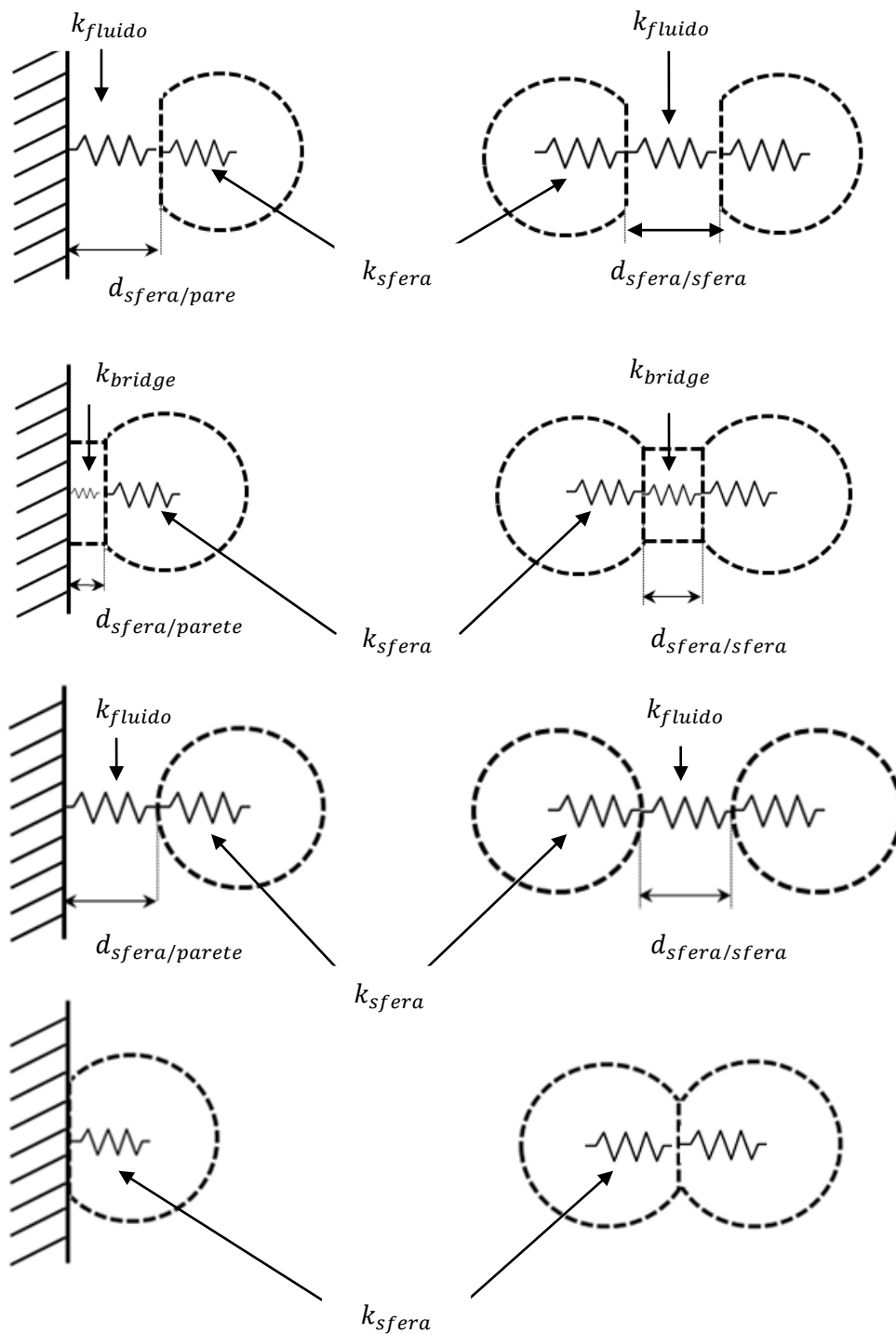
**Figura 4.23** Distribuzione della temperatura all'interno delle sfere per il caso bridge.

Il sistema può essere considerato come un insieme di resistenze termiche disposte in serie [5], dove ciascuna regione contribuisce alla resistenza termica complessiva dell'intero sistema.



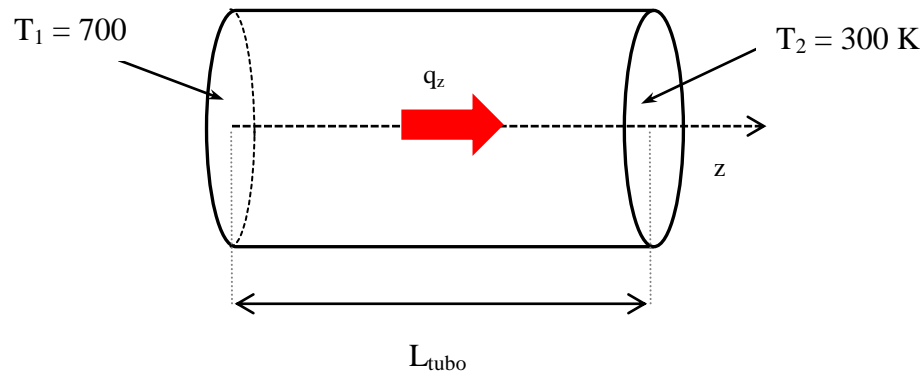
**Figura 4.24** Modello per descrivere il meccanismo di scambio termico per un sistema costituito da 4 sfere in serie applicando le 4 modifiche dei punti di contatto sfera/sfera e sfera/parete.

A seconda del metodo implementato, la regione tra le sfere può essere fluido oppure il bridge con conducibilità termiche assegnate nelle Tabelle 4.9 e 4.10. A ciascuna modifica sfera/sfera e sfera/parete compete una resistenza  $\hat{R}_{sfera/sfera}$  e  $\hat{R}_{sfera/parete}$  rispettivamente, caratterizzate dalle conducibilità termiche e relative distanze mostrate in Figura 4.25.



**Figura 4.25** Modelli utilizzati per modellare i punti di contatto sfera/sfera e sfera/parete per lo studio dello scambio termico.





**Figura 4.26** Discretizzazione del fenomeno di scambio termico lungo la coordinata  $z$  del tubo.

Il flusso termico, secondo la legge di Fourier lungo la coordinata  $z$ , vale:

$$q_z = \frac{(T_1 - T_2)}{\hat{R}_{tot}} \quad (4.2)$$

dove  $T_1$  e  $T_2$  sono le due temperature fissate sulle due pareti del tubo, mentre  $R_{tot}$  è la resistenza totale che incontra il flusso termico nell'attraversare il tubo lungo il tratto  $L_{tubo}$ . La resistenza totale è data dalla somma di ciascuna resistenza presente nel tratto  $L_{tubo}$ . Dopo aver applicato le modifiche globali e locali alle sfere, è possibile scrivere i contributi derivanti dai punti di contatto sfera/parete, dalle resistenze delle  $n$  sfere ed anche dai punti di contatto sfera/sfera, secondo quanto segue:

$$R_{tot} = \sum_i^{ss} \left( \frac{d_{parete/sfera}}{k_{mezzo}} \right)_i + \sum_k^{N_{sfere}} \left( \frac{d_{sfera}}{k_{sfera}} \right)_k + \sum_j^{sp} \left( \frac{d_{sfera/sfera}}{k_{mezzo}} \right)_j \quad (4.3)$$

Per qualsiasi metodo deve ovviamente valere:

$$L_{tubo} = \sum_i^{ss} (d_{parete/sfera})_i + \sum_k^{N_{sfere}} (d_{sfera})_k + \sum_j^{sp} (d_{sfera/sfera})_j \quad (4.4)$$

In questo sistema sono presenti 2 punti di contatto sfera/parete, 4 sfere e 3 punti di contatto sfera/sfera. Pertanto:

$$\hat{R}_{tot} = \left( 2 \cdot \frac{d_{parete/sfera}}{k_{mezzo}} + 4 \cdot \frac{d_{sfera}}{k_{sfera}} + 3 \cdot \frac{d_{sfera/sfera}}{k_{mezzo}} \right) \quad (4.5)$$

La distanza  $d_{sfera}$  è il tratto lungo la coordinata  $z$  che attraversa solo la sfera interessata dopo la modifica applicata.

	$d_{parete/sfera}$ [mm]	$d_{sfera}$ [mm]	$d_{sfera/sfera}$ [mm]
<i>gap</i>	0.065	25.27	0.13
<i>cap</i>	0.26	24.89	0.51
<i>bridge</i>	0.26	24.89	0.51
<i>overlap</i>	0	25.40	0

**Tabella 4.11** Valori delle distanze sfera/sfera e sfera/parete utilizzate dopo la modifica dei punti di contatto.

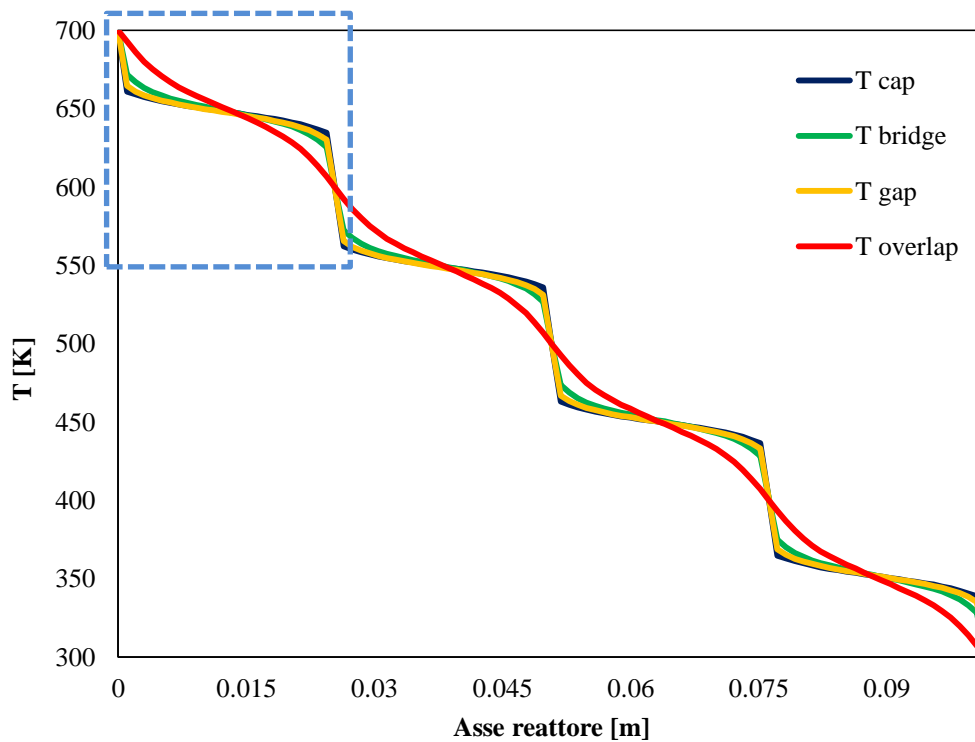
Le 4 diverse resistenze termiche calcolate sono riportate in Tabella 4.12.

	$\hat{R}_{tot}$ [(K · m <sup>2</sup> )/W]	$\hat{R}_{sfera/parete}$ [(K · m <sup>2</sup> )/W]	$\hat{R}_{sfera/sfera}$ [(K · m <sup>2</sup> )/W]
<i>gap</i>	0.122	0.003	0.005
<i>cap</i>	0.185	0.011	0.021
<i>bridge</i>	0.124	0.003	0.006
<i>overlap</i>	0.102	0	0

**Tabella 4.12** Valori delle resistenze totali, resistenza sfera/parete e sfera/sfera calcolate per il sistema studiato.

Il metodo che descrive il fenomeno di scambio termico con la minima resistenza termica possibile è l'*overlap*. La sovrapposizione delle sfere garantisce l'assenza di resistenze termiche tra le sfere e tra le sfere e la parete ed impone una forte conduzione termica che risulta tanto maggiore tanto più alti sono i valori di conducibilità termica delle sfere e le dimensioni delle superfici condivise. Questo si ripercuote sul profilo termico che si sviluppa all'interno delle sfere, come mostrato in Figura 4.27. I metodi di modifica locale sono stati creati imponendo la stessa distanza tra le sfere e sfera e parete. Dalla Tabella 4.12 si dimostra che l'introduzione del

bridge diminuisce la resistenza termica di circa 3.5 volte a causa della conducibilità termica del bridge che è maggiore rispetto a quella del fluido. Avvicinando considerevolmente le sfere tra di loro si riduce la loro distanza, diminuendo così la dissipazione di energia termica verso la regione fluida circostante. Questa piccola distanza è stata assegnata riducendo il diametro delle sfere di circa lo 0.5% rispetto al loro diametro nominale: la distanza tra le sfere è stata ridotta ad un quarto rispetto a quella utilizzata per i metodi di modifica locale.



**Figura 4.27** Profili delle temperature lungo l'asse z del tubo passante per i centri delle sfere.

Pertanto confrontando il *cap* con il *gap method* si può affermare che a parità di conducibilità termica utilizzata il parametro che discrimina lo scambio termico è la distanza tra i corpi solidi che scambiano calore tra loro. Quest'effetto è ben visibile nella Figura 4.28.

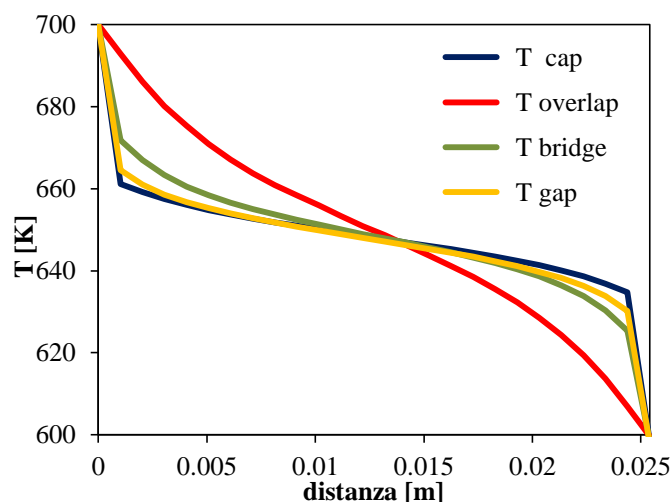


Figura 4.28 Particolare dei salti termici a distanza pari a  $z=25.4 \text{ mm}$  dalla parete mantenuta a  $T_1$ .

Sia il *cap* che il *gap* trasferiscono il calore attraverso la regione fluida che ha un coefficiente di scambio termico di gran lunga inferiore rispetto a quelle delle sfere. La conducibilità termica delle sfere è circa 35 volte maggiore rispetto a quella del fluido. La loro resistenza termica è tanto maggiore tanto più grande è la distanza tra le sfere dopo la modifica effettuata. Riportando in Figura 4.29 e 4.30 i valori raggiunti dalla temperatura sul punto della superficie della sfera 1 e sfera 4 più vicino alle pareti mantenute a  $T_1$  e  $T_2$  rispettivamente, si nota che la distanza ed i coefficienti di conducibilità termica del fluido e *bridge* giocano un ruolo importante nel caratterizzare i valori di temperatura raggiunti da entrambe le sfere.

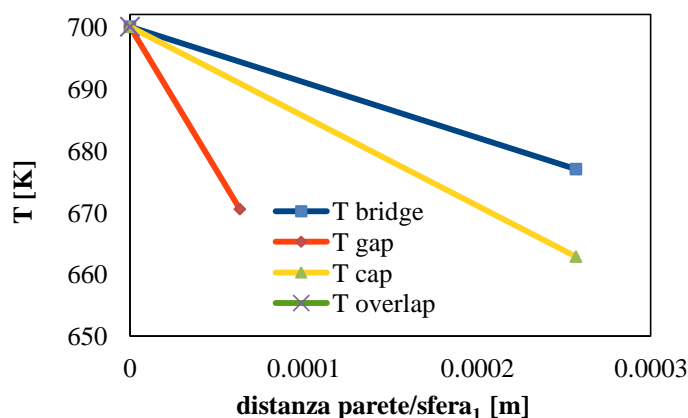
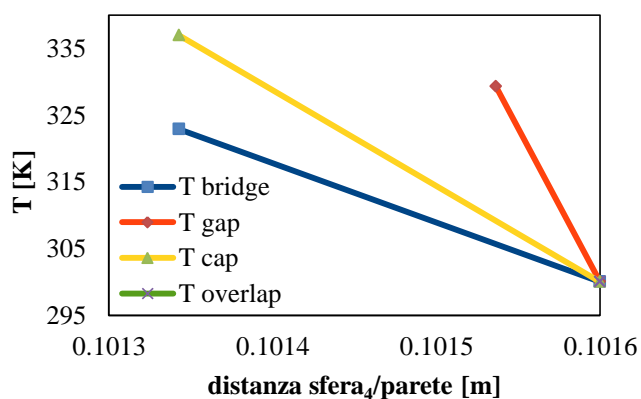


Figura 4.29 Valori di temperatura raggiunti per i 4 metodi sulla superficie della sfera 1 che si trova vicino la parete mantenuta a  $T_1$ .



**Figura 4.30** Valori di temperatura raggiunti per i 4 metodi sulla superficie della sfera 4 che si trova vicino la parete mantenuta a  $T_2$ .

Come d'accordo con i risultati ottenuti in Tabella 4.12 l'ordine delle resistenze termiche calcolate rispetta i valori delle temperature raggiunte riportate in Figura 4.30. L'*overlap* con resistenza nulla ha un valore di temperatura superiore rispetto agli altri, pari a quella imposta sulla parete. Il *bridge* che ha una resistenza termica inferiore rispetto al *cap*, ma superiore rispetto all'*overlap*, garantisce un valore più elevato di temperatura sulla superficie della sfera vicino la parete rispetto a quello raggiunto dalle sfere con un *cap* tra la sfera e parete. Il *gap* ed il *bridge* hanno delle resistenze termiche approssimativamente uguali anche se la distanza tra la sfera e la parete per il *bridge* è pari a circa 4 volte rispetto a quella del *gap*. L'analisi svolta sulla determinazione delle resistenze termiche può essere pienamente utilizzata per confrontare i casi generati con le modifiche locali dei punti di contatto, ma non risulta appropriata per confrontare in maniera quantitativa gli approcci di modifica locale con quelli di modifica globale, perché tali metodi intervengono in maniera diversa sulla sfera per eliminare i punti di contatto sfera/sfera e sfera/parete.

## 4.4 Conclusioni

I metodi di modifica locale sono stati così implementati e attentamente confrontati per capire quale dei due approcci (*cap* o *bridge*) è il più appropriato per modellare punti di contatto sfera/sfera e sfera/parete per casi studio in presenza di una reazione chimica. Le prove sono state svolte con il flusso convettivo dei reagenti in ingresso al

---

reattore diretto come i diametri delle sfere a differenza dello studio descritto nel Capitolo 3. In questo modo così è stata studiata anche la seconda configurazione limite in cui i punti di contatto sfera/sfera si trovano lontano rispetto al campo di moto periferico interessato da forti flussi convettivi. Pertanto, per verificare se i punti di contatto sfera/sfera sono delle zone di ristagno anche a regimi fluidodinamici più spinti, le analisi sono state condotte a diversi numeri di Reynolds ed i risultati ottenuti hanno evidenziato che se a bassi numeri di Reynolds il trasporto di materia dalla fase fluida alla superficie catalitica nelle zone interessate dalla modifica locale tra le sfere è principalmente governato dai flussi diffusivi, ad alti numeri di Reynolds invece anche il flussi convettivi giocano un ruolo importante garantendo zone di ricircolo e quindi maggiori tempi di contatto dei reagenti rispetto alle zone più periferiche. È stato osservato che la modifica della superficie delle sfere gioca un ruolo significativo sulla stima della conversione dei reagenti interessati alla reazione catalitica superficiale. Valori del *cap* compresi tra  $0.005 \cdot d_s$  e  $0.03 \cdot d_s$  garantiscono la minima variazione possibile di errore percentuale di conversione di ossigeno di circa il 1.65% rispetto al caso di riferimento per bassi valori di Reynolds. Il *bridge method* offre delle diminuzioni maggiori di area catalitica rispetto al *cap* e per valori del raggio del *bridge* tra  $0.05 \cdot d_s$  e  $0.15 \cdot d_s$  si ottengono errori contenuti tra 2.5 e 3% nelle stesse condizioni operative del *cap*. Aumentando il numero di Reynolds si ha una riduzione della conversione poiché diminuisce il tempo di contatto delle specie chimiche nel reattore, così il campo di moto in prossimità dei punti di contatto sfera/sfera diventa più importante e la scelta dell'approccio per modificare i punti di contatto gioca un ruolo discriminante. A valori più alti del numero di Reynolds gli errori percentuali della conversione aumentano considerevolmente per il *bridge* ed in maniera più moderata per il *cap method* come si vede dai risultati ottenuti nelle Tabelle 4.6 e 4.7. La configurazione geometrica proposta impone che i punti di contatto sfera/parete siano pienamente coinvolti dalla reazioni chimica superficiale in ragione del flusso convettivo che si trova costretto ad attraversare quelle zone. Se da un lato il *cap method* garantisce dei risultati più attendibili rispetto al *bridge method*, dall'altro la generazione della mesh per un caso *cap* richiede un maggior numero di celle in ragione della regione fluida che si trova tra le sfere e sfera/parete dopo la

modifica. Pertanto la generazione della mesh per un sistema con un maggior numero di sfere con il *cap method* sicuramente richiede un numero maggiore di celle rispetto al *bridge method*. Ciò non toglie che a spese computazionali superiori il *cap method* garantisce comunque dei risultati più accurati rispetto al *bridge*.

Parallelamente lo studio dello scambio termico ha dimostrato che il modello Dixon *et al.* [4] è il più appropriato per descrivere lo scambio termico tra sistemi di sfere con punti di contatto sfera/sfera e sfera/parete. Attraverso un'analisi fatta introducendo delle resistenze termiche fittizie per punti di contatto sfera/sfera e sfera/parete per i 4 metodi di modifica si è osservato che per i metodi di modifica locale il confronto tra rapporto distanza sfera/sfera e conducibilità termica del mezzo è fondamentale per comprendere il meccanismo di scambio termico. Utilizzando il modello di Dixon *et al.* [4] si hanno bassi valori della resistenza termica, a causa della più alta conducibilità termica del mezzo tra le sfere (che è appunto il *bridge*). All'aumentare della distanza tra le sfere a causa del *cap* si riscontrano elevate resistenze termiche con conseguenti aumenti delle dissipazioni di flusso termico nella regione fluida circostante. Per un sistema costituito da più sfere (ad esempio un letto impaccato di sfere, dove la reazione chimica è fortemente esotermica) la modellazione dei punti di contatto è critica. Generalmente in casi industriali per reazioni fortemente esotermiche la parete del reattore deve essere mantenuta a valori di temperatura più bassi rispetto alla temperatura di reazione per garantire una certa differenza di temperatura. Pertanto è critica la modellazione dei punti di contatto sfera/parete che è la zona interessata dai forti gradienti termici. Grazie ai risultati ottenuti nel Capitolo 3, da cui si evince che il modello Dixon *et al.* [4] garantisce la minima dissipazione possibile di calore nella regione fluida circostante e ai risultati ottenuti in questo Capitolo, è possibile affermare che il metodo *bridge* è il più soddisfacente per garantire una corretta modellazione del sistema considerato.

Il prossimo studio sarà soggetto di confronto dei due metodi di modifica locale dei punti di contatto sfera/sfera e sfera/parete per un sistema semplice costituito da due sfere in presenza di una reazione chimica superficiale esotermica. I risultati ottenuti in questo Capitolo per entrambi i fenomeni di scambio di materia e scambio termico verranno così utilizzati per studiare sistematicamente questo nuovo caso studio.

## Bibliografia

- [1] Maestri M., e Cuoci A., *Coupling CFD with detailed microkinetic modeling in heterogeneous catalysis*,. Chemical Engineering Science, 96, pp. 106-117, (2013).
- [2] Maestri M., Cuoci A., [www.catalyticfoam.polimi.it](http://www.catalyticfoam.polimi.it).
- [3] <http://www.openfoam.com/>
- [4] Dixon A.G., Nijemeisland M., Stitt E.H., “*Systematic mesh development for 3D CFD simulation of fixed beds: Contact points study*”, Computers and Chemical Engineering, 48, pp. 135-153, (2013).
- [5] Incropera F.P., DeWitt D.P., Bergman T.L. Lavine A.S., “*Fundamentals of Heat and Mass Transfer*”, 7th Edition, John Wiley, (2011).



# 5 Confronto dei metodi di modifica locale su un caso sfera/sfera in presenza di una reazione chimica superficiale eterogenea esotermica

## 5.1 Introduzione

I fenomeni di scambio di materia affrontati nei precedenti Capitoli sono stati condotti a temperatura costante e uniforme ed hanno dimostrato che per studi CFD in presenza di una reazione chimica catalitica sulla superficie delle sfere poste a contatto tra loro la soluzione migliore è quella di modificare i punti di contatto attraverso il *cap method*, in grado di offrire la minima variazione possibile di area catalitica. Gli studi svolti sullo scambio termico sono stati fatti imponendo la condizione *no slip* per la velocità sulle pareti del tubo e sulla superficie delle sfere ed è stato visto che il modello elaborato da Dixon *et al.* [1] descrive meglio i fenomeni di scambio termico rispetto agli altri metodi di modifica dei punti di contatto. Uno studio sistematico sullo scambio termico in condizioni di regime laminare e turbolento è stato eseguito da Dixon, Nijemeisland e Stitt per un caso semplice sfera/sfera e sfera/parete. Questi autori hanno studiato l'effetto dei differenti approcci di modifica dei punti di contatto sul campo di moto locale e sullo scambio termico. In particolare, Dixon *et al.* hanno dimostrato che il *gap method* fornisce dei profili di temperatura non realistici, mentre l'*overlap* per la presenza di superfici di contatto offre elevati flussi termici tra le sfere. Gli autori pertanto, raccomandano di sostituire

il punto di contatto con il bridge con rapporto raggio bridge su diametro sfera di  $r_b/d_s \leq 0.1$  per valori di Reynolds  $\leq 2000$  e valori più piccoli di  $r_b/d_s \leq 0.05$  in condizione di regime turbolento,  $Re \geq 2000$ . Lo studio svolto dagli autori non è stato esteso per casi in presenza di una reazione chimica catalitica eterogenea. La reattività chimica accoppiata allo scambio termico per effetto del calore di reazione è una situazione tipica dei processi catalitici industriali odierni. Gentile e Manelli [2] nel loro lavoro di Tesi hanno svolto degli studi CFD con reazione catalitica intraparticellare su un letto impaccato di sfere in condizioni adiabatiche. Hanno dimostrato che il livello di *refinement* della mesh strutturata è fondamentale per descrivere in maniera accurata lo scambio termico tra la parete del reattore e le sfere con esso a contatto. Infatti un basso livello di *refinement* fornisce una *mesh* con elevati valori dell'indice di *skewness* e conseguentemente dei picchi di temperatura non realistici in prossimità della parete. Sebbene questo problema sia stato risolto rendendo la *mesh* più fitta nelle zone tra la parete e le sfere, lo sforzo computazionale richiesto per eliminare i bridges tra le sfere è talmente notevole da rendere quasi impossibile la loro eliminazione completa.



**Figura 5.1** Sezione del letto impaccato di sfere lungo la lunghezza del reattore [2].

Questo problema può essere risolto utilizzando una *mesh* di tipo non strutturata che, come visto nel Capitolo 2, si adatta molto meglio alla superficie geometrica, specialmente se esistono molti dettagli geometrici, e modificando opportunamente i punti di contatto tra le sfere per evitare una bassa qualità della *mesh*. I metodi di

modifica locale sembrano essere i più appropriati per modellare accettabilmente i punti di contatto. In questo Capitolo verranno confrontati il *cap method* e il *bridge method* per un sistema costituito da due sfere catalitiche. Tale studio prevede una reazione catalitica eterogenea superficiale con sviluppo di calore di reazione e quindi propagazione del calore nelle regioni solide e nel fluido circostante. I due metodi così applicati verranno confrontati e successivamente si discuteranno dei limiti di applicabilità dei metodi proposti. Le simulazioni sono state condotte con il solver dinamico `catalyticConductivePimpleFoam` [3].

## 5.2 Creazione del caso: dominio geometrico e generazione della mesh

Le due sfere disposte all'interno del reattore sono a contatto tra loro come mostrato in Figura 5.2.

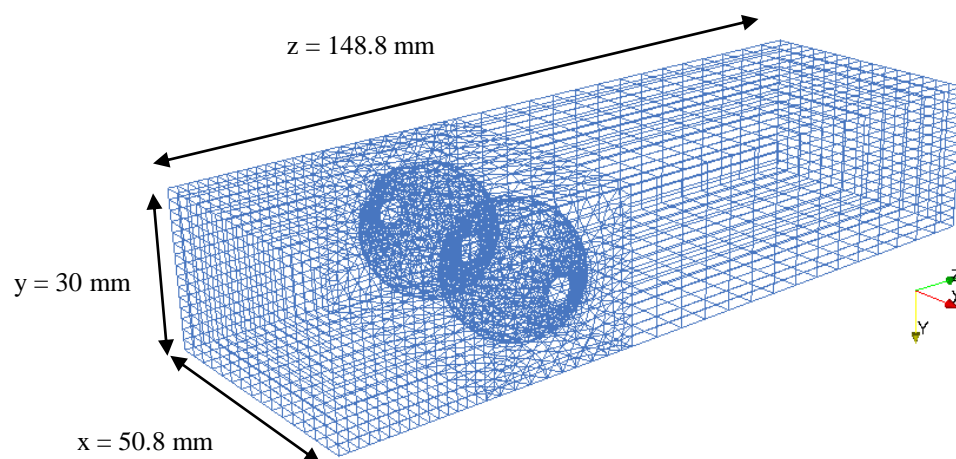
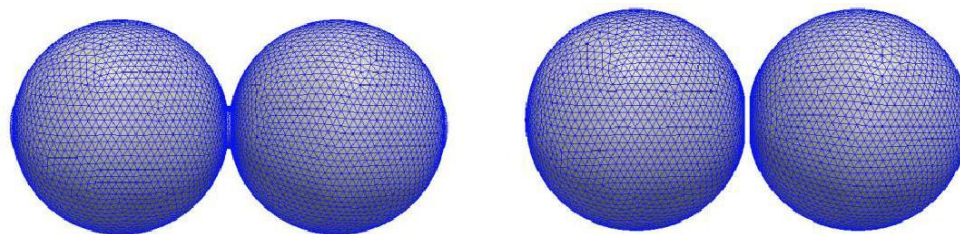


Figura 5.2. Mesh ibrida 3D per i casi *cap method* e *bridge method*

Nella geometria proposta esistono due punti di contatto sfera/parete ed un punto di contatto sfera/sfera.

In accordo con i risultati ottenuti nei precedenti Capitoli sono stati utilizzati i due metodi di modifica locale: il *cap* ed il *bridge*.



**Figura 5.3. Modifica dei punti di contatto sfera/sfera e sfera/parete con *bridges* e *caps* rispettivamente.**

I valori utilizzati sono riportati in Tabella 5.1.

	<i>bridges</i>	<i>caps</i>
<b>metodo</b>	<b><math>r_b/d_s</math> [-]</b>	<b><math>2h/d_s</math> [-]</b>
<i>sfera/sfera</i>	0.1	0.0202
<i>sfera/parete</i>	0.1	0.0202

**Tabella 5.1 Valori utilizzati per modificare i punti di contatto sfera/sfera e sfera/parete con i metodi *bridge* e *cap*.**

I valori delle dimensioni del *cap* e *bridge* utilizzati rientrano in quelli raccomandati nei Capitoli 3 e 4 che garantiscono minime variazioni possibili di conversione rispetto al caso ideale. Il numero di Reynolds per questa simulazione è pari a 70 circa. Pertanto anche per lo scambio termico secondo quanto riportato dal lavoro di Dixon *et al.* [1] il valore di  $r_b/d_s$  pari a 0.1 è accettabile. L'approccio utilizzato per creare il *cap* ed il *bridge* è lo stesso a quello descritto nel Capitolo 3. Di seguito in Tabella 5.2 si riportano i valori caratteristici delle *mesh* ottenute per entrambi i metodi.

	<b>N° di celle</b>	<b>MAX skewness</b>	<b>MAX non-ortogonalità</b>	<b>MAX Aspect Ratio</b>
<i>cap</i>	984556	0.649	56.589	5.52
<i>bridge</i>	887124	0.561	63.156	4.89

**Tabella 5.2 Valori caratteristici delle mesh ottenute per il *cap* ed il *bridge*.**

Le condizioni operative utilizzate sono riportate in Tabella 5.3:

<i>Specie</i>	<i>Frazione Massiva</i>	<i>condizioni operative</i>
H <sub>2</sub>	$5.3721 \cdot 10^{-4}$	Da = 0,015 kcin = $53.59 \cdot 10^{-6}$
O <sub>2</sub>	$2.993 \cdot 10^{-2}$	P = 101325 Pa
N <sub>2</sub>	$9.7628 \cdot 10^{-1}$	T = 373.15 K

**Tabella 5.3** Condizioni operative utilizzate per il caso studio.

Lo studio è stato condotto in condizioni adiabatiche, imponendo una temperatura al fluido in ingresso di 373.15 K. Le proprietà fisiche del fluido e delle regioni solide utilizzate sono riportate nelle Tabelle 5.4 e 5.5.

<i>Proprietà del fluido</i>			
$\rho_f$ [kg/m <sup>3</sup> ]	C <sub>p</sub> [J/(kg · K)]	$\mu$ [Pa · s]	k <sub>f</sub> [W/(m · K)]
1.225	1006.43	$1.789 \cdot 10^{-5}$	0.0242

**Tabella 5.4** Proprietà fisiche del fluido utilizzate.

	$\rho_s$ [kg/m <sup>3</sup> ]	C <sub>p</sub> [J/(kg · K)]	k <sub>s</sub> [W/(m · K)]
<i>sfera</i>	1947	1000	1
<i>bridges sfera/sfera</i>	1947	1000	0.0842
<i>bridges sfera/parete</i>	1947	1000	0.04205

**Tabella 5.5** Proprietà fisiche delle regioni solide utilizzate.

## 5.3 Risultati e discussioni

Di seguito si riportano i risultati ottenuti per questa simulazione.

metodo	area catalitica [cm <sup>2</sup> ]	conversione % O <sub>2</sub>
<i>bridge</i>	39.815	25.83
<i>cap</i>	40.426	26.05

**Tabella 5.6** Valori di conversione percentuale di O<sub>2</sub> in funzione dell'area catalitica offerta dai due metodi di modifica locale.

Utilizzando il *bridge method* si nota come la variazione di area catalitica comporta una sottostima della conversione. Le Figure 5.5 e 5.6 mostrano le frazioni massive delle specie chimiche sviluppatasi lungo il reattore per entrambi i metodi di modifica.

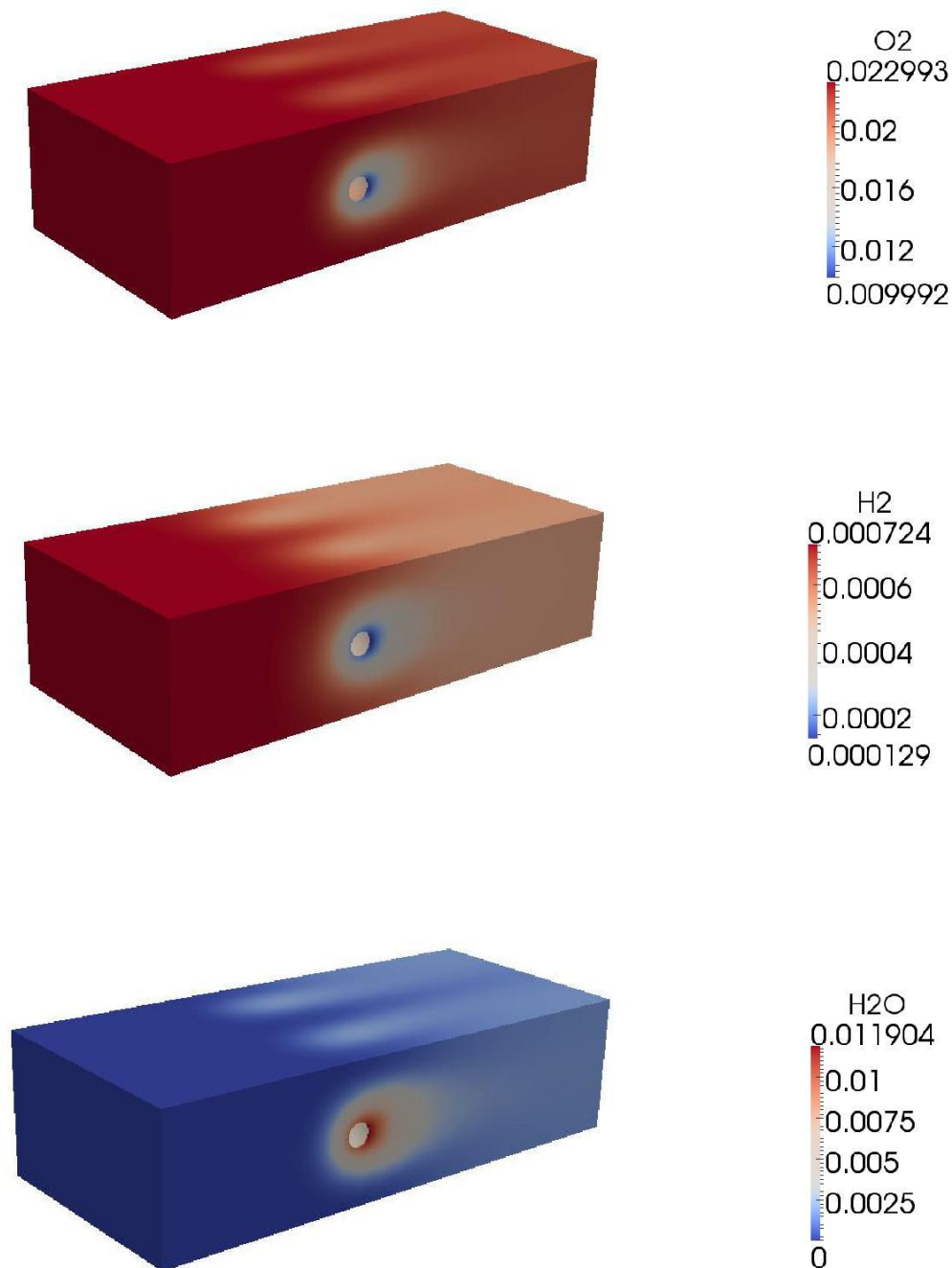
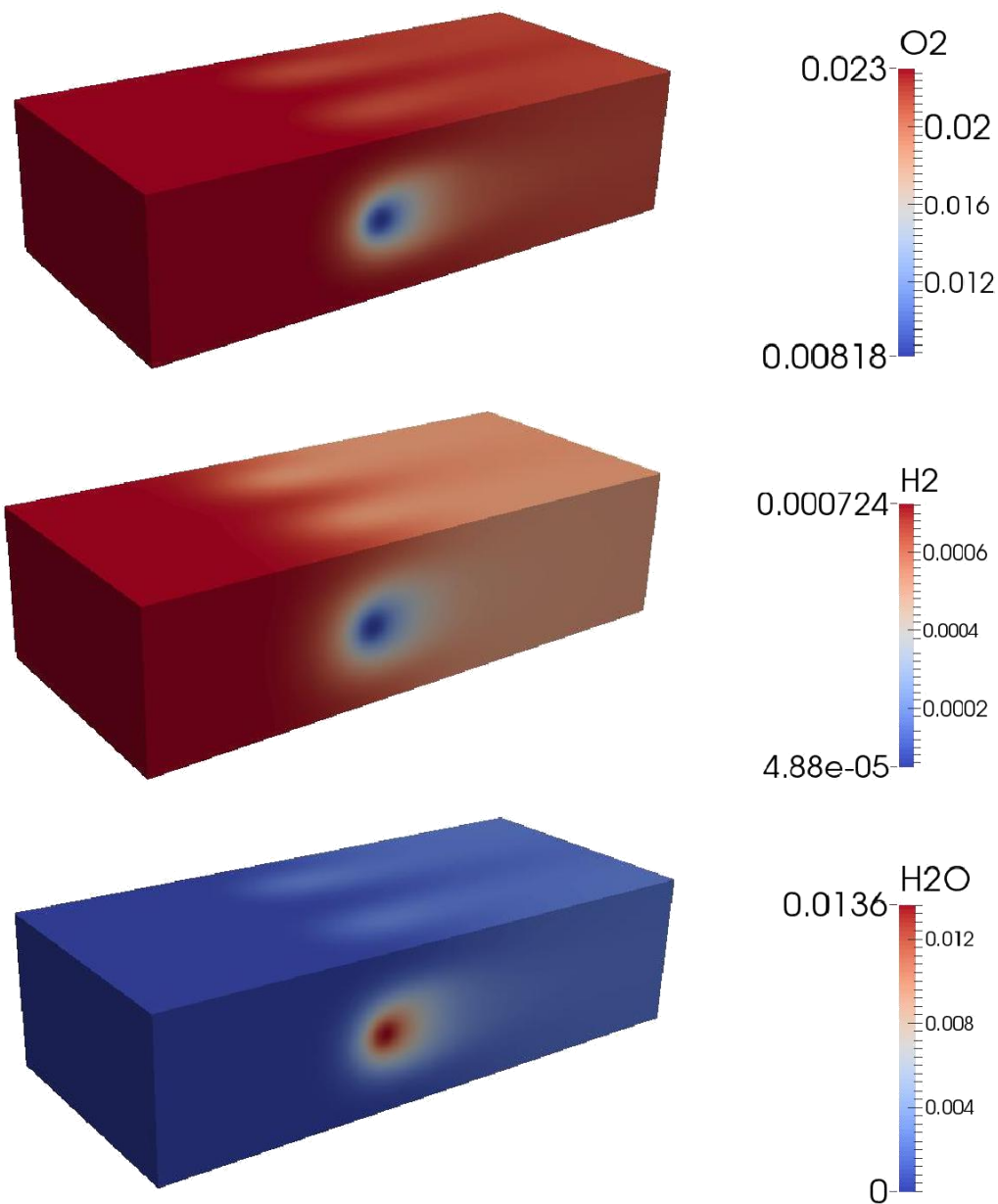


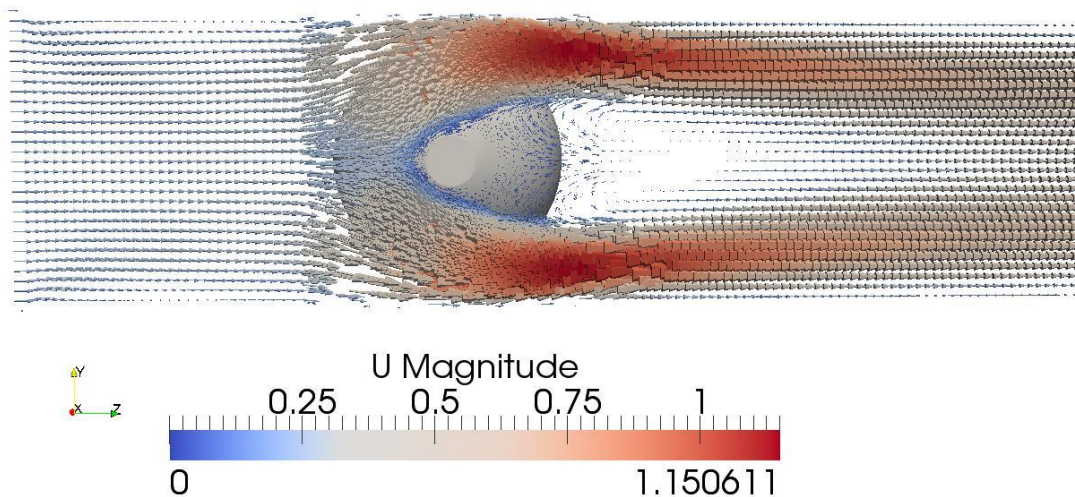
Figura 5.4 Frazioni massive delle specie chimiche lungo il reattore per il *bridge method*.



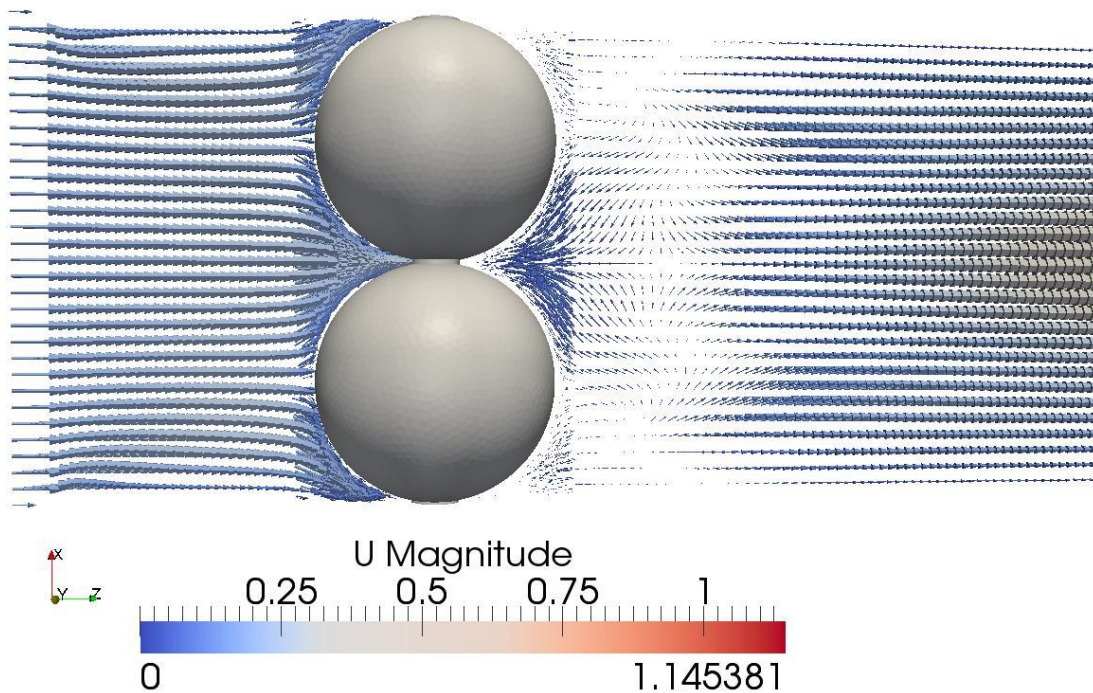
**Figura 5.5** Frazioni massive delle specie chimiche lungo il reattore per il *cap method*.

La geometria impone una fluidodinamica tipica di un flusso che si sviluppa attorno a due sfere stazionarie poste a contatto. A valle delle sfere si generano dei vortici e quindi degli effetti di *backflow* del fluido, come mostrato nelle Figure 5.7 e 5.8.

Le linee di flusso interagiscono con le sfere creando così delle zone di ristagno a valle le sfere. In tali zone i tempi di contatto sono considerevolmente più alti rispetto alla zona direttamente coinvolta dal flusso convettivo del fluido.



**Figura 5.6** Campo di moto del fluido lungo il piano yz passante per il centro del bridge sfera/sfera.



**Figura 5.7** Campo di moto del fluido lungo il piano xz passante per il centro del bridge sfera/sfera.



La presenza di vortici laminari garantisce un notevole consumo di reagenti in ragione degli elevati tempi di contatto soprattutto nelle zone dove i punti di contatto sono stati modificati e quindi un maggiore sviluppo di calore.

Questo dimostra la più alta temperatura rispetto alle zone a monte delle sfere.

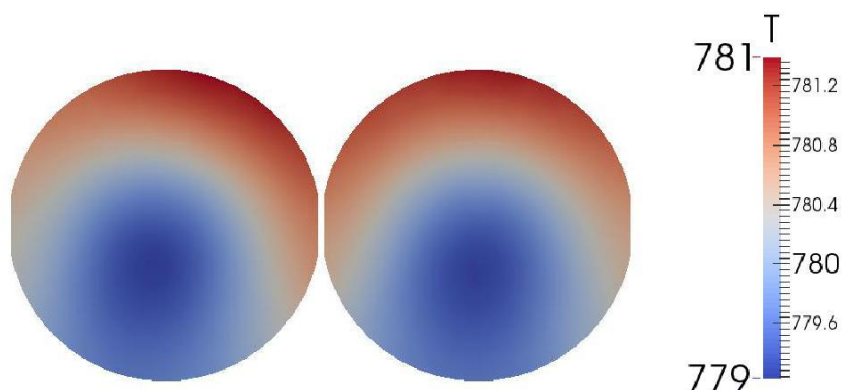


Figura 5.8 Distribuzione della temperatura all'interno delle regioni solide per il *cap*.

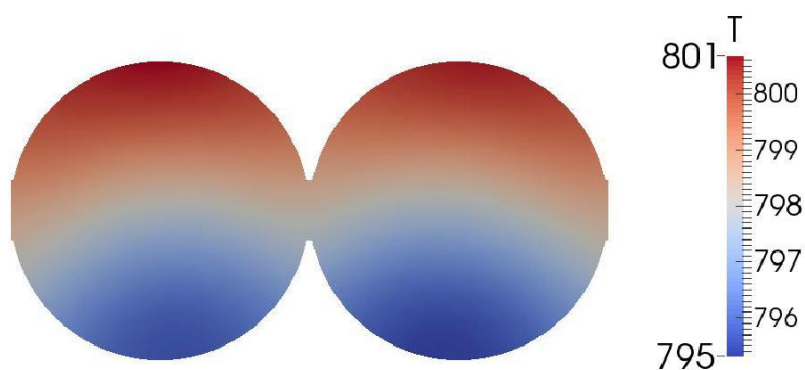
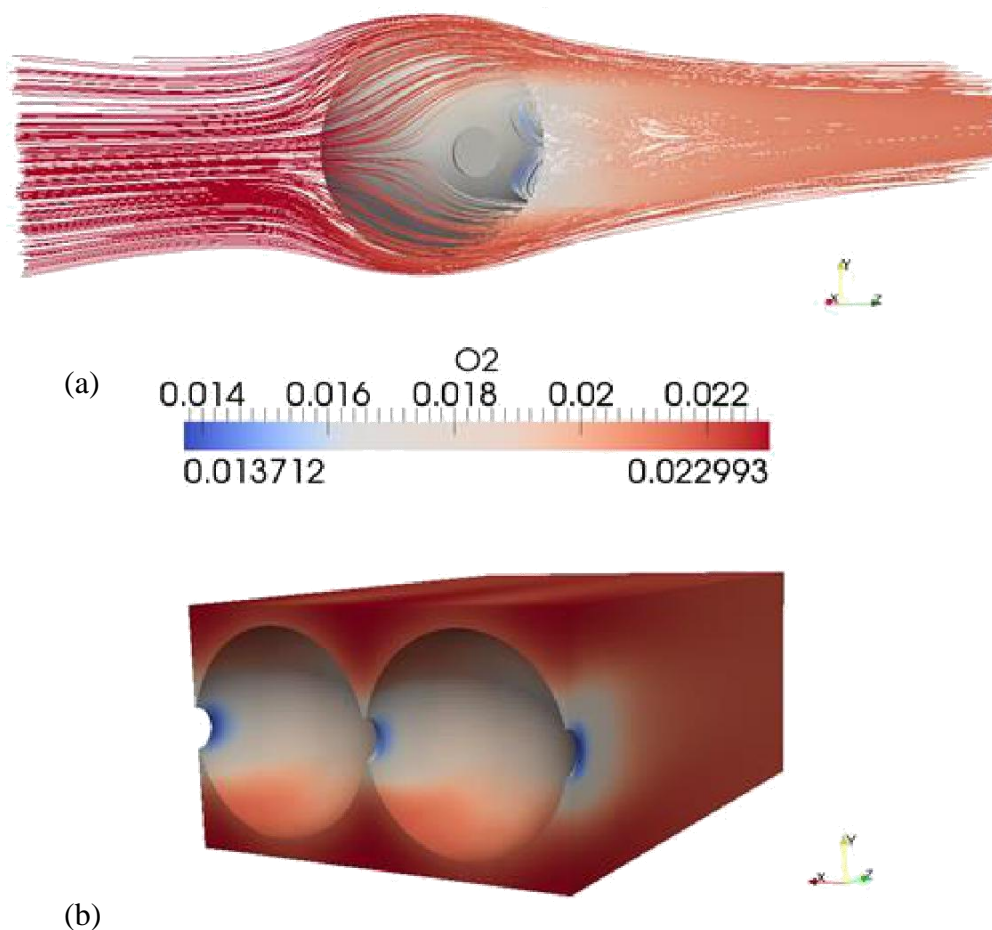


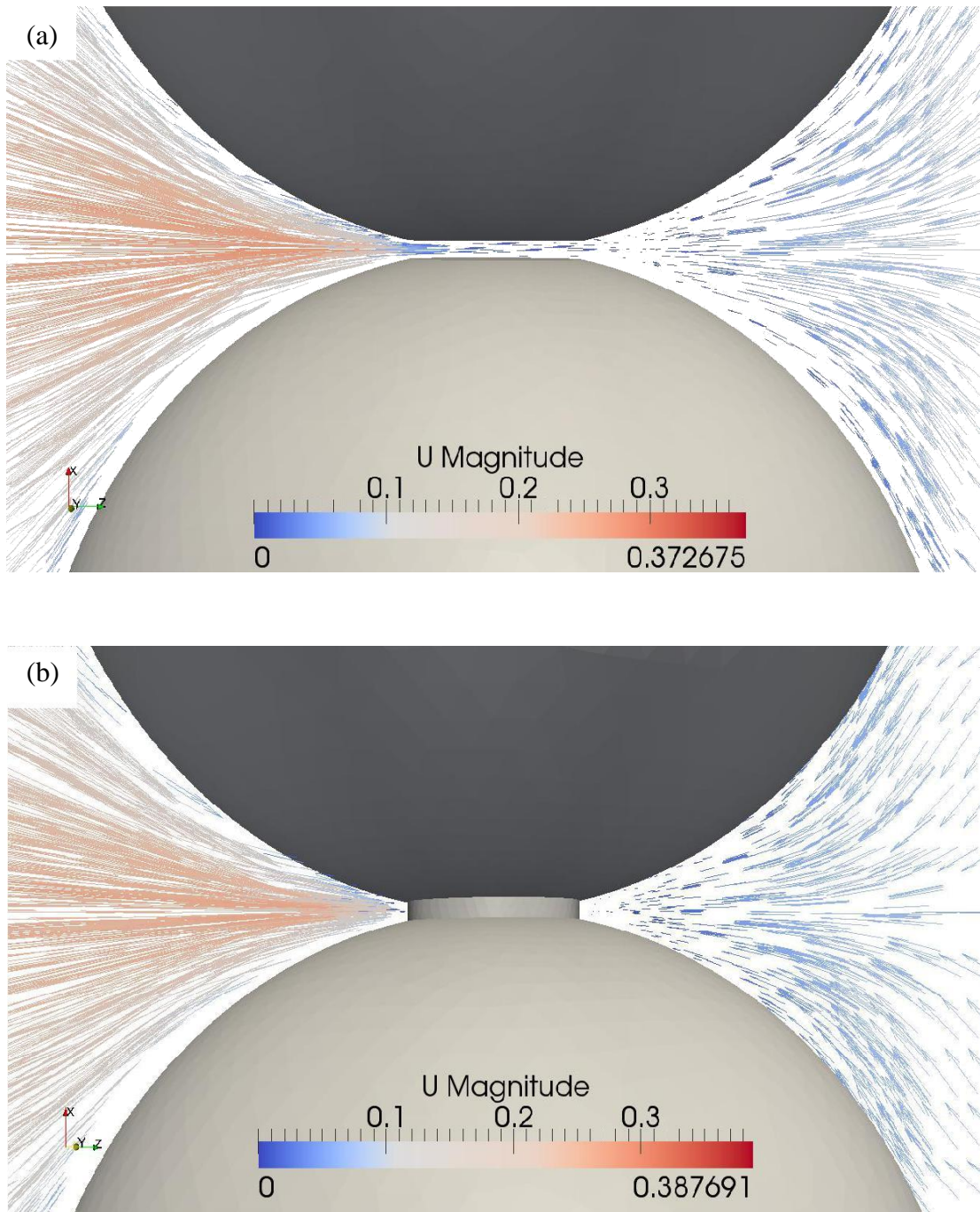
Figura 5.9 Distribuzione della temperatura all'interno delle regioni solide per il *bridge*.

Pertanto le zone interessate dalle modifiche dei punti di contatto prendono pienamente parte alla reazione catalitica superficiale e si nota dalla Figura 5.11 (b) il consumo di ossigeno attorno ai *bridges*.



**Figura 5.10** Consumo di ossigeno in prossimità dei *bridges*.

Una giustificazione dei diversi valori di conversione ottenuti per i due approcci è stata data più volte: l'area catalitica offerta dai diversi approcci di modifica dei punti di contatto ha effetti sulla stima della conversione delle specie chimiche. In Figura 5.12 si riportano i campi di moto locali per i due casi *cap* e *bridge*. Si nota che il flusso convettivo che investe le sfere è di gran lunga maggiore in modulo rispetto al flusso che si genera dietro le sfere per effetto dei vortici. Nelle zone del *cap* e del *bridge* esistono pertanto delle zone di ristagno dove i fenomeni di scambio di materia sono principalmente dovuti ai flussi diffusivi molto piccoli a confronto con il campo di moto convettivo lontano dai punti contatto.



**Figura 5.11** Campo di moto in prossimità della modifica effettuata tra le sfere per il cap (a) ed il bridge (b).

Questo effetto fluidodinamico si ripercuote pertanto sui valori di conversione ottenuti. Per comprendere meglio le modalità di scambio termico tra le regioni solide ed il fluido risulta conveniente analizzare i profili di temperatura ottenuti lungo una

linea passante per i due diametri delle sfere (nella direzione  $x$ ) riportati nelle Figure 5.13 e 5.14. Il trasferimento di calore tra i corpi solidi e la regione fluida avviene attraverso diversi meccanismi. La conduzione tra le regioni solide avviene tra le superfici di contatto, mentre per il cap attraverso i filetti di fluido vicino i punti di contatto.

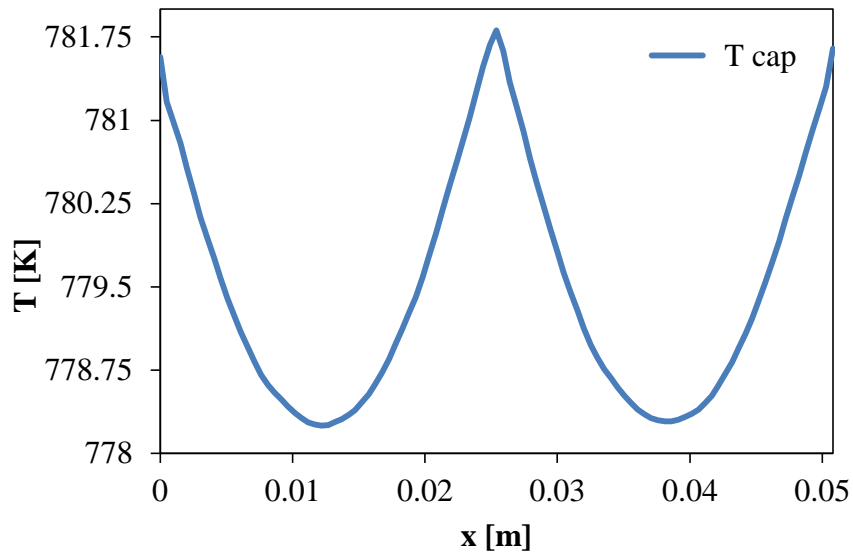


Figura 5.12 Profilo di temperatura all'interno delle sfere per il *cap method*.

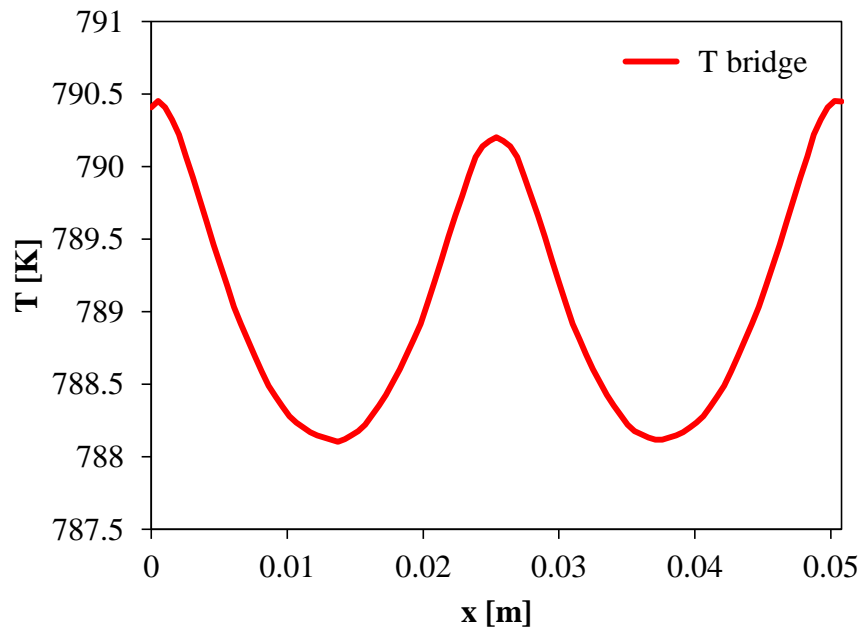


Figura 5.13 Profilo di temperatura all'interno delle sfere per il *bridge method*.

I due approcci influenzano considerevolmente lo stato termico delle due sfere. Dall'analisi delle Figure 5.13 e 5.14 è possibile fare interessanti osservazioni. Il *bridge* sfera/sfera si trova a contatto con due corpi solidi (le sfere) ed il fluido circostante rispetto al *bridge* sfera/parete che invece scambia calore con una sola sfera e la regione fluida. Inoltre la superficie laterale esposta è pari al doppio rispetto a quella del *bridge* sfera/parete, e questo si ripercuote sulla velocità dello smaltimento del calore di reazione diversa per entrambi i casi.

Sia il contributo di scambio di calore attraverso le superfici laterali del *bridge* che quello conduttivo attraverso le superfici di contatto si riflettono sui valori di temperatura ottenuti nei punti dove le modifiche sono state applicate. Un valore di temperatura più elevato in prossimità della parete si giustifica attraverso la velocità di smaltimento del calore di reazione, che è più lenta nella zona vicino la parete poiché, avendo imposto la condizione `zeroGradient` per la temperatura sulle pareti del reattore, il *bridge* sfera/parete è costretto a scambiare calore solo con la sfera adiacente e la regione fluida per conduzione termica.

Il *cap* offre invece un'altra distribuzione della temperatura. Dalla Figura 5.13 si nota che la temperatura più alta si trova per  $x=25.4\text{ mm}$  dove è stato applicato il *cap* tra le sfere. Entrambe le facce del *cap* si trovano ad una distanza pari a  $0.513\text{ mm}$ , e scambiano calore di reazione solo attraverso il filetto di fluido che le separa, in entrambe le zone sia tra la parete e la sfera che tra le sfere il fluido è stagnante pertanto lo scambio termico avviene sostanzialmente per conduzione ed il contributo di entrambe le sfere in quel punto garantisce una più alta temperatura rispetto a quella che si raggiunge nella modifica sfera/parete. Un'altra importante osservazione sono i salti termici che si sviluppano nelle zone sfera/sfera e sfera/parete: in accordo con quanto visto nel Capitolo 3, il *cap* fornisce dei salti termici molto pronunciati e non realistici soprattutto tra la parete e le sfere.

## 5.4 Conclusioni

I metodi locali di modifica dei punti di contatto sono stati confrontati tra loro per un caso semplice sfera/sfera. Questo studio ha dimostrato che sebbene il *cap method*

---

abbia risolto il problema eliminando celle con alti valori di *skewness* nei punti di contatto, comunque non è il metodo più appropriato per modellare correttamente i punti di contatto tra le sfere per casi di reazioni chimica catalitica superficiale in presenza di scambio termico. Infatti la presenza del filetto di fluido tra le sfere non garantisce l'integrale trasferimento di calore tra le sfere con conseguenza di dissipazione di calore di reazione nella zona fluida. La discontinuità dei valori dei coefficienti di conducibilità termica tra le regioni solide e il fluido provoca dei salti termici poco realistici e quindi una non corretta distribuzione della temperatura all'interno delle sfere. Si suggerisce di utilizzare il *bridge method* con valori del rapporto  $r_b/d_s \leq 0.1$ . Inoltre considerando che l'applicazione dei bridges provoca una diminuzione dell'area catalitica che per un sistema costituito da tante sfere influisce significativamente è consigliabile utilizzare dei *bridges* con valori più piccoli di 0.1 per garantire la minima variazione possibile di area catalitica. Qualora la reazione chimica sia condotta all'interno del reattore in condizioni isoterme allora è preferibile utilizzare il *cap method* visto che garantisce la minima variazione possibile di area catalitica.

## Bibliografia

- [1] Dixon A.G., Nijemeisland M., Stitt E.H., “*Systematic mesh development for 3D CFD simulation of fixed beds: Contact points study*”, Computers and Chemical Engineering, 48, pp. 135-153, (2013).
- [2] Gentile G. e Manelli F., “*An efficient computational framework for the advanced modeling and design of industrial catalytic reactors*”, Tesi di Laurea Magistrale in Ingegneria Chimica, Politecnico di Milano, (2013).
- [3] Maestri M. e Cuoci A., [www.catalyticfoam.polimi.it](http://www.catalyticfoam.polimi.it).

# 6 Analisi sistematica di reattori impaccati di sfere

## 6.1 Introduzione

La catalisi ed i processi catalitici sono degli aspetti di grande importanza nell'Ingegneria Chimica. La maggior parte delle reazioni catalitiche di interesse industriale vengono condotte in reattori eterogenei a letto fisso.

Un reattore eterogeneo a letto fisso (o semplicemente reattore impaccato) può essere definito come uno strato o un insieme di particelle catalitiche arrangiate all'interno di un vessel che promuove il contatto tra le particelle catalitiche ed i reagenti che si trovano nella fase fluida [1].

Attraverso l'utilizzo della CFD, i reattori impaccati possono essere descritti in maniera dettagliata. Uno degli attuali limiti per la descrizione dei reattori attraverso la CFD è quello di modellare in maniera dettagliata reattori di interesse industriale a causa delle elevate risorse computazionali richieste. È molto importante trovare un ottimo compromesso tra accuratezza dei risultati e tempi computazionali richiesti per garantire una soddisfacente conduzione della simulazione. Normalmente, dove è possibile, vengono studiate porzioni di letti impaccati per indagare, ad esempio, l'effetto della dimensione e forma del pellet, oppure l'effetto delle modifiche dei punti di contatto tra le particelle catalitiche sulla conversione dei reagenti in prodotti. Lo studio svolto nei precedenti Capitoli ha dimostrato l'effetto della modifica dei punti di contatto su simulazioni CFD in presenza di una reazione chimica superficiale e fenomeni di scambio termico per configurazioni geometriche molto semplici. Lo studio è stato fondamentale per caratterizzare il meccanismo dei fenomeni di scambio di materia e di calore tra le sfere. In questo Capitolo si



affronteranno delle analisi volte a caratterizzare i letti impaccati di sfere, utilizzando i 4 metodi descritti nel Capitolo 1 per modificare i punti di contatto.

I 4 *script* che consentono l'implementazione dei 4 metodi di modifica dei punti di contatto per i casi semplici sfera/sfera sono stati così sviluppati per modificare i punti di contatto sfera/sfera e sfera/parete in letti impaccati con  $n$  sfere. Una volta ottenute le modifiche con le dimensioni volute è stato possibile confrontare l'effetto delle modifiche sulle caratteristiche geometriche dei letti impaccati.

Lo scopo di questo Capitolo è quello di illustrare delle metodologie in grado di modellare letti impaccati con il metodo di modifica che garantisca le minime variazioni possibili delle proprietà geometriche, rispetto al letto impaccato non modificato. L'Approccio DEM-SALOME ha consentito di generare la mesh di letti impaccati attraverso script in linguaggio `python` utilizzando le librerie di SALOME. Infine, è stato analizzato un *caso studio*, di simulazione di un reattore impaccato di 20 sfere. La simulazione è stata effettuata con il solver `catalyticConductivePimpleFOAM` [2] per studiare i fenomeni di scambio di materia e di calore che si sviluppa per effetto della reazione catalitica superficiale.

## 6.2 Proprietà geometriche dei letti impaccati di sfere

I letti impaccati di sfere hanno molteplici applicazioni nell'Industria Chimica. Oltre ai processi catalitici, vengono utilizzati in diverse operazioni unitarie, come ad esempio nella filtrazione oppure nell'operazione di scambio ionico. Generalmente, il dimensionamento dei reattori impaccati tiene conto di determinate grandezze geometriche del letto che dipendono dal tipo di studio che si vuole condurre. Spesso la scelta di tali grandezze è fondamentale per il corretto funzionamento del reattore. Ad esempio, i reattori a letto fluido necessitano di particelle catalitiche molto più piccole rispetto ai reattori a letto fisso per garantire una buona sospensione del letto e quindi un corretto funzionamento del reattore. I reattori a letto fisso di sfere possono essere caratterizzati tra tre grandezze geometriche [3]:

- ✓ il rapporto diametro tubo su diametro delle sfere denominato con il termine  $N$  definito anche come *aspect ratio*

- ✓ il rapporto tra l'area superficiale delle particelle catalitiche e il volume del reattore  $a_v$
- ✓ il grado di vuoto  $\varepsilon$  del letto impaccato

### 6.2.1 Aspect Ratio

La prima proprietà importante che caratterizza un reattore impaccato di sfere è il rapporto tra il diametro (o *diametro equivalente*) del vessel ed il diametro delle particelle.

$$N = \frac{d_{tubo}}{d_{particella}} \quad (6.1)$$

I reattori impaccati vengono pertanto classificati come reattori con alti e/o bassi valori di  $N$ . Per rendere il concetto si possono immaginare due situazioni limite. Un tubo impaccato con sabbia avrebbe elevati valori di  $N$ , ma lo stesso tubo impaccato con delle palline da golf avrebbe valori più bassi di  $N$ . Questo fenomeno si ripercuote sui diversi profili di velocità che si sviluppano all'interno del reattore. Reattori con elevati valori di  $N$  garantiscono un profilo uniforme di velocità lungo la sezione trasversale al reattore.

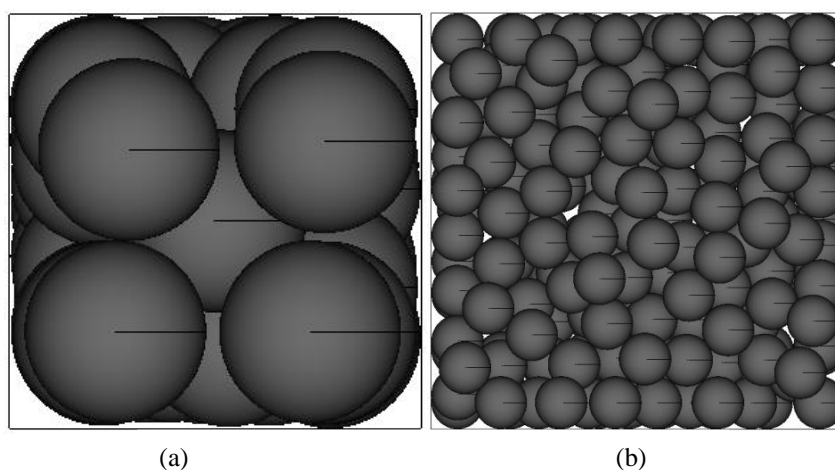


Figura 6.1 reattore con  $N=2.2$ (a); reattore con  $N=7.87$ (b).

Quando il valore di  $N$  diventa molto basso allora il profilo di velocità non può essere considerato uniforme, a causa degli elevati valori di porosità locali nel letto impaccato. Per reazioni fortemente esotermiche si preferisce utilizzare valori di  $N$  compresi tra 2-8 perché garantiscono un soddisfacente scambio termico per mantenere la temperatura all'interno del reattore a valori di processo ragionevoli.

### 6.2.2 Grado di vuoto

Quando un tubo viene impaccato con delle sfere, si distinguono due zone: la regione solida, che è data dall'insieme delle particelle che costituiscono il letto impaccato ed una regione fluida che è il vuoto che si crea nel letto impaccato. Il rapporto tra il volume del fluido rispetto a quello totale del reattore rappresenta il grado di vuoto  $\varepsilon$ :

$$\varepsilon = \frac{V_{vuoto}}{V_{totale}} = \frac{V_{totale} - V_{sfere}}{V_{totale}} \quad (6.2)$$

dove  $V_{vuoto}$  è il volume della regione fluida, mentre  $V_{totale}$  è il volume totale del tubo. Nel caso in cui le sfere abbiano la stessa dimensione nel letto impaccato, la 6.2 diventa:

$$\varepsilon = 1 - n \frac{\pi d_{sfere}^3}{6 \cdot d_{tubo}^2 L} \quad (6.3)$$

dove  $n$  è il numero delle sfere presenti all'interno del letto impaccato. Generalmente per modellare reattori industriali oppure *micro-channel* il numero delle particelle è un'incognita. Il grado di vuoto può essere così calcolato attraverso delle correlazioni sperimentali, che consentono di stimare il grado di vuoto senza conoscere il numero delle particelle [4].

### 6.2.3 Area catalitica per unità di volume

Per reattori catalitici eterogenei è importante valutare la superficie catalitica esposta per favorire le reazioni catalitiche superficiali. Si preferisce introdurre l'area catalitica per unità di volume, definita come il rapporto tra la superficie catalitica delle particelle che costituiscono il letto impaccato ed il volume del reattore. In formula:

$$a_v = \frac{n \cdot A_{sfera}}{V_{totale}} = \frac{n \cdot \pi d_{sfera}^2}{V_{totale}} \quad (6.4)$$

Conoscendo il numero delle sfere  $n$  ed il grado di vuoto  $\varepsilon$  del letto impaccato, è possibile calcolare  $V_{totale}$  nel seguente modo:

$$V_{pieno} = n \cdot V_{sfera} = V_{totale}(1 - \varepsilon) \quad (6.5)$$

$$V_{totale} = \frac{n \cdot V_{sfera}}{(1 - \varepsilon)} \quad (6.6)$$

Pertanto, l'equazione 6.5 diventa:

$$a_v = \frac{n \cdot \pi d_{sfera}^2}{n \cdot V_{sfera}} (1 - \varepsilon) \leftrightarrow a_v = \frac{6(1 - \varepsilon)}{d_s} \quad (6.7)$$

Qualora, la forma delle particelle sia differente, bisogna introdurre un altro parametro che è il diametro equivalente, definito come il diametro di una sfera fittizia che ha lo stesso volume della particella non sferica che si trova nel letto impaccato. Un metodo alternativo è quello di utilizzare un parametro  $\beta$ , chiamato fattore di forma, che definisce lo scostamento dalla sfera ideale [5]. Quando le particelle sono sferiche, il diametro equivalente si riduce semplicemente al diametro delle particelle.

Questa equazione non può essere utilizzata per determinare l' $a_v$  di reattori impaccati, dove i punti di contatto sono stati modificati con i metodi *bridge*, *cap* ed *overlap*, poiché sia il volume che l'area delle sfere sono stati modificati. In questo caso, poiché le sfere non sono perfettamente sferiche, bisognerebbe ricorrere ad una

espressione più complessa che tenga conto della quantità di area e volume rimossa/aggiunta per effetto della modifica effettuata.

### 6.3 Approccio DEM–SALOME per la generazione di letti impaccati

La più comune tipologia di letto impaccato utilizzato nella Chimica Industriale è il letto con un arrangiamento casuale di particelle all'interno di un tubo con sezione circolare oppure quadrata. Un impaccamento random può essere generato attraverso degli algoritmi di calcolo capaci di considerare delle dinamiche realistiche di impaccamento di sfere una alla volta all'interno di un tubo. In questo lavoro è stato utilizzato l'approccio DEM (*Discrete Element Method*) per la generazione dei letti impaccati e SALOME per effettuare le modifiche dei punti di contatto e generazione della mesh.

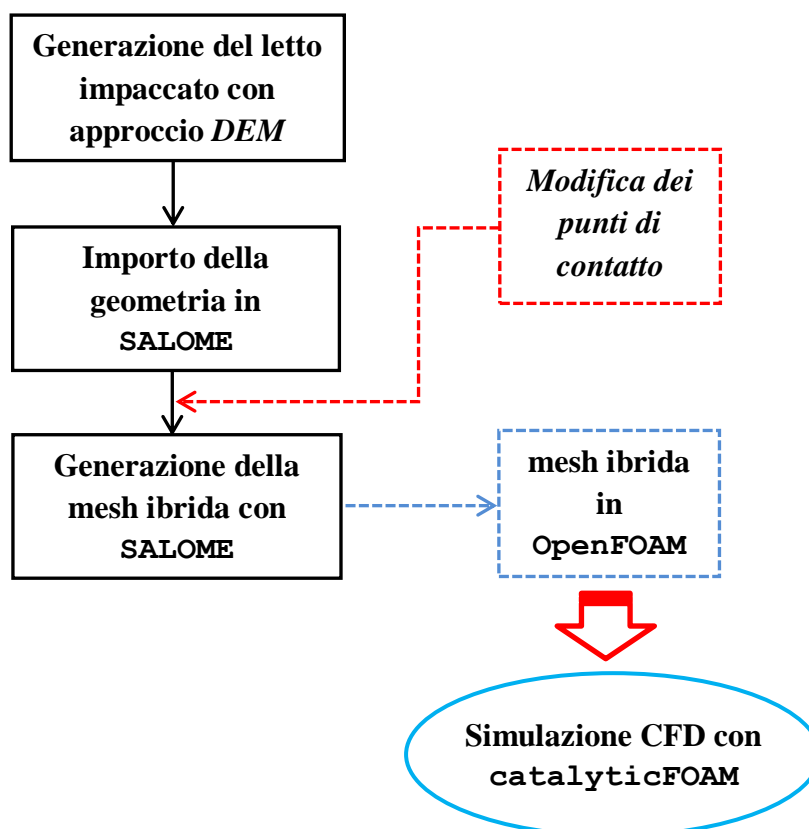


Figura 6.2 Diagramma di lavoro utilizzato per generare la mesh ibrida di letti impaccati.

Il diagramma rappresentato in Figura 6.2 rappresenta la metodologia utilizzata per generare la mesh di letti impaccati di sfere dopo aver modificato opportunamente i punti di contatto. Il primo step è quello di definire le dimensioni delle sfere e del box che contiene le sfere. Una volta definite le grandezze del letto impaccato si procede con la simulazione *DEM*.

L'approccio *DEM* è una famiglia di metodi numerici basato sul calcolo del moto di una grande quantità di particelle su uno spazio delimitato. Sostanzialmente il codice risolve le equazioni del moto di Newton di ciascuna particella fornendo un orientamento spaziale random per impaccare un tubo di dimensioni assegnate. Le forze che vengono considerate nelle equazioni del bilancio di forze sono:

- forza di attrito: tra due o più particelle che si toccano
- contatto plastico: quando due particelle collidono
- gravità: attrazione tra le particelle a causa delle loro forze peso
- attrazione potenziale: come adesione, coesione o attrazione elettrostatica

Tutte le forze elencate vengono considerate all'interno dell'equazione che governa il bilancio di forze per ciascuna particella. Il codice di calcolo discretizza queste equazioni nel tempo secondo un time step assegnato e vengono risolte fino a quando il letto impaccato raggiunge la stabilità meccanica. Le equazioni del moto delle particelle vengono risolte utilizzando il metodo del secondo ordine di Adams-Bashforth [6] che stima i valori delle forze di contatto nel calcolo al time step successivo attraverso un'estrapolazione lineare o attraverso metodi multi-step come *il predictor-corrector method* [7]. Esistono diversi codici open-source in grado di implementare la metodologia descritta:

- *SDEC*: Spherical Discrete Element [8]
- *Yade*: Yet Another Dynamic Engine, un codice di calcolo che si basa sull'approccio DEM scritto in C++ [9]
- *LAMMPS*: consente delle simulazioni DEM molto veloci anche in parallelo [10]

- *LIGGGTHS*: è un codice che si basa sull'approccio LAMMPS e DEM che consente di importare file CAD [11]

Il modello *DEM* considera la particella con due tipi di moto: il moto rotazionale e quello traslazionale, che possono essere rappresentati attraverso le leggi del moto di Newton [12]. Queste equazioni, basate su forze e coppie originate dall'interazione con le particelle vicine, sono date:

$$m_i \frac{d\mathbf{v}_i}{dt} = \sum_j \mathbf{f}_{ij} + m_i \mathbf{g} \quad (6.8)$$

$$I_i \frac{d\boldsymbol{\omega}_i}{dt} = \sum_j \mathbf{t}_{ij} \quad (6.9)$$

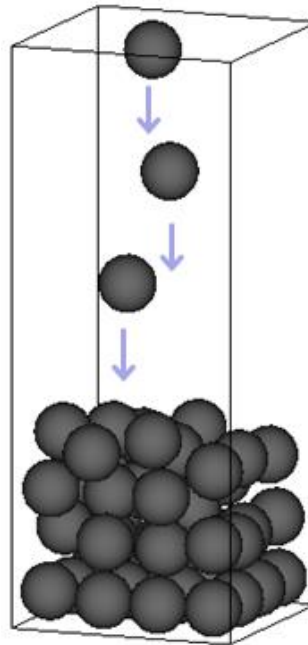
dove  $m_i$ ,  $I_i$ ,  $\mathbf{v}_i$ ,  $\boldsymbol{\omega}_i$  sono rispettivamente la massa, il momento di inerzia, la velocità ed il momento angolare della particella  $i$ . Le forze coinvolte sono la forza gravitazionale e la forza di interazione  $\mathbf{f}_{ij}$  tra le particelle  $i$  e  $j$  che dipende dalla deformazione della particella  $i$ . La coppia agente sulla particella  $i$  per effetto della particella  $j$ ,  $\mathbf{t}_{ij}$ , deriva dalla forza di contatto e provoca la rotazione oppure il rallentamento delle rotazioni relative delle due particelle. Le pareti vengono trattate come particelle di dimensioni infinite. Le componenti delle forze e delle coppie agenti sulla particella  $i$  sono:

- la forza normale di smorzamento
- le forze tangenziali
- le forze di attrito Columbiano
- le coppie generate dalle forze tangenziali
- le coppie di attrito volvente

Pertanto, le traiettorie, le velocità e le velocità angolari delle particelle possono essere determinate attraverso le equazioni 6.8 e 6.9.

La simulazione *DEM* si arresta quando le velocità delle particelle all'interno del letto impaccato sono vicine allo zero. In questo modo si garantisce un bilancio delle forze agenti su ciascuna particella e quindi una stabilità meccanica del letto impaccato.

La Figura 6.3 mostra il processo di generazione di un letto impaccato di sfere.



**Figura 6.3** Generazione di un letto impaccato attraverso una simulazione *DEM*.

Quando si genera un letto impaccato di sfere attraverso una simulazione *DEM* la distanza tra i punti di contatto tra le sfere non è esattamente pari a zero: alcune delle sfere del letto impaccato possono trovarsi a distanze molto piccole e quindi non toccarsi, altre possono addirittura sovrapporsi. Questa conseguenza va ricercata nel tipo di modello utilizzato per descrivere il moto delle particelle all'interno del tubo prima e dopo la collisione. Implementando le equazioni di Newton per descrivere il moto delle particelle, si utilizza un modello chiamato *soft contact model* [13]. Questo metodo fornisce meno informazioni circa l'interazione della particella *i* con le altre particelle, ma ha il grande vantaggio di avere dei dispendi computazionali notevolmente inferiori rispetto ai modelli più dettagliati. Attraverso il modello *DEM* non è pertanto possibile ottenere un impaccamento dove la distanza tra le sfere che si toccano è esattamente pari a zero. Il metodo *DEM* introduce un certo errore nella



generazione del letto impaccato. Questi errori sono stati ridotti al minimo lavorando con dimensioni molto piccole delle sfere nell'ordine dei  $10^{-2}$  m e successivamente per riportare la mesh nelle dimensioni desiderate, è stata moltiplicata per un fattore di  $10^2$  prima di importarla in ambiente OpenFOAM.

Generato il letto impaccato di sfere, è stato possibile importare la geometria considerata in SALOME in un file di testo contenente le coordinate dei centri di tutte le sfere che costituiscono il letto impaccato. Le proprietà del letto impaccato come ad esempio minima distanza tra le sfere, grado di vuoto,  $a_v$ , ecc. sono state valutate da SALOME. Una volta scritto lo *script* `properties_packedBed-localMethods.py` è stato possibile lanciare lo script dal terminale per ottenere le informazioni desiderate sul letto impaccato in studio. In Tabella 6.1 si riportano le caratteristiche geometriche di un letto impaccato di 200 sfere.

<i>Proprietà del letto impaccato non modificato</i>	
Numero di sfere	200
constantDistance [m]	$6.4 \cdot 10^{-5}$
punti di contatto sfera/sfera	444
punti di contatto sfera/parete	6
diametro delle sfere [m]	0.254
min distanza sfera/sfera [m]	0
min distanza sfera/parete [m]	$9.999 \cdot 10^{-8}$
superficie catalitica [m <sup>2</sup> ]	40.536
volume sfere [m <sup>3</sup> ]	1.716
volume box [m <sup>3</sup> ]	3.222
grado di vuoto [-]	0.483
$a_v$ teorica [m <sup>2</sup> /m <sup>3</sup> ]	12.202
$a_v$ reale [m <sup>2</sup> /m <sup>3</sup> ]	12.202
Err % $a_v$ [-]	$7.9 \cdot 10^{-4}$

**Tabella 6.1** Valutazione delle proprietà geometriche di un letto impaccato random di 200 sfere.

Per identificare il numero dei punti di contatto è stato necessario definire una distanza minima chiamata `constantDistance`. Il punto di contatto viene identificato quando la distanza sfera/sfera e sfera/parete è inferiore rispetto al valore definito dal `constantDistance`. Il valore scelto pari a  $6.4 \cdot 10^{-5}$  m è stato stimato

sulla base dello studio svolto nel Capitolo 3 per la generazione della *mesh* per il *gap method*. È stato visto che generando la *mesh* con una riduzione del diametro delle sfere maggiore uguale allo 0.25% si ha un dispendio computazionale ragionevolmente inferiore rispetto al caso di riferimento. Imponendo una variazione dello 0.25% del diametro delle sfere si ottiene una minima distanza pari a:

$$d_{nominale} - d_{attuale} = \frac{0.25 \cdot d_p}{100} = 6.35 \cdot 10^{-5} \text{ m} = \text{min}_{distanza} \quad (6.10)$$

E' possibile così definire una costante:

$$\frac{\text{min}_{distanza}}{d_p} = 0.0025 = SR \quad (6.11)$$

Il risultato ottenuto indica che quando due sfere si trovano ad una distanza inferiore o pari a quella calcolata nella 6.11 le due sfere si possono considerare a contatto. Una giustificazione a questa importante affermazione è quella di asserire che valori inferiori alla  $\text{min}_{distanza}$  forniscono delle *mesh* di bassa qualità vicino il punto di contatto tra le sfere. Questo risultato può essere esteso anche per letti impaccati con dimensioni del reattore e delle particelle diversi, moltiplicando semplicemente la 6.12 con il diametro delle particelle utilizzato. Infine, per verificare che l'impaccamento di sfere fornisce un valore di  $a_v$  ragionevolmente vicino a quello di un *impaccamento ideale di sfere* (cioè dove tra i punti di contatto la distanza tra le sfere è pari a zero) il risultato ottenuto con la 6.7 è stato confrontato con il valore stimato da SALOME. Per questo impaccamento l'errore percentuale ottenuto è circa pari allo  $7.9 \cdot 10^{-4} \%$  che può essere ritenuto accettabile.

Nel Paragrafo successivo si discuterà la metodologia utilizzata per implementare i 4 metodi di modifica dei punti di contatto per i letti impaccati di sfere.

### 6.3.1 Implementazione dei metodi di modifica dei punti di contatto in letti impaccati di sfere

Identificati i punti di contatto è stato possibile eliminarli con i metodi di modifica globale e locale.

➤ *Metodi di modifica globale: gap e overlap*

L'implementazione dei metodi di modifica globale è stata più semplice rispetto ai metodi locali. Infatti, una volta importate le coordinate dei centri di tutte le sfere in SALOME, è stata definita una variazione percentuale del diametro delle sfere, generando così il nuovo letto impaccato (Figura 6.5).

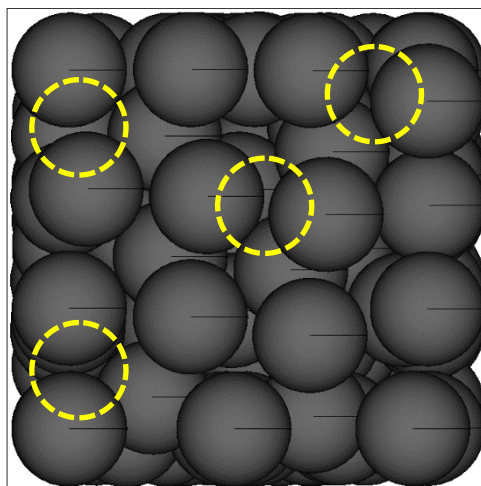


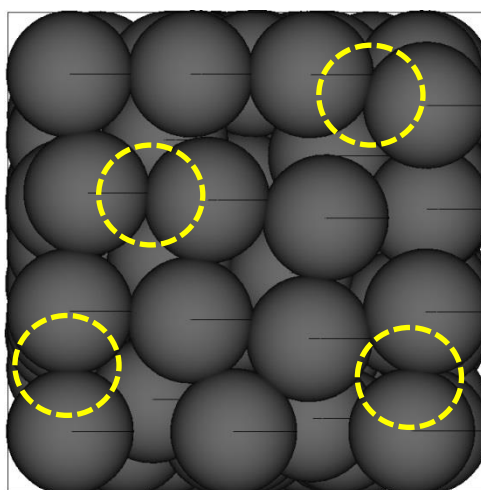
Figura 6.4 Vista dall'alto di un letto impaccati dopo aver applicato il *gap method*.

Una volta effettuata la modifica dei punti di contatto è stato possibile valutare le proprietà geometriche dei letti impaccati, fornendo dei confronti quantitativi con il letto impaccato non modificato. Ad esempio, confrontando i risultati ottenuti in Tabella 6.2, si nota che imponendo una diminuzione dei diametri delle sfere del 5% i punti di contatto sono stati eliminati del tutto. Conseguentemente si ha un aumento del grado di vuoto del letto pari al 15.24% ed una variazione di  $a_v$  (oppure area catalitica) pari a circa il 9.8%.

<b>Proprietà del letto impaccato -Metodo globale GAP -</b>	
Numero di sfere	200
Variazione % diametro	5.0
diámetro delle sfere [m]	0.254
min distanza sfera/sfera [m]	$7.366 \cdot 10^{-3}$
min distanza sfera/parete [m]	$6.35 \cdot 10^{-3}$
superficie catalitica [m <sup>2</sup> ]	36.584
grado di vuoto [-]	0.557
$a_v$ teorica [m <sup>2</sup> /m <sup>3</sup> ]	11.012
Err % $a_v$ [-]	-9.75
Err % grado di vuoto[-]	15.24

**Tabella 6.2** Valutazione delle proprietà geometriche del letto impaccato dopo aver applicato il *gap method*.

Sebbene questi due metodi siano molto semplici da implementare, non sempre garantiscono una buona accuratezza della modellazione. Applicando infatti il *gap method*, il valore minimo della diminuzione percentuale del diametro delle sfere è stato opportunamente scelto per garantire delle distanze sfera/sfera e sfera/parete non più piccole del valore impostato (`constantDistance`). Per l'*overlap method* invece, si potrebbero creare dei punti di contatto a causa dell'espansione del letto (Tabella 6.3-Figura 6.5).



**Figura 6.5** Vista dall'alto di un letto impaccati dopo aver applicato l'*overlap method*.

<b><i>Proprietà del letto impaccato -Metodo globale OVERLAP -</i></b>	
Numero di sfere	200
Variazione % diametro	5.0
diámetro delle sfere [m]	0.254
min distanza sfera/sfera [m]	0.0
min distanza sfera/parete [m]	0.0
superficie catalitica [m <sup>2</sup> ]	30.889
grado di vuoto [-]	0.425
a <sub>v</sub> teorica [m <sup>2</sup> /m <sup>3</sup> ]	9.295
Err % a <sub>v</sub> [-]	-23.822
Err % grado di vuoto[-]	-12.127

**Tabella 6.3** Valutazione delle proprietà geometriche del letto impaccato dopo aver applicato l'*overlap method*.

Lo *script* è stato riportato in Appendice con il nome di `properties_packedBed-globalMethods.py`

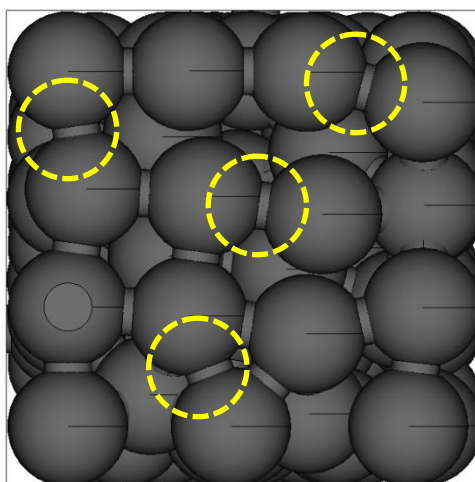
➤ *Metodi di modifica locale: cap e bridge*

Per modificare i punti di contatto nei letti impaccati con i metodi di modifica locale sono stati utilizzati gli *script* creati per generare le geometrie descritte nel Capitolo 3. La creazione del *bridge*, oppure del *cap* è stata resa automatica per tutti i punti di contatto identificati nel letto impaccato. In entrambi gli *script* è possibile definirerispettivamente i rapporti  $r_b/d_p$ , per il bridge method e  $2h/d_p$  per il *cap method*.

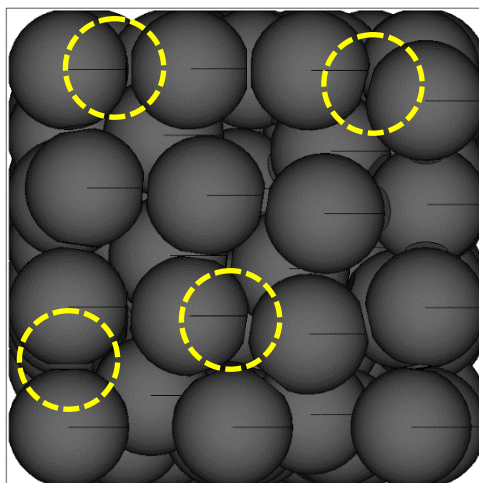
<b>Proprietà del letto impaccato -Metodo locale BRIDGE -</b>	
Numero di sfere	200
diametro delle sfere [m]	0.254
numero dei bridges sfera/sfera	444
numero dei bridges sfera/parete	7
min distanza sfera/sfera [m]	$7.4 \cdot 10^{-3}$
min distanza sfera/parete [m]	$6.4 \cdot 10^{-3}$
rb/ds [-]	0.2
rb [m]	0.0508
lunghezza bridge sfera/sfera [m]	0.0212
lunghezza bridge sfera/parete [m]	0.0106
superficie catalitica [m <sup>2</sup> ]	35.998
grado di vuoto [-]	0.472
a <sub>v</sub> teorica [m <sup>2</sup> /m <sup>3</sup> ]	10.836
Err % a <sub>v</sub> [-]	-11.195
Err % grado di vuoto[-]	-2.358

**Tabella 6.4** Valutazione delle proprietà geometriche del letto impaccato dopo aver applicato il *bridge method*.

Le Tabella 6.4 e 6.5 mostrano gli output degli *script* contenenti le proprietà dei letti impaccati calcolate.



(a)



(b)

Figura 6.6 Vista dall'alto di letti impaccati modificati. *Bridge method* (a); *cap method* (b).

<b>Proprietà del letto impaccato -Metodo locale CAP -</b>	
Numero di sfere	200
diametro delle sfere [m]	0.254
$2h/ds$ [-]	0.005
lunghezza cap sfera/sfera [m]	0.00127
lunghezza cap sfera/parete [m]	$6.4 \cdot 10^{-4}$
superficie catalitica [m <sup>2</sup> ]	36.584
grado di vuoto [-]	0.557
$a_v$ teorica [m <sup>2</sup> /m <sup>3</sup> ]	11.012
Err % $a_v$ [-]	9.75
Err % grado di vuoto[-]	15.24

**Tabella 6.5** Valutazione delle proprietà geometriche del letto impaccato dopo aver applicato il *cap method*.

Il *cap method* è stato implementato utilizzando il secondo modello più complesso descritto nel Capitolo 3. Sebbene sia un metodo più lento rispetto al primo (*bridge/cap method*) garantisce una migliore precisione locale.

Il primo metodo è stato implementato come segue: una volta aver sostituito i punti di contatto con i *bridges* è stata applicata la funzione booleana “*cut operation*” che consente di eliminare la porzione di solido tra le sfere, creando così un *cap*. Il problema di questo metodo è che, quando alcune sfere si sovrappongono, le sfere

risultanti non sono perfettamente uniformi vicino il cap applicato. Questo effetto si ripercuote sulla generazione della mesh perché si hanno delle imperfezioni sulla superficie delle sfere.

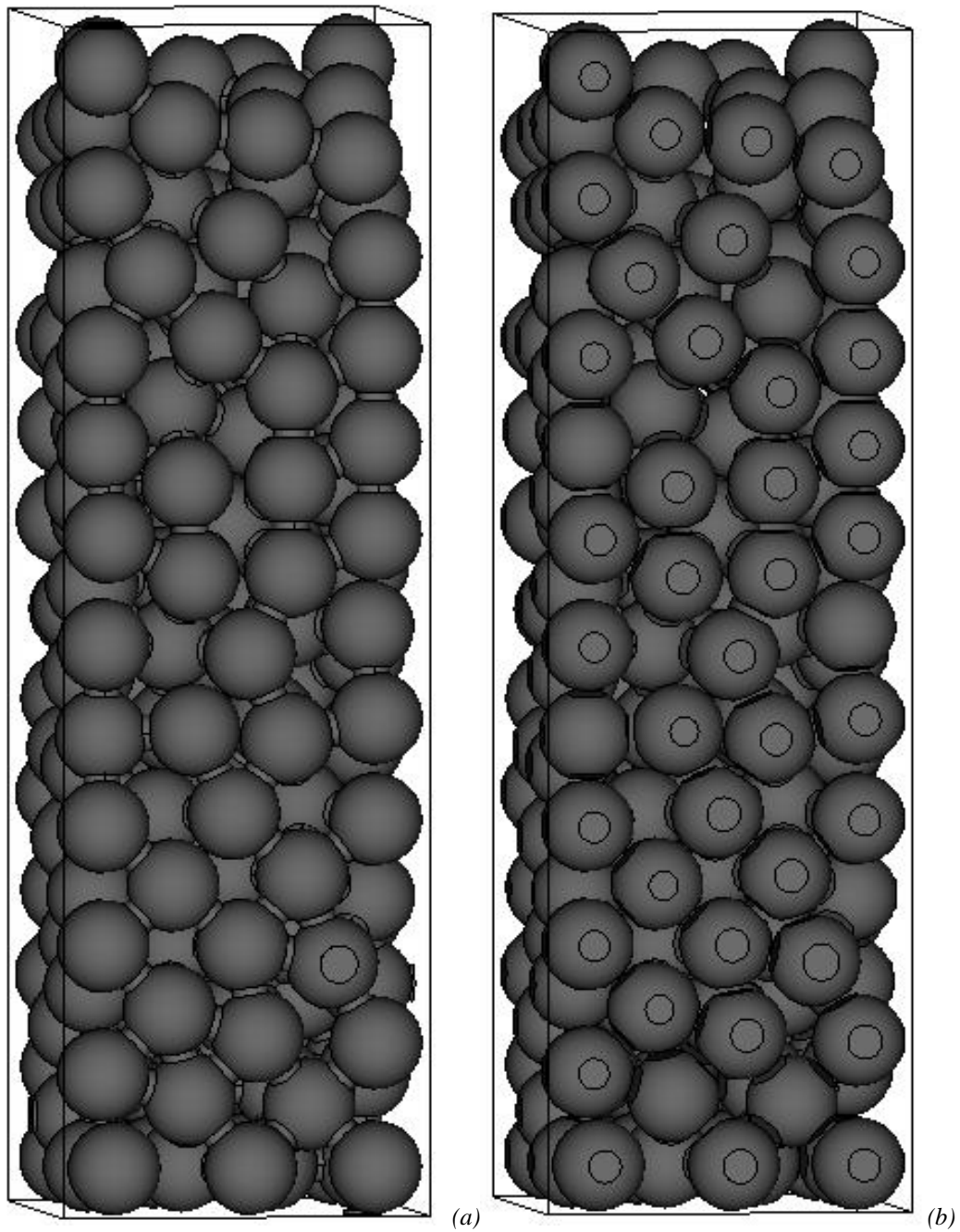


Figura 6.7 Vista laterale di letti impaccati di 200 sfere. *bridge method* (a); *cap method* (b).



### 6.3.2 Generazione della mesh

Una volta aver modificata la geometria, è stato possibile generare la *mesh* ibrida. L'algoritmo utilizzato per generare la mesh non strutturata è il NETGEN-1D2D3D che si basa sul metodo della triangolazione di Delaunay. In Figura 6.8 si riporta un mesh ibrida per un letto impaccato di 20 sfere.

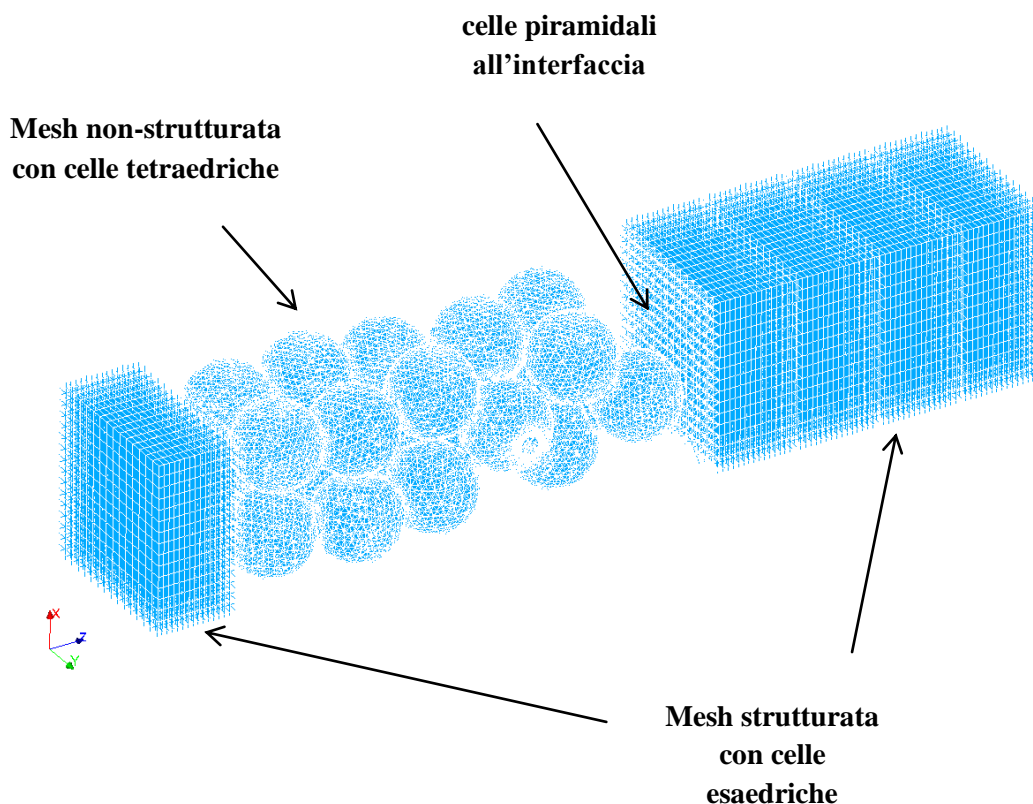
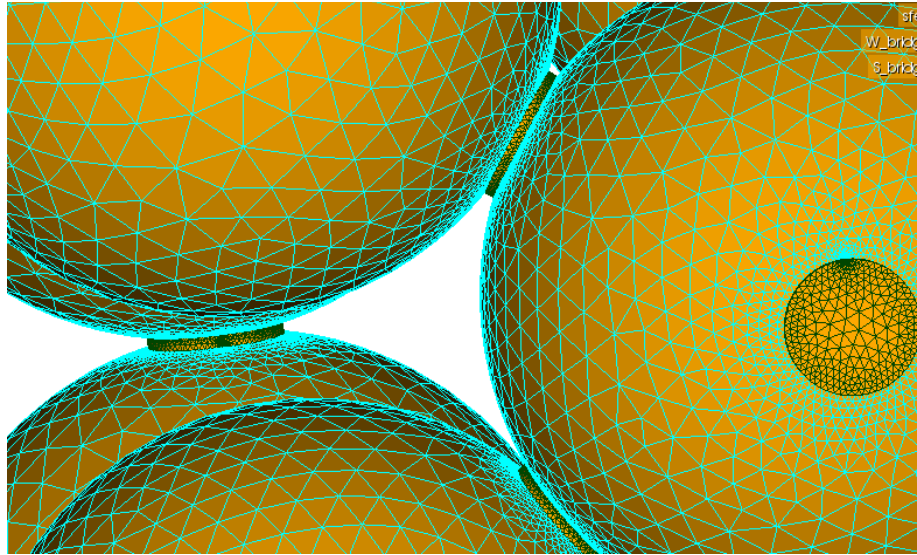


Figura 6.8 Mesh ibrida 3D ottenuta per un letto impaccato di 20 sfere.

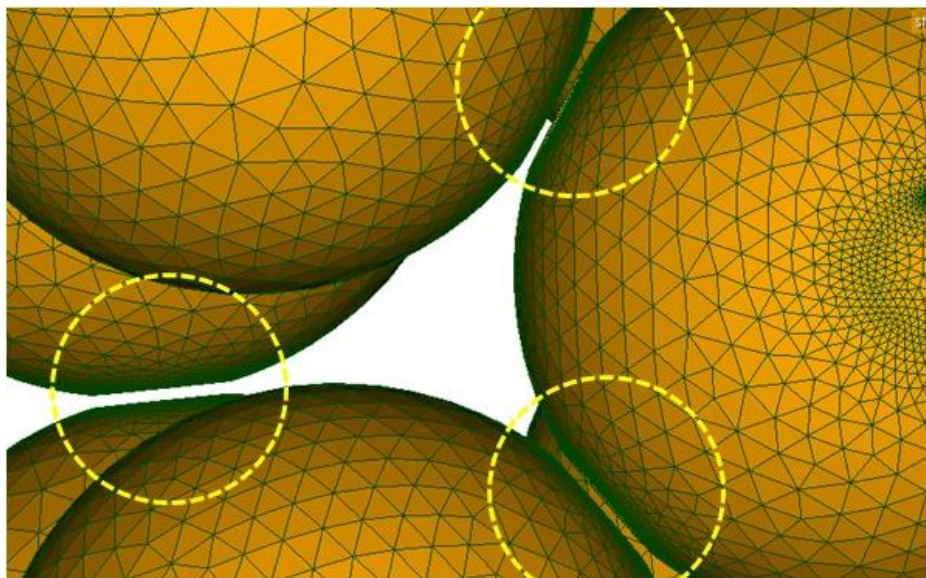
Per ottenere delle mesh più fitte è stato necessario impostare la dimensione lineare delle celle tetraedriche sulla superficie laterale del *bridge* pari a circa un terzo della lunghezza dei *bridges*. In questo modo la *mesh* si sviluppa progressivamente prima sulla superficie delle sfere e poi nella regione fluida circostante. La mesh multiregione riportata in Figura 6.8 contiene circa 1 milione e 500 mila celle, ed è stata generata in pochi minuti utilizzando un singolo processore. Per ottenere delle mesh multiregione per letti impaccati di centinaia o addirittura migliaia di sfere è

necessario parallelizzare lo *script* in più processori per garantire tempi di calcolo soddisfacenti.

Nelle Figure 6.9-6.10 si riportano dei particolari della *mesh*, evidenziando le modifiche locali effettuate nei punti di contatto tra le sfere.



**Figura 6.9** Modifica dei punti di contatto attraverso il *bridge method* per un letto impaccato di 20 sfere.



**Figura 6.10** Modifica dei punti di contatto attraverso il *cap method* per un letto impaccato di 20 sfere.

## 6.4 Valutazione delle proprietà geometriche dei letti impaccati

Gli scripts creati per implementare i 4 metodi di modifica dei punti di contatto in letti impaccati di sfere sono stati utilizzati per confrontare le proprietà geometriche con quelle di un letto impaccato di riferimento (cioè senza nessuna modifica nei punti di contatto). Questa valutazione è stata molto importante al fine di valutare le variazioni percentuali di grado di vuoto e di  $a_v$  rispetto al letto impaccato non modificato. L'analisi è stata condotta su 4 diversi letti impaccati a diversi valori di  $N$  imponendo le dimensioni del letto ed il numero di sfere riportate in Tabella 6.6.

$N$	Numero di sfere	$L_{tubo}/d_s$
2.2	100	25.44
3.94	200	12.86
5.96	400	10.77
7.99	800	11.96

Tabella 6.6 Valori delle dimensioni dei letti impaccati e numero di sfere utilizzate per ottenere impaccamenti a 4 diversi valori di  $N$ .

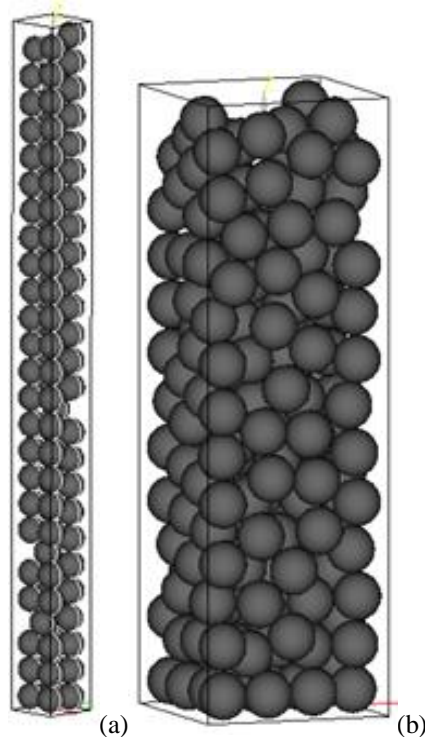
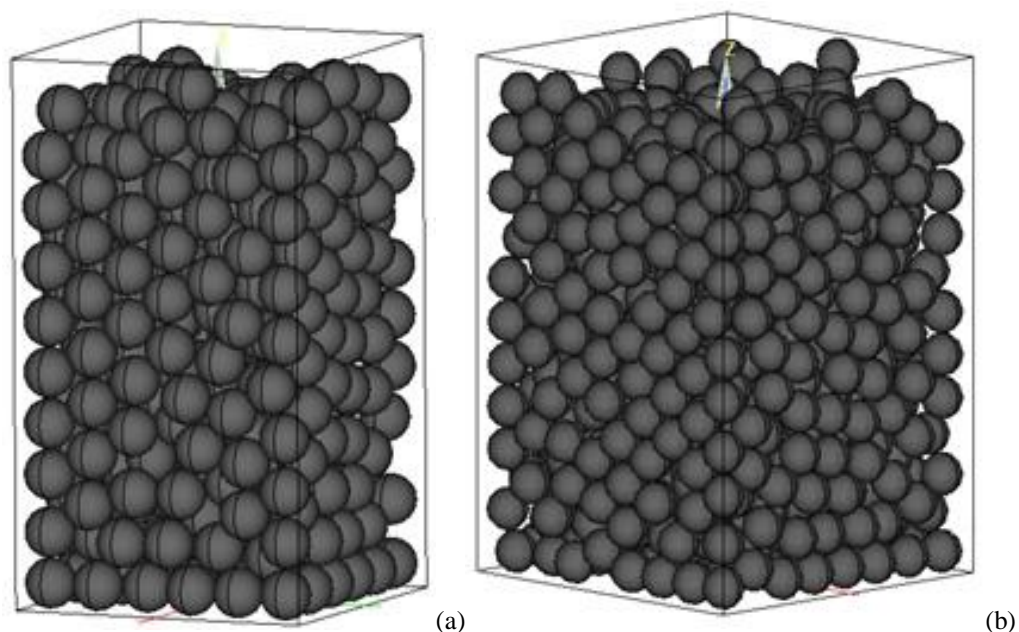


Figura 6.11 Impaccamenti utilizzati a diversi valori di  $N$ :  $N=2.2$ (a);  $N=3.94$ (b).



**Figura 6.12 Impaccamenti utilizzati a diversi valori di  $N$ . (a):  $N=5.96$ ; (b):  $N=7.99$ .**

L'analisi ha mostrato che, all'aumentare di  $N$  aumentano considerevolmente i punti di contatto sfera/sfera si hanno così valori più bassi del grado di vuoto poiché viene meno l'effetto sfera/parete.

$N$	$N^\bullet$ di sfere	$L_{tubo}/d_s$	<i>punti di contatto sfera/sfera</i>	<i>punti di contatto sfera/parete</i>	<i>grado di vuoto</i>
2.2	100	25.44	167	6	0.562
3.94	200	12.86	444	7	0.483
5.96	400	10.77	978	6	0.458
7.99	800	11.96	2080	8	0.455

**Tabella 6.7 Proprietà ottenute per i 4 letti impaccati a diversi valori di  $N$ .**

Ciascun metodo di modifica dei punti di contatto muta diversamente le proprietà geometriche dei letti impaccati. Pertanto, per valutare le variazioni percentuali del grado di vuoto e di  $a_v$  rispetto ad un caso di riferimento, sono stati confrontati i 4 approcci di modifica dei punti di contatto.

Al fine di confrontare la variazione di  $a_v$  per i due metodi di modifica locale, gli errori percentuali sono stati confrontati rispetto alla variazione della distanza  $h$  tra le

sfere. Come mostrato in Figura 6.13 il valore  $h$  tra le sfere è stato scelto della stessa lunghezza per entrambi i metodi.

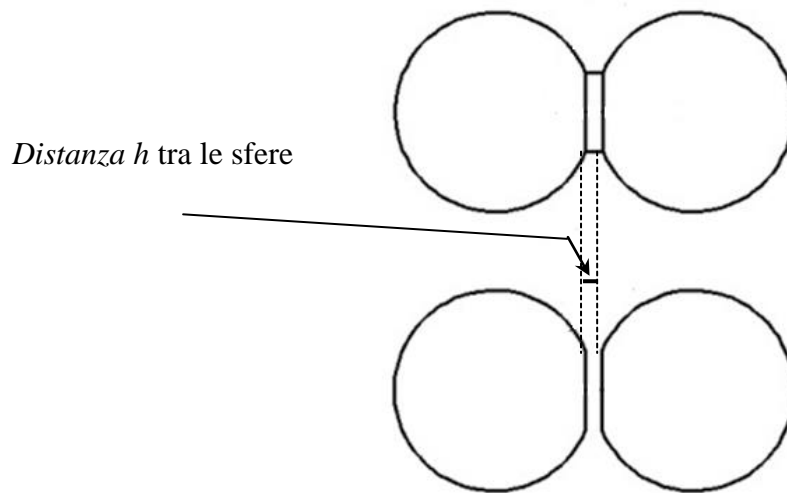


Figura 6.13 Distanza  $h$  tra le sfere ottenuta attraverso i metodi di modifica locale.

La Figura 6.14 mostra come implementando il *cap method*, si ottengono degli errori contenuti fino allo 0.91% circa.

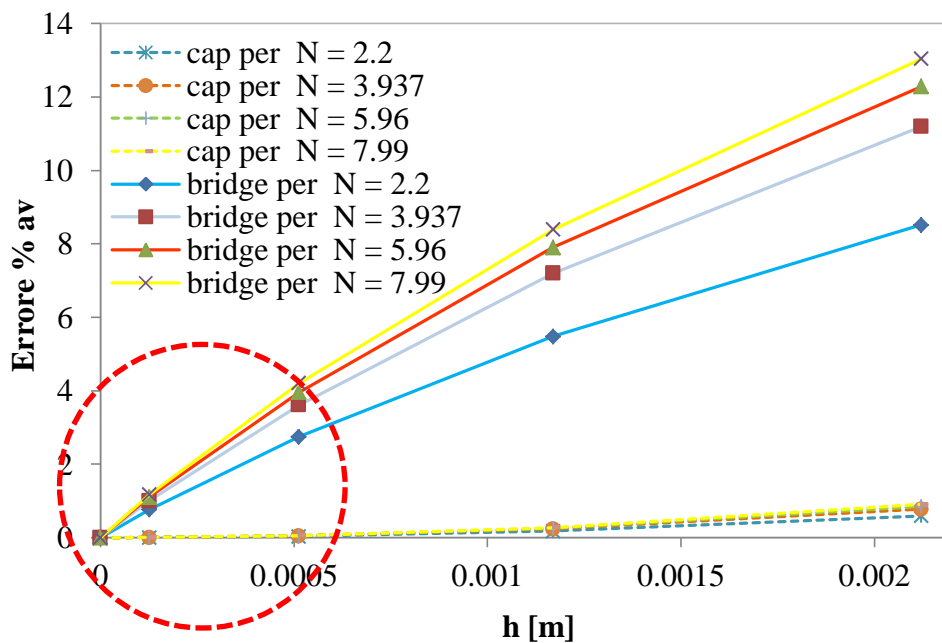


Figura 6.14 Confronto dell'errore % di area catalitica per i metodi di modifica locale.

A parità di distanza  $h$  tra le sfere, il *bridge method* invece fornisce degli errori percentuali considerevolmente maggiori. Sebbene la sostituzione di un punto di contatto con un *bridge* garantisce la presenza dell'area laterale come superficie catalitica disponibile, è comunque di gran lunga inferiore rispetto a quella offerta dal *cap*. Inoltre si nota che a parità di valore del *cap* utilizzato nei diversi letti impaccati e all'aumentare dell'aspect ratio, (quindi del numero di punti di contatto sfera/sfera) si ha un aumento dell'errore percentuale causato dal maggior numero di modifiche effettuate alle sfere e quindi maggior perdita di area catalitica.

In Tabella 6.8 si riportano gli errori percentuali per i il *cap* ed il *bridge* a due valori distinti di  $N$  pari a 2.2 e 7.99 rispettivamente. Si osserva come il *cap method* dia degli errori più contenuti rispetto al *bridge method*.

	<b>Err % Area per N =2.2</b>	<b>Err % Area per N =7.99</b>
<i>cap 1</i>	0.001	0.002
<i>cap 2</i>	0.034	0.052
<i>bridge 1</i>	0.766	1.174
<i>bridge 2</i>	2.747	4.208

**Tabella 6.8 Valori di Errore % di area catalitica per i due metodi di modifica locale a due valori distinti di  $N$ .**

Il *gap* e l'*overlap* methods sono stati tra loro confrontati riportando i valori degli errori percentuali di  $a_v$  rispetto alla variazione percentuale del diametro delle sfere. Per il *gap method* si parla di diminuzione del diametro, mentre per l'*overlap* di aumento percentuale del diametro. I due metodi forniscono delle variazioni percentuali lineari e si nota che gli errori sono contenuti entro il 10% circa, quindi più contenuti rispetto a quelli forniti dal *bridge method*.

Applicando una diminuzione oppure un aumento dei diametri delle sfere fino al 0.5% per entrambi i metodi di modifica gli errori percentuali si mantengono sotto il 2% circa. Per variazioni maggiori, l'*overlap method* fornisce degli errori maggiori dovuti alla sovrapposizione delle sfere con conseguente perdita di area catalitica.

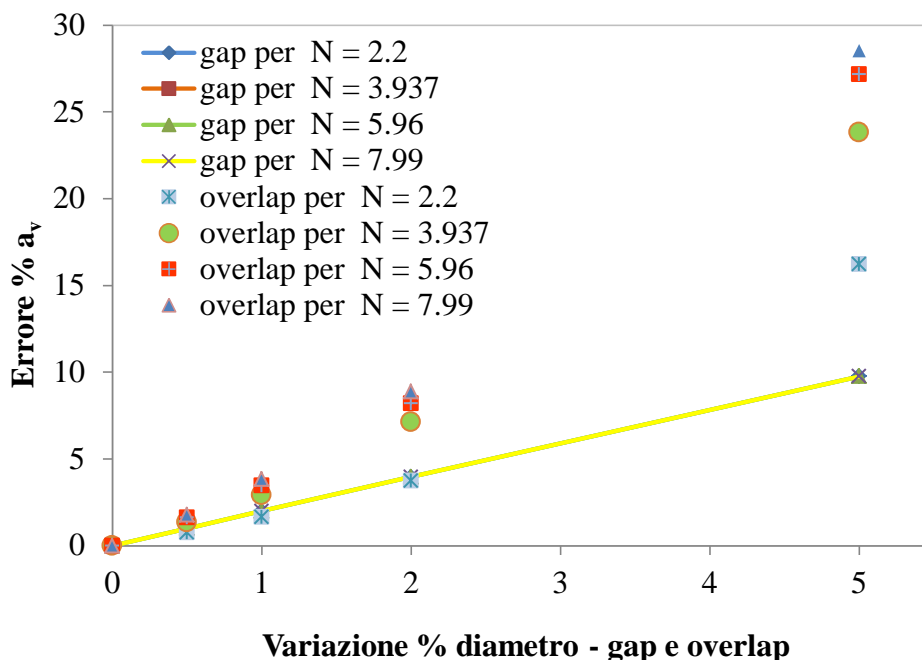


Figura 6.15 Variazione % di  $a_v$  rispetto al riferimento per i letti impaccati a diversi valori di  $N$ , implementando il *gap* e l'*overlap method*.

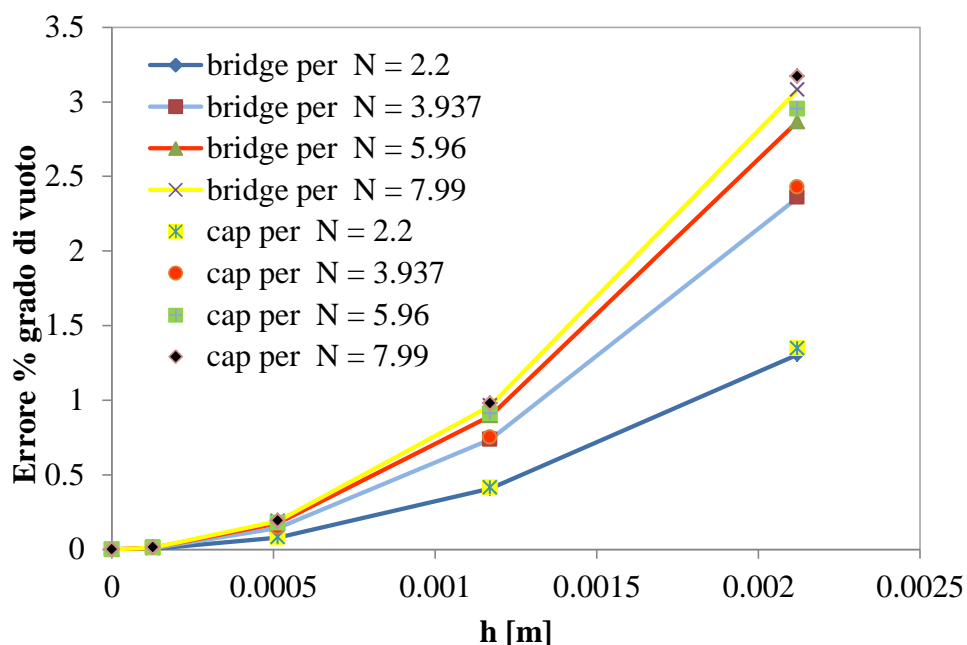


Figura 6.16 Variazione % dell'area catalitica rispetto al riferimento per i letti impaccati a diversi valori di  $N$ , implementando il *gap* e l'*overlap method*.

Le stesse analisi sono state svolte anche per il grado di vuoto. È stato visto che i metodi di modifica locale offrono una variazione minore rispetto ai metodi globali.

Come ci si aspettava, a parità della dimensione del *bridge* o del *cap* applicato tra le sfere, gli errori percentuali aumentano all'aumentare del numero dei punti di contatto sfera/sfera come per il caso dell'area catalitica.

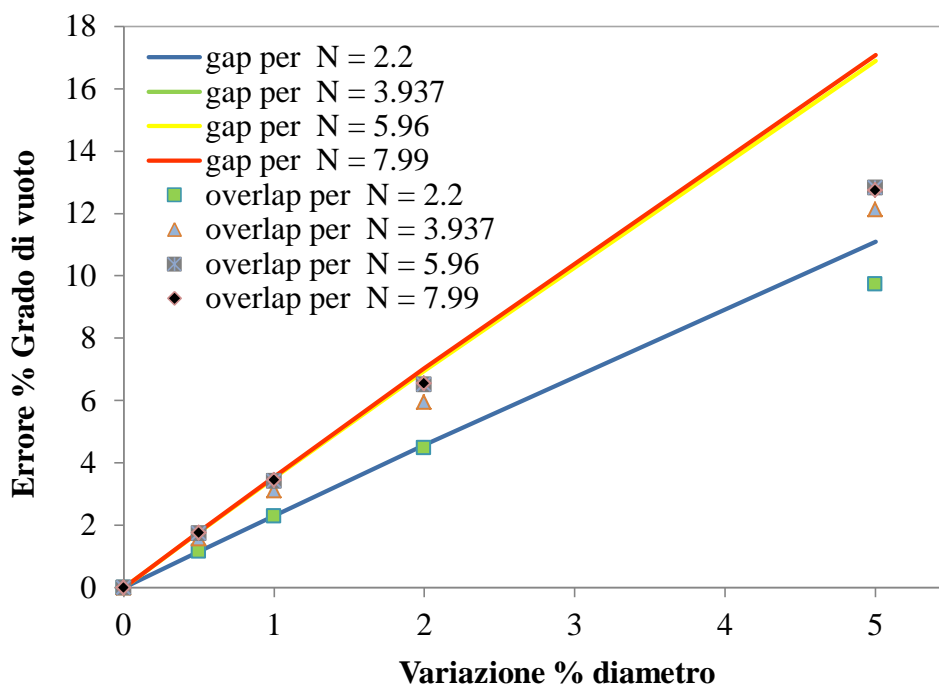


Figura 6.17 Variazione % dell'area catalitica rispetto al riferimento per i letti impaccati a diversi valori di  $N$ , implementando il *gap* e l'*overlap method*.

Analizzando i risultati ottenuti, è stato visto che i metodi di modifica locale garantiscono la minor variazione di area catalitica e di grado di vuoto di un letto impaccato di sfere. Facendo riferimento alla Figura 6.16 e alla Tabella 6.9 si nota che a parità di distanza  $h$  tra le sfere entrambi i metodi forniscono circa lo stesso errore sul grado di vuoto.

	<i>Err % Area per N = 2.2</i>	<i>Err % Area per N = 7.99</i>	<i>Err % vuoto per N = 2.2</i>	<i>Err % vuoto per N = 7.99</i>
<i>cap 1</i>	0.001	0.002	0.005	0.012
<i>cap 2</i>	0.034	0.052	0.080	0.189
<i>bridge 1</i>	0.766	1.174	0.005	0.012
<i>bridge 2</i>	2.747	4.208	0.08	0.188

Tabella 6.9 Errori di  $a_v$  e del grado di vuoto ottenuti per i metodi di modifica locale.



## 6.5 Simulazione di un caso studio

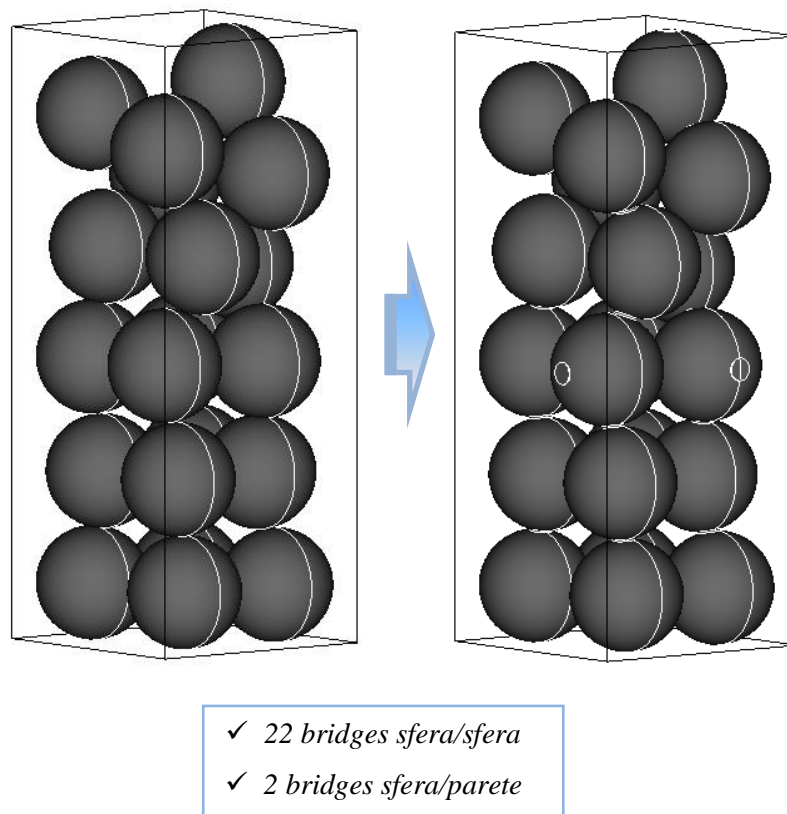
Lo studio dei fenomeni di scambio di materia e di calore con la CFD è uno strumento molto utilizzato per simulare reattori catalitici di importanza industriale. Il presente studio, si pone come obiettivo quello di condurre una simulazione CFD per un reattore catalitico eterogeneo attraverso il solver Multiregion catalyticConductivePimpleFOAM [2]. La metodologia descritta nel Paragrafo 3 per la modellazione CFD dei letti impaccati è stata implementata per modificare i punti di contatto e generare così la mesh ibrida.

### 6.5.1 Simulazione DEM e generazione della mesh

Un importante aspetto da considerare per condurre una soddisfacente simulazione CFD è quello di modificare, se è necessario, i punti di contatto. È stato visto che i metodi di modifica locale assicurano dei risultati più accurati rispetto a quelli globali, poiché offrono minime variazioni delle proprietà geometriche dei letti impaccati. Dai risultati ottenuti negli studi precedenti è stato possibile affermare che quando sono presenti anche i fenomeni di scambio termico è consigliabile implementare il modello elaborato da Dixon *et al.* [14], dal momento che è il più appropriato per garantire una soddisfacente modellazione del meccanismo di scambio termico. Pertanto, sulla base di questi risultati, il letto impaccato è stato modificato sostituendo i punti di contatto con dei bridges di dimensione assegnata. È stata scelta una dimensione dei bridges con rapporto  $r_b/d_s$  pari a 0.075 per garantire la minima perdita di area catalitica. Le caratteristiche del letto utilizzato sono riportate in Tabella 6.10.

$N$	$d_{\text{sfera}}$ [mm]	$r_b$ [mm]	lunghezza bridge sfera/sfera [mm]	lunghezza bridge sfera/parete [mm]
2.2	25.4	19.05	2.874	1.436

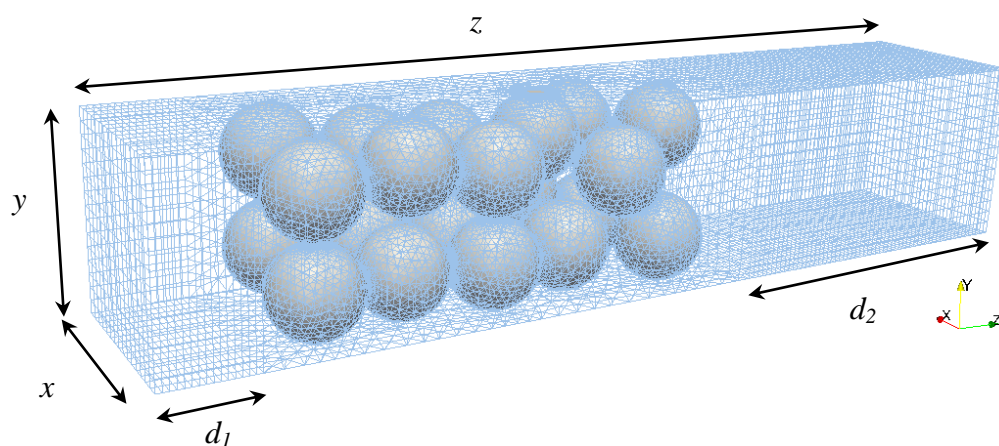
Tabella 6.10 Caratteristiche geometriche del letto impaccato random di 20 sfere.



**Figura 6.18** Impaccamento attraverso una simulazione *DEM* di un letto impaccato di 20 sfere e successiva modifica dei punti di contatto.

Dalle analisi condotte nei precedenti Capitoli si è visto che, al fine di poter confrontare i 4 metodi di modifica dei punti di contatto, l'analisi di convergenza della mesh è risultata fondamentale. Infatti per indagare in maniera dettagliata i campi di moto locali e i fenomeni di trasferimento di materia per i reagenti dalla fase fluida alla superficie catalitica è stato necessario infittire la mesh nelle zone interessate alle modifiche effettuate nei punti di contatto. Il presente studio non si pone come obiettivo quello di confrontare i risultati ottenuti con altre simulazioni CFD oppure con correlazioni sperimentali, ma quello di dimostrare la validità del modello implementato analizzando le variazioni delle proprietà geometriche e la qualità della mesh ottenute.

La mesh ibrida è stata generata attraverso lo *script* `packedbed_bridge-methodMultiRegion.py`. Le caratteristiche del reattore sono riportate in Figura 6.19.



$$x = y = 54 \text{ mm}$$

$$z = 203 \text{ mm}$$

$$d_s = 25.4 \text{ mm}$$

$$d_1 = d_s$$

$$d_2 = 3 d_s$$

Figura 6.19 Mesh ibrida 3D di un letto impaccato di 20 sfere con le dimensioni sopra riportate.

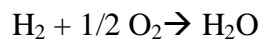
I parametri che descrivono la qualità della mesh sono riportate in Tabella 6.6.

	<i>N° di celle</i>	<i>MAX skewness</i>	<i>MAX non-ortogonalità</i>	<i>MAX Aspect Ratio</i>
<b>fluidRegion</b>	340808	0.761	71.014	6.98
<b>sfere</b>	845595	0.610	66.523	5.23
<b>bridges sfera/sfera</b>	155331	0.582	64.986	6.12
<b>bridges sfera/parete</b>	4308	0.632	61.033	5.01

Tabella 6.11 Caratteristiche della mesh ibrida utilizzata.

La mesh in totale contiene 1346042 celle e, come si può notare dai risultati riportati in Tabella 6.11, si hanno degli ottimi valori di indice di skewness e non ortogonalità grazie alla modifica dei punti di contatto.

La simulazione è stata condotta in condizioni adiabatiche utilizzando una cinetica semplice del primo ordine:



$$r = k_{cin} \cdot [\text{O}_2] \quad (6.12)$$

le condizioni operative utilizzate sono le stesse di quelle dello studio descritto nel Capitolo 5.

### 6.5.2 Risultati della simulazione

Per ridurre i tempi di calcolo la simulazione è stata effettuata in parallelo utilizzando 6 cores.

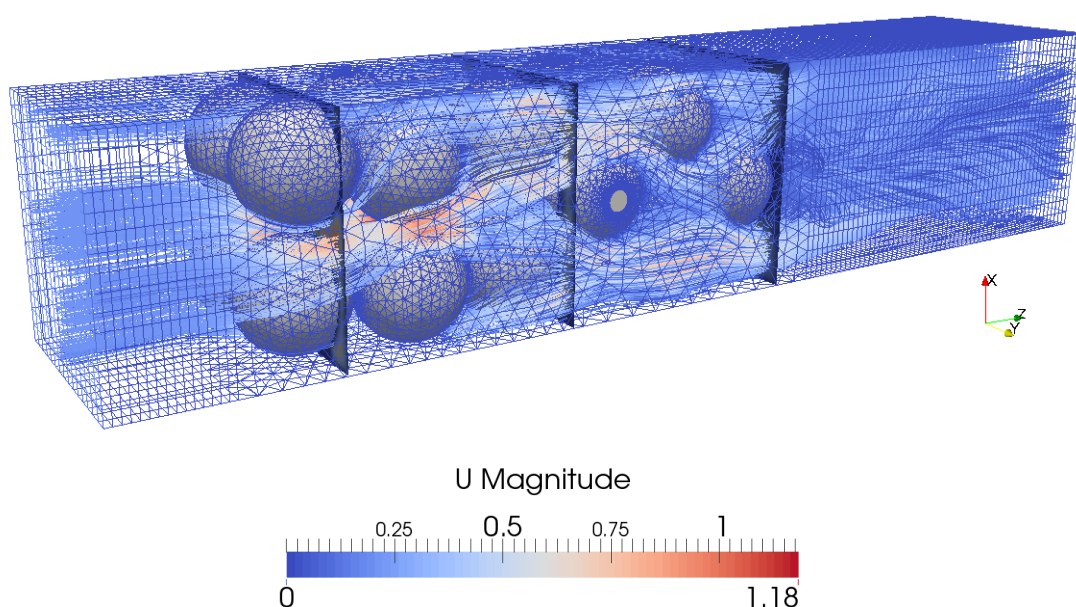


Figura 6.20 Campo di moto che si sviluppa all'interno del letto impaccato.

$a_v$	$Err \% a_v$	$\varepsilon$	$Err \% \varepsilon$	conversione % $\text{O}_2$
10.019	1.442	0.570	-0.022	26.86

Tabella 6.12 Caratteristiche geometriche del letto impaccato e risultati della simulazione.

Sostituendo i punti di contatto sfera/sfera e sfera/parete con *bridges* delle dimensioni riportate in Tabella 6.12 si ottengono degli errori percentuali sull'area catalitica per unità di volume ed sul grado di vuoto pari a rispettivamente 1.44% e -0.022%. Pertanto, si mostra come l'utilizzo del *bridge method* comporta una perdita di area

catalitica e di grado di vuoto a causa del volume occupato dai *bridges* nel letto impaccato. Analizzando in una porzione del letto impaccato, l'effetto della modifica dei punti di contatto rispetto alle frazioni massive delle specie chimiche che sono interessate ai fenomeni di scambio di materia, si dimostra che anche in una configurazione random si ha conversione dei reagenti in prodotti nelle zone dove sono presenti i *bridges*.

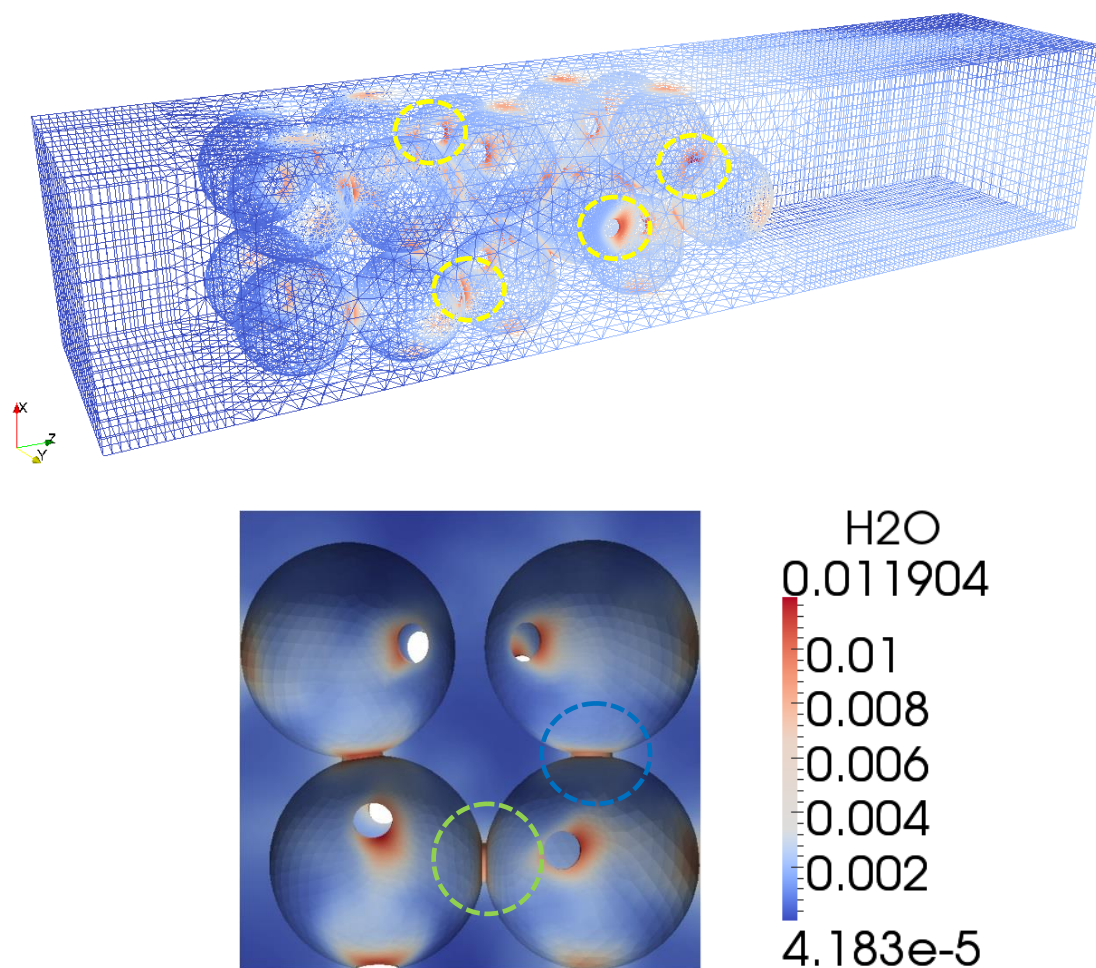
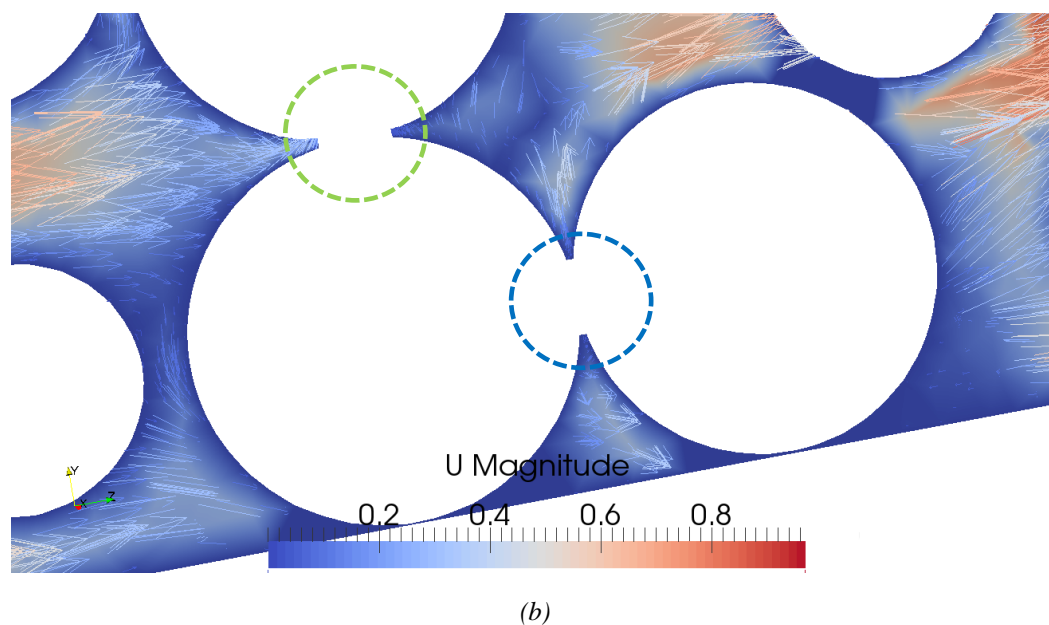
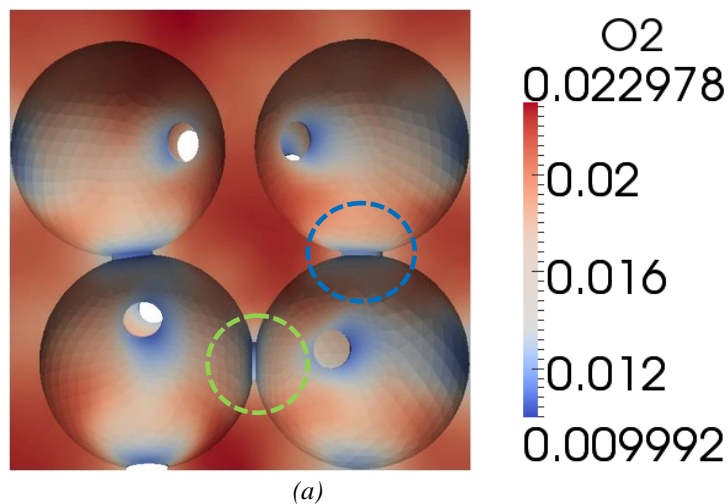


Figura 6.21 Frazione massiva di  $H_2O$  che si sviluppa sulla superficie delle sfere e nei *bridges*.

La Figura 6.22 (b) evidenzia i campi di moto locale per due configurazioni distinte dei *bridges*. Nella zona evidenziata in verde il flusso convettivo è diretto sul *bridge* sfera/sfera, mentre la zona evidenziata in blu mostra un *bridge* che non è direttamente coinvolto dal flusso convettivo, ma si hanno delle zone di ristagno e back mixing locali. In entrambi i casi, la superficie catalitica dei *bridges* è coinvolta

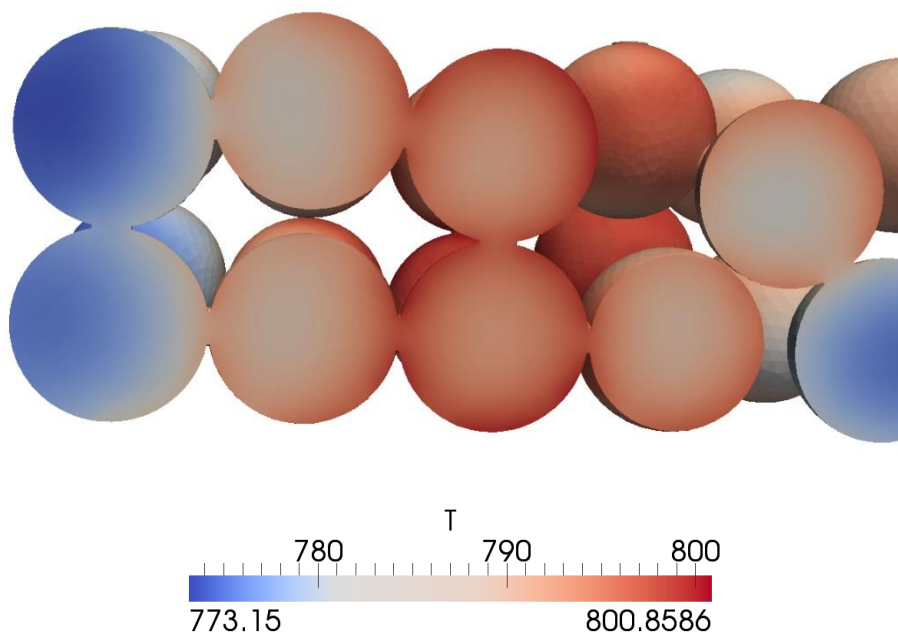
dalla reazione catalitica superficiale come dimostrato negli studi svolti nei Capitoli 3 e 4. Le zone di ristagno nelle zone a valle le sfere, garantiscono elevati tempi di contatto delle specie chimiche e quindi elevata disponibilità per reagire sulla superficie catalitica delle sfere.



**Figura 6.22** Frazione massiva di  $O_2$  che si sviluppa nei *bridges* (a); campo di moto locale attorno ai *bridges* (b).

Una volta che i reagenti reagiscono sulla superficie catalitica, la reazione catalitica superficiale genera calore che si propaga all'interno le sfere e nella regione fluida

circostante. La distribuzione di temperatura all'interno le sfere è fortemente influenzata dalle condizioni fluidodinamiche che si creano nel letto impaccato. In Figura 6.23 si mostra come le sfere catalitiche siano coinvolte dai fenomeni di scambio di calore in maniera diversa. Come ci si aspettava, le sfere direttamente coinvolte dal flusso convettivo scambiano più facilmente calore con la regione fluida, poiché il flusso entrante si trova a temperatura più bassa di quella di reazione e quindi raffredda le sfere. Le sfere che si trovano nella zona più interna del reattore si trovano a temperature maggiori.



**Figura 6.23** Distribuzione della temperatura all'interno le sfere.

Per la valutazione delle conducibilità termiche dei bridges sono stati utilizzati i valori suggeriti da Dixon *et al.* [14] e questo modello già convalidato negli studi precedenti ha permesso di modellare in maniera soddisfacente lo scambio termico tra le sfere e la regione fluida in un reattore catalitico eterogeneo.

## 6.6 Conclusioni

L'analisi sistematica affrontata in questo Capitolo è stata fondamentale per confrontare il grado di accuratezza con cui i letti impaccati sono stati modellati utilizzando i 4 metodi di modifica dei punti di contatto. Si è osservato che l'aspect ratio è una grandezza fondamentale che fornisce il numero dei punti di contatto sfera/sfera e sfera/parete. Dal punto di vista della modellazione CFD, operare con elevati valori di  $N$ , influisce sulle altre proprietà geometriche quali, il grado di vuoto e l' $a_v$ , ed obbliga l'utente a considerare la necessità di utilizzare il metodo di modifica dei punti di contatto più appropriato che garantisca il più alto grado di accuratezza nella modellazione del letto impaccato considerato. È stato dimostrato che i metodi di modifica locale sono dei perfetti candidati per essere implementati nelle simulazioni CFD che si pongono l'obiettivo di studiare il moto del fluido e le perdite di carico poiché offrono la medesima percentuale di errore. Dixon *et al.* [14] hanno dimostrato che con errori dell'1% sui gradi di vuoto le perdite di carico forniscono una variazione del 3% rispetto al valore esatto. Pertanto i risultati mostrati in Tabella 6.4 dimostrano che utilizzando piccole dimensioni del *bridge* e del *cap* tra le sfere si ottengono errori sulle perdite di carico inferiori allo 0.1% circa sulle perdite di carico. Per condurre studi CFD sui reattori catalitici eterogenei, bisogna verificare la variazione di superficie catalitica rispetto al reale letto impaccato non modificato, al fine di non ottenere dei risultati che sottostimano i valori di conversione reali delle specie chimiche coinvolte nelle reazioni catalitiche superficiali. Il *cap method* è il candidato ideale per modellare i punti di contatto nei reattori catalitici eterogenei isotermei, per simulazioni CFD in presenza di reazioni chimiche catalitiche superficiali. Quando, oltre ai fenomeni di scambio di materia, sono coinvolti i fenomeni di scambio termico, si consiglia di utilizzare il *bridge method*, poiché descrive meglio i fenomeni di scambio termico tra le sfere in accordo con i risultati ottenuti nei Capitoli precedenti.



---

## Bibliografia

- [1] Froment G.F. e Bischoff, K. B., “*Chemical Reactor Analysis and Design*”, 2nd Edition, John Wiley & Sons, (1990).
- [2] Maestri M. e Cuoci A., [www.catalyticfoam.polimi.it](http://www.catalyticfoam.polimi.it).
- [3] Fogler H.S., “*Elements of Chemical Reaction Engineering*”, Prentice-Hall, pp.645-738, New Jersey, (2006).
- [4] Rebughini S., “*Fundamental assessment of gas-to-particle mass transfer in micro-channel packed bed reactors*”, Tesi di Laurea Magistrale in Ingegneria Chimica, Politecnico di Milano, (2013).
- [5] Sissom L.E. e Pitts D.R., “*Elements of Transport Phenomena*”, McGraw-Hill, Kogakush Japan, (1972).
- [6] Kruggel-Emden, H., Sturm M., Wirtz S., Scherer V., “*Selection of an appropriate time integration scheme for the discrete element method (DEM)*”, *Computers & Chemical Engineering*, 32(10), pp. 2263-2279, (2008).
- [7] Diethelm K., Ford N., Freed A., “*A Predictor-Corrector Approach for the Numerical Solution of Fractional Differential Equations*”, *Nonlinear Dynamics*, 29(1), pp.3-22, (2002).
- [8] SDEC. “*An open-source software using a discrete element method to simulate granular material*”, Link website: [http://geo.hmg.inpg.fr/frederic/Research\\_project\\_Discrete\\_Element\\_Software.html](http://geo.hmg.inpg.fr/frederic/Research_project_Discrete_Element_Software.html).
- [9] Yade “*An extensible open-source framework for discrete numerical models, focused on Discrete Element Method*”, Link website: <https://www.yadedem.org/doc/>.
- [10] LAMMPS “*Molecular Dynamics Simulator*”, Link website: <http://lammps.sandia.gov/>.
- [11] LIGGGTHS “*Open Source Discrete Element Method Particle Simulation Code*”, Link website: <http://cfdem.dcs-computing.com/?q=OpenSourceDEM>

- [12] Zhu H.P., Rahman M., Aibing Y., Bridgwater J., Zulli P., “*Effect of particle properties on particle percolation behaviour in a packed bed*”, Minerals Engineering, 22(11), pp.961-969, (2009).
- [13] Padros C.B., “*Discrete element simulations with LIGGGHTS*”, Master of Science Thesis, Swansea University Prifysgol Abertawe, (2014).
- [14] Dixon A.G., Nijemeisland M., Stitt E.H., “*Systematic mesh development for 3D CFD simulation of fixed beds: Contact points study*”. Computers and Chemical Engineering, 48, pp. 135-153, (2013).

## 7 Conclusioni

Al fine di eliminare la problematica relativa alla generazione della mesh per reattori impaccati di sfere catalitiche, sono stati proposti i metodi di modifica globale e locale dei punti di contatto. Lo studio più recente condotto da Dixon *et al.* [1] ha mostrato che per condurre delle simulazioni CFD in regime turbolento con scambio termico la soluzione migliore è quella di sostituire il punto di contatto con un *bridge* con un rapporto  $r_b/d_s \leq 0.1$ .

In questo lavoro di Tesi, sono state presentati nuovi studi, volti ad indagare l'effetto della modifica dei punti di contatto sui fenomeni di trasporto in reattori catalitici eterogenei, in presenza di reazioni catalitiche eterogenee.

Per interpretare tale effetto sui fenomeni di trasporto di materia e di calore, sono stati eseguiti diversi studi CFD con configurazioni geometriche a complessità crescente.

Le geometrie e le *mesh* sono state generate attraverso il software SALOME che ha permesso di modificare i punti di contatto e generare delle *mesh* ibride in maniera automatica attraverso *script* in linguaggio `python`. Inizialmente lo studio è stato effettuato su sistemi semplici sfera/sfera per comprendere l'effetto della modifica locale e globale delle sfere catalitiche sulla reattività chimica e sul meccanismo di scambio termico. Successivamente per approfondire la precedente analisi è stata studiata una configurazione geometrica più complessa. In questa seconda configurazione geometrica proposta, i punti di contatto tra le sfere si trovano nelle zone stagnanti del reattore. Confrontando questi risultati ottenuti con i casi di riferimento si è visto che modificando i punti di contatto con il *cap method* si ottiene una soddisfacente modellazione, grazie al confronto sulla mesh non modificata che è la descrizione più accurata possibile.

In entrambe le configurazioni geometriche, i fenomeni di scambio di materia sono fortemente influenzati dalla variazione di superficie catalitica delle sfere, in

particolare, il *cap method*, che offre la minima variazione di superficie catalitica, fornisce dei risultati più accurati rispetto agli altri metodi di modifica dei punti di contatto.

Parallellamente sono stati indagati anche i fenomeni di scambio termico ed è stato dimostrato che il modello Dixon *et al.* [1] fornisce dei valori di resistenza termica inferiore rispetto agli altri metodi. Infatti, i due metodi *cap* e *gap* forniscono dei profili di temperatura poco realistici a causa della forte variazione della conducibilità termica nel punto di contatto. L'*overlap method*, a causa della sovrapposizione delle sfere offre un elevato contributo di trasferimento di calore per conduzione termica, che comporta un appiattimento dei profili di temperatura tra le sfere ed elevati valori di flusso termico. È stato dimostrato pertanto, che il modello Dixon *et al.* [1] è il più appropriato per descrivere correttamente il meccanismo di scambio termico tra le sfere.

Inoltre, al fine di indagare quale metodo sia il più appropriato per modellare i punti di contatto per studi CFD con la presenza di entrambi i fenomeni di scambio di materia e di calore, i due metodi di modifica locale sono stati tra loro confrontati utilizzando il risolutore *catalyticConductivePimpleFOAM* [2]. Lo studio ha dimostrato che il *bridge method* sebbene offra delle variazioni di area catalitica maggiore rispetto al *cap method*, tuttavia porta ad una migliore modellazione dello scambio termico tra le sfere e sfera/parete poichè consente una migliore modellazione dei fenomeni di scambio di materia e di calore in presenza di una reazione catalitica eterogenea.

Nell'ultima parte della Tesi i metodi di modifica locale e globale sono stati implementati in *script* in grado di modificare in maniera automatica i punti di contatto e generare successivamente la *mesh* ibrida per letti impaccati di sfere catalitiche. È stato possibile pertanto valutare le proprietà geometriche dei letti impaccati una volta modificati, ed è stato dimostrato che i metodi locali offrono una buona modellazione geometrica dei letti impaccati. In particolare, il metodo locale *cap* provoca la minima variazione della superficie catalitica delle sfere come ci si aspettava, il grado di vuoto, invece, viene descritto in maniera soddisfacente da entrambi i metodi di modifica locale.

Il presente lavoro di Tesi ha dimostrato che i metodi di modifica locale sono degli ottimi candidati per la modellazione geometrica dei punti di contatto in reattori impaccati di sfere catalitiche. È stato visto che per condurre delle simulazioni CFD in presenza dei soli fenomeni di scambio di materia in presenza di una reazione catalitica superficiale, il *cap method* è il modello che descrive meglio i fenomeni di scambio di materia e le perdite di carico per reattori catalitici di sfere. La presenza dei fenomeni di scambio termico costringe l'implementazione del modello Dixon *et al.* [1] al fine di descrivere correttamente i fenomeni di scambio termico all'interno le sfere.

Ulteriori sviluppi futuri che permettano una migliore modellazione dei reattori catalitici potrebbero essere:

- Parallelizzazione degli *script* creati per la generazione della *mesh* per letti impaccati di sfere al fine di minimizzare i tempi computazionali richiesti per la generazione di *mesh* di letti di centinaia o migliaia di sfere.
- Studio dell'effetto della modifica dei punti di contatto sui meccanismi di trasporto di materia e di calore all'interno delle sfere catalitiche.

## Bibliografia

- [1] Dixon A.G., Nijemeisland M., Stitt E.H., “*Systematic mesh development for 3D CFD simulation of fixed beds: Contact points study*”, *Computers and Chemical Engineering*, 48, pp. 135-153, (2013).
- [2] Maestri M. e Cuoci A., *Coupling CFD with detailed microkinetic modeling in heterogeneous catalysis*. *Chemical Engineering Science*, 96, pp. 106-117, (2013).
- [3] Maestri M. e Cuoci A., [www.catalyticfoam.polimi.it](http://www.catalyticfoam.polimi.it).
- [4] Web link: <http://www.openfoam.com/>

## APPENDICE - Script per le generazione delle mesh in SALOME

In questa sezione si riportano gli *script* in `python` utilizzati per generare le mesh ibride. Gli *scripts* sono stati scritti per la versione di SALOME 7.2.0. Il seguente link offre tutti i comandi che si possono utilizzare in SALOME attraverso linguaggio `python`:

[http://docs.salome-platform.org/5/geom/user/geompy\\_page.html](http://docs.salome-platform.org/5/geom/user/geompy_page.html)

Una volta aver generato lo *script* è possibile lanciarlo dal terminale (in *batch mode*) con il seguente comando:

```
./runAppli -t <nome_script.py>
```

Per semplicità di trattazione, i metodi di modifica locale *cap* e *bridge* sono stato accorpati in un unico *script*. Questo ha permesso così di evidenziare le differenze tra i due codici. In particolare, le righe dello *script* che appartengono al solo *bridge method* sono state evidenziate in verde, mentre le sole righe del codice che appartengono al *cap method* sono state evidenziate in rosso. Ovviamente, le righe lasciate in nero sono le stesse per entrambi i metodi di modifica locale.

Di seguito si riportano gli *script* utilizzati per generare le geometrie e mesh dei diversi casi studio.

### Scripts Capitolo 3

➤ *Studio dei fenomeni di scambio di materia*

```

#-----#
# Implementazione dei metodi di modifica locale: bridge e cap #
#
# >>bridge_method-cata_2sfere.py & cap_method-cata_2sfere.py <<#
##
#-----#

import sys
import salome
salome.salome_init()
theStudy = salome.myStudy
import salome_notebook
notebook = salome_notebook.NoteBook(theStudy)
#####
###      GENERAZIONE DELLA GEOMETRIA      ###
#####
import GEOM
from salome.geom import geomBuilder
import math
import SALOMEDS
geompy = geomBuilder.New(theStudy)
# Origine assi cartesiani
O = geompy.MakeVertex(0, 0, 0)
OX = geompy.MakeVectorDXDYDZ(1, 0, 0)
OY = geompy.MakeVectorDXDYDZ(0, 1, 0)
OZ = geompy.MakeVectorDXDYDZ(0, 0, 1)
# Scegli la dimensione del cap/bridge tra le sfere
radius = 0.0127 # m
d_p = 2 * radius # m
rb_dp = 0.05
rb = rb_dp * d_p
arcoseno = math.asin(rb/radius)
coseno = math.cos(arcoseno)
cap = radius * (1 - coseno )
## Creazione reattore
p1_box1 = geompy.MakeVertex( -0.0127,-0.01524, 0)
p2_box1 = geompy.MakeVertex( 0.0127, 0.01524, 0.011)
box1 = geompy.MakeBoxTwoPnt(p1_box1, p2_box1)
p1_box2 = geompy.MakeVertex( -0.0127,-0.01524, 0.011)
p2_box2 = geompy.MakeVertex( 0.0127, 0.01524, 0.045)
box2 = geompy.MakeBoxTwoPnt(p1_box2, p2_box2)
box3 = geompy.MakeTranslation(box1, 0, 0, 0.045)
p1_box4 = geompy.MakeVertex(-0.0127,-0.01524, 0.056)
p2_box4 = geompy.MakeVertex( 0.0127, 0.01524, 0.1524)
box4 = geompy.MakeBoxTwoPnt(p1_box4, p2_box4)
radius = 0.0127
vertexSfera = geompy.MakeVertex(0.0127, 0, 0.028)
sfera = geompy.MakeSpherePntR(vertexSfera, radius)
boxcut = geompy.MakeBoxDXDYDZ(0.05, 0.05, 0.05)
geomObj_5 = geompy.MakeTranslation(boxcut, 0.0127, -0.025, 0)
sfera_R = geompy.MakeCut(sfera, geomObj_5)

```



```

sfera_L = geompy.MakeMirrorByAxis(sfera_R, OZ)
semisfere = geompy.MakeFuse(sfera_R, sfera_L)
# Creazione bridge/cap
contactPoint = geompy.MakeVertex(0, 0, 0.028)
normalVector = geompy.MakeVectorDXDYDZ(1, 0, 0)
bridge = geompy.MakeCylinder(contactPoint, normalVector, rb, 2*cap)
bridgel = geompy.MakeTranslationVectorDistance(bridge, normalVector,
-cap)
cap_sfere = geompy.MakeCut(semisfere, bridgel)
sfereBridges = geompy.MakeFuse(cap_sfere, bridgel)

fluido_2BRIDGE = geompy.MakeCut(box2, sfereBridges)
partition = geompy.MakePartition([box1, fluido_2BRIDGE, box3, box4], [])

fluido_2CAP = geompy.MakeCut(box2, cap_sfere)
partition = geompy.MakePartition([box1, fluido_2CAP, box3, box4], [])

geompy.addToStudy(partition, "Partition")
# fluido 1
pc_box1 = geompy.MakeVertex( 0, 0, 0.0055)
fluid_1 = geompy.GetShapesNearPoint(partition, pc_box1,
geompy.ShapeType["SOLID"], 1e-6)
# fluido 2
pc_box2 = geompy.MakeVertex( 0, 0, 0.0195)
fluid_2 = geompy.GetShapesNearPoint(partition, pc_box2,
geompy.ShapeType["SOLID"], 1e-6)
# fluido 3
pc_box3 = geompy.MakeVertex( 0, 0, 0.0505)
fluid_3 = geompy.GetShapesNearPoint(partition, pc_box3 ,
geompy.ShapeType["SOLID"], 1e-6)
# fluido 4
pc_box4 = geompy.MakeVertex( 0, 0, 0.1042)
fluid_4 = geompy.GetShapesNearPoint(partition, pc_box4 ,
geompy.ShapeType["SOLID"], 1e-6)
## Assegnazione delle patches
# Groups Fluid_1
left = geompy.GetFaceNearPoint( fluid_1, geompy.MakeVertex( 0, 0, -
100 ))
inlet = geompy.CreateGroup( fluid_1, geompy.ShapeType["FACE"],
"inlet" )
geompy.UnionList( inlet, [ left ])
z1 = 0.011/2
front_box1 = geompy.GetFaceNearPoint( fluid_1, geompy.MakeVertex(
100, 0, z1 ))
frontWall_box1 = geompy.CreateGroup( fluid_1,
geompy.ShapeType["FACE"])
geompy.UnionList( frontWall_box1, [ front_box1 ])
back_box1 = geompy.GetFaceNearPoint( fluid_1, geompy.MakeVertex( -
100, 0, z1))
backWall_box1 = geompy.CreateGroup( fluid_1,
geompy.ShapeType["FACE"])
geompy.UnionList( backWall_box1, [ back_box1 ])
up_box1 = geompy.GetFaceNearPoint( fluid_1, geompy.MakeVertex( 0,
100, z1 ))
upWall_box1 = geompy.CreateGroup( fluid_1, geompy.ShapeType["FACE"])
geompy.UnionList( upWall_box1, [ up_box1 ])

```

```
down_box1 = geompy.GetFaceNearPoint( fluid_1, geompy.MakeVertex( 0,
-100, z1 ))
downWall_box1 = geompy.CreateGroup( fluid_1,
geompy.ShapeType["FACE"])
geompy.UnionList( downWall_box1, [ down_box1 ])
fixedWall_fluid1 = geompy.UnionListOfGroups( [frontWall_box1,
backWall_box1, upWall_box1, downWall_box1] )
geompy.addToStudyInFather(fluid_1,
fixedWall_fluid1, 'fixedWall_fluid1')
# Groups sul Fluid_2
all = geompy.SubShapeAll( fluid_2, geompy.ShapeType["FACE"] )
z2 = 0.0125
left_fluid2 = geompy.GetFaceNearPoint( fluid_2, geompy.MakeVertex(
0, 0, -100 ))
inlet_fluid2 = geompy.CreateGroup( fluid_2,
geompy.ShapeType["FACE"])
geompy.UnionList( inlet_fluid2, [ left_fluid2 ])
right_fluid2 = geompy.GetFaceNearPoint( fluid_2, geompy.MakeVertex(
0, 0, 100 ))
outlet_fluid2 = geompy.CreateGroup( fluid_2,
geompy.ShapeType["FACE"])
geompy.UnionList( outlet_fluid2, [ right_fluid2 ])
front_fluid2 = geompy.GetFaceNearPoint( fluid_2, geompy.MakeVertex(
200, 0, z2 ))
frontWall_box2 = geompy.CreateGroup( fluid_2,
geompy.ShapeType["FACE"])
geompy.UnionList( frontWall_box2, [ front_fluid2 ])
back_fluid2 = geompy.GetFaceNearPoint( fluid_2, geompy.MakeVertex( -
200, 0, z2 ))
backWall_box2 = geompy.CreateGroup( fluid_2,
geompy.ShapeType["FACE"])
geompy.UnionList( backWall_box2, [ back_fluid2 ])
up_fluid2 = geompy.GetFaceNearPoint( fluid_2, geompy.MakeVertex( 0,
200, z2 ))
upWall_box2 = geompy.CreateGroup( fluid_2, geompy.ShapeType["FACE"])
geompy.UnionList( upWall_box2, [ up_fluid2 ])
down_fluid2 = geompy.GetFaceNearPoint( fluid_2, geompy.MakeVertex(
0, -200, z2 ))
downWall_box2 = geompy.CreateGroup( fluid_2,
geompy.ShapeType["FACE"])
geompy.UnionList( downWall_box2, [ down_fluid2 ])
reactingWall = geompy.CreateGroup( fluid_2,
geompy.ShapeType["FACE"], "catalyticWall")
geompy.UnionList( reactingWall, all)
geompy.DifferenceList( reactingWall, [ inlet_fluid2, outlet_fluid2,
frontWall_box2, backWall_box2, upWall_box2, downWall_box2] )
# Valutazione della superficie catalitica #
catalytic_Area = geompy.BasicProperties(reactingWall)
print " Area catalitica [m2] = ", catalytic_Area[1]
# Union of groups
fixedWall_fluid2 = geompy.UnionListOfGroups( [frontWall_box2,
backWall_box2, upWall_box2, downWall_box2] )
geompy.addToStudyInFather(fluid_2, fixedWall_fluid2, 'fixedWall_fluid2
')
# Groups Fluid_3
z3 = 0.0505
```

---

```
front_box3 = geompy.GetFaceNearPoint( fluid_3, geompy.MakeVertex(
100, 0, z3 ))
frontWall_box3 = geompy.CreateGroup( fluid_3,
geompy.ShapeType["FACE"])
geompy.UnionList( frontWall_box3, [ front_box3 ])
back_box3 = geompy.GetFaceNearPoint( fluid_3, geompy.MakeVertex( -
100, 0, z3 ))
backWall_box3 = geompy.CreateGroup( fluid_3,
geompy.ShapeType["FACE"])
geompy.UnionList( backWall_box3, [ back_box3 ])
up_box3 = geompy.GetFaceNearPoint( fluid_3, geompy.MakeVertex( 0,
100, z3 ))
upWall_box3 = geompy.CreateGroup( fluid_3, geompy.ShapeType["FACE"])
geompy.UnionList( upWall_box3, [ up_box3 ])
down_box3 = geompy.GetFaceNearPoint( fluid_3, geompy.MakeVertex( 0,
-100, z3 ))
downWall_box3 = geompy.CreateGroup( fluid_3,
geompy.ShapeType["FACE"])
geompy.UnionList( downWall_box3, [ down_box3 ])
fixedWall_fluid3 = geompy.UnionListOfGroups( [frontWall_box3,
backWall_box3, upWall_box3, downWall_box3] )
geompy.addToStudyInFather(fluid_3,
fixedWall_fluid3, 'fixedWall_fluid3')
# Groups Fluid_4
z4 = 0.1032
front_box4 = geompy.GetFaceNearPoint( fluid_4, geompy.MakeVertex(
100, 0, z4 ))
frontWall_box4 = geompy.CreateGroup( fluid_4,
geompy.ShapeType["FACE"])
geompy.UnionList( frontWall_box4, [ front_box4 ])
back_box4 = geompy.GetFaceNearPoint( fluid_4, geompy.MakeVertex( -
100, 0, z4 ))
backWall_box4 = geompy.CreateGroup( fluid_4,
geompy.ShapeType["FACE"])
geompy.UnionList( backWall_box4, [ back_box4 ])
up_box4 = geompy.GetFaceNearPoint( fluid_4, geompy.MakeVertex( 0,
100, z4 ))
upWall_box4 = geompy.CreateGroup( fluid_4, geompy.ShapeType["FACE"])
geompy.UnionList( upWall_box4, [ up_box4 ])
down_box4 = geompy.GetFaceNearPoint( fluid_4, geompy.MakeVertex( 0,
-100, z4 ))
downWall_box4 = geompy.CreateGroup( fluid_4,
geompy.ShapeType["FACE"])
geompy.UnionList( downWall_box4, [ down_box4 ])
right = geompy.GetFaceNearPoint( fluid_4, geompy.MakeVertex( 0, 0,
100 ))
outlet = geompy.CreateGroup( fluid_4, geompy.ShapeType["FACE"],
"outlet" )
geompy.UnionList( outlet, [ right ])
# Union of groups
fixedWall_fluid4 = geompy.UnionListOfGroups( [frontWall_box4,
backWall_box4, upWall_box4, downWall_box4] )
geompy.addToStudyInFather(fluid_4,
fixedWall_fluid4, 'fixedWall_fluid4')
```

---

```

inertWall = geompy.CreateGroup(partition, geompy.ShapeType["FACE"])
geompy.UnionIDs(inertWall, [34, 4, 31, 14, 55, 45, 68, 38, 109, 89,
106, 96, 133, 113, 130, 120])
geompy.addToStudyInFather( partition, inertWall, 'inertWall' )

```

```

inertWall = geompy.CreateGroup(partition, geompy.ShapeType["FACE"])
geompy.UnionIDs(inertWall, [34, 4, 31, 14, 55, 45, 68, 38, 110, 90,
107, 97, 134, 114, 131, 121])
geompy.addToStudyInFather( partition, inertWall, 'inertWall')

```

```

#####
###          GENERAZIONE DELLA MESH          ###
#####
import SMESH, SALOMEDS
from salome.smesh import smeshBuilder
smesh = smeshBuilder.New(theStudy)
from salome.NETGENPlugin import NETGENPluginBuilder
mesh = smesh.Mesh( partition, "Fluid mesh")
# NETGEN on fluid - to be computed after boxes
mesh.Tetrahedron( smeshBuilder.NETGEN_3D )
ng12 = mesh.Triangle( smeshBuilder.NETGEN_1D2D )
ng12.SetMaxSize( 0.08 )
# sub-meshes on boxes - to be meshed first
algo1D = mesh.Segment( fluid_1 )
algo1D.NumberOfSegments( 20 )
mesh.Quadrangle( fluid_1 )
mesh.Hexahedron( fluid_1 )
algo1D = mesh.Segment( fluid_3 )
algo1D.NumberOfSegments( 20 )
mesh.Quadrangle( fluid_3 )
mesh.Hexahedron( fluid_3 )
algo1D = mesh.Segment( fluid_4 )
algo1D.NumberOfSegments( 20 )
mesh.Quadrangle( fluid_4 )
mesh.Hexahedron( fluid_4 )
NETGEN_2D3D = mesh.Tetrahedron(algo=smeshBuilder.NETGEN_1D2D3D )
NETGEN_3D_Parameters = NETGEN_2D3D.Parameters()
NETGEN_3D_Parameters.SetMaxSize( 0.08 )
NETGEN_3D_Parameters.SetSecondOrder( 0 )
NETGEN_3D_Parameters.SetOptimize( 1 )
NETGEN_3D_Parameters.SetFineness( 4 )
NETGEN_3D_Parameters.SetMinSize( 0.001 )
mesh.Compute()
# Assegnazione delle patches
aStudyBuilder = theStudy.NewBuilder()
inlet = mesh.GroupOnGeom(inlet, 'inlet', SMESH.FACE)
outlet = mesh.GroupOnGeom(outlet, 'outlet', SMESH.FACE)
reactingWall =
mesh.GroupOnGeom(reactingWall, 'reactingWall', SMESH.FACE)
inertWall = mesh.GroupOnGeom(inertWall, 'inertWall', SMESH.FACE)

# Export the mesh
sys.path.insert( 0, r'/home/salvo/Scrivania')
import salomeToOpenFOAM

salomeToOpenFOAM.exportToFoam(mesh, '/home/salvo/Mesh_1/polyMesh')

```

```

#-----#
# Implementazione dei metodi di modifica globale: gap e overlap #
# #
# >>gap_method-cata_2sfere.py &overlap_method-cata_2sfere.py <<#
##
#-----#

import sys
import salome
salome.salome_init()
theStudy = salome.myStudy
import salome_notebook
notebook = salome_notebook.NoteBook(theStudy)
#####
###          GENERAZIONE DELLA GEOMETRIA          ###
#####
import GEOM
from salome.geom import geomBuilder
import math
import SALOMEDS
geompy = geomBuilder.New(theStudy)
# Origine assi cartesiani
O = geompy.MakeVertex(0, 0, 0)
OX = geompy.MakeVectorDXDYDZ(1, 0, 0)
OY = geompy.MakeVectorDXDYDZ(0, 1, 0)
OZ = geompy.MakeVectorDXDYDZ(0, 0, 1)
# Dimensioni sfere
radius = 0.0127 # m
d_nom = 2 * radius # m
#-----#
# Scegli la dimensione del gap/overlap tra le sfere
Var = 0.05 # %

d_gap = d_nom * (1 - Var)
radius_gap = d_gap/2

d_overlap = d_nom / (1 - Var)
radius_overlap = d_overlap/2

#-----#
## Creazione reattore
p1_box1 = geompy.MakeVertex(-0.0127,-0.01524, 0)
p2_box1 = geompy.MakeVertex( 0.0127, 0.01524, 0.011)
box1 = geompy.MakeBoxTwoPnt(p1_box1, p2_box1)
p1_box2 = geompy.MakeVertex(-0.0127,-0.01524, 0.011)
p2_box2 = geompy.MakeVertex( 0.0127, 0.01524, 0.045)
box2 = geompy.MakeBoxTwoPnt(p1_box2, p2_box2)
box3 = geompy.MakeTranslation(box1, 0, 0, 0.045)
p1_box4 = geompy.MakeVertex(-0.0127,-0.01524, 0.056)
p2_box4 = geompy.MakeVertex( 0.0127, 0.01524, 0.1524)
box4 = geompy.MakeBoxTwoPnt(p1_box4, p2_box4)
radius = 0.0127
vertexSfera = geompy.MakeVertex(0.0127, 0, 0.028)
sfera = geompy.MakeSpherePntR(vertexSfera, radius_gap)
boxcut = geompy.MakeBoxDXDYDZ(0.05, 0.05, 0.05)
geomObj_5 = geompy.MakeTranslation(boxcut, 0.0127, -0.025, 0)

```

---

```

sfera_R = geompy.MakeCut(sfera, geomObj_5)
sfera_L = geompy.MakeMirrorByAxis(sfera_R, OZ)
semisfere = geompy.MakeFuse(sfera_R, sfera_L)
fluido_2 = geompy.MakeCut(box2, semisfere)
partition = geompy.MakePartition( [box1, fluido_2, box3, box4],[ ] )
geompy.addToStudy(partition, "Partition")
# fluido 1
pc_box1 = geompy.MakeVertex( 0, 0, 0.0055)
fluid_1 = geompy.GetShapesNearPoint(partition, pc_box1,
geompy.ShapeType["SOLID"], 1e-6)
geompy.addToStudyInFather(partition, fluid_1, 'fluid_1')
# fluido 2
pc_box2 = geompy.MakeVertex( 0, 0, 0.0195)
fluid_2 = geompy.GetShapesNearPoint(partition, pc_box2,
geompy.ShapeType["SOLID"], 1e-6)
geompy.addToStudyInFather(partition, fluid_2, 'fluid_2')
# fluido 3
pc_box3 = geompy.MakeVertex( 0, 0, 0.0505)
fluid_3 = geompy.GetShapesNearPoint(partition, pc_box3 ,
geompy.ShapeType["SOLID"], 1e-6)
geompy.addToStudyInFather(partition, fluid_3, 'fluid_3')
# fluido 4
pc_box4 = geompy.MakeVertex( 0, 0, 0.1042)
fluid_4 = geompy.GetShapesNearPoint(partition, pc_box4 ,
geompy.ShapeType["SOLID"], 1e-6)
geompy.addToStudyInFather(partition, fluid_4, 'fluid_4')
## Assegnazione delle patches
# Groups Fluid_1
left = geompy.GetFaceNearPoint( fluid_1, geompy.MakeVertex( 0, 0, -
100 ))
inlet = geompy.CreateGroup( fluid_1, geompy.ShapeType["FACE"],
"inlet" )
geompy.UnionList( inlet, [ left ])
z1 = 0.011/2
front_box1 = geompy.GetFaceNearPoint( fluid_1, geompy.MakeVertex(
100, 0, z1 ))
frontWall_box1 = geompy.CreateGroup( fluid_1,
geompy.ShapeType["FACE"])
geompy.UnionList( frontWall_box1, [ front_box1 ])
back_box1 = geompy.GetFaceNearPoint( fluid_1, geompy.MakeVertex( -
100, 0, z1))
backWall_box1 = geompy.CreateGroup( fluid_1,
geompy.ShapeType["FACE"])
geompy.UnionList( backWall_box1, [ back_box1 ])
up_box1 = geompy.GetFaceNearPoint( fluid_1, geompy.MakeVertex( 0,
100, z1 ))
upWall_box1 = geompy.CreateGroup( fluid_1, geompy.ShapeType["FACE"])
geompy.UnionList( upWall_box1, [ up_box1 ])
down_box1 = geompy.GetFaceNearPoint( fluid_1, geompy.MakeVertex( 0,
-100, z1 ))
downWall_box1 = geompy.CreateGroup( fluid_1,
geompy.ShapeType["FACE"])
geompy.UnionList( downWall_box1, [ down_box1 ])
fixedWall_fluid1 = geompy.UnionListOfGroups( [frontWall_box1,
backWall_box1, upWall_box1, downWall_box1] )

```

---

---

```

geompy.addToStudyInFather(fluid_1,
fixedWall_fluid1,'fixedWall_fluid1')
# Groups sul Fluid_2
all = geompy.SubShapeAll( fluid_2, geompy.ShapeType["FACE"] )
z2 = 0.0125
left_fluid2 = geompy.GetFaceNearPoint( fluid_2, geompy.MakeVertex(
0, 0, -100 ))
inlet_fluid2 = geompy.CreateGroup( fluid_2,
geompy.ShapeType["FACE"])
geompy.UnionList( inlet_fluid2, [ left_fluid2 ])
right_fluid2 = geompy.GetFaceNearPoint( fluid_2, geompy.MakeVertex(
0, 0, 100 ))
outlet_fluid2 = geompy.CreateGroup( fluid_2,
geompy.ShapeType["FACE"])
geompy.UnionList( outlet_fluid2, [ right_fluid2 ])
front_fluid2 = geompy.GetFaceNearPoint( fluid_2, geompy.MakeVertex(
200, 0, z2 ))
frontWall_box2 = geompy.CreateGroup( fluid_2,
geompy.ShapeType["FACE"])
geompy.UnionList( frontWall_box2, [ front_fluid2 ])
back_fluid2 = geompy.GetFaceNearPoint( fluid_2, geompy.MakeVertex( -
200, 0, z2 ))
backWall_box2 = geompy.CreateGroup( fluid_2,
geompy.ShapeType["FACE"])
geompy.UnionList( backWall_box2, [ back_fluid2 ])
up_fluid2 = geompy.GetFaceNearPoint( fluid_2, geompy.MakeVertex( 0,
200, z2 ))
upWall_box2 = geompy.CreateGroup( fluid_2, geompy.ShapeType["FACE"])
geompy.UnionList( upWall_box2, [ up_fluid2 ])
down_fluid2 = geompy.GetFaceNearPoint( fluid_2, geompy.MakeVertex(
0, -200, z2 ))
downWall_box2 = geompy.CreateGroup( fluid_2,
geompy.ShapeType["FACE"])
geompy.UnionList( downWall_box2, [ down_fluid2 ])
reactingWall = geompy.CreateGroup( fluid_2,
geompy.ShapeType["FACE"], "catalyticWall")
geompy.UnionList( reactingWall, all)
geompy.DifferenceList( reactingWall, [ inlet_fluid2, outlet_fluid2,
frontWall_box2, backWall_box2, upWall_box2, downWall_box2] )
# Valutazione della superficie catalitica #
catalytic_Area = geompy.BasicProperties(reactingWall)
print " Area catalitica [m2] = ", catalytic_Area[1]
# Union of groups
fixedWall_fluid2 = geompy.UnionListOfGroups( [frontWall_box2,
backWall_box2, upWall_box2, downWall_box2] )
geompy.addToStudyInFather(fluid_2,
fixedWall_fluid2,'fixedWall_fluid2')
# Groups Fluid_3
z3 = 0.0505
front_box3 = geompy.GetFaceNearPoint( fluid_3, geompy.MakeVertex(
100, 0, z3 ))
frontWall_box3 = geompy.CreateGroup( fluid_3,
geompy.ShapeType["FACE"])
geompy.UnionList( frontWall_box3, [ front_box3 ])
back_box3 = geompy.GetFaceNearPoint( fluid_3, geompy.MakeVertex( -
100, 0, z3 ))

```

---

```

backWall_box3 = geompy.CreateGroup( fluid_3,
geompy.ShapeType["FACE"])
geompy.UnionList( backWall_box3, [ back_box3 ])
up_box3 = geompy.GetFaceNearPoint( fluid_3, geompy.MakeVertex( 0,
100, z3 ))
upWall_box3 = geompy.CreateGroup( fluid_3, geompy.ShapeType["FACE"])
geompy.UnionList( upWall_box3, [ up_box3 ])
down_box3 = geompy.GetFaceNearPoint( fluid_3, geompy.MakeVertex( 0,
-100, z3 ))
downWall_box3 = geompy.CreateGroup( fluid_3,
geompy.ShapeType["FACE"])
geompy.UnionList( downWall_box3, [ down_box3 ])
# Union of groups
fixedWall_fluid3 = geompy.UnionListOfGroups( [frontWall_box3,
backWall_box3, upWall_box3, downWall_box3] )
geompy.addToStudyInFather(fluid_3,
fixedWall_fluid3, 'fixedWall_fluid3')
# Groups Fluid_4
z4 = 0.1032
front_box4 = geompy.GetFaceNearPoint( fluid_4, geompy.MakeVertex(
100, 0, z4 ))
frontWall_box4 = geompy.CreateGroup( fluid_4,
geompy.ShapeType["FACE"])
geompy.UnionList( frontWall_box4, [ front_box4 ])
back_box4 = geompy.GetFaceNearPoint( fluid_4, geompy.MakeVertex( -
100, 0, z4 ))
backWall_box4 = geompy.CreateGroup( fluid_4,
geompy.ShapeType["FACE"])
geompy.UnionList( backWall_box4, [ back_box4 ])
up_box4 = geompy.GetFaceNearPoint( fluid_4, geompy.MakeVertex( 0,
100, z4 ))
upWall_box4 = geompy.CreateGroup( fluid_4, geompy.ShapeType["FACE"])
geompy.UnionList( upWall_box4, [ up_box4 ])
down_box4 = geompy.GetFaceNearPoint( fluid_4, geompy.MakeVertex( 0,
-100, z4 ))
downWall_box4 = geompy.CreateGroup( fluid_4,
geompy.ShapeType["FACE"])
geompy.UnionList( downWall_box4, [ down_box4 ])
right = geompy.GetFaceNearPoint( fluid_4, geompy.MakeVertex( 0, 0,
100 ))
outlet = geompy.CreateGroup( fluid_4, geompy.ShapeType["FACE"],
"outlet" )
geompy.UnionList( outlet, [ right ])
fixedWall_fluid4 = geompy.UnionListOfGroups( [frontWall_box4,
backWall_box4, upWall_box4, downWall_box4] )
geompy.addToStudyInFather(fluid_4,
fixedWall_fluid4, 'fixedWall_fluid4')

inertWall = geompy.CreateGroup(partition, geompy.ShapeType["FACE"])
geompy.UnionIDs(inertWall, [34, 4, 31, 14, 55, 45, 68, 38, 55, 45,
68, 97, 80, 100, 87, 111, 124, 104, 121])

inertWall = geompy.CreateGroup(partition, geompy.ShapeType["FACE"])
geompy.UnionIDs(inertWall, [34, 4, 31, 14, 55, 45, 68, 38, 101, 84,
104, 91, 115, 128, 125, 108])

```



```
#####  
###          GENERAZIONE DELLA MESH          ###  
#####  
import SMESH, SALOMEDS  
from salome.smesh import smeshBuilder  
smesh = smeshBuilder.New(theStudy)  
from salome.NETGENPlugin import NETGENPluginBuilder  
mesh = smesh.Mesh( partition, "Fluid mesh")  
# NETGEN on fluid - to be computed after boxes  
mesh.Tetrahedron( smeshBuilder.NETGEN_3D )  
ng12 = mesh.Triangle( smeshBuilder.NETGEN_1D2D )  
ng12.SetMaxSize( 0.08 )  
# sub-meshes on boxes - to be meshed first  
algo1D = mesh.Segment( fluid_1 )  
algo1D.NumberOfSegments( 20 )  
mesh.Quadrangle( fluid_1 )  
mesh.Hexahedron( fluid_1 )  
algo1D = mesh.Segment( fluid_3 )  
algo1D.NumberOfSegments( 20 )  
mesh.Quadrangle( fluid_3 )  
mesh.Hexahedron( fluid_3 )  
algo1D = mesh.Segment( fluid_4 )  
algo1D.NumberOfSegments( 20 )  
mesh.Quadrangle( fluid_4 )  
mesh.Hexahedron( fluid_4 )  
NETGEN_2D3D = mesh.Tetrahedron(algo=smeshBuilder.NETGEN_1D2D3D )  
NETGEN_3D_Parameters = NETGEN_2D3D.Parameters()  
NETGEN_3D_Parameters.SetMaxSize( 0.08 )  
NETGEN_3D_Parameters.SetSecondOrder( 0 )  
NETGEN_3D_Parameters.SetOptimize( 1 )  
NETGEN_3D_Parameters.SetFineness( 4 )  
NETGEN_3D_Parameters.SetMinSize( 0.001 )  
mesh.Compute()  
aStudyBuilder = theStudy.NewBuilder()  
inlet = mesh.GroupOnGeom(inlet, 'inlet', SMESH.FACE)  
outlet = mesh.GroupOnGeom(outlet, 'outlet', SMESH.FACE)  
reactingWall =  
mesh.GroupOnGeom(reactingWall, 'reactingWall', SMESH.FACE)  
inertWall = mesh.GroupOnGeom(inertWall, 'inertWall', SMESH.FACE)  
  
# Export the mesh  
sys.path.insert( 0, r'/home/salvo/Scrivania')  
import salomeToOpenFOAM  
salomeToOpenFOAM.exportToFoam(mesh, '/home/salvo/Mesh_1/polyMesh')
```

## Scripts Capitolo 6

- Scripts per la determinazione delle proprietà geometriche dei letti impaccati.

```
#####
# -----#
#   Determinazione delle proprieta' geometriche di letti impaccati#
#   Modellazione dei punti di contatto i metodi di modifica locale #
# -----#
#   properties_packedBed-localMethods.py #
#
#####
import csv
import salome
salome.salome_init()
theStudy = salome.myStudy
import GEOM
from salome.geom import geomBuilder
geompy = geomBuilder.New(theStudy)
#-----#
#           Importa le sfere
#-----#
inputFileName = "200" # nome file input
vertexes = []
spheres = []
centres = []

with open(inputFileName, 'rb') as csvFile:

    csvReader = csv.reader(csvFile, delimiter=' ')

    thisIsHeader = True
    for row in csvReader:

        if thisIsHeader == True:

            if "'radius'" in str(row) and "'x', 'y', 'z'" in
str(row):

                radiusColumnIndex = row.index('radius') - 2

                xColumnIndex = row.index('x') - 2
                yColumnIndex = row.index('y') - 2
                zColumnIndex = row.index('z') - 2

                thisIsHeader = False

            else:

                [x, y, z] = [
                    float(row[xColumnIndex]),
                    float(row[yColumnIndex]),
                    float(row[zColumnIndex]),
                    ]

                radius = float(row[radiusColumnIndex])
```

```

vertex = geompy.MakeVertex(x, y, z)
vertexes.append(vertex)

sphere = geompy.MakeSpherePntR(vertex, radius)
spheres.append(sphere)
centres.append(vertex)

compound = geompy.MakeCompound(spheres)
geompy.addToStudy(compound, "spheres")
N_sfere = len(spheres)
#-----#
#           Scegli il rapporto rb/dp desiderato
#-----#
import math
d_p = 2 * radius # m
rb_dp = 0.05
rb = rb_dp * d_p
arcoseno = math.asin(rb/radius)
coseno = math.cos(arcoseno)
cap = radius * (1 - coseno)
#-----#
#           Scegli il rapporto 2h/dp desiderato
#-----#
# h = cap applicato su ciascuna sfere
# hh = cap applicato su entrambe le sfere
d_p = radius * 2
hh_dp = 0.00501256289338
h = hh_dp * d_p/2
cap = h
contactDistance = 6.4e-05
infiniteNumber = 9999
#-----#
#           Creazione del box contenente le sfere
#-----#
searchBox = geompy.BoundingBox(compound)
aBB = geompy.MakeBoundingBox(compound) # coordinate
Xmin = searchBox[0]
Xmax = searchBox[1]
Ymin= searchBox[2]
Ymax= searchBox[3]
Zmin =searchBox[4]
Zmax = searchBox[5]
p1_box = geompy.MakeVertex( searchBox[0], searchBox[2],
searchBox[4])
p2_box = geompy.MakeVertex( searchBox[1], searchBox[3],
searchBox[5])
box = geompy.MakeBoxTwoPnt(p1_box, p2_box)
geompy.addToStudy(box, "Box")
boxFaces1 = geompy.SubShapeAll(box, geompy.ShapeType["FACE"][0]
boxFaces2 = geompy.SubShapeAll(box, geompy.ShapeType["FACE"][1]
boxFaces3 = geompy.SubShapeAll(box, geompy.ShapeType["FACE"][2]
boxFaces4 = geompy.SubShapeAll(box, geompy.ShapeType["FACE"][3]
boxFaces5 = geompy.SubShapeAll(box, geompy.ShapeType["FACE"][4]
boxFaces6 = geompy.SubShapeAll(box, geompy.ShapeType["FACE"][5]

```

---

```

Faces = [boxFaces1, boxFaces2, boxFaces3, boxFaces4, boxFaces5,
boxFaces6]
#-----#
#           Generazione del cap sfera/sfera
#-----#
nbSpheres = len(spheres)
newSpheres = []

for i in range(nbSpheres):

    sphere1 = spheres[i]
    sphere1Face = geompy.SubShapeAll(sphere1,
geompy.ShapeType["FACE"])[0]

    for j in range(nbSpheres):

        if i != j:

            sphere2 = spheres[j]

            distance = geompy.MinDistance(sphere1, sphere2)

            if distance <= contactDistance:

                [x, y, z] = geompy.ClosestPoints(sphere1,
sphere2)[1][0:3]

                contactPoint = geompy.MakeVertex(x, y, z)
                normalVector = geompy.GetNormal(sphere1Face,
contactPoint)

                tangentePlane =
geompy.MakePlane(contactPoint, normalVector, infiniteNumber)
                extrudedPlane =
geompy.MakePrismVecH(tangentePlane, normalVector, infiniteNumber)
                extrudedPlane =
geompy.MakeTranslationVectorDistance(extrudedPlane, normalVector, -
cap)

                sphere1 = geompy.MakeCut(sphere1,
extrudedPlane)

            newSpheres.append(sphere1)

compound_SS = geompy.MakeCompound(newSpheres)
#-----#
#           Generazione del cap sfera/parete
#-----#
## vertici in Zmin per il box
p1 = geompy.MakeVertex(Xmin, Ymin, Zmin)
p2 = geompy.MakeVertex(Xmax, Ymin, Zmin)
p3 = geompy.MakeVertex(Xmax, Ymax, Zmin)
p4 = geompy.MakeVertex(Xmin, Ymax, Zmin)
## vertici in Zmax per il box
p5 = geompy.MakeVertex(Xmin, Ymin, Zmax)
p6 = geompy.MakeVertex(Xmax, Ymin, Zmax)
p7 = geompy.MakeVertex(Xmax, Ymax, Zmax)
p8 = geompy.MakeVertex(Xmin, Ymax, Zmax)

```

---

---

```

## Wires face down
line1d = geompy.MakeEdge(p1, p2)
line2d = geompy.MakeEdge(p1, p5)
line3d = geompy.MakeEdge(p2, p6)
line4d = geompy.MakeEdge(p5, p6)
## Wires face up
line1u = geompy.MakeEdge(p3, p4)
line2u = geompy.MakeEdge(p8, p7)
line3u = geompy.MakeEdge(p4, p8)
line4u = geompy.MakeEdge(p3, p7)
## Wires face back
line1b = geompy.MakeEdge(p8, p4)
line2b = geompy.MakeEdge(p1, p5)
line3b = geompy.MakeEdge(p5, p8)
line4b = geompy.MakeEdge(p1, p4)
## Wires face front
line1f = geompy.MakeEdge(p3, p2)
line2f = geompy.MakeEdge(p7, p6)
line3f = geompy.MakeEdge(p2, p6)
line4f = geompy.MakeEdge(p7, p3)
## Pareti del letto
down = geompy.MakeFaceWires([line1d, line2d, line3d, line4d], 1)
up = geompy.MakeFaceWires([line1u, line2u, line3u, line4u], 2)
back = geompy.MakeFaceWires([line1b, line2b, line3b, line4b], 3)
front = geompy.MakeFaceWires([line1f, line2f, line3f, line4f], 4)
## Vettori normali alle pareti del letto
normalVector_down = geompy.GetNormal(down, p5)
normalVector_up = geompy.GetNormal(up, p7)
normalVector_back = geompy.GetNormal(back, p8)
normalVector_front = geompy.GetNormal(front, p6)
## Estrusione delle facce lungo il vettore "normalVector"
extruded_down = geompy.MakePrismVecH(down, normalVector_down, cap)
extruded_up = geompy.MakePrismVecH(up, normalVector_up, cap)
extruded_back = geompy.MakePrismVecH(back, normalVector_back, cap)
extruded_front = geompy.MakePrismVecH(front, normalVector_front, -
cap)
front_back = geompy.MakeFuse(extruded_front, extruded_back)
up_down = geompy.MakeFuse(extruded_up, extruded_down)
Extruded_Walls = geompy.MakeFuse(front_back, up_down)
# Cut-operation sfere/parete
Cut_WallSpheres = geompy.MakeCut(compound_SS, Extruded_Walls)
geompy.addToStudy(Cut_WallSpheres, "Cut Spheres")
#-----#
#           Generazione dei bridges
#-----#
# → Bridges sfera/sfera
nbSpheres = len(spheres)
bridges = []
contactPoints = []
for i in range(nbSpheres):

    sphere1 = spheres[i]
    sphere1Face = geompy.SubShapeAll(sphere1,
geompy.ShapeType["FACE"])[0]

    for j in range(i, nbSpheres):

```

---

---

```

        if i != j:
            sphere2 = spheres[j]

            distance = geompy.MinDistance(sphere1, sphere2)

            if distance <= contactDistance:

                [x, y, z] = geompy.ClosestPoints(sphere1,
sphere2) [1] [0:3]

                contactPoint = geompy.MakeVertex(x, y, z)
                contactPoints.append(contactPoint)
                normalVector = geompy.GetNormal(sphere1Face,
contactPoint)

                bridge =
geompy.MakeCylinder(contactPoint, normalVector, rb, 2*cap)
                bridge1 =
geompy.MakeTranslationVectorDistance(bridge, normalVector, -cap)
                bridges.append(bridge1)

bridges_SS = geompy.MakeCompound(bridges)
cp1 = len(bridges)
cp_ss = float(cp1)
# → Bridges sfera/parete
nbSpheres2 = len(spheres)
nFaces = len(Faces)
NewSpheres2 = []
bridges2 = []
contactPoints2 = []

for i in range(nbSpheres2):
    sphere = spheres[i]
    sphereFace = geompy.SubShapeAll(sphere,
geompy.ShapeType["FACE"]) [0]

    for j in range(nFaces):

        FaceBox = Faces[j]

        distance = geompy.MinDistance(sphere, FaceBox)

        if distance <= contactDistance:

            [x, y, z] = geompy.ClosestPoints(sphere,
FaceBox) [1] [0:3]

            contactPoint2 = geompy.MakeVertex(x, y, z)
            contactPoints2.append(contactPoint2)
            normalVector = geompy.GetNormal(sphereFace,
contactPoint2)

            bridge =
geompy.MakeCylinder(contactPoint2, normalVector, rb, cap)

```

---

```

                                bridge2 =
geompy.MakeTranslationVectorDistance(bridge, normalVector, -cap)
                                bridges2.append(bridge2)

NewSpheres2.append(sphere)

bridges_WS = geompy.MakeCompound(bridges2)
cp2 = len(bridges2)
cp_ws = float(cp2)
Bridges = geompy.MakeFuse(bridges_SS, bridges_WS)
BridgesSfere = geompy.MakeCompound([Bridges, compound],[])
geompy.addToStudy(BridgesSfere, "BridgesSfere")
#-----#
# Valutazione delle proprieta' dei letti impaccati #
#           RIFERIMENTO E BRIDGE METHOD           #
#-----#
# --> Letto impaccato senza modifiche
volume_box = geompy.BasicProperties(box)
Proprieta_sfereNB = geompy.BasicProperties(compound)
void_fractionNB = (volume_box[2] - Proprieta_sfereNB[2])/
volume_box[2]
av_1t = 6*(1-void_fractionNB)/d_p
av_1n = Proprieta_sfereNB[1]/volume_box[2]
# --> Letto impaccato e modificato con bridge method
volume_box = geompy.BasicProperties(box)
Proprieta_sfereBB = geompy.BasicProperties(BridgesSfere)
area_capSS = 2*math.pi*cap*radius*2
area_bridgeSS = 2*math.pi*rb*2*cap
area_CAP = 2*(math.pi*rb**2)
area_capWS = area_capSS/2
area_bridgeWS = area_bridgeSS/2
S_sfera = 4*math.pi*radius**2
Area_catalitica_bridge = N_sfere * S_sfera - cp_ss*(area_capSS-
area_bridgeSS) - cp_ws*(area_capWS-area_bridgeWS)
Area_catalitica_cap = N_sfere * S_sfera - cp_ss*(area_capSS-
area_CAP) - cp_ws*(area_capWS-area_CAP/2)
V_sfereCAP = Proprieta_sfereNB[2] - (cp_ws*(3.14159*cap**2*(d_p/2-
cap/3))+cp_ss*(2*3.14159*cap**2*(d_p/2-cap/3)))
V_bridges = cp_ws*(3.14159*rb**2*cap)+cp_ss*(3.14159*rb**2*2*cap)
void_fractionBB = (volume_box[2] - V_sfereCAP - V_bridges)/
volume_box[2]
av_2n_b = Area_catalitica_bridge/volume_box[2]
#-----#
# Valutazione delle proprieta' dei letti impaccati #
#           RIFERIMENTO E CAP METHOD           #
#-----#
# → Letto impaccato senza modifiche
volume_box = geompy.BasicProperties(box)
Proprieta_sfereNB = geompy.BasicProperties(compound)
void_fractionNB = (volume_box[2] - Proprieta_sfereNB[2])/
volume_box[2]
av_1t = 6*(1-void_fractionNB)/d_p
av_1n = Proprieta_sfereNB[1]/volume_box[2]
# → Letto impaccato e modificato con cap method
volume_box = geompy.BasicProperties(box)
Proprieta_sfereCAP = geompy.BasicProperties(Cut_WallSpheres)

```

```

Area_catalitica = Proprieta_sfereCAP[1]
void_fraction = (volume_box[2] - Proprieta_sfereCAP[2])/
volume_box[2]
av_2n = Proprieta_sfereCAP[1]/volume_box[2]
#-----#
# Metti a Display le proprieta' dei letti impaccati #
#-----#
print "
=====
print "          Proprieta' del letto impaccato          "
print "          - metodo locale BRIDGE -          "
print "
=====

print "          "
print "    Numero di sfere =", N_sfere
print "    diametro delle sfere [m] =", d_p
print "          "
print "    numero di bridges sfera/sfera =", cp_ss
print "    numero di bridges sfera/parete =", cp_ws
print "    rb/ds [-] =", rb/d_p
print "    rb [m] = ", rb
print "    lunghezza bridge sfera/sfera [m] =", 2*cap
print "    lunghezza bridge sfera/parete [m] =", cap
print "          "
print "    superficie catalitica [m2] =", Area_catalitica_bridge
print "    grado di vuoto [-] =", void_fractionBB
print "    av [m2/m3] =", av_2n_b
print "          "
print " -----"
print "          >>> Err % superficie catalitica =",
(Proprieta_sfereNB[1] - Area_catalitica_bridge)/Proprieta_sfereNB[1]
*100
print "          >>> Err % av =", (av_1n - av_2n_b)/av_1n *100
print "          >>> Err % grado di vuoto =", -(void_fractionNB -
void_fractionBB)/void_fractionNB *100
print " -----"
print "          ===== - metodo locale CAP - ====="
print "          "
print "    Numero di sfere =", N_sfere
print "    diametro delle sfere [m] =", d_p
print "          "
print "    2h/ds [-] =", 2*cap/d_p
print "    lunghezza cap sfera/sfera [m] =", 2*cap
print "    lunghezza cap sfera/parete [m] =", cap
print "          "
print "    grado di vuoto [-] =", void_fraction
print "    av [m2/m3] =", av_2n
print "          "
print " -----"
print "          >>> Err % superficie catalitica =",
(Proprieta_sfereNB[1] - Area_catalitica)/Proprieta_sfereNB[1] *100
print "          >>> Err % av =", (av_1n - av_2n)/av_1n *100
print "          >>> Err % grado di vuoto =", -(void_fractionNB -
void_fraction)/void_fractionNB *100
print " -----"
print "          "

```



```
#####
# ----- #
#   Determinazione delle proprieta' geometriche di letti impaccati#
#   Modellazione dei punti di contatto i metodi di modifica globale #
# ----- #
#   properties_packedBed-globalMethods.py #
#   #
#####

import csv
import salome
import math
salome.salome_init()
theStudy = salome.myStudy
import GEOM
from salome.geom import geomBuilder
geompy = geomBuilder.New(theStudy)
#-----#
#   Importa le sfere e scegli la modifica %
#-----#
inputFileName = "100" # nome file input
Var = 0.05           # mettere il valore %
contactDistance = 6.4e-5
vertexes = []
spheres = []
spheres1 = []
spheres2 = []
centres = []
infiniteNumber = 9999

with open(inputFileName, 'rb') as csvFile:

    csvReader = csv.reader(csvFile, delimiter=' ')

    thisIsHeader = True
    for row in csvReader:

        if thisIsHeader == True:

            if "'radius'" in str(row) and "'x', 'y', 'z'" in
str(row):

                radiusColumnIndex = row.index('radius') - 2

                xColumnIndex = row.index('x') - 2
                yColumnIndex = row.index('y') - 2
                zColumnIndex = row.index('z') - 2

                thisIsHeader = False

            else:

                [x, y, z] = [
                    float(row[xColumnIndex]),
                    float(row[yColumnIndex]),
```

---

```

        float(row[zColumnIndex]),
        ]

        radius = float(row[radiusColumnIndex])
        d_nom = 2 * radius
        vertex = geompy.MakeVertex(x, y, z)
        vertexes.append(vertex)

        sphere = geompy.MakeSpherePntR(vertex, radius)
        spheres.append(sphere)
        centres.append(vertex)

        d_overlap = d_nom / (1 - Var)
radius_overlap = d_overlap/2
        sphere3 = geompy.MakeSpherePntR(vertex,
radius_overlap)
        spheres2.append(sphere3)

        d_gap = d_nom * (1 - Var)
        radius_gap = d_gap/2
        sphere2 = geompy.MakeSpherePntR(vertex,
radius_gap)
        spheres1.append(sphere2)
        centres.append(vertex)

compound1 = geompy.MakeCompound(spheres)
geompy.addToStudy(compound1, "nominal spheres")
compound2 = geompy.MakeCompound(spheres1)
geompy.addToStudy(compound2, "gap spheres")
compound3 = geompy.MakeCompound(spheres2)
geompy.addToStudy(compound3, "overlap spheres")
N_sfere = len(spheres)
#-----#
#           Creazione del box contenente le sfere
#-----#
searchBox = geompy.BoundingBox(compound1)
aBB = geompy.MakeBoundingBox(compound1)           # coordinate
Xmin = searchBox[0]
Xmax = searchBox[1]
Ymin= searchBox[2]
Ymax= searchBox[3]
Zmin =searchBox[4]
Zmax = searchBox[5]
p1_box = geompy.MakeVertex( searchBox[0], searchBox[2],
searchBox[4])
p2_box = geompy.MakeVertex( searchBox[1], searchBox[3],
searchBox[5])
box = geompy.MakeBoxTwoPnt(p1_box, p2_box)
geompy.addToStudy(box, "Box")
boxFaces1    = geompy.SubShapeAll(box, geompy.ShapeType["FACE"])[0]
boxFaces2    = geompy.SubShapeAll(box, geompy.ShapeType["FACE"])[1]
boxFaces3    = geompy.SubShapeAll(box, geompy.ShapeType["FACE"])[2]
boxFaces4    = geompy.SubShapeAll(box, geompy.ShapeType["FACE"])[3]
boxFaces5    = geompy.SubShapeAll(box, geompy.ShapeType["FACE"])[4]
boxFaces6    = geompy.SubShapeAll(box, geompy.ShapeType["FACE"])[5]

```

---

---

```

Faces = [boxFaces1, boxFaces2, boxFaces3, boxFaces4, boxFaces5,
boxFaces6]
#-----#
#     Calcola la minima distanza sfera/sfera
#     sul letto impaccato non modificato
#-----#
nbSpheres = len(spheres)
nFaces = len(Faces)
distances1s = []
distances1w = []

for i in range(nbSpheres):

    sphere1 = spheres[i]
    sphere1Face = geompy.SubShapeAll(sphere1,
geompy.ShapeType["FACE"])[0]

    for j in range(i, nbSpheres):

        if i != j:

            sphere2 = spheres[j]
            distance = geompy.MinDistance(sphere1, sphere2)
            distances1s.append(distance)

min_distance_s1 = min(distances1s)
#-----#
#     Calcola la minima distanza sfera/parete
#     sul letto impaccato non modificato
#-----#
for i in range(nbSpheres):

    sphere = spheres[i]
    sphereFace = geompy.SubShapeAll(sphere,
geompy.ShapeType["FACE"])[0]

    for j in range(nFaces):

        FaceBox = Faces[j]
        distance = geompy.MinDistance(sphere, FaceBox)
        distances1w.append(distance)

min_distance_w1 = min(distances1w)
#-----#
#     Determinazione del numero dei punti di contatto #
#-----#
# → Punti di contatto sfera/sfera
nbSpheres = len(spheres)
contactPoints = []

for i in range(nbSpheres):

    sphere1 = spheres[i]
    sphere1Face = geompy.SubShapeAll(sphere1,
geompy.ShapeType["FACE"])[0]

```

---

```

for j in range(i, nbSpheres):
    if i != j:
        sphere2 = spheres[j]
        distance = geompy.MinDistance(sphere1, sphere2)
        if distance <= contactDistance:
            [x, y, z] = geompy.ClosestPoints(sphere1,
sphere2) [1] [0:3]
            contactPoint = geompy.MakeVertex(x, y, z)
            contactPoints.append(contactPoint)

cp_SS = geompy.MakeCompound(contactPoints)
cp1 = len(contactPoints)
cp_ss = float(cp1)
# → Punti di contatto sfera/parete
nbSpheres2 = len(spheres)
nFaces = len(Faces)
contactPoints2 = []

for i in range(nbSpheres2):
    sphere = spheres[i]
    sphereFace = geompy.SubShapeAll(sphere,
geompy.ShapeType["FACE"]) [0]
    for j in range(nFaces):
        FaceBox = Faces[j]
        distance = geompy.MinDistance(sphere, FaceBox)
        if distance <= contactDistance:
            [x, y, z] = geompy.ClosestPoints(sphere,
FaceBox) [1] [0:3]
            contactPoint2 = geompy.MakeVertex(x, y, z)
            contactPoints2.append(contactPoint2)

cp_WS = geompy.MakeCompound(contactPoints2)
cp2 = len(contactPoints2)
cp_ws = float(cp2)
#-----#
#     Calcola la minima distanza sfera/parete
#     sul letto impaccato modificato
#-----#
# → Punti di contatto sfera/sfera
nbSpheres = len(spheres1)
nFaces = len(Faces)
distances1ss = []
distances1ws = []
contactPoints1ss = []
contactPoints1ws = []

```

```

for i in range(nbSpheres):
    sphere1 = spheres1[i]
    sphere1Face = geompy.SubShapeAll(sphere1,
geompy.ShapeType["FACE"])[0]

    for j in range(i, nbSpheres):

        if i != j:

            sphere2 = spheres1[j]
            distance = geompy.MinDistance(sphere1, sphere2)
            distances1ss.append(distance)

            if distance <= contactDistance:

                [x, y, z] = geompy.ClosestPoints(sphere1,
sphere2)[1][0:3]

                contactPointss = geompy.MakeVertex(x, y, z)
                contactPoints1ss.append(contactPointss)

cplss = len(contactPoints1ss)
cp_ss1 = float(cplss)
min_distance_ss1 = min(distances1ss)
# → Punti di contatto sfera/parete
for i in range(nbSpheres):

    sphere = spheres1[i]
    sphereFace = geompy.SubShapeAll(sphere,
geompy.ShapeType["FACE"])[0]

    for j in range(nFaces):

        FaceBox = Faces[j]
        distance = geompy.MinDistance(sphere, FaceBox)
        distances1ws.append(distance)

        if distance <= contactDistance:

            [x, y, z] = geompy.ClosestPoints(sphere,
FaceBox)[1][0:3]

            contactPointws = geompy.MakeVertex(x, y, z)
            contactPoints1ws.append(contactPointws)

cplws = len(contactPoints1ws)
cp_ws1 = float(cplws)
min_distance_ws1 = min(distances1ws)
#-----#
# Valutazione delle proprieta' dei letti impaccati #
#-----#
# → Letto impaccato senza modifiche
volume_box = geompy.BasicProperties(box)
Proprieta_sfere = geompy.BasicProperties(compound1)

```

```

void_fractionNB = (volume_box[2] - Proprieta_sfere[2])/
volume_box[2]
volume_Box = geompy.BasicProperties(box)
av_1t = 6*(1-void_fractionNB)/d_nom
av_1n = Proprieta_sfere[1]/volume_box[2]
# → Letto impaccato con gap method
Proprieta_sfereCAP = geompy.BasicProperties(compound2)
void_fraction = (volume_Box[2] - Proprieta_sfereCAP[2])/
volume_Box[2]
av_2t = 6*(1-void_fraction)/d_gap
av_2n = Proprieta_sfereCAP[1]/volume_Box[2]
# → Letto impaccato con overlap method
volume_box = geompy.BasicProperties(box)
Proprieta_sfereOVERLAP = geompy.BasicProperties(compound3)
r_over = d_overlap/2
h = (d_overlap-d_nom)
V_sferaover = 1.33333*3.14159*r_over**3
area_cata = N_sfere*(4 * 3.14159*r_over**2) -
cp_ss*(2*2*3.14159*r_over*h) - cp_ws*(2*3.14159*r_over*h)
V_over = N_sfere*V_sferaover - cp_ss*(2*(3.14159*(h**2)*(r_over-
h/3))) - cp_ws*(3.14159*(h**2)*(r_over-h/3))
void_fraction = (volume_box[2] - V_over)/ volume_box[2]
av_2n = area_cata/volume_box[2]
#-----#
# Metti a Display le proprieta' dei letti impaccati #
#-----#
print "          == - metodo blobale GAP -          ==          "
print "      Numero di sfere =", N_sfere
print "      Var % diametro =", 100*(d_nom - d_gap)/d_nom
print "      diametro delle sfere [m] =", d_gap
print "      min distanza sfera/sfera [m] =", min_distance_ssl
print "      min distanza sfera/parete [m] =", min_distance_wsl
print "      superficie catalitica [m2] =", Proprieta_sfereCAP[1]
print "      grado di vuoto [-] =", void_fraction
print "      av_teo [m2/m3] =", av_2t
print "      av_nom [m2/m3] =", av_2n
print "      Err % av =", 100*(av_2t-av_2n)/av_2t
print "      >>> Err % superficie catalitica =",
(Proprieta_sfere[1]-Proprieta_sfereCAP[1])/Proprieta_sfere[1] *100
print "      >>> Err % av =", (av_1n - av_2n)/av_1n *100
print "      >>> Err % grado di vuoto =", (void_fractionNB-
void_fraction)/void_fractionNB *100
print "          "
print "          == - metodo globale OVERLAP -==          "
print "      Numero di sfere =", N_sfere
print "      Var % diametro = +", -100*(d_nom - d_overlap)/d_nom
print "      diametro delle sfere [m] =", d_overlap
print "      superficie catalitica [m2] =", area_cata
print "      grado di vuoto [-] =", void_fraction
print "      av_nom [m2/m3] =", av_2n
print "      >>> Err % superficie catalitica =",
(Proprieta_sfere[1]-area_cata)/Proprieta_sfere[1] *100
print "      >>> Err % av =", (av_1n - av_2n)/av_1n *100
print "      >>> Err % grado di vuoto =", (void_fractionNB-
void_fraction)/void_fractionNB *100
print "          "

```

```
#####
# ----- Script per la generazione della mesh ibrida ----- #
#           di letti impaccati di sfere Multiregion           #
#           Modellazione dei punti di contatto con il BRIDGE METHOD #
#                                                                 #
# -----packedbed_bridge-methodMultiRegion.py----- #
#####
import csv
import salome
salome.salome_init()
theStudy = salome.myStudy
import GEOM
from salome.geom import geomBuilder
geompy = geomBuilder.New(theStudy)
#####
###           GENERAZIONE DELLA GEOMETRIA           ###
#####
# Origine assi cartesiani
O = geompy.MakeVertex(0, 0, 0)
OX = geompy.MakeVectorDXDYDZ(1, 0, 0)
OY = geompy.MakeVectorDXDYDZ(0, 1, 0)
OZ = geompy.MakeVectorDXDYDZ(0, 0, 1)
#-----#
#           Importa le sfere           #
#-----#
inputFileName = "20sfere_ok" # nome file input

vertexes = []
spheres = []
centres = []

with open(inputFileName, 'rb') as csvFile:

    csvReader = csv.reader(csvFile, delimiter=' ')

    thisIsHeader = True
    for row in csvReader:

        if thisIsHeader == True:

            if "'radius'" in str(row) and "'x', 'y', 'z'" in
str(row):

                radiusColumnIndex = row.index('radius') - 2

                xColumnIndex = row.index('x') - 2
                yColumnIndex = row.index('y') - 2
                zColumnIndex = row.index('z') - 2

                thisIsHeader = False

            else:

                [x, y, z] = [
                    float(row[xColumnIndex]),
```

---

```

        float(row[yColumnIndex]),
        float(row[zColumnIndex]),
    ]

    radius = float(row[radiusColumnIndex])

    vertex = geompy.MakeVertex(x, y, z)
    vertexes.append(vertex)

    sphere = geompy.MakeSpherePntR(vertex, radius)
    spheres.append(sphere)
    centres.append(vertex)

compounds = geompy.MakeCompound(spheres)
N_sfere = len(spheres)
#-----#
#       Scegli il rapporto rb/dp desiderato
#-----#
import math
d_p = 2 * radius # m
rb_dp = 0.05
rb = rb_dp * d_p
arcoseno = math.asin(rb/radius)
coseno = math.cos(arcoseno)
cap = radius * (1 - coseno)
contactDistance = 6.4e-05
#-----#
#       Creazione del box contenente le sfere
#-----#
searchBox = geompy.BoundingBox(compounds)
aBB = geompy.MakeBoundingBox(compounds) # coordinate
Xmin = searchBox[0]
Xmax = searchBox[1]
Ymin = searchBox[2]
Ymax = searchBox[3]
Zmin = searchBox[4]
Zmax = searchBox[5]
z_min = Zmin - radius/2.3
z_max = Zmax + radius/2.3
p1_box = geompy.MakeVertex( Xmin, Ymin, z_min)
p2_box = geompy.MakeVertex( Xmax, Ymax, z_max)
box = geompy.MakeBoxTwoPnt(p1_box, p2_box)
boxFaces1 = geompy.SubShapeAll(box, geompy.ShapeType["FACE"][0]
boxFaces2 = geompy.SubShapeAll(box, geompy.ShapeType["FACE"][1]
boxFaces3 = geompy.SubShapeAll(box, geompy.ShapeType["FACE"][2]
boxFaces4 = geompy.SubShapeAll(box, geompy.ShapeType["FACE"][3]
boxFaces5 = geompy.SubShapeAll(box, geompy.ShapeType["FACE"][4]
boxFaces6 = geompy.SubShapeAll(box, geompy.ShapeType["FACE"][5]
Faces = [boxFaces1, boxFaces2, boxFaces3, boxFaces4, boxFaces5,
boxFaces6]
#-----#
#       Generazione dei bridges
#-----#
# --> Bridges sfera/sfera
nbSpheres = len(spheres)
bridges = []

```

---



---

```

contactPoints = []

for i in range(nbSpheres):
    sphere1 = spheres[i]
    sphere1Face = geompy.SubShapeAll(sphere1,
geompy.ShapeType["FACE"])[0]

    for j in range(i, nbSpheres):
        if i != j:
            sphere2 = spheres[j]

            distance = geompy.MinDistance(sphere1, sphere2)

            if distance <= contactDistance:
                [x, y, z] = geompy.ClosestPoints(sphere1,
sphere2)[1][0:3]

                contactPoint = geompy.MakeVertex(x, y, z)
                contactPoints.append(contactPoint)
                normalVector = geompy.GetNormal(sphere1Face,
contactPoint)

                h = 2 * cap
                bridge =
geompy.MakeCylinder(contactPoint, normalVector, rb, h)
                bridge1 =
geompy.MakeTranslationVectorDistance(bridge, normalVector, -h/2)
                bridges.append(bridge1)

bridges_SS = geompy.MakeCompound(bridges)
cut_SS = geompy.MakeCut(compounds, bridges_SS)
# --> Bridges sfera/parete
nbSpheres2 = len(spheres)
nFaces = len(Faces)
NewSpheres2 = []
bridges2 = []
contactPoints2 = []

for i in range(nbSpheres2):
    sphere = spheres[i]
    sphereFace = geompy.SubShapeAll(sphere,
geompy.ShapeType["FACE"])[0]

    for j in range(nFaces):
        FaceBox = Faces[j]

        distance = geompy.MinDistance(sphere, FaceBox)

        if distance <= contactDistance:

```

---

---

```

[x, y, z] = geompy.ClosestPoints(sphere,
FaceBox) [1] [0:3]

contactPoint2 = geompy.MakeVertex(x, y, z)
contactPoints2.append(contactPoint2)
normalVector = geompy.GetNormal(sphereFace,
contactPoint2)

        bridge =
geompy.MakeCylinder(contactPoint2, normalVector, rb, cap)
        bridge2 =
geompy.MakeTranslationVectorDistance(bridge, normalVector, -cap)
        bridges2.append(bridge2)

NewSpheres2.append(sphere)

bridges_WS = geompy.MakeCompound(bridges2)
Bridges = geompy.MakeCompound([bridges_SS, bridges_WS],[])
BridgesSfere = geompy.MakeCompound([bridges_SS, bridges_WS,
compounds],[])
cut_WS_spheres = geompy.MakeCut(compounds, Bridges)
# -----#
fluidRegion = geompy.MakeCut(box, BridgesSfere)
# fluidRegion = geompy.MakeCut(box, compounds)
# fluidRegion = geompy.MakeCut(fluidRegion, Bridges)
# -----#
#fluidRegion = geompy.MakeCut(box, compounds)
#fluidRegion = geompy.MakeCut(fluidRegion, Bridges)
#-----#
# Generazione di altri box per allungare il letto impaccato #
#-----#
# ---> box 1
z_inlet = z_min - d_p
p1_box1 = geompy.MakeVertex( Xmin, Ymin, z_min)
p2_box1 = geompy.MakeVertex( Xmax, Ymax, z_inlet)
box1 = geompy.MakeBoxTwoPnt(p1_box1, p2_box1)
# ---> box 3
z_box3 = z_max + d_p
p1_box3 = geompy.MakeVertex( Xmax, Ymax, z_max)
p2_box3 = geompy.MakeVertex( Xmin, Ymin, z_box3)
box3 = geompy.MakeBoxTwoPnt(p1_box3, p2_box3)
# ---> box 4
k = z_box3 - z_max
box4 = geompy.MakeTranslation(box3, 0, 0, k)
# ---> box 5
box5 = geompy.MakeTranslation(box4, 0, 0, k)
# ---> box 6
box6 = geompy.MakeTranslation(box5, 0, 0, k)
#-----#
#   Partizione delle diverse regioni: solidi e fluido   #
#-----#
partition = geompy.MakePartition( [box1, fluidRegion, bridges_SS,
bridges_WS, cut_WS_spheres, box3, box4, box5, box6],[] )
# fluido 1

```

---

---

```

pc_box1 = geompy.MakeVertex( (Xmin + Xmax)/2, (Ymin + Ymax)/2,
(z_min + z_inlet)/2)
fluid_1 = geompy.GetShapesNearPoint(partition, pc_box1,
geompy.ShapeType["SOLID"], 1e-6)
geompy.addToStudyInFather(partition, fluid_1, 'fluid_1')
# fluido 2
pc_box2 = geompy.MakeVertex( (Xmin + Xmax)/2, (Ymin + Ymax)/2,
(z_max + Zmax)/2) # non centrato perche' ci sono le sfere
fluid_2 = geompy.GetShapesNearPoint(partition, pc_box2,
geompy.ShapeType["SOLID"], 1e-6)
geompy.addToStudyInFather(partition, fluid_2, 'fluid_2')
# sfere
spheres = geompy.GetInPlace( partition, cut_WS_spheres )
geompy.addToStudyInFather(partition, spheres, 'spheres')
# Bridges ws e Bridges ss
bridges_ws = geompy.GetInPlace( partition, bridges_WS )
geompy.addToStudyInFather(partition, bridges_ws, 'bridges_WS')
bridges_ss = geompy.GetInPlace( partition, bridges_SS )
geompy.addToStudyInFather(partition, bridges_ss, 'bridges_SS')
# fluido 3
pc_box3 = geompy.MakeVertex( (Xmin + Xmax)/2, (Ymin + Ymax)/2,
(z_max + z_box3)/2)
fluid_3 = geompy.GetShapesNearPoint(partition, pc_box3 ,
geompy.ShapeType["SOLID"], 1e-6)
geompy.addToStudyInFather(partition, fluid_3, 'fluid_3')
# fluido 4
pc_box4 = geompy.MakeTranslation(pc_box3, 0, 0, k)
fluid_4 = geompy.GetShapesNearPoint(partition, pc_box4 ,
geompy.ShapeType["SOLID"], 1e-6)
geompy.addToStudyInFather(partition, fluid_4, 'fluid_4')
# fluido 5
pc_box5 = geompy.MakeTranslation(pc_box4, 0, 0, k)
fluid_5 = geompy.GetShapesNearPoint(partition, pc_box5 ,
geompy.ShapeType["SOLID"], 1e-6)
geompy.addToStudyInFather(partition, fluid_5, 'fluid_5')
# fluido 6
pc_box6 = geompy.MakeTranslation(pc_box5, 0, 0, k)
fluid_6 = geompy.GetShapesNearPoint(partition, pc_box6 ,
geompy.ShapeType["SOLID"], 1e-6)
geompy.addToStudyInFather(partition, fluid_6, 'fluid_6')
#-----#
#                               Assegnazione delle patches                               #
#-----#
# Faces
z1 = ( z_inlet + z_min) / 2
## ---> Fluido 1
left = geompy.GetFaceNearPoint( fluid_1, geompy.MakeVertex( 0, 0, -
100 ))
inlet = geompy.CreateGroup( fluid_1, geompy.ShapeType["FACE"],
"inlet" )
geompy.UnionList( inlet, [ left ])
front_box1 = geompy.GetFaceNearPoint( fluid_1, geompy.MakeVertex(
100, 0, z1 ))
frontWall_box1 = geompy.CreateGroup( fluid_1,
geompy.ShapeType["FACE"])
geompy.UnionList( frontWall_box1, [ front_box1 ])

```

---

---

```

back_box1 = geompy.GetFaceNearPoint( fluid_1, geompy.MakeVertex( -
100, 0, z1))
backWall_box1 = geompy.CreateGroup( fluid_1,
geompy.ShapeType["FACE"])
geompy.UnionList( backWall_box1, [ back_box1 ])
up_box1 = geompy.GetFaceNearPoint( fluid_1, geompy.MakeVertex( 0,
100, z1 ))
upWall_box1 = geompy.CreateGroup( fluid_1, geompy.ShapeType["FACE"])
geompy.UnionList( upWall_box1, [ up_box1 ])
down_box1 = geompy.GetFaceNearPoint( fluid_1, geompy.MakeVertex( 0,
-100, z1 ))
downWall_box1 = geompy.CreateGroup( fluid_1,
geompy.ShapeType["FACE"])
geompy.UnionList( downWall_box1, [ down_box1 ])
fixedWall_fluid1 = geompy.UnionListOfGroups( [frontWall_box1,
backWall_box1, upWall_box1, downWall_box1] )
geompy.addToStudyInFather(fluid_1,
fixedWall_fluid1, 'fixedWall_fluid1')
fluidRegion_1 = geompy.CreateGroup( partition,
geompy.ShapeType["SOLID"],"fluidRegion_1")
geompy.UnionList( fluidRegion_1, [ fluid_1 ])
## ---> Fluido 2
all = geompy.SubShapeAll( fluid_2, geompy.ShapeType["FACE"] )
z2 = ( z_min + z_max ) / 3
left_fluid2 = geompy.GetFaceNearPoint( fluid_2, geompy.MakeVertex(
0, 0, -100 ))
inlet_fluid2 = geompy.CreateGroup( fluid_2,
geompy.ShapeType["FACE"])
geompy.UnionList( inlet_fluid2, [ left_fluid2 ])
right_fluid2 = geompy.GetFaceNearPoint( fluid_2, geompy.MakeVertex(
0, 0, 100 ))
outlet_fluid2 = geompy.CreateGroup( fluid_2,
geompy.ShapeType["FACE"])
geompy.UnionList( outlet_fluid2, [ right_fluid2 ])
front_fluid2 = geompy.GetFaceNearPoint( fluid_2, geompy.MakeVertex(
200, 0, z2 ))
frontWall_box2 = geompy.CreateGroup( fluid_2,
geompy.ShapeType["FACE"])
geompy.UnionList( frontWall_box2, [ front_fluid2 ])
back_fluid2 = geompy.GetFaceNearPoint( fluid_2, geompy.MakeVertex( -
200, 0, z2 ))
backWall_box2 = geompy.CreateGroup( fluid_2,
geompy.ShapeType["FACE"])
geompy.UnionList( backWall_box2, [ back_fluid2 ])
up_fluid2 = geompy.GetFaceNearPoint( fluid_2, geompy.MakeVertex( 0,
200, z2 ))
upWall_box2 = geompy.CreateGroup( fluid_2, geompy.ShapeType["FACE"])
geompy.UnionList( upWall_box2, [ up_fluid2 ])
down_fluid2 = geompy.GetFaceNearPoint( fluid_2, geompy.MakeVertex(
0, -200, z2 ))
downWall_box2 = geompy.CreateGroup( fluid_2,
geompy.ShapeType["FACE"])
geompy.UnionList( downWall_box2, [ down_fluid2 ])
reactingWall = geompy.CreateGroup( fluid_2,
geompy.ShapeType["FACE"], "catalyticWall")
geompy.UnionList( reactingWall, all)

```

---

---

```

geompy.DifferenceList( reactingWall, [ inlet_fluid2, outlet_fluid2,
frontWall_box2, backWall_box2, upWall_box2, downWall_box2] )
## ---> Calcolo dell'area catalitica
catalytic_Area = geompy.BasicProperties(reactingWall)
print " catalytic Area = ", catalytic_Area[1]
fixedWall_fluid2 = geompy.UnionListOfGroups( [frontWall_box2,
backWall_box2, upWall_box2, downWall_box2] )
geompy.addToStudyInFather(fluid_2,
fixedWall_fluid2,'fixedWall_fluid2')
fluidRegion_2 = geompy.CreateGroup( partition,
geompy.ShapeType["SOLID"],"fluidRegion_2")
geompy.UnionList( fluidRegion_2, [ fluid_2 ])
# solids
Sfere = geompy.CreateGroup( partition,
geompy.ShapeType["SOLID"],"sfere_cut")
geompy.UnionList( Sfere, [ spheres])
Bridges_ss = geompy.CreateGroup( partition,
geompy.ShapeType["SOLID"],"bridge_ss")
geompy.UnionList( Bridges_ss, [ bridges_ss ])
Bridges_ws = geompy.CreateGroup( partition,
geompy.ShapeType["SOLID"],"bridge_ws")
geompy.UnionList( Bridges_ws, [ bridges_ws])
## ---> Fluido 3
z3 = z_max + k / 2
front_box3 = geompy.GetFaceNearPoint( fluid_3, geompy.MakeVertex(
100, 0, z3 ))
frontWall_box3 = geompy.CreateGroup( fluid_3,
geompy.ShapeType["FACE"])
geompy.UnionList( frontWall_box3, [ front_box3 ])
back_box3 = geompy.GetFaceNearPoint( fluid_3, geompy.MakeVertex( -
100, 0, z3 ))
backWall_box3 = geompy.CreateGroup( fluid_3,
geompy.ShapeType["FACE"])
geompy.UnionList( backWall_box3, [ back_box3 ])
up_box3 = geompy.GetFaceNearPoint( fluid_3, geompy.MakeVertex( 0,
100, z3 ))
upWall_box3 = geompy.CreateGroup( fluid_3, geompy.ShapeType["FACE"])
geompy.UnionList( upWall_box3, [ up_box3 ])
down_box3 = geompy.GetFaceNearPoint( fluid_3, geompy.MakeVertex( 0,
-100, z3 ))
downWall_box3 = geompy.CreateGroup( fluid_3,
geompy.ShapeType["FACE"])
geompy.UnionList( downWall_box3, [ down_box3 ])
fixedWall_fluid3 = geompy.UnionListOfGroups( [frontWall_box3,
backWall_box3, upWall_box3, downWall_box3] )
geompy.addToStudyInFather(fluid_3,
fixedWall_fluid3,'fixedWall_fluid3')
fluidRegion_3 = geompy.CreateGroup( partition,
geompy.ShapeType["SOLID"],"fluidRegion_3")
geompy.UnionList( fluidRegion_3, [ fluid_3 ])
## ---> Fluido 4
z4 = z3 + k
front_box4 = geompy.GetFaceNearPoint( fluid_4, geompy.MakeVertex(
100, 0, z4 ))
frontWall_box4 = geompy.CreateGroup( fluid_4,
geompy.ShapeType["FACE"])

```

---

---

```

geompy.UnionList( frontWall_box4, [ front_box4 ])
back_box4 = geompy.GetFaceNearPoint( fluid_4, geompy.MakeVertex( -
100, 0, z4 ))
backWall_box4 = geompy.CreateGroup( fluid_4,
geompy.ShapeType["FACE"])
geompy.UnionList( backWall_box4, [ back_box4 ])
up_box4 = geompy.GetFaceNearPoint( fluid_4, geompy.MakeVertex( 0,
100, z4 ))
upWall_box4 = geompy.CreateGroup( fluid_4, geompy.ShapeType["FACE"])
geompy.UnionList( upWall_box4, [ up_box4 ])
down_box4 = geompy.GetFaceNearPoint( fluid_4, geompy.MakeVertex( 0,
-100, z4 ))
downWall_box4 = geompy.CreateGroup( fluid_4,
geompy.ShapeType["FACE"])
geompy.UnionList( downWall_box4, [ down_box4 ])
fixedWall_fluid4 = geompy.UnionListOfGroups( [frontWall_box4,
backWall_box4, upWall_box4, downWall_box4] )
geompy.addToStudyInFather(fluid_4,
fixedWall_fluid4, 'fixedWall_fluid4')
fluidRegion_4 = geompy.CreateGroup( partition,
geompy.ShapeType["SOLID"], "fluidRegion_4")
geompy.UnionList( fluidRegion_4, [ fluid_4 ])
z5 = z4 + k
front_box5 = geompy.GetFaceNearPoint( fluid_5, geompy.MakeVertex(
100, 0, z5 ))
frontWall_box5 = geompy.CreateGroup( fluid_5,
geompy.ShapeType["FACE"])
geompy.UnionList( frontWall_box5, [ front_box5 ])
back_box5 = geompy.GetFaceNearPoint( fluid_5, geompy.MakeVertex( -
100, 0, z5 ))
backWall_box5 = geompy.CreateGroup( fluid_5,
geompy.ShapeType["FACE"])
geompy.UnionList( backWall_box5, [ back_box5 ])
up_box5 = geompy.GetFaceNearPoint( fluid_5, geompy.MakeVertex( 0,
100, z5))
upWall_box5 = geompy.CreateGroup( fluid_5, geompy.ShapeType["FACE"])
geompy.UnionList( upWall_box5, [ up_box5 ])
down_box5 = geompy.GetFaceNearPoint( fluid_5, geompy.MakeVertex( 0,
-100, z5 ))
downWall_box5 = geompy.CreateGroup( fluid_5,
geompy.ShapeType["FACE"])
geompy.UnionList( downWall_box5, [ down_box5 ])
fixedWall_fluid5 = geompy.UnionListOfGroups( [frontWall_box5,
backWall_box5, upWall_box5, downWall_box5] )
geompy.addToStudyInFather(fluid_5,
fixedWall_fluid5, 'fixedWall_fluid5')
fluidRegion_5 = geompy.CreateGroup( partition,
geompy.ShapeType["SOLID"], "fluidRegion_5")
geompy.UnionList( fluidRegion_5, [ fluid_5 ])
z6 = z5 + k
right = geompy.GetFaceNearPoint( fluid_6, geompy.MakeVertex( 0, 0,
100 ))
outlet = geompy.CreateGroup( fluid_6, geompy.ShapeType["FACE"],
"outlet" )
geompy.UnionList( outlet, [ right ])

```

---

---

```

front_box6 = geompy.GetFaceNearPoint( fluid_6, geompy.MakeVertex(
100, 0, z6 ))
frontWall_box6 = geompy.CreateGroup( fluid_6,
geompy.ShapeType["FACE"])
geompy.UnionList( frontWall_box6, [ front_box6 ])
back_box6 = geompy.GetFaceNearPoint( fluid_6, geompy.MakeVertex( -
100, 0, z6 ))
backWall_box6 = geompy.CreateGroup( fluid_6,
geompy.ShapeType["FACE"])
geompy.UnionList( backWall_box6, [ back_box6 ])
up_box6 = geompy.GetFaceNearPoint( fluid_6, geompy.MakeVertex( 0,
100, z6 ))
upWall_box6 = geompy.CreateGroup( fluid_6, geompy.ShapeType["FACE"])
geompy.UnionList( upWall_box6, [ up_box6 ])
down_box6 = geompy.GetFaceNearPoint( fluid_6, geompy.MakeVertex( 0,
-100, z6 ))
downWall_box6 = geompy.CreateGroup( fluid_6,
geompy.ShapeType["FACE"])
geompy.UnionList( downWall_box6, [ down_box6 ])
fixedWall_fluid6 = geompy.UnionListOfGroups( [frontWall_box6,
backWall_box6, upWall_box6, downWall_box6] )
geompy.addToStudyInFather(fluid_6,
fixedWall_fluid6, 'fixedWall_fluid6')
fluidRegion_6 = geompy.CreateGroup( partition,
geompy.ShapeType["SOLID"],"fluidRegion_6")
geompy.UnionList( fluidRegion_6, [ fluid_6 ])
fluidRegion = geompy.UnionListOfGroups( [fluidRegion_1,
fluidRegion_2, fluidRegion_3, fluidRegion_4, fluidRegion_5,
fluidRegion_6] )
geompy.addToStudyInFather(partition, fluidRegion, 'fluidRegion')
inertWall = geompy.CreateGroup(partition, geompy.ShapeType["FACE"])
geompy.UnionIDs(inertWall, [4, 1094, 1039, 56, 881, 1077, 48, 31,
1063, 1042, 14, 1101, 1022, 72, 1087, 1066, 38, 67, 1046, 896, 1111,
1090, 1029, 1070, 890, 34, 1114, 1053]) # Pareti del reattore da
modificare per ciascun impaccamaneto!
geompy.addToStudyInFather( partition, inertWall, 'inertWall' )
#####
###          GENERAZIONE DELLA MESH          ###
#####
# seleziona le superfici laterali dei bridge nella superficie
catalitica totale
#locSizeFaces = []
#for face in geompy.SubShapeAll( catalyticWall,
geompy.ShapeType["FACE"] ):
#     kind = geompy.KindOfShape( face )
#     print kind[0]
#     if ( kind[0] == geompy.kind.CYLINDER or
#         kind[0] == geompy.kind.CYLINDER2D or
#         kind[0] == geompy.kind.ARC_CIRCLE or
#         kind[0] == geompy.kind.PLANAR ):
#         locSizeFaces.append( geompy.GetSubShapeID(
partition, face ))
#         pass
#print locSizeFaces
#locSizeGroup = geompy.CreateGroup( partition,
geompy.ShapeType["FACE"], "locSizeGroup")

```

---

---

```

#geompy.UnionIDs( locSizeGroup, locSizeFaces )
import SMESH, SALOMEDS
from salome.smesh import smeshBuilder
smesh = smeshBuilder.New(theStudy)
from salome.NETGENPlugin import NETGENPluginBuilder
mesh = smesh.Mesh( partition, "Fluid mesh")
mesh.Tetrahedron( smeshBuilder.NETGEN_3D )
ng12 = mesh.Triangle( smeshBuilder.NETGEN_1D2D )
ng12.SetMaxSize( 0.18 )
algo1D = mesh.Segment( fluid_1 )
algo1D.NumberOfSegments( 20 )
mesh.Quadrangle( fluid_1 )
mesh.Hexahedron( fluid_1 )
algo1D = mesh.Segment( fluid_3 )
algo1D.NumberOfSegments( 20 )
mesh.Quadrangle( fluid_3 )
mesh.Hexahedron( fluid_3 )
algo1D = mesh.Segment( fluid_4 )
algo1D.NumberOfSegments( 20 )
mesh.Quadrangle( fluid_4 )
mesh.Hexahedron( fluid_4 )
algo1D = mesh.Segment( fluid_5 )
algo1D.NumberOfSegments( 20 )
mesh.Quadrangle( fluid_5 )
mesh.Hexahedron( fluid_5 )
algo1D = mesh.Segment( fluid_6 )
algo1D.NumberOfSegments( 20 )
mesh.Quadrangle( fluid_6 )
mesh.Hexahedron( fluid_6 )
NETGEN_2D3D = mesh.Tetrahedron(algo=smeshBuilder.NETGEN_1D2D3D,
geom=spheres )
NETGEN_3D_Parameters = NETGEN_2D3D.Parameters()
NETGEN_3D_Parameters.SetMaxSize( 0.18 )
NETGEN_3D_Parameters.SetSecondOrder( 0 )
NETGEN_3D_Parameters.SetOptimize( 1 )
NETGEN_3D_Parameters.SetFineness( 2 )
NETGEN_3D_Parameters.SetMinSize( 0.001 )
# 'local size'
#localSize = 0.005
#mesh.Segment( locSizeGroup ).LocalLength( localSize )
#mesh.Triangle( locSizeGroup )
mesh.Compute()
inlet = mesh.GroupOnGeom(inlet, 'inlet', SMESH.FACE)
outlet = mesh.GroupOnGeom(outlet, 'outlet', SMESH.FACE)
reactingWall =
mesh.GroupOnGeom(reactingWall, 'catalyticWall', SMESH.FACE)
inertWall = mesh.GroupOnGeom(inertWall, 'inertWall', SMESH.FACE)
fluidRegion =
mesh.GroupOnGeom(fluidRegion, 'fluidRegion', SMESH.VOLUME)
sfere = mesh.GroupOnGeom(Sfere, 'sfere', SMESH.VOLUME)
Bridges_SS = mesh.GroupOnGeom(Bridges_ss, 'bridges_SS', SMESH.VOLUME)
Bridges_WS = mesh.GroupOnGeom(Bridges_ws, 'bridges_WS', SMESH.VOLUME)
# Export la mesh
sys.path.insert( 0, r'/home/salvo/Scrivania')
import salomeToOpenFOAM
salomeToOpenFOAM.exportToFoam(mesh, '/home/salvo/Mesh_1/polyMesh')

```

---