



**POLITECNICO
DI MILANO**

Scuola di Ingegneria Industriale e dell'Informazione
Corso di Laurea Magistrale in Ingegneria Biomedica

**Progetto di un sistema software modulare per la simulazione
fluidodinamica della circolazione coronarica basata su modelli
0D e 3D**

Relatore:

Prof. Gabriele Dubini

Tesi di Laurea di:

Davide Pasini

Matr: 682456

Anno Accademico 2012 – 2013

Alla mia famiglia.....

*«Il Signore è il mio pastore: non manco di nulla;
su pascoli erbosi mi fa riposare, ad acque tranquille mi conduce.
Mi rinfranca, mi guida per il giusto cammino,
per amore del suo nome.
Se dovessi camminare in una valle oscura,
non temerei alcun male, perché tu sei con me.
Il tuo bastone e il tuo vincastro mi danno sicurezza.
Davanti a me tu prepari una mensa sotto gli occhi dei miei nemici;
cospargi di olio il mio capo. Il mio calice trabocca.
Felicità e grazia mi saranno compagne tutti i giorni della mia vita,
e abiterò nella casa del Signore per lunghissimi anni.»*

Salmo 23

Ringraziamenti

Un ringraziamento speciale va, prima di tutti, al professor Gabriele Dubini: una persona capace di farti capire in ogni momento che “ce la puoi fare” e che le cose non sono così complicate come sembrano.

Un grazie a Raffaella e a tutta la mia famiglia, per aver sostenuto/sopportato il mio lavoro in questi anni, anche nei momenti più difficili.

Grazie a Flavio e Sergio, capaci, con esempio e passione, di far nascere, in un bimbo di otto anni, il sogno che un giorno sarebbe diventato ingegnere.

Un grazie a nonna Ines che ha sempre creduto in me e mi ha insegnato che lo studio è il cibo della mente senza il quale non si può vivere.

Grazie mille a tutti gli amici storici e a quelli che ho incontrato in questa avventura universitaria, vi porto tutti nel mio cuore ...

Davide

Indice

Capitolo 1: L'albero coronarico: cenni di anatomia e fisiopatologia.....	3
1.1. Anatomia delle coronarie.....	4
Arteria coronaria di sinistra.....	4
Arteria coronaria di destra.....	5
1.2. Patologie coronariche: coronaropatie.....	5
Cause.....	5
Terapia farmacologica.....	7
Intervento di by-pass.....	7
Trattamenti mininvasivi.....	8
1.3. Statistiche europee riguardanti l'incidenza delle malattie vascolari sulla mortalità e sui costi per la collettività.....	9
Fattori di Rischio.....	11
Capitolo 2: I modelli multiscala 0D/3D del sistema cardiovascolare.....	13
Capitolo 3: Il software Open Source. In particolare i software scientifici Open Source utilizzati.....	17
3.1. Introduzione al software open source.....	18
3.2. Strumenti di modellazione : Modelica e OpenModelica.....	18
a) Approccio causale rispetto all'approccio acausale.....	18
b) La Modellazione Orientata agli Oggetti e Modelica.....	22
c) Modellazione fisica vs. Modellazione causale.....	24
3.3. Strumento di simulazione CFD: OpenFOAM.....	26
Struttura di un case in OpenFOAM.....	28
Capitolo 4: Modello matematico 0D.....	30
a) Analogia elettrica del modello a parametri concentrati.....	35
4.2. Modello matematico del circuito sistemico e polmonare.....	38
4.3. Modello matematico del cuore.....	41
4.4. Modello matematico dell'albero coronarico.....	43
Capitolo 5: Implementazione tramite il linguaggio Modelica del modello 0D.....	45
5.1. Attivazione cardiaca.....	46
5.2. Vasi arteriosi.....	47
5.3. Circuito Sistemico e muscolo cardiaco.....	47
5.4. Albero coronarico (coronaria sinistra).....	48

5.5.Sistema 0D completo.....	49
Capitolo 6: Modello fluidodinamico 3D.....	50
6.1.Definizione della geometria.....	51
a)Geometria e mesh del vaso fisiologico.....	51
b)Geometria e mesh del vaso patologico.....	53
6.2.Modello fluidodinamico adottato.....	55
a)Implementazione del modello in OpenFOAM.....	56
Capitolo 7: Il problema accoppiato 0D/3D.....	58
Capitolo 8: Implementazione software di gestione della simulazione CoSimO: COronary SIMulation Organizer.....	63
a)Introduzione a Python.....	64
b)Le librerie OMPython e PyFoam.....	65
c)Funzionamento di CoSimO.....	65
d)Blocco di interscambio dati tra modello 0D e CoSimO.....	67
Capitolo 9: Risultati.....	70
9.1.Risultati modello 0D, caso 1.....	72
a)Impostazioni della simulazione.....	72
9.2.Risultati modello 0D, caso 2.....	76
9.3.Risultati modello accoppiato 0D/3D, caso 1.....	80
a)Impostazioni della simulazione.....	80
9.4.Risultati modello accoppiato 0D/3D, caso 2.....	84
a)Impostazioni della simulazione.....	84
Capitolo 10: Discussione, limitazioni e sviluppi futuri.....	90
10.1.Obiettivo 1.....	91
10.2.Obiettivo 2.....	93

Sommario

1. Introduzione

Le malattie cardiovascolari (cardiovascular disease: CVD) sono la principale causa di mortalità e morbilità in Europa: contano più di 4 milioni di decessi ogni anno, circa la metà di tutte le morti sono causate da CVD (52% delle morti nelle donne e il 42% delle morti negli uomini). Viene stimato un costo per l'economia europea, dovuto alle malattie cardiovascolari, pari a 196 miliardi di Euro all'anno. Del costo totale per le CVD in Europa, circa il 54% è dovuto a costi per cure dirette, 24% per perdite produttività e il 22% per le cure informali di persone affette da CVD.

Le maggiori forme di malattie cardiovascolari sono le malattie cardiache coronariche (coronary heart disease: CHD) e l'infarto del miocardio. Solo poco la metà delle morti dovute a malattie cardiovascolari sia per gli uomini che per le donne sono malattie cardiache coronariche.

Le coronaropatie sono causate da un accumulo di depositi di grasso di consistenza cerosa sulle pareti interne delle arterie. Questi depositi sono costituiti da colesterolo, calcio e altre sostanze che sono trasportate dal sangue e il prodotto del loro accumulo è definito "placca aterosclerotica" o semplicemente "placca". Le placche possono ostruire le arterie coronarie e renderle rigide e irregolari, provocando il cosiddetto "indurimento delle arterie" o aterosclerosi.

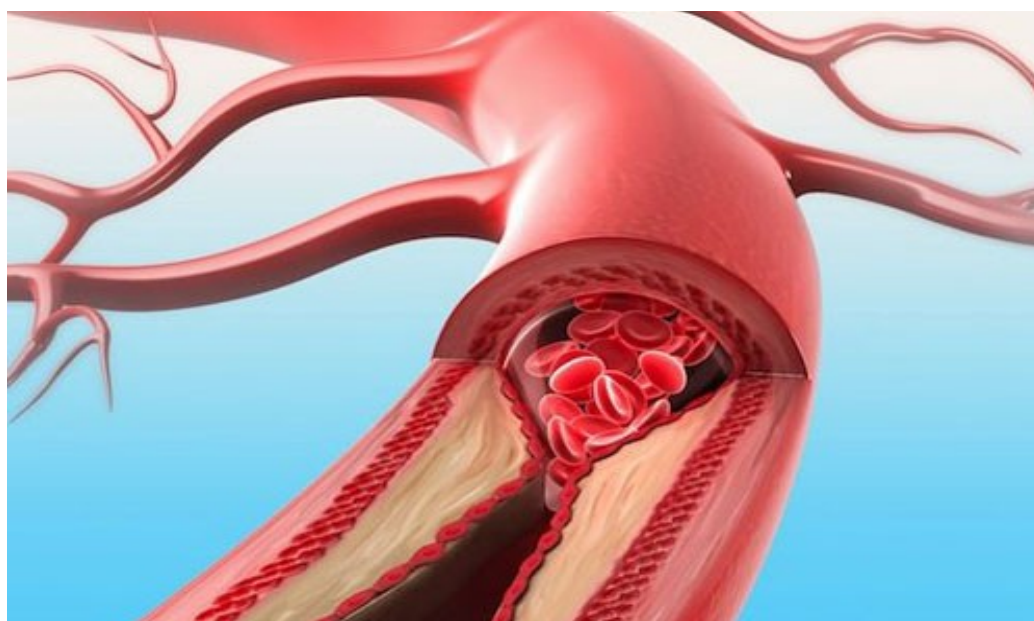


Figura 1: Rappresentazione di un vaso aterosclerotico [<http://www.curiomed.it/larteriosclerosi/>].

Lentamente, i depositi restringono il lume delle arterie coronarie riducendo così l'apporto di sangue e ossigeno al muscolo cardiaco (Figura 1). Tale riduzione del flusso ematico può causare dolore al torace (angina), difficoltà respiratorie (dispnea) e altri sintomi, mentre l'ostruzione completa può indurre un attacco cardiaco.

Esistono svariate misure da adottare per controllare una malattia cardiovascolare: assumere particolari farmaci, modificare l'alimentazione e praticare attività fisica. Quando tutto ciò non è sufficiente, una delle tecniche chirurgiche maggiormente utilizzate per ripristinare il corretto flusso ematico al miocardio consiste nell'impianto di uno stent in un intervento chirurgico di by-pass.

La fluidodinamica computazionale risulta essere un valido strumento di indagine sia in ambito clinico, per quanto riguarda nella diagnosi medica, che industriale nella progettazione di dispositivi medici.

Negli ultimi anni, grazie alla crescente capacità di calcolo, sono stati sviluppati studi basati su modelli multiscala che prevedono quindi l'accoppiamento di modelli a parametri concentrati con modelli tridimensionali.

La maggior parte de software commerciali o open source normalmente utilizzati, prevedono la descrizione matematica sia del modello a parametri concentrati che del modello tridimensionale all'interno di un unico ambiente di sviluppo.

In letteratura esistono diversi lavori riguardanti modelli multiscala del sistema cardiovascolare. A titolo di esempio nel lavoro di K. Laganà [1] è stato utilizzato l'approccio sopra descritto relativamente allo studio della perfusione polmonare e coronarica nel caso di circolazione univentricolare.

In questo lavoro di tesi viene proposto il progetto di un sistema software modulare per la simulazione fluidodinamica della circolazione coronarica basata su modelli 0D e 3D. L'obiettivo ultimo è quello di creare uno strumento attraverso il quale si possa accoppiare con facilità modelli predefiniti del sistema cardiocircolatorio con un dominio 3D di interesse.

Open source (termine inglese che significa codice sorgente aperto), in informatica, indica un software i cui autori (più precisamente i detentori dei diritti) ne permettono e favoriscono il libero studio e l'apporto di modifiche da parte di altri programmatori indipendenti. Questo è realizzato mediante l'applicazione di apposite licenze d'uso.

L'open source non va considerato come una scappatoia per aggirare il problema dell'acquisto o dell'utilizzo di un software proprietario, ma diviene una vera e propria scelta di

campo in favore della libertà di circolazione e dello scambio di idee.

Questo principio ha determinato la scelta di utilizzare il software open source, per quanto possibile, nello sviluppo di questo lavoro di tesi.

2. Materiali e metodi

Attraverso il linguaggio di programmazione open source Modelica è stato sviluppato un modello a-causale del sistema cardiovascolare composto da: circolo sistemico, circolo polmonare e pompa cardiaca. Il modello a parametri concentrati sfrutta l'analogia idraulica/elettrica dei modelli fluidodinamici.

OpenModelica è un framework di interpretazione e risoluzione dei modelli creati in linguaggio Modelica. Attraverso questo framework, che comprende anche un editor grafico OMEdit, sono stati realizzati i blocchi funzionali del modello.

Con lo stesso sistema è stato sviluppato un modello dettagliato dell'albero coronarico, in particolare il ramo sinistro.

Questi blocchi, opportunamente accoppiati costituiscono un modello di sistema cardiovascolare indipendente e funzionante.

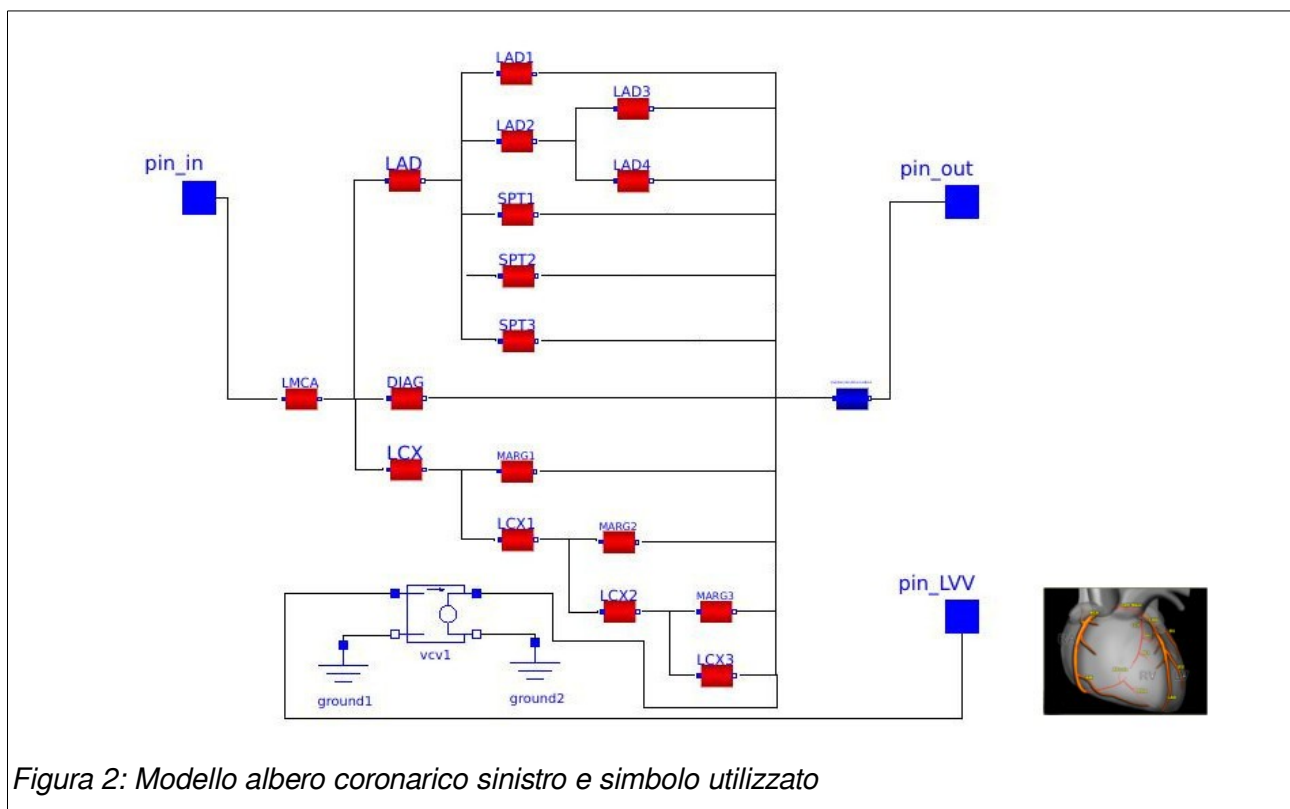


Figura 2: Modello albero coronarico sinistro e simbolo utilizzato

Parallelamente, attraverso il framework open source OpenFOAM, per la simulazione fluidodinamica computazionale CFD (Computational Fluid Dynamics), è stato realizzato un modello di arteria coronarica che costituisce il dominio di interesse.

Tale modello è stato realizzato in due configurazioni: fisiologica e con presenza di stenosi.

I modelli fluidodinamici adottati sono stati:

Fluido Newtoniano incomprimibile e pareti del vaso indeformabili. Nel caso della geometria con presenza di stenosi è stato introdotto il modello di turbolenza a due equazioni $k - \omega$.

Il modello 3D, configurato all'interno di OpenFOAM, costituisce un modello indipendente che può essere risolto stand alone.

La separazione del codice tra i modelli 0D e 3D permette di poterli sviluppare in modo autonomo l'uno dall'altro e effettuare tutte le operazioni preliminari di preprocessing.

E' stato realizzato il software CoSimO (Coronary Simulation Organizer) per la gestione della simulazione multiscala ottenuta accoppiando il modello 0D a quello 3D.

CoSimO è realizzato tramite il linguaggio di scripting Python. CoSimO gestisce la simulazione accoppiata dei due modelli: 3D (OpenFOAM) e 0D (OpenModelica).

Per fare questo è necessaria la definizione di "parametri di accoppiamento" tra i quali:

- passo della simulazione (timestep);
- tempo finale della simulazione (endTime);
- variabili di interfaccia tra i modelli 0D e 3D;
- coefficienti legati all' algoritmo la risoluzione dei modelli accoppiati.

Tali parametri sono contenuti all'interno di un file di configurazione modificabile dall'utente utilizzatore.

L'algoritmo utilizzato per l'accoppiamento e l'esecuzione delle simulazione è di tipo implicito. Ad ogni passo della simulazione, per poter passare al passo successivo, i due sistemi devono convergere a valori comuni delle variabili di interfaccia.

3. Risultati

L'esecuzione del modello 0D non accoppiato presenta dei risultati comparabili rispetto ad altri lavori precedenti, in termini di grandezze fluidodinamiche.

Nella rete elettrica è stata simulata la condizione di stenosi nel ramo discendente anteriore interventricolare (LAD), attraverso la variazione del parametro elettrico "resistenza".

I risultati hanno evidenziato una diminuzione della portata di sangue nel vaso interessato e

un aumento della pressione a monte del vaso.

Non sono state evidenziate variazioni nel comportamento del modello cardiaco. Questo risultato è determinato dal fatto che non è presente un modello di attivazione del muscolo cardiaco dipendente dalla perfusione coronarica.

L'esecuzione dei modelli 0D e 3D accoppiati presenta dei risultati generali comparabili con gli andamenti di pressione e portata ricavati dal solo modello 0D. Risultano però forme d'onda diverse dovute al tipo di modello 3D utilizzato che non comprende effetti della deformabilità della parete del vaso che invece erano presenti, seppur come parametro concentrato, nel modello 0D.

4. Conclusioni

I due principali obiettivi di questo lavoro di tesi sono stati:

- Produrre un modello modulare 0D del sistema cardiovascolare che fosse attendibile, modulare e quindi migliorabile attraverso implementazioni aggiuntive senza dover ridefinire quanto già implementato.
- Mettere a punto un sistema software in grado di accoppiare modelli 0D e 3D indipendenti tra loro, il tutto sfruttando codice open source.

I risultati hanno evidenziato la bontà del modello 0D, la sua velocità di esecuzione e la facilità nel modificare i suoi parametri interni. Quest'ultima caratteristica è dovuta al tipo di codice che è molto simile alle equazioni matematiche rappresentanti il modello fisico e non ad un codice di programmazione procedurale.

Lavori futuri potranno migliorare il modello 0D, introducendo blocchi aggiuntivi o modificando quelli esistenti, e, soprattutto, validare tale modello attraverso la raccolta e il confronto con dati clinici.

Il software CoSimO, nella versione attuale, rappresenta un punto di partenza nell'accoppiamento modulare di sistemi 0D e 3D. Lavori futuri potranno studiare altri algoritmi di interfacciamento e di esecuzione della simulazione al fine di migliorarne le prestazioni temporali.

Summary

1. Introduction

Diseases of the heart and circulatory system (cardiovascular disease or CVD) are the main cause of death in Europe: accounting for over 4 million deaths each year. Nearly half of all deaths are from CVD (52% of deaths in women and 42% of deaths in men).

Is estimated to cost the EU economy almost €196 billion a year. Of the total cost of CVD in the EU, around 54% is due to direct health care costs, 24% to productivity losses and 22% to the informal care of people with CVD.

The main forms of CVD are coronary heart disease (CHD) and stroke. Just under half of all deaths from CVD in both men and women are from CHD.

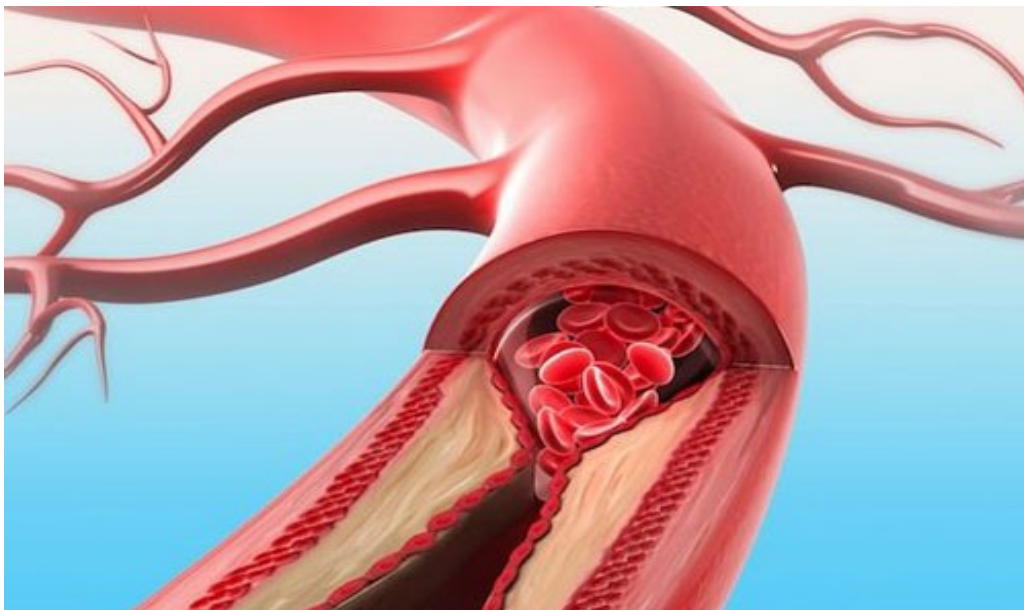


Figure 3: Representation of an atherosclerotic vessel [<http://www.curiomed.it/larteriosclerosi/>].

Coronary heart diseases are caused by an accumulation of waxy fat deposits on inner walls of arteries. These deposits are composed of cholesterol, calcium and other substances that are transported by blood and the product of their accumulation is defined "atherosclerotic plaque" or simply "plaque". The plaques can block the coronary arteries and make them stiff and uneven, causing the so-called "hardening of the arteries" or atherosclerosis.

Slowly, deposits narrow the lumen of the coronary arteries, reducing supply of blood and oxygen to the heart muscle (Figure 3). This reduction in blood flow may cause chest pain (angina), breathing difficulty (dyspnoea) and other symptoms, while the complete obstruction

can cause a heart attack.

There are several measures to control cardiovascular disease: take particular medicines, change diet and physical activity. When all this is not enough, one of the most used surgical techniques to restore the proper blood flow to the myocardium consists of an implantation of a stent in a surgical by-pass.

Computational fluid dynamics turns out to be a valuable tool both in clinical investigation, with regard to the medical diagnosis, and in industrial design of medical devices.

In the last few years, thanks to the increasing computing power, several studies based on multi-scale models have been developed; they predict the coupling of lumped parameter models with three-dimensional models.

Most of commercial software and commonly used open-source systems provide a mathematical description both of the lumped parameter model and the three-dimensional model within a single development environment.

In the literature there are several works about multi-scale models of cardiovascular system. As an example, in the work of K. Lagana [1] the above mentioned approach was used to study the pulmonary and coronary perfusion in case of univentricular circulation.

This thesis proposes the development of a modular software system for fluid dynamics simulation of the coronary circulation based on 0D and 3D models. The main goal is to create a tool through which you can easily couple predefined templates of cardiovascular system to a 3D domain of interest.

Open source (English word that means open source code), in Informatics, indicates a software whose authors (more precisely the rights holders) allow and encourage free study and modifications made by other independent programmers. This is achieved through the application of appropriate licenses.

Open source should not be considered as a loophole to get around the problem of the purchase or the use of proprietary software, but it becomes a real choice in favor of the freedom to exchange ideas.

This principle has led to the decision to use open source software, as far as possible, in the development of this thesis.

2. Materials and methods

Through the open source programming language Modelica, acausal model of the cardiovascular system has been developed. It consists of: systemic circulation, pulmonary circulation and cardiac pump. The lumped parameter model uses the hydraulic / electric analogy of fluid dynamic models.

OpenModelica is a framework of interpretation and resolution of models created in Modelica language. Through this framework, which also includes a graphical editor (OMEdit), functional blocks of the model have been realized.

Through the same system a detailed model of the coronary tree, especially the left branch, has been developed.

These functional blocks, suitably coupled, form a functioning and independent model of cardiovascular system.

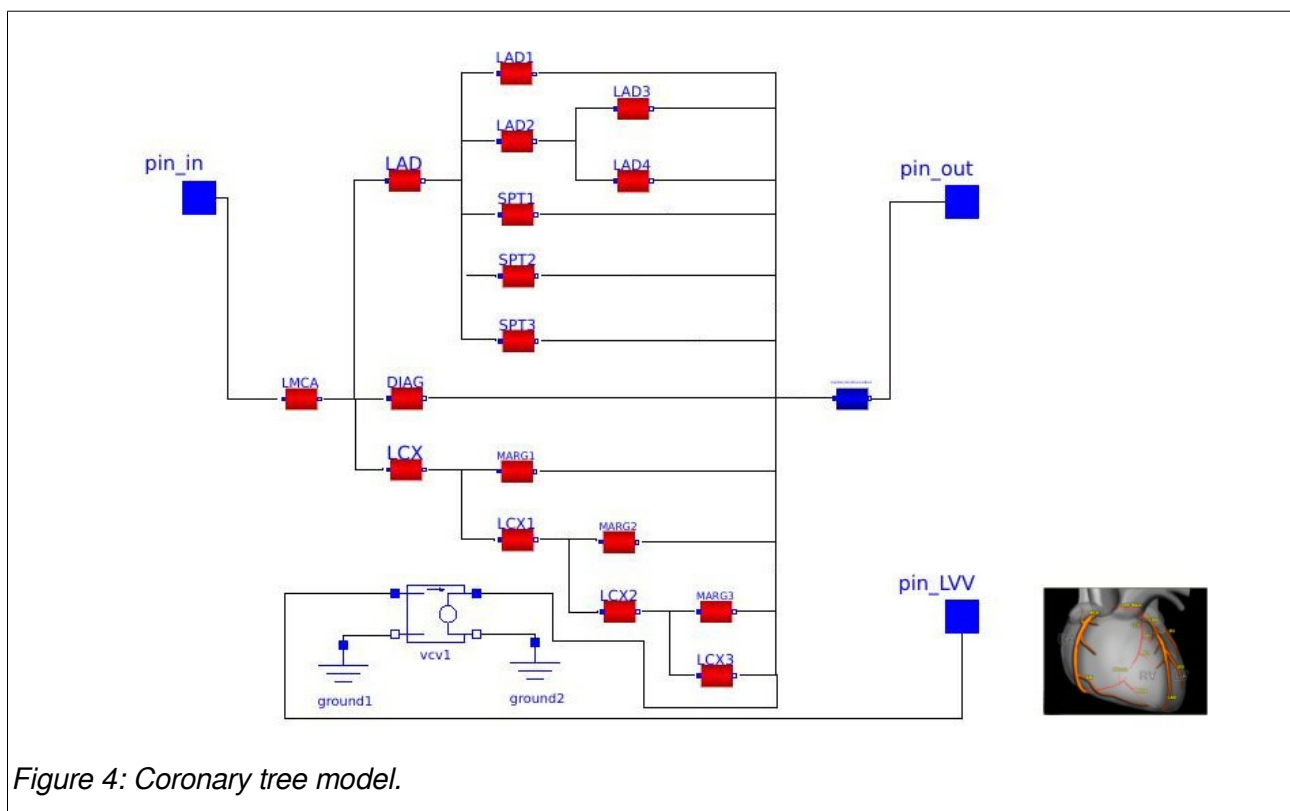


Figure 4: Coronary tree model.

At the same time, through the open source framework OpenFOAM, used for the computational fluid dynamics simulation - CFD (Computational Fluid Dynamics) - a model of coronary artery has been created and it is the domain of interest.

This model has been developed within two configurations: physiological and in the presence of stenosis.

The adopted fluid dynamic models have been:

Incompressible Newtonian fluid and deformable vessel walls. In case of the geometry with presence of stenosis, the turbulence model with two equations has been introduced.

The 3D model, configured in OpenFOAM, is an independent model that can be solved 'stand alone'.

The separation of code between 0D and 3D models allows to develop them independently of each other and to do all the preliminary preprocessing.

CoSimO software (Coronary Simulation Organizer) has been developed to manage the multiscale simulation obtained by coupling model 0D to 3D.

CoSimO is developed through the scripting language Python and it manages the coupled simulation of the two models: 3D (OpenFOAM) and 0D (OpenModelica).

To do this you need the definition of "coupling parameters" such as:

- simulation step (timestep);
- end time of the simulation (endTime);
- interface variables between 0D and 3D models;
- coefficients related to the algorithm to solve coupled models.

These parameters are contained in a configuration file that user can modify.

The algorithm used to match and execute the simulation is implicit type. At each step of the simulation, in order to move to the next step, the two systems have to converge to common values of interface variables.

3. Results

The execution of not paired 0D model offers results compared to other previous works, in terms of fluid-dynamic.

In the electricity network the condition of stenosis in the left anterior descending artery (LAD) was simulated through the variation of the electrical parameter "resistance".

The results showed a decrease in the blood flow in the affected vessel and a pressure increase upstream of the vessel.

There were no changes in the behavior of the heart model. This result is due to the fact that there is no model of activation of the heart muscle dependent on coronary perfusion.

The performance of coupled 0D and 3D models shows general results comparable with the trends of pressure and flow obtained from 0D model. However, there are different waveforms due to the kind of 3D model used which does not include effects of the deformability of the

vessel wall. On the contrary they were present, even as a concentrated parameter , in 0D model.

4. Conclusions

The two main aims of this thesis have been:

- to create a 0D model of cardiovascular system that is reliable, modular and improvable through additional implementations without having to redefine what has already been implemented.
- to develop a software able to match 0D and 3D models independent of each other, by using open source code.

The results have showed the accuracy of 0D model, its execution speed and the simplicity in changing its internal parameters. This latter feature is due to the kind of code very similar to mathematical equations representing the physical model, and not similar to a code of procedural programming.

Future work will improve 0D model, introducing additional blocks or modifying the existing ones, and, above all, will validate this model through the collection and comparison with clinical data.

CoSimO software, in the current version, is a starting point in the modular coupling of 0D and 3D systems. Future work will investigate other algorithms to interface and execute simulation in order to improve its timing performance.

Introduzione

Le arterie coronarie sono i vasi sanguigni responsabili dell'apporto di nutrienti al muscolo cardiaco. Tali vasi possono andare incontro a diverse patologie, in particolare alla formazione di occlusioni. E' ben nota la correlazione tra alterazioni dell'emodinamica locale e la formazione di placche aterosclerotiche.

Durante gli ultimi decenni, sono state introdotte terapie e dispositivi al fine di prevenire e curare le malattie cardiovascolari. La fluidodinamica computazionale risulta essere un valido strumento di indagine, sia in ambito clinico per quanto riguarda la diagnosi medica, che industriale nella progettazione di dispositivi medici.

I modelli fluidodinamici cardiovascolari sono indispensabili in questi settori. La comune pratica consiste nello studio fluidodinamico di modelli isolati ai quali vengono imposti opportuni valori di pressione e portata come condizioni al contorno. Negli ultimi anni, grazie alla crescente capacità di calcolo, sono stati sviluppati studi basati su modelli multiscala che prevedono l'accoppiamento di modelli a parametri concentrati (0D) con modelli tridimensionali (3D).

Un modello a parametri concentrati con un buon livello di dettaglio permette di studiare comportamenti di sistemi nel loro complesso e di porre le condizioni al contorno adeguate per lo studio di domini tridimensionali dettagliati. Spesso, nei modelli multiscala, la soluzione del dominio 0D è inglobata nel codice che risolve il dominio tridimensionale. Questo approccio implica una notevole difficoltà nel riutilizzare il codice sviluppato per altre simulazioni.

In questo ambito si pone questo lavoro di tesi che ha lo scopo di:

- sviluppare un modello a parametri concentrati del sistema cardiovascolare focalizzato soprattutto al circuito coronarico.
- mettere a punto un sistema software in grado di accoppiare modelli 0D e 3D indipendenti tra loro, sfruttando codice open source.

Nel primo capitolo è presente una breve introduzione riguardante l'anatomia e la fisiopatologia del circuito coronarico. Vengono proposte alcune statistiche che indicano come le malattie cardiovascolari siano ancora un grande costo per la nostra società prima di tutto in termini umani ma anche in termini economici.

Il secondo capitolo introduce il software open source che verranno utilizzati in questo lavoro.

Il terzo ed il quarto capitolo descrivono il modello matematico 0D e la sua implementazione computazionale di tipo a-causale attraverso il linguaggio open source Modelica ed il tool OpenModelica.

Nel quinto capitolo è presentato il modello matematico del sistema 3D e la sua implementazione computazionale attraverso l'utilizzo del software OpenFOAM.

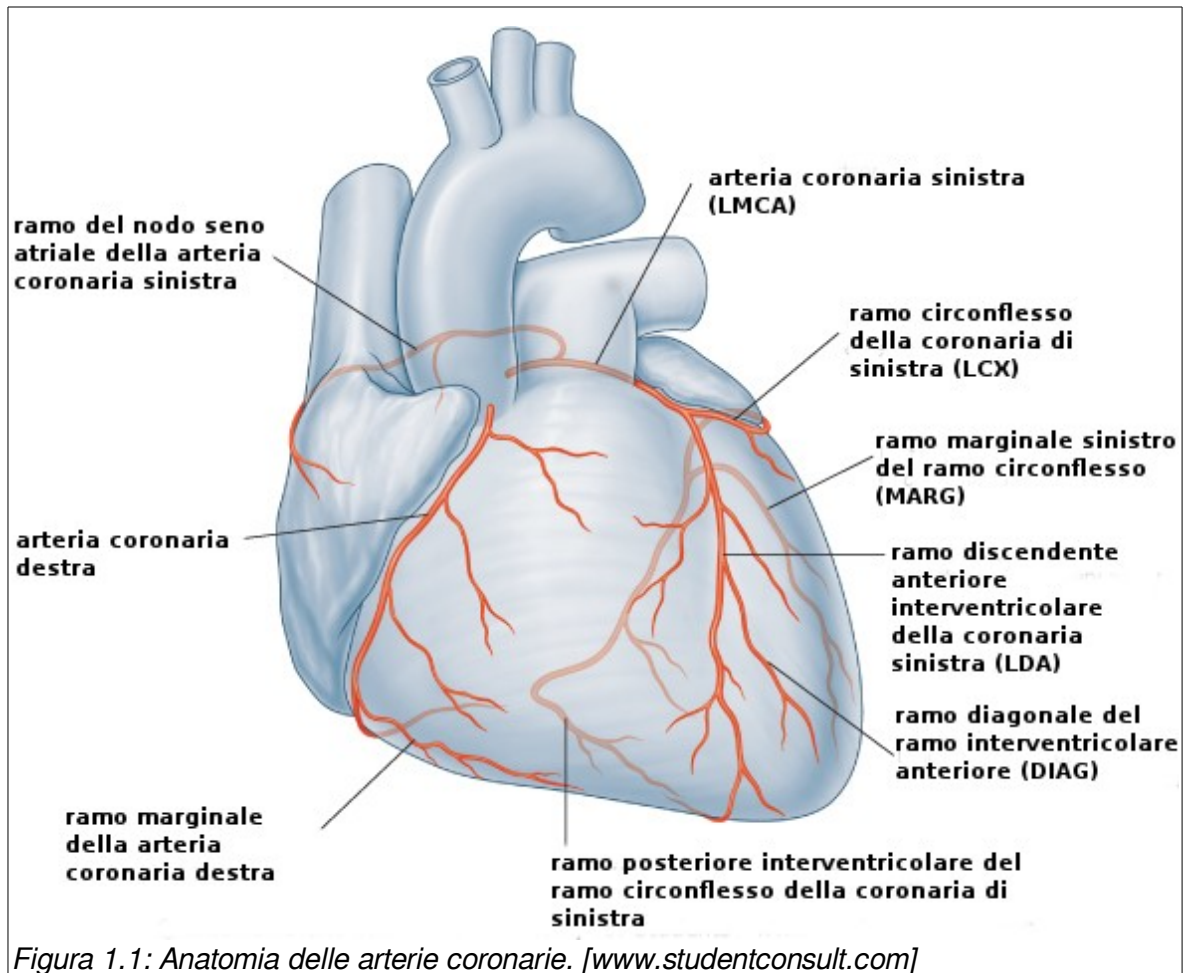
Il sesto ed il settimo capitolo descrivono la strategia utilizzata nell'accoppiamento dei sistemi 0D e 3D e la sua implementazione che da origine al software CoSimO (Coronary Simulation Organizer)

Nei successivi due capitoli, ottavo e nono, sono presentati i risultati delle simulazioni numeriche effettuate e la loro analisi.

Capitolo 1:

L'albero coronarico: cenni di anatomia e fisiopatologia

In questo capitolo vengono descritti i principi di anatomia e fisiopatologia delle coronarie e il grave impatto che le malattie cardiovascolari hanno sulla società, in particolare quelle coronariche.



1.1. Anatomia delle coronarie

Nella porzione sopravvalvolare, l'aorta toracica ascendente presenta un segmento dilatato, detto bulbo aortico, il quale è delimitato dalla rimanente aorta dall'anello valvolare. Tale anello, comprende i seni di Valsalva (destro sinistro e non coronarico), dai quali originano le arterie coronarie. Gli osti di queste, sono generalmente situati al centro dei seni di Valsalva.

Arteria coronaria di sinistra

L'arteria coronaria sinistra (LCA), dopo la sua origine dal seno di Valsalva sinistro, presenta un tronco comune (left main coronary artery: LMCA) che decorre per 5\20 mm tra l'arteria polmonare e la parete anteriore dell'atrio sinistro, per poi dividersi nei due rami principali:

- Ramo discendente anteriore (left anterior descending: LAD) che decorre nel solco interventricolare anteriore, fino a raggiungere la punta del cuore. Tale arteria, durante il suo decorso emette due diversi tipi di ramificazioni: i rami settali (SPTx) o detti perforanti (che irrorano il fascio di His) ed i rami diagonali (DIAG) (con decorso

obliquo).

- Ramo circonflesso (Circumflex: LCX), che decorre lungo il solco atrio-ventricolare sinistro e da origine ai rami ottusi marginali (MARGx). Tale vaso, durante il suo percorso continua nel ramo postero- laterale (che si distribuisce sulla parete posteriore del ventricolo sinistro) e nel ramo atriale sinistro [2].

Arteria coronaria di destra

L'arteria coronaria di destra (RCA), dopo la sua origine dal seno di Valsalva destro, decorre lungo il solco atrio-ventricolare destro (anteriore), che percorre in tutta la sua lunghezza, fino alla superficie diaframmatica del cuore.

All'altezza della crux (unione tra solco atrioventricolare e solco interventricolare posteriore), si divide in due rami terminali:

- Ramo discendente posteriore (che dà origine ai rami settali).
- Rami per la parete del ventricolo sinistro.

Entrambe le arterie coronarie, come indica il loro nome, descrivono nel loro decorso due distinte corone: Corona anteriore composta da, dall'arteria discendente anteriore (LAD) e dall'arteria discendente posteriore destra (RPD) Corona posteriore, composta dal tronco comune della coronaria sinistra, arteria circonflessa (LCX) e suoi rami terminali, branca postero-laterale sinistra e arteria coronaria destra(RCA).

Entrambe le corone vascolari sono ampiamente anastomizzate fra loro, attraverso rami terminali e collaterali [2].

1.2. Patologie coronariche: coronaropatie.

L'ostruzione di un'arteria coronaria può essere all'origine di un attacco di cuore. Le coronaropatie rappresentano una delle principali cause di morte.

Si definisce coronaropatia una malattia del cuore responsabile di un apporto ematico inadeguato al muscolo cardiaco, condizione che può avere risvolti nefasti. Un altro termine usato per fare riferimento alle coronaropatie è "cardiopatia coronarica".

Cause

Le coronaropatie sono causate da un accumulo di depositi di grasso di consistenza cerosa sulle pareti interne delle arterie. Questi depositi sono costituiti da colesterolo, calcio e altre sostanze che sono trasportate dal sangue e il prodotto del loro accumulo è definito "placca aterosclerotica" o semplicemente "placca". Le placche possono ostruire le arterie coronarie e

renderle rigide e irregolari, provocando il cosiddetto "indurimento delle arterie" o aterosclerosi.

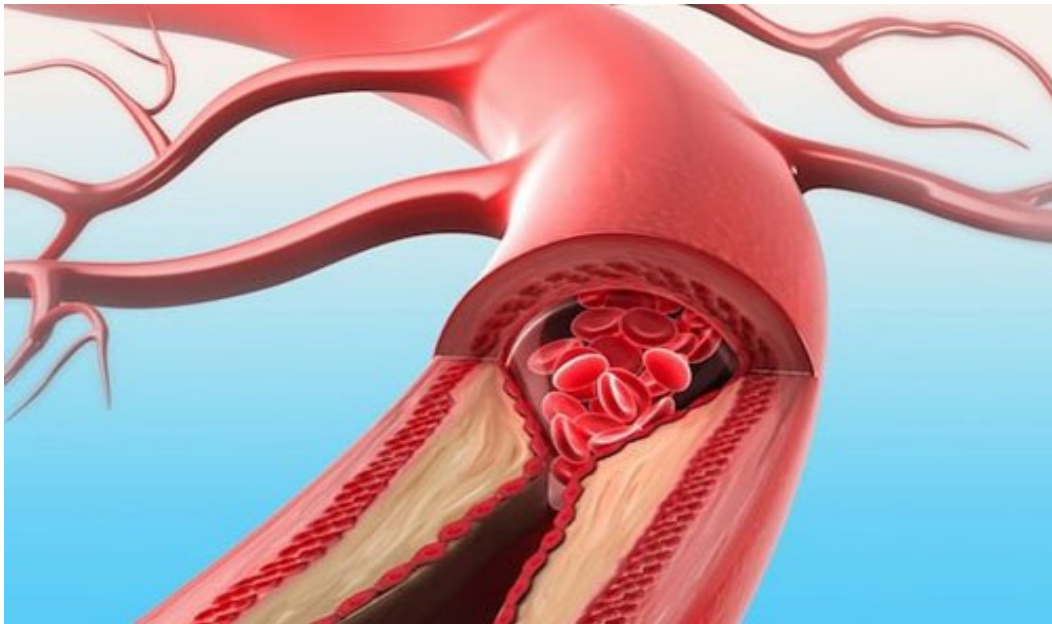


Figura 1.2: Rappresentazione di un vaso aterosclerotico [<http://www.curiomed.it/larteriosclerosi/>].

Le ostruzioni possono essere singole o multiple e presentare vari livelli di gravità e differenti ubicazioni. A poco a poco, i depositi restringono il lume delle arterie coronarie riducendo così l'apporto di sangue e ossigeno al muscolo cardiaco (Figura 1.3). Tale riduzione del flusso ematico può causare dolore al torace (angina), difficoltà respiratorie (dispnea) e altri sintomi, mentre l'ostruzione completa può indurre un attacco cardiaco.

Il ridotto apporto ematico al miocardio è generalmente dovuto all'aterosclerosi, patologia caratterizzata da lesioni infiltrate di lipidi (placche ateromatose) che si sviluppano nella superficie interna dei vasi coronarici, riducendone il lume e provocando un inspessimento delle pareti delle arterie stesse.

Esistono svariate misure da adottare per controllare una malattia cardiovascolare: assumere particolari farmaci, modificare l'alimentazione e praticare attività fisica. Quando tutto ciò non è sufficiente, una delle tecniche chirurgiche maggiormente utilizzate per ripristinare il corretto flusso ematico al miocardio consiste nell'impianto di uno stent in un intervento chirurgico di by-pass.

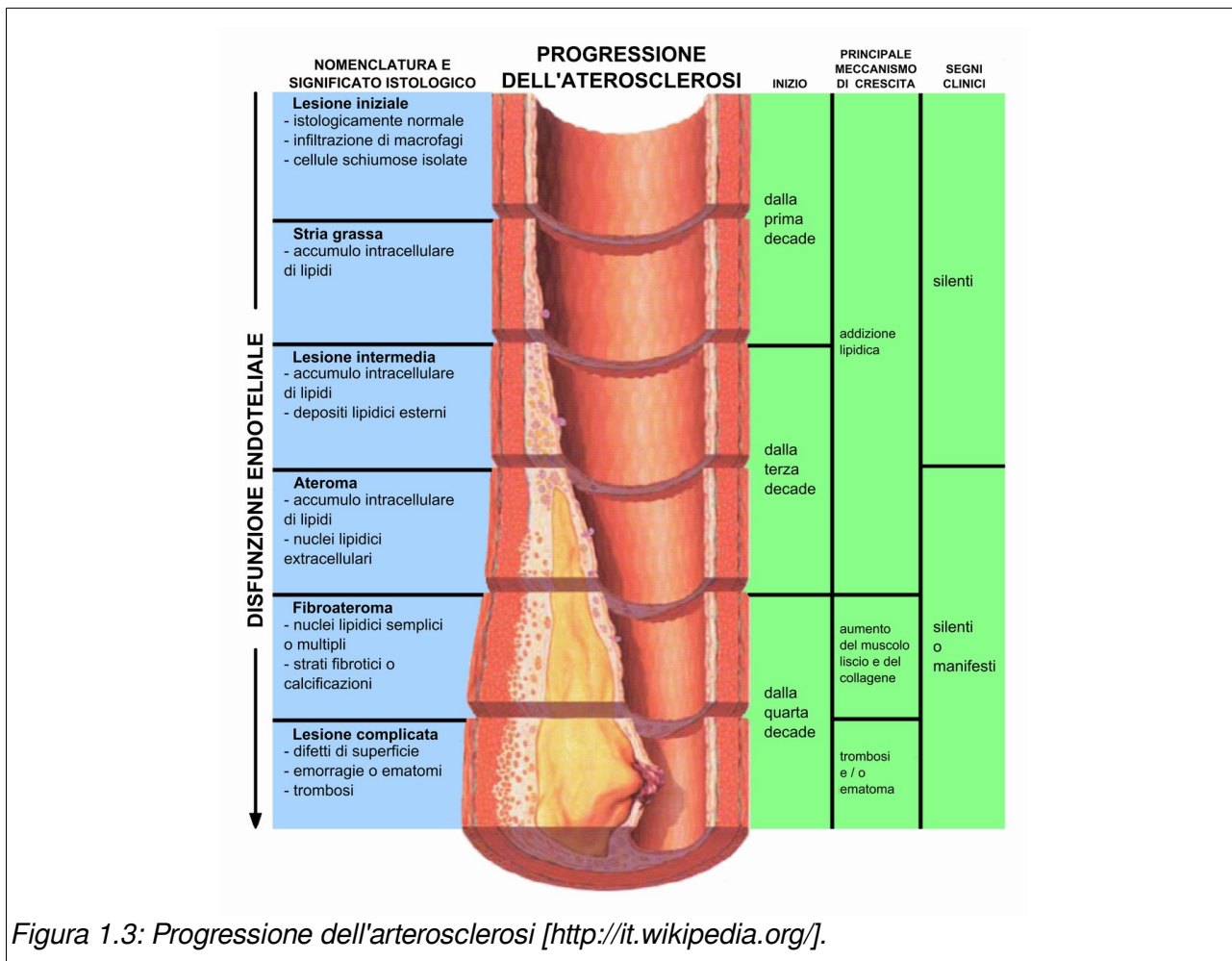


Figura 1.3: Progressione dell'arterosclerosi [http://it.wikipedia.org/].

Terapia farmacologica

Se l'aterosclerosi viene identificata a uno stadio precoce, vengono utilizzati farmaci quali nitrati, betabloccanti, calcio-antagonisti, acido acetilsalicilico e ipocolesterolizzanti (statine). Questi agenti riducono la progressione della malattia o ne alleviano i sintomi.

Intervento di by-pass

Il by-pass aorto-coronarico o "CABG" (pronuncia inglese "cabbage") è una procedura cardiaca comune. Il chirurgo rimuove una sezione di un vaso sanguigno sano dalla gamba, dal torace o dal braccio. Il vaso viene quindi collegato all'arteria coronaria appena a monte del punto interessato dall'ostruzione. Si crea così un percorso alternativo attraverso il quale il sangue potrà fluire evitando il tratto ostruito dell'arteria e raggiungere il cuore.

L'intervento di by-pass viene eseguito in anestesia totale e il paziente rimane incosciente durante la procedura. Due diverse procedure di by-pass chirurgico per le coronaropatie sono: l'intervento a cuore battente e l'intervento a cuore fermo.

Intervento a cuore battente: definito anche intervento off-pump, viene eseguito mentre il cuore batte. La procedura spesso richiede degli strumenti speciali che consentono al chirurgo di operare sul cuore mentre questo si muove. L'intervento a cuore battente risulta indicato per alcuni pazienti per i quali la circolazione extracorporea comporterebbe un rischio aggiuntivo di complicanze postoperatorie.

Intervento a cuore fermo: la maggior parte degli interventi di by-pass aorto-coronarico viene eseguita praticando un'incisione nel torace mentre il cuore è arrestato e una macchina cuore-polmone si occupa di garantire la circolazione del sangue all'interno del sistema cardiocircolatorio. Questa procedura viene definita chirurgia a cuore fermo ed è l'intervento di by-pass tradizionale.

Trattamenti mininvasivi

Per alcuni pazienti, un'alternativa al by-pass aorto-coronarico è rappresentata dalla chirurgia aorto-coronarica mininvasiva. L'angioplastica coronarica con palloncino, l'impianto di stent e la chirurgia di by-pass aorto-coronarico mininvasiva sono tre opzioni di trattamento mininvasivo delle coronaropatie.

Angioplastica coronarica con palloncino: definita anche intervento coronarico percutaneo (ossia eseguito attraverso la cute), utilizza un palloncino per dilatare il lume interno dell'arteria e permettere al sangue di tornare a fluire a una velocità normale o pressoché normale.

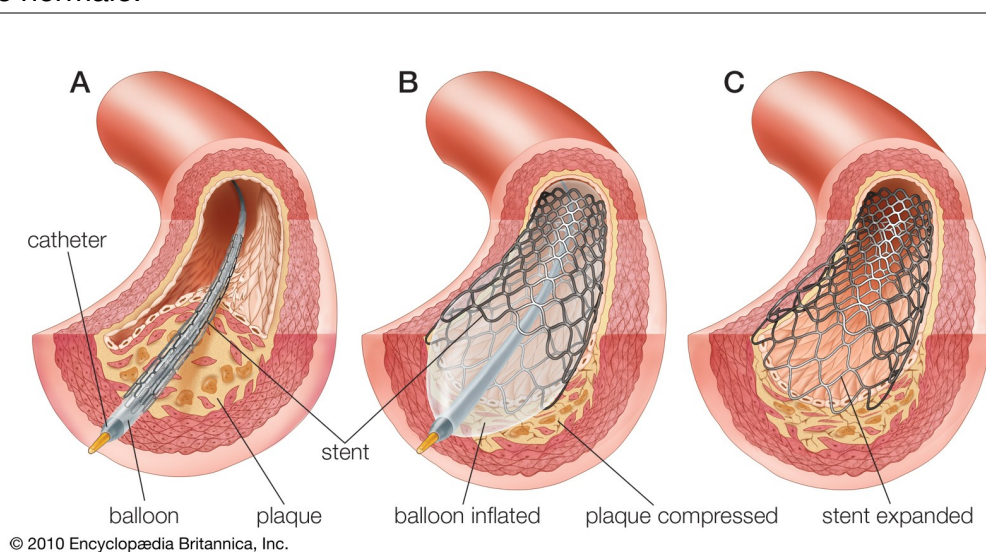


Figura 1.4: Angioplastica con applicazione di stent. Fasi: A – posizionamento del catetere. B – rigonfiamento del palloncino e deformazione dello stent. C – rimozione del catetere [http://www.informasalute.net].

Impianto di stent: in questa procedura si utilizza un dispositivo chiamato per l'appunto stent per ripristinare il flusso sanguigno nell'arteria coronaria. Uno stent è un tubicino espandibile con una struttura a rete metallica, in acciaio, lega di cobalto o lega nichel-titanio. Come nell'angioplastica, lo stent, montato su un palloncino, viene espanso all'interno dell'arteria per comprimere la placca e ripristinare il flusso sanguigno (Figura 1.4).

By-pass aorto-coronarico mininvasivo: la procedura a cuore battente descritta sopra può essere eseguita praticando una piccola incisione intercostale piuttosto che con una sternotomia mediana.

1.3. Statistiche europee riguardanti l'incidenza delle malattie vascolari sulla mortalità e sui costi per la collettività.

Le malattie cardiovascolari (cardiovascular disease: CVD) sono la principale causa di mortalità e morbidità in Europa: contano più di 4 milioni di decessi ogni anno, circa la metà di tutte le morti sono causate da CVD (52% delle morti nelle donne e il 42% delle morti negli

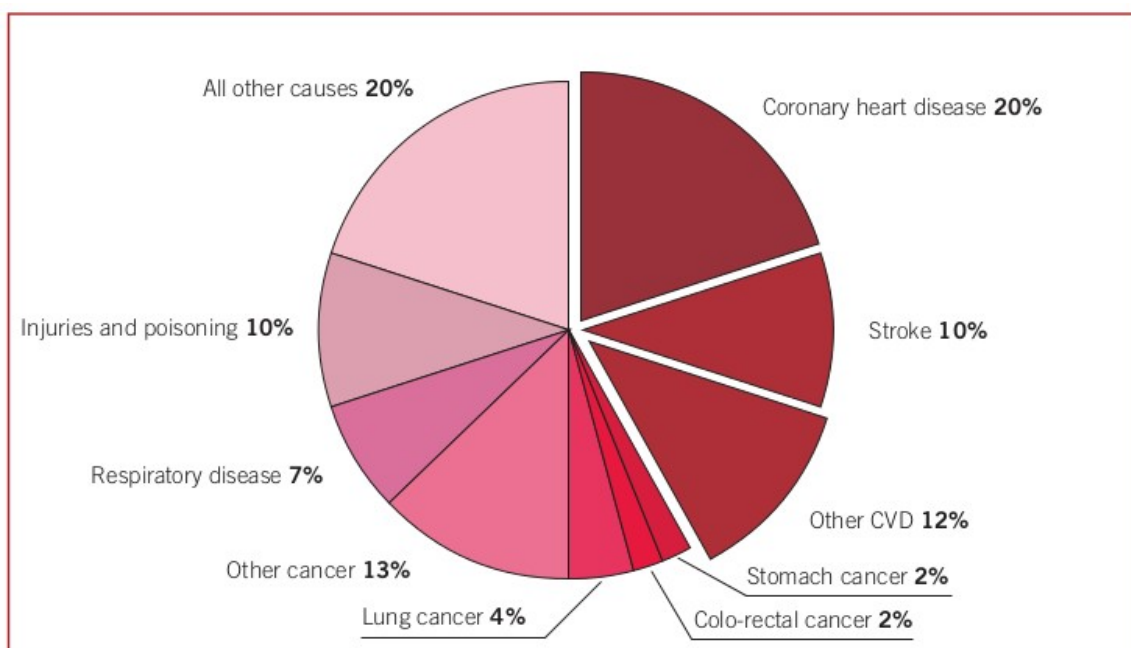


Figura 1.5: Mortalità Europea per causa (Uomini), ultimo anno disponibile [3].

uomini). Le maggiori forme di malattie cardiovascolari sono le malattie cardiache coronariche (coronary heart disease: CHD) e l'infarto del miocardio. Solo poco la metà delle morti dovute a malattie cardiovascolari sia per gli uomini che per le donne sono malattie cardiache coronariche (Figura 1.5 e Figura 1.6) [3].

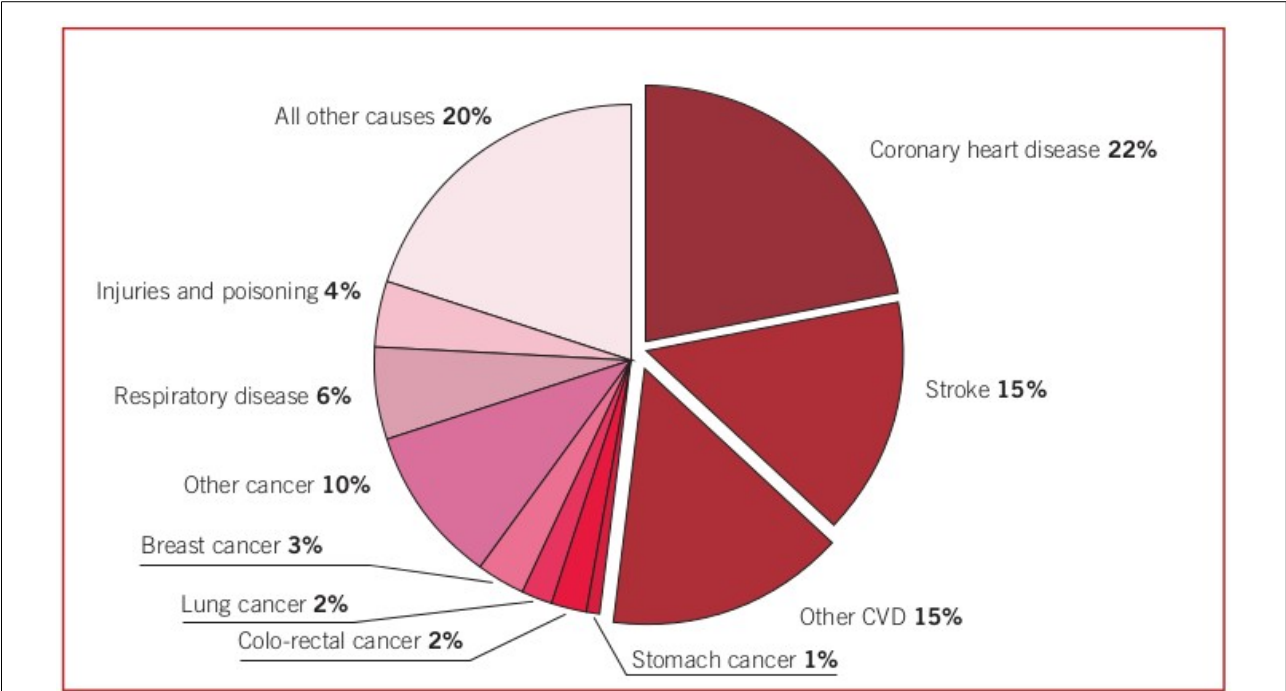


Figura 1.6: Mortalità Europea per causa (Donne), ultimo anno disponibile [3].

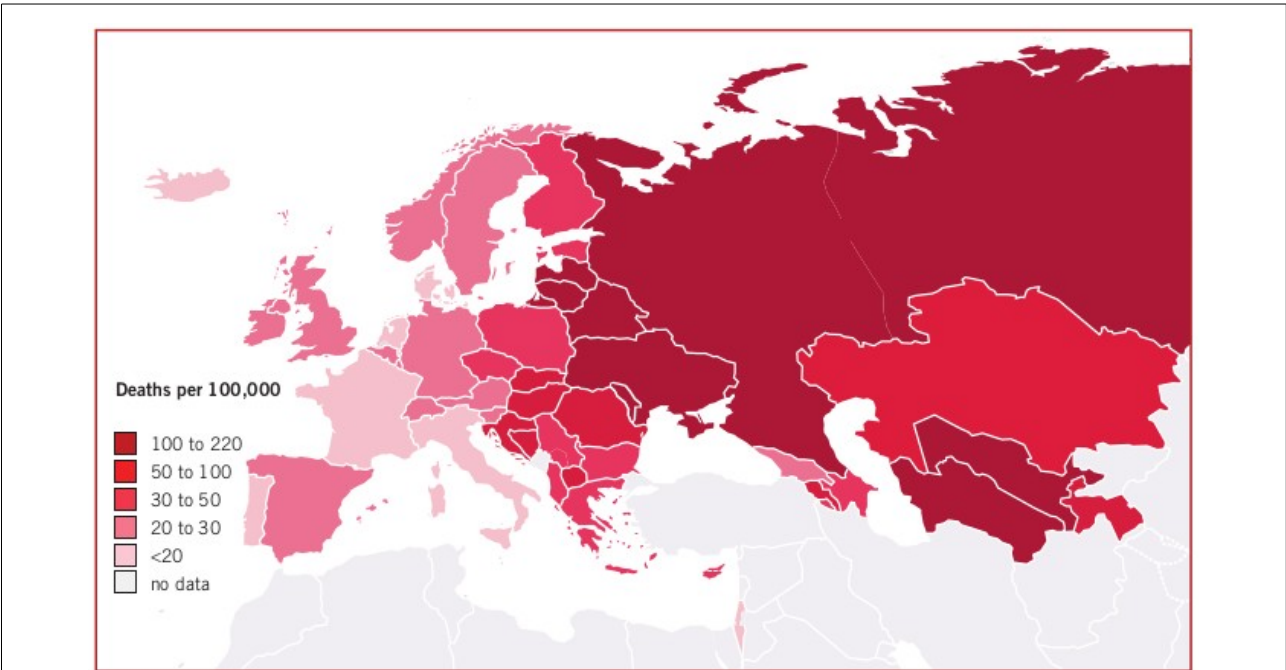
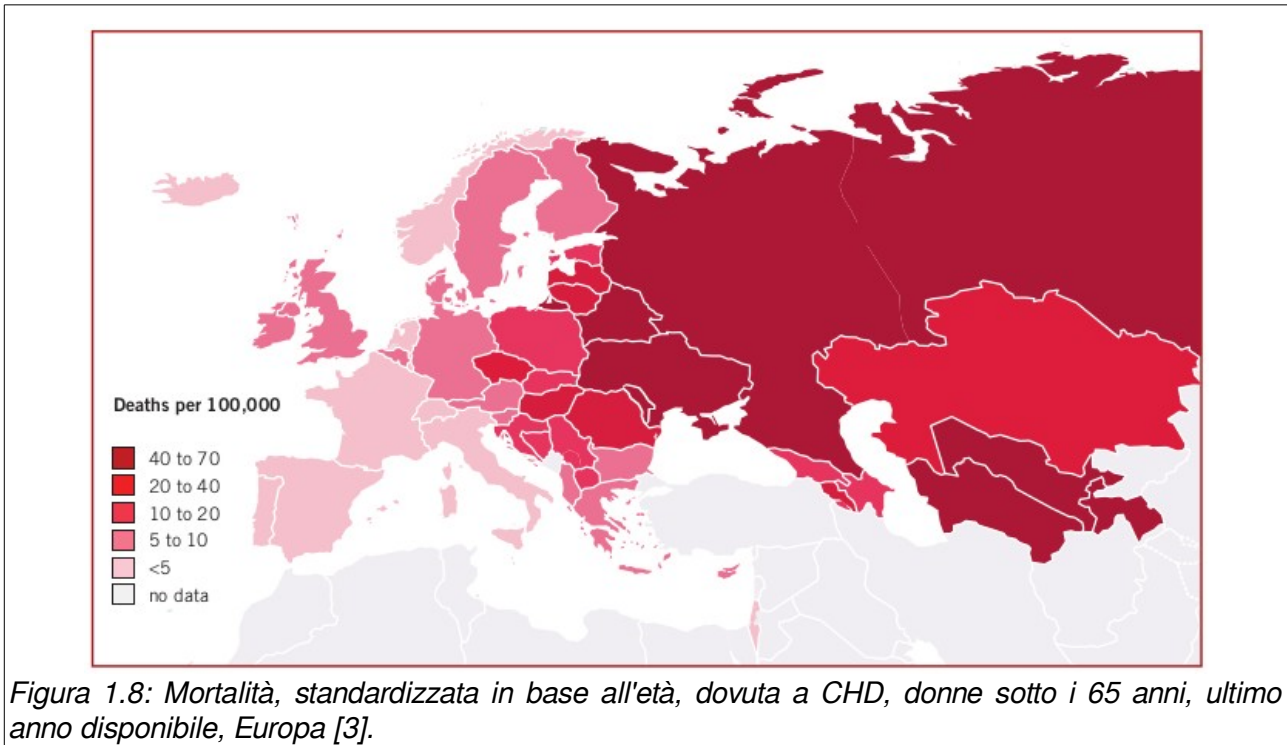


Figura 1.7: Mortalità, standardizzata in base all'età, dovuta a CHD, uomini sotto i 65 anni, ultimo anno disponibile, Europa [3].

CVD sono la maggiore causa di morte nell'Unione Europea contando 1.9 milioni di decessi ogni anno. Le malattie cardiache coronariche, CHD, da sole sono la maggiore causa di morte in Europa, contando 1.8 milioni di morti ogni anno.

Oltre che al più importante costo umano resta da considerare anche il costo economico che ogni anno incide sulle nazioni Europee, infatti, CVD sono il maggiore costo economico per l'Europa.



Viene stimato un costo per l'economia europea, dovuto alle malattie cardiovascolari, pari a 196 miliardi di Euro all'anno. Del costo totale per le CVD in Europa, circa il 54% è dovuto a costi per cure dirette, 24% per perdite produttività e il 22% per le cure informali di persone affette da CVD.

CHD è stimato costare all'economia europea circa 60 miliardi di euro all'anno: il 31% di tutto i costi per le CVD. Di tutto il costo delle CHD circa il 33% è dovuto a cure dirette , il 29% a perdite di produttività e il 38% a cure informali di persone affette da CHD.

Si stima che l'infarto costi all'economia europea più di 30 miliardi di Euro all'anno. Del costo totale per gli infarti in EU, circa il 50% è dovuto a costi per cure dirette, il 22% a perdite di produttività e il 29% a cure informali per persone affette da infarto [3].

Fattori di Rischio

I principali fattori di rischio per le malattie cardiovascolari sono: fumo, ipertensione e colesterolo, fattori direttamente correlati allo stile di vita e alle abitudini alimentari così come all'attività fisica abituale. Altri fattori che si associano alle malattie cardiovascolari sono sovrappeso e obesità, diabete mellito, abuso di alcool e stress correlato a situazioni psicosociali. Dei principali fattori che influenzano la presenza delle malattie cardiovascolari la loro situazione Europea è:

- **Fumo:** Il fumo rimane il maggiore problema pubblico per la salute in Europa anche se il numero di fumatori è diminuito in molte nazioni europee, mentre in altre è rimasto

stabile ed in alcuni casi è incrementato, soprattutto per quanto riguarda le donne. Le donne, in molte nazioni, fumano quanto gli uomini.

- **Dieta:** Il consumo di frutta e verdura si è incrementato in tutta Europa durante l'ultima decade, mentre il consumo di grassi è rimasto stabile.
- **Attività fisica:** Nelle nazioni europee, molti adulti, partecipano ad adeguati livelli di attività fisica, maggiormente per quanto riguarda le donne.
- **Obesità:** i livelli di obesità in Europa sia per i bambini che per adulti variano di molto da nazione a nazione con un incremento notevole negli ultimi anni.
- **Diabete:** la presenza di forme di diabete in Europa è elevata e sta aumentando rapidamente negli ultimi dieci anni, incrementandosi di più del 50% in molte nazioni.

Capitolo 2:

I modelli multiscala 0D/3D del sistema cardiovascolare.

In questo capitolo vengono introdotti i modelli multiscala e la loro importanza in ambito bioingegneristico. Vengono proposti alcuni esempi presenti in letteratura.

Durante gli ultimi decenni, sono state introdotte terapie e dispositivi in grado di prevenire e curare le malattie cardiovascolari. La fluidodinamica computazionale risulta essere un valido strumento di indagine sia in ambito clinico, per quanto riguarda la diagnosi medica, che industriale, nella progettazione di dispositivi medici.

I modelli fluidodinamici cardiovascolari sono indispensabili in questi ambiti. La comune pratica consiste nello studio fluidodinamico di modelli isolati ai quali vengono imposti opportuni valori di pressione e portata come condizioni al contorno. Negli ultimi anni, grazie alla crescente capacità di calcolo, sono stati sviluppati studi basati su modelli multiscala che prevedono l'accoppiamento di modelli a parametri concentrati con modelli tridimensionali.

Il sistema circolatorio è una rete chiusa in cui l'emodinamica in un singolo distretto è correlata alla dinamica globale dell'intero sistema. Il distretto di interesse viene studiato mediante un modello tridimensionale (3D) che permette uno studio locale delle caratteristiche del flusso ematico e delle sollecitazioni che esso esercita sulla parete. Il restante albero ematico viene modellato tramite modelli semplificati ottenuti integrando le equazioni di Navier-Stokes per ottenere un "equivalente elettrico" in termini di effetti resistivi, di impedenza e capacitivi.

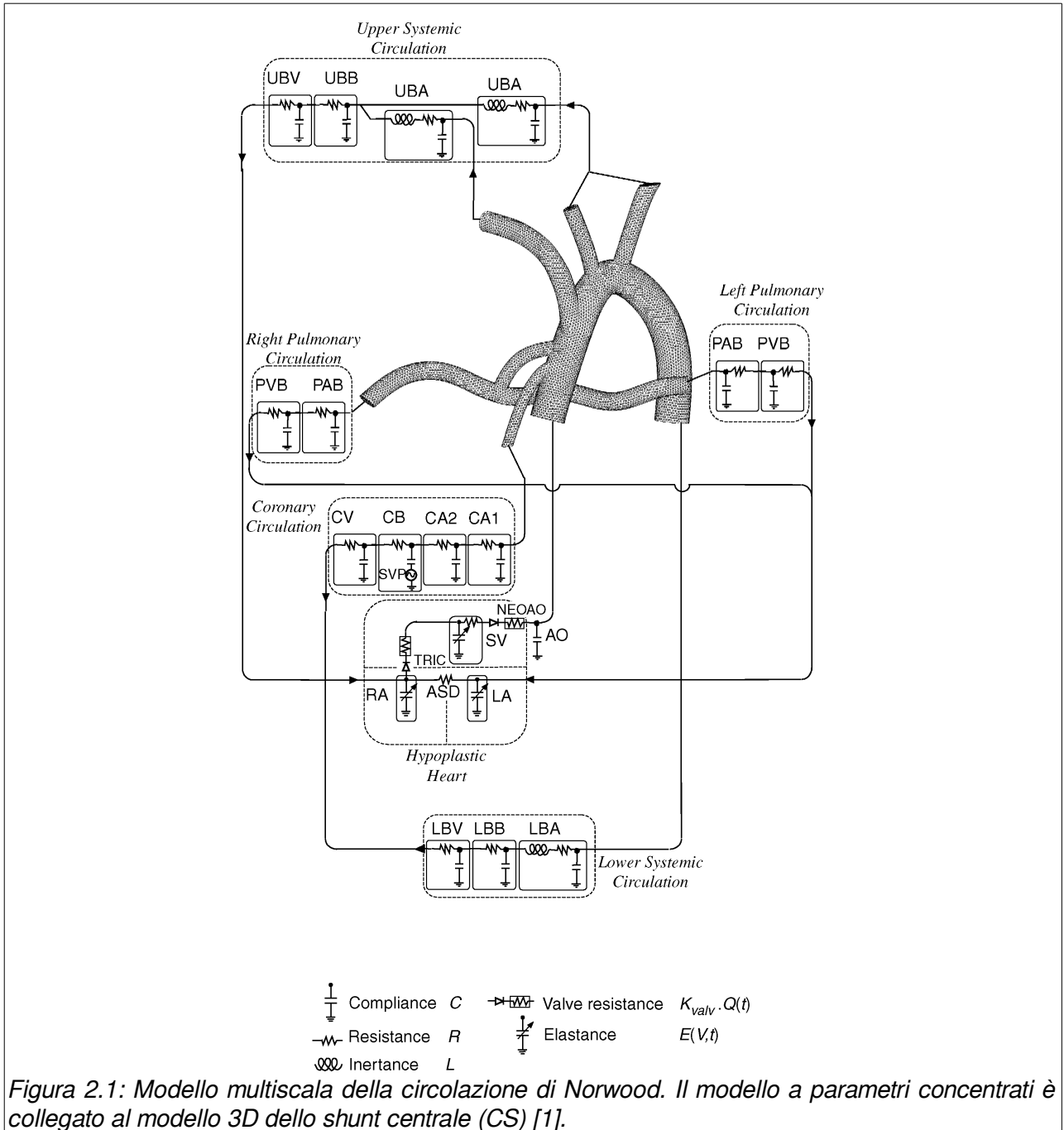
La maggior parte dei software commerciali o open source, normalmente utilizzati, prevede la descrizione matematica sia del modello a parametri concentrati che del modello tridimensionale all'interno di un unico ambiente di sviluppo.

Normalmente il modello a parametri concentrati è descritto in funzioni matematiche, costituite da equazioni differenziali ordinarie, che vengono poste come condizioni al contorno del modello tridimensionale di cui si vuole studiare la fluidodinamica. Queste funzioni vanno modellate di volta in volta ad hoc a seconda del problema fluidodinamico allo studio.

Questo approccio pone un'oggettiva difficoltà nel riutilizzo del codice sviluppato. Situazioni in cui sarebbe interessante lo studio del modello tridimensionale in un'altra posizione all'interno del sistema cardiovascolare implicano riscrittura di codice. Inoltre tale codice assume un carattere molto "informatico/procedurale" e quindi perde di leggibilità in senso ingegneristico. D'altra parte l'inserimento del codice relativo al modello 0D come funzione matematica all'interno del risolutore del modello tridimensionale ne aumenta l'efficienza perché il risolutore del modello tridimensionale non viene mai arrestato durante il calcolo della simulazione.

In letteratura esistono diversi lavori riguardanti modelli multiscala del sistema cardiovascolare. A titolo di esempio ne vengono riportati tre in cui il modello 0D è definito come funzione all'interno del risolutore del modello tridimensionale.

Nel lavoro di K. Laganà [1] è stato utilizzato l'approccio sopra descritto relativamente allo studio della perfusione polmonare e coronarica nel caso di circolazione univentricolare (Figura 2.1).



Nel lavoro di H.J. Kim [4] è presentato un modello multiscala nel quale il dominio 3D rappresenta i principali vasi collegati all'Aorta.

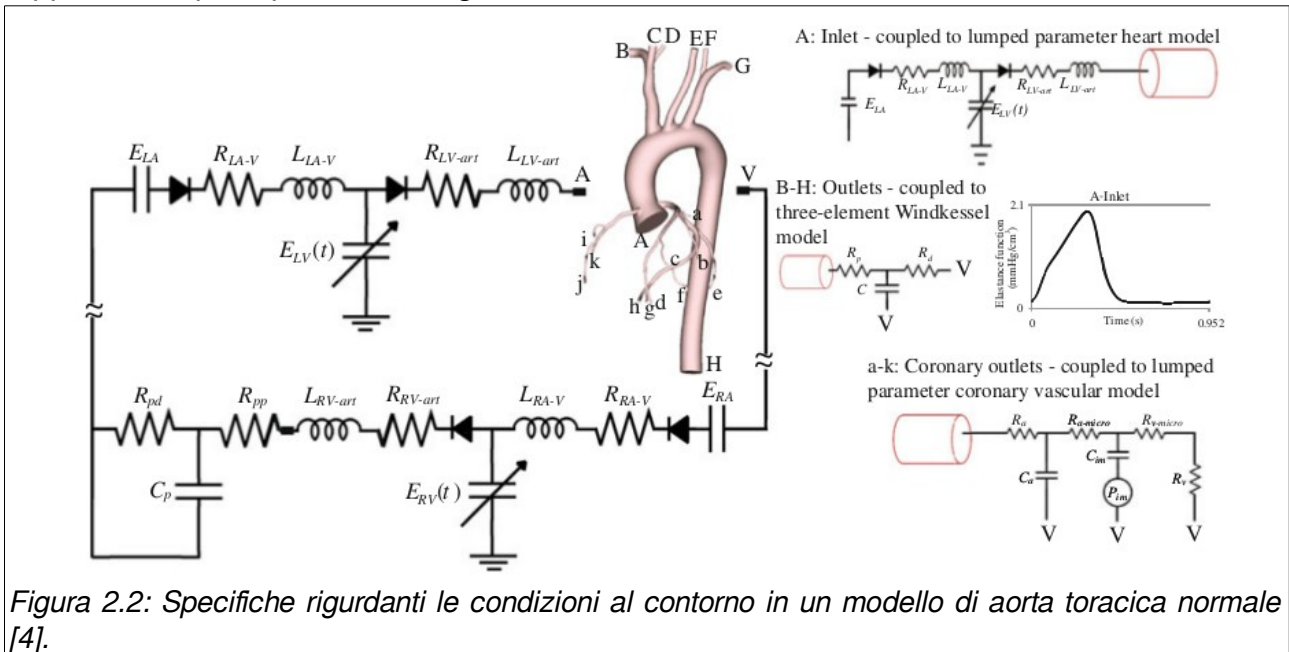


Figura 2.2: Specifiche riguardanti le condizioni al contorno in un modello di aorta toracica normale [4].

Nel lavoro di R. Ponzini [5] è proposto un modello di bypass coronarico con graft a forma di Y (coronary artery bypass graft (CABG)).

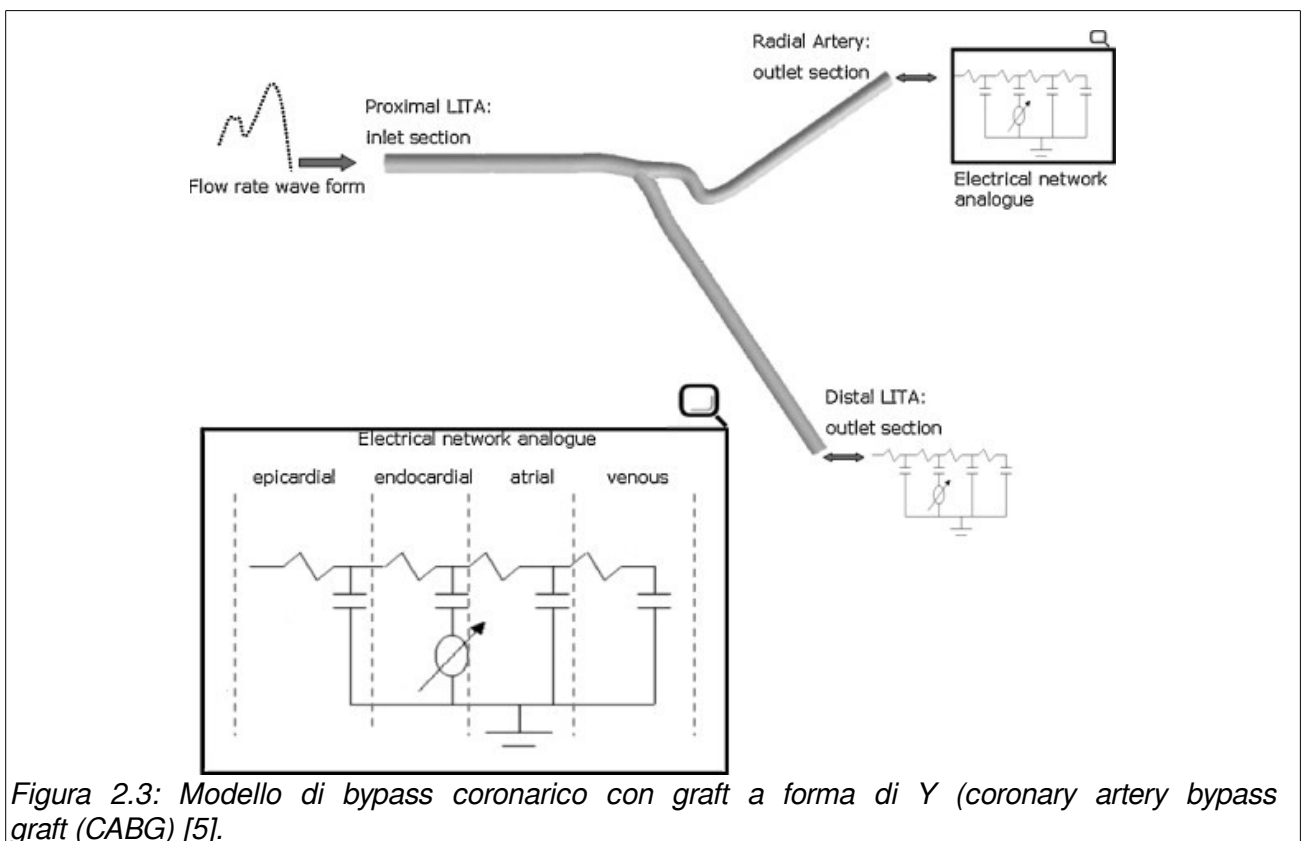


Figura 2.3: Modello di bypass coronarico con graft a forma di Y (coronary artery bypass graft (CABG)) [5].

Capitolo 3:

Il software Open Source. In particolare i software scientifici Open Source utilizzati.

In questo capitolo viene fornita una breve introduzione al software open source. Vengono poi descritti più nel dettaglio il linguaggio acausale Modelica, il tool OpenModelica e il tool OpenFOAM

3.1. Introduzione al software open source

Open source (termine inglese che significa codice a sorgente aperto), in informatica, indica un software i cui autori (più precisamente i detentori dei diritti) ne permettono e favoriscono il libero studio e l'apporto di modifiche da parte di altri programmatori indipendenti. Questo è realizzato mediante l'applicazione di apposite licenze d'uso.

Il fenomeno ha tratto grande beneficio da Internet, perché esso permette a programmatori geograficamente distanti di coordinarsi e lavorare allo stesso progetto.

L'utilizzo del software open source offre diversi vantaggi ma anche svantaggi. Tra i vantaggi sicuramente si può ricordare che, essendo il codice sorgente consultabile e modificabile è possibile adattarlo alle proprie esigenze. Spesso gli utilizzatori di un certo pacchetto software formano una comunità di sviluppo e di assistenza tecnica.

Il software open source, soprattutto se utilizzato in ambito scientifico, obbliga a conoscere il funzionamento interno degli algoritmi che compongono il software stesso e quindi non tutti sono in grado di apportarvi modifiche, inoltre l'usabilità spesso non è ottimale per un utente inesperto, infatti, anche se spesso è disponibile molta documentazione, è sovente incompleta o non aggiornata.

La complessità e le dimensioni di alcuni software liberi (ad esempio, i sistemi operativi) è tale che è necessario il supporto commerciale di un'azienda; il software libero si presta a creare nuove opportunità di business nel campo della formazione e del supporto, oltre che della eventuale personalizzazione del software [6].

L'open source quindi, non va considerato semplicemente come una scappatoia per aggirare il problema dell'acquisto o dell'utilizzo di un software proprietario, ma diviene una vera e propria scelta di campo in favore della libertà di circolazione e dello scambio delle idee, in modo da consentire una crescita che coinvolga tutta l'utenza e non sia legata alla disponibilità economica. Lo scambio di idee consente, tra l'altro, di apporre significativi miglioramenti al programma; un open source messo "sul mercato", infatti, è già in parte testato e vagliato da numerosi programmatori che ne possono scoprire i bug ed eliminarli.

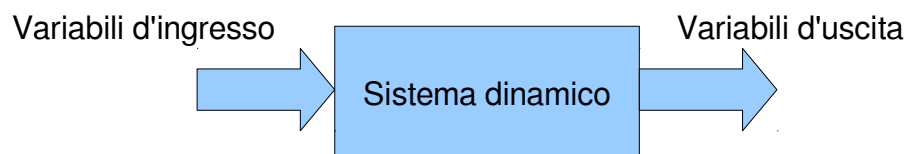
La libertà di circolazione e dello scambio delle idee ha determinato la scelta di utilizzare il software open source, per quanto possibile, nello sviluppo di questo lavoro di tesi.

3.2. Strumenti di modellazione : Modelica e OpenModelica

a) Approccio causale rispetto all'approccio acausale

Nello studio di sistemi dinamici lineari costituiti da più sottosistemi collegati fra loro è possibile realizzare modelli ingresso-uscita della porzione di sistema di interesse utilizzando

l'approccio causale: il problema viene formulato in base ai nessi di causalità presenti nel sistema, affinché le equazioni che esprimono le grandezze di interesse (variabili di uscita), siano espresse in funzione di quantità note (variabili di ingresso), attraverso una serie definita e rigidamente codificata di operazioni, impiegate per risolvere quel particolare problema. Le variabili di ingresso rappresentano le azioni che vengono compiute sull'oggetto in esame da agenti esterni che ne influenzano il comportamento, mentre le variabili di uscita rappresentano quanto del comportamento del sistema è di interesse: in altri termini, tra le variabili di ingresso e quelle di uscita vi è un rapporto di causa ed effetto, dato che l'evoluzione delle seconde descrive il modo in cui il sistema risponde alle sollecitazioni impresse dalle prime [7].



Disegno 1: Sistema dinamico

In generale, però, la conoscenza del valore in un certo istante delle variabili di ingresso non è sufficiente ad individuare il valore nello stesso istante delle variabili di uscita: si comprende che, per descrivere compiutamente l'oggetto da modellare, è necessario introdurre un terzo vettore di variabili, chiamate variabili di stato, che ne descrivano la situazione interna. La formulazione causale del problema sfrutta una rappresentazione grafica basata sui cosiddetti schemi a blocchi, che consentono di mettere in luce con chiarezza le interazioni tra i diversi sottosistemi: le componenti usate per descrivere il sistema fisico, chiamate appunto blocchi, utilizzano quantità note al fine di definirne altre incognite e possono essere espresse nella seguente forma generale:

$$\dot{x}(t) = f(x(t), y(t), u(t), t) \quad (3.1)$$

$$y(t) = g(x(t), y(t), u(t), t) \quad (3.2)$$

dove $u \in \mathbb{R}^m$, $x \in \mathbb{R}^n$ e $y \in \mathbb{R}^p$ rappresentano, rispettivamente, i vettori delle variabili di ingresso, stato e uscita del sistema, mentre f e g sono due funzioni vettoriali. L'equazione (3.1) costituisce l'equazione di stato: si tratta di un'equazione differenziale di tipo ODE che mette in relazione con l'ingresso le variabili che descrivono la situazione interna del sistema; invece, l'equazione (3.2) va sotto il nome di trasformazione d'uscita ed è un'equazione algebrica che consente di determinare l'uscita in uno specifico istante di tempo

in base alla conoscenza dello stato e degli ingressi agenti sul sistema. In particolare, nella modellistica si studiano quasi sempre sistemi invarianti nel tempo, caratterizzati dal fatto che le funzioni f e g non dipendono esplicitamente dal tempo. In questo modo si ottiene

$$\begin{cases} \dot{x}(t) = f(x(t), y(t), u(t)) \\ y(t) = g(x(t), y(t), u(t)) \end{cases} \quad (3.3)$$

Quando vengono connessi fra loro, i blocchi così definiti, sono in grado di descrivere e simulare grandi sistemi di equazioni differenziali. La formulazione del problema della modellazione secondo tale approccio costituisce uno strumento molto efficace per rappresentare il comportamento matematico dei sistemi dinamici: nella rappresentazione sul diagramma di stato, ad esempio, gli jacobiani delle funzioni f e g che legano lo stato agli ingressi e alle uscite possono essere usati per determinare poli, zeri e funzioni di trasferimento del sistema linearizzato. Inoltre, i diagrammi a blocchi sono ideali per descrivere il sistema di controllo, in quanto permettono di focalizzare immediatamente le grandezze sulle quali è possibile esercitare un'azione per gestire l'evoluzione del sistema e in che modo queste ne influenzeranno le uscite. Infine la necessità di dover lavorare direttamente sulle equazioni, al fine di codificarle ed implementarle nell'ambiente di simulazione, porta ad una maggiore comprensione della dinamica del sistema e dei processi caratteristici del sistema. Tuttavia l'aver imposto a priori, prima ancora di realizzare il modello, quali grandezze saranno note e quali incognite non permette di operare modifiche, poiché i nuovi contesti in cui tale modello sarà adottato porteranno invariabilmente a violare tali assunzioni. Riassumiamo quindi i punti deboli di tale formulazione:

- Le equazioni usate devono essere ricavate manualmente a partire dalle equazioni costitutive e dai principi di conservazione. Dovendo trattare come minimo sistemi di equazioni differenziali del primo ordine, questo lavoro può essere molto complesso e può generare errori nel modello finale.
- Le assunzioni di causalità devono essere fatte a livello delle componenti piuttosto che a livello del sistema: questo inquadra molto rigidamente il problema e il flusso di informazioni, circoscrivendo la sua applicabilità.

L'impiego dei diagrammi a blocchi nella modellazione è dovuto al fatto che si riescono ad ottenere dei modelli facilmente simulabili con un calcolatore, dato che la formulazione del problema in una serie di operazioni matematiche viene svolta dall'utente a priori e quindi, per implementare il modello, è sufficiente codificare una serie di operazioni, sebbene per risolvere sistemi di equazioni differenziali si debba ricorrere a metodi d'integrazione per le equazioni ODE che possono risultare anche abbastanza complessi. L'aspetto più

penalizzante di questa formulazione consiste nell'impossibilità di effettuare modifiche sul modello realizzato: se ad esempio fosse necessario dover monitorare l'andamento temporale di un'ulteriore grandezza del sistema non prevista nel modello concepito inizialmente, sarebbe necessario ridefinire il vettore delle grandezze di interesse in uscita e riformulando il problema; risulterebbe vantaggioso invece avere a disposizione durante la simulazione tutte le variabili e le grandezze caratteristiche del sistema, anche e soprattutto quelle che non sono osservabili e accessibili nel sistema reale.

Negli ultimi anni, grazie all'aumento delle capacità di calcolo, si è diffusa un'altra tipologia di modellazione, basata su un approccio a-causale. Si tratta di un modo strutturato di fare modellazione, basato sulle equazioni costitutive che descrivono i processi coinvolti che unitamente alle leggi di conservazione determinano il sistema di equazioni che deve essere risolto. Realizzare modelli dinamici secondo questa formulazione del problema non richiede di specificare a priori quali grandezze agiranno come ingressi e quali come uscite: le equazioni per ciascun modello vengono scritte in forma dichiarativa indipendentemente dalle condizioni al contorno. I nessi di causalità rimangono non specificati e vengono fissati solo quando le equazioni vengono risolte, in base alle condizioni al contorno imposte dal contesto applicativo. Per questo motivo, si parla di modellazione a-causale: il termine modellazione fisica viene impiegato in quanto questo tipo di approccio è adatto per rappresentare la struttura fisica del sistema descritto. Il vantaggio principale della modellazione a-causale è che la direzione secondo la quale le equazioni vengono risolte si può facilmente adattare di volta in volta in base al flusso di dati, caratteristico del contesto applicativo cui si fa riferimento. In questo modo, l'attenzione rimane focalizzata sui componenti fisici del sistema, il cui comportamento è descritto da sistemi DAE (Differential Algebraic Equations). Per realizzare dei modelli a-causali è necessario selezionare i componenti e collegarli in uno schema: i modelli di ciascun componente sono formulati indipendentemente dalla conoscenza dei legami o delle equazioni che regolano altre parti del sistema. Partendo dal modello generale, che inquadri solo gli elementi fondamentali, è possibile arricchirlo (aggiungendo sensori, attuatori, ecc.) fino a renderlo completo. Con questo approccio, si riducono notevolmente le tempistiche e le difficoltà di modellazione, producendo modelli più comprensibili e vicini al mondo fisico. Per contro, è necessario che il simulatore sia in grado di effettuare un'analisi simbolica su sistemi DAE di grandi dimensioni [7].

Tabella 1: Riassunto delle principali caratteristiche dei due principali approcci alla modellistica dei sistemi dinamici.

Approccio Causale	Approccio a-causale
È necessario indicare sin dall'inizio quali grandezze agiranno come ingressi e quali come uscite del sistema.	Non è necessario specificare a priori ingressi e uscite del sistema, né fissare le condizioni al contorno.
Bisogna riscrivere le equazioni, nella rappresentazione del diagramma di stato, in funzione del contesto applicativo.	I nessi di causalità caratteristici del contesto rimangono non specificati: emergono solo quando le equazioni devono essere risolte.
La formulazione del problema in una serie di operazioni da codificare è svolta dall'utente.	I modelli dei componenti sono realizzati indipendentemente dalla conoscenza dei legami che questi hanno fra loro.
Grandi sforzi per modificare anche leggermente la configurazione del modello.	Possibilità di realizzare modelli meno complessi e più vicini alla realtà fisica descritta.
Difficoltà nel riutilizzare il lavoro già svolto sul modello.	Possibilità di riutilizzare componenti e sottosistemi sviluppati in altri modelli.

b) La Modellazione Orientata agli Oggetti e Modelica

La modellazione ad oggetti (OOM, Object Oriented Modelling) è una tecnica largamente accettata che ha già prodotto sia linguaggi di modellazione che pacchetti software.

L'approccio ad oggetti è basato su alcuni paradigmi, fra i fondamentali troviamo:

- La definizione di porte fisiche (chiamate connettori) come interfacce standard dotate di ingressi e uscite, contenenti le equazioni che definiscono il rapporto fra le grandezze delle entità che vanno ad unire.
- L'indipendenza reciproca fra l'interfaccia del modello e la descrizione interna.
- La definizione dei modelli in forma a-causale: in questo modo, modelli già realizzati possono essere agevolmente modificati e riutilizzati.

Lo stato d'arte della OOM è ben rappresentato dallo sviluppo del progetto Modelica [<http://www.modelica.org/>, Modelica and Modelica Association].

Nel settembre del 1996 un gruppo di scienziati ed esperti programmatori di software unì le forze per lavorare allo sviluppo della programmazione a oggetti. L'obiettivo iniziale fu quello di scrivere una sorta di carta sulla programmazione object-oriented, che prevedesse una

possibile unificazione dei concetti finalizzata a creare un linguaggio di programmazione comune. Il linguaggio di programmazione alla base del tool di modellazione Dymola (software attualmente di proprietà della società francese Dassault Systèmes) fu scorporato e reso open source. Nascevano così OpenModelica [www.openmodelica.org] e l'organizzazione no profit Modelica Association. Nel 1997 fu elaborata la libreria standard di Modelica (presente oggi in tutti i software che sfruttano un linguaggio adatto alla modellazione di generici sistemi fisici, descritti da equazioni algebriche e differenziali).

Le caratteristiche principali di questo linguaggio sono:

- *Modellazione dichiarativa a-causale*: il modello di ogni singolo componente fisico è descritto da un insieme di equazioni algebriche differenziali che rappresentano il comportamento dell'oggetto, la cui forma non dipende dal metodo numerico con il quale verranno risolte. Non è necessario definire dall'inizio le grandezze che agiranno come ingressi e porre a priori le condizioni al contorno: questo è essenziale per raggiungere un'effettiva modellazione orientata agli oggetti fisici. La causalità del modello è determinata automaticamente dal compilatore ad un livello aggregato, dove il modello del sistema viene assemblato partendo da modelli di componenti elementari.
- *Trasparenza del codice*: l'approccio dichiarativo permette di scrivere il codice in modo facilmente leggibile, in un modo molto simile a come le equazioni vengono scritte sulla carta. Nel sistema virtuale è immediatamente riconoscibile il sistema esistente, la topologia viene mantenuta e questo ne agevola la riferibilità. Le icone realizzate rappresentano un componente del sistema e le linee che collegano le varie parti indicano le connessioni fisiche esistenti (una linea elettrica, una tubatura, un flusso di potenza termica).
- *Incapsulamento*: i modelli dei componenti del sistema sono legati attraverso connettori, che rappresentano delle interfacce rigorosamente definite dotate di ingressi e uscite contenenti le equazioni che definiscono il rapporto fra le grandezze delle entità che vanno ad unire. Qualunque coppia di oggetti compatibili può essere unita per mezzo di connettori, a prescindere dai loro dettagli interni. Questa caratteristica è fondamentale per lo sviluppo di librerie di modelli riutilizzabili: in più, permette di sostituire agevolmente una parete del modello del sistema con una più dettagliata o più semplificata, senza condizionare il resto del modello.
- *Modellazione multi-fisica*: il linguaggio Modelica non è finalizzato in particolare ad alcuna branca dell'ingegneria, come la meccanica, termodinamica o ingegneria

elettrica. Permette la modellazione di sistemi di varia natura, purché descritti da equazioni DAE: in questo modo, relazioni provenienti da contesti molto dissimili fra loro possono essere trattate nello stesso modo e risulta immediato combinare fra loro modelli che provengono da campi diversi dell'ingegneria, con sistemi di controllo continui o a tempo discreto. Nell'impiego di questo linguaggio nella descrizione di un reattore nucleare di potenza, vengono trattati fenomeni eterogenei, quali lo scambio termico, la fluidodinamica, la neutronica con il medesimo formalismo.

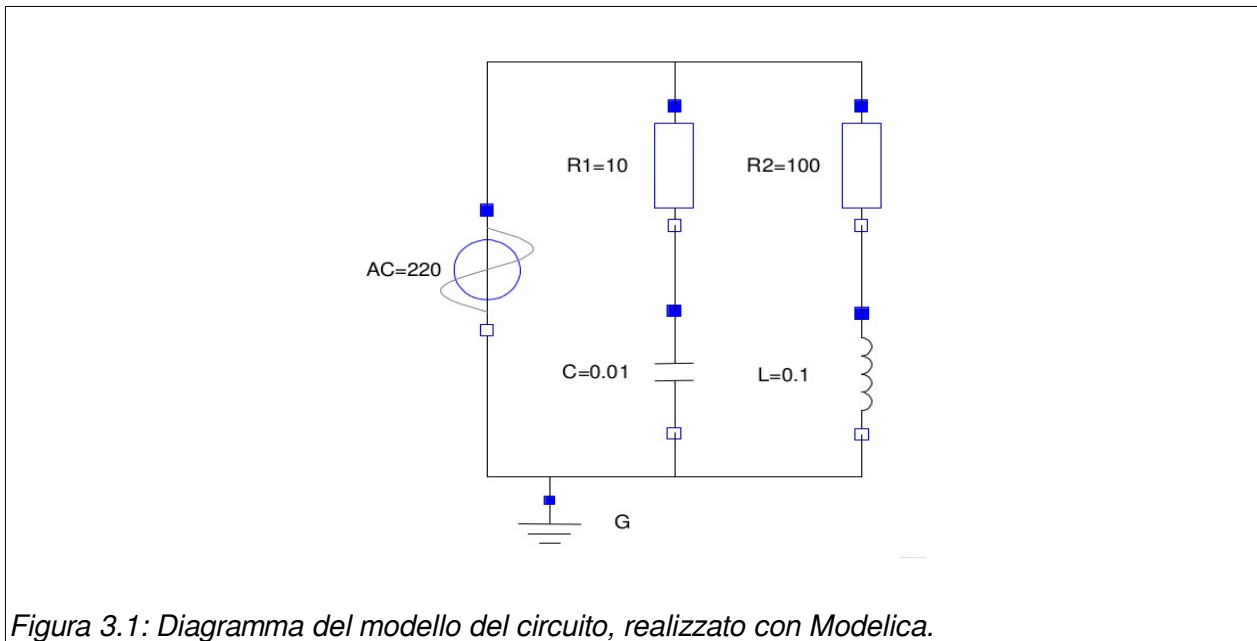
- *Ereditarietà*: i componenti possono essere definiti in una struttura gerarchica, dove le parti più complesse sono ottenute dai modelli base aggiungendo variabili specifiche, equazioni o altri modelli.
- *Riutilizzabilità*: la modellazione a-causale, l'incapsulamento e l'ereditarietà favoriscono e incentivano la riutilizzabilità di modelli e sottosistemi esistenti in altri progetti. A livello di componente, è spesso possibile riutilizzare modelli disponibili in librerie standard, introducendo modifiche dove necessario. [7]

c) **Modellazione fisica vs. Modellazione causale**

Al fine di evidenziare le differenze che la separano l'approccio tradizionale da quello a-causale, si riporta l'esempio di un semplice circuito elettrico. Per questo sistema elementare è stato realizzato con Modelica un modello, `SimpleCircuit`, il cui diagramma è rappresentato in Figura 3.1. Come si può vedere, ad ogni componente che compare nel sistema (resistori, capacitori, induttori, un generatore di tensione e una messa a terra) viene fatta corrispondere un'icona, un oggetto grafico identificativo che lo rappresenta. Questo aspetto è caratteristico del Component Based Approach, cioè l'approccio più congeniale che permette di sfruttare i principali vantaggi di questo linguaggio: prima viene creato, per ciascun elemento presente nel sistema, un modello che lo caratterizza e, una volta realizzati, questi modelli vengono collegati fra loro ricreando la configurazione esistente. Si noti, infatti, come le connessioni logiche corrispondano sostanzialmente alle connessioni fisiche presenti nel circuito esistente: proprio per questa ragione, ci si riferisce alla modellazione orientata agli oggetti con il termine modellazione fisica.

Il modello può essere considerato a-causale, dato che non c'è alcun flusso predefinito di informazione, nessun nesso causale riscontrabile a priori.

Connessioni tra gli oggetti sono specificate utilizzando la dichiarazione di connessione che è una speciale forma sintattica di equazione.



model SimpleCircuit

```

Resistor R1(R=10);
Capacitor C(C=0.01);
Resistor R2(R=100);
Inductor L(L=0.1);
VsourceAC AC;
Ground G;

```

equation

```

connect (AC.p, R1.p); // Capacitor circuit
connect (R1.n, C.p);
connect (C.n, AC.n);
connect (R1.p, R2.p); // Inductor circuit
connect (R2.n, L.p);
connect (L.n, C.n);
connect (AC.n, G.p); // Ground

```

end SimpleCircuit;

A confronto viene mostrato in Figura 3.2, lo stesso circuito in un altro modello, realizzato però secondo l'approccio causale, facendo uso di un diagramma a blocchi. Si nota subito che, in questa rappresentazione, la topologia fisica non è più preservata: fra la struttura del diagramma del modello e il circuito esistente non è più possibile intravedere un'immediata corrispondenza.

Inoltre, il modello è causale, in quanto si percepisce subito la direzione assunta dal segnale: il generatore di tensione svolge la funzione di ingresso e la corrente circolante ricopre il ruolo di grandezza di interesse. Tuttavia, anche per questo semplice circuito, l'analisi che deve essere svolta per convertire il modello fisico in un modello a blocchi non è immediata. Un

altro svantaggio è che le rappresentazioni dei componenti non sono indipendenti dal contesto. Ad esempio, i resistori R1 e R2 vengono definiti in modi differenti, il che rende il loro riutilizzo all'interno di una libreria estremamente arduo. Infine, tali modelli del sistema sono di solito difficili da conservare, dato che anche lievi cambiamenti nel sistema reale possono comportare stravolgimenti nel diagramma a blocchi corrispondente [8].

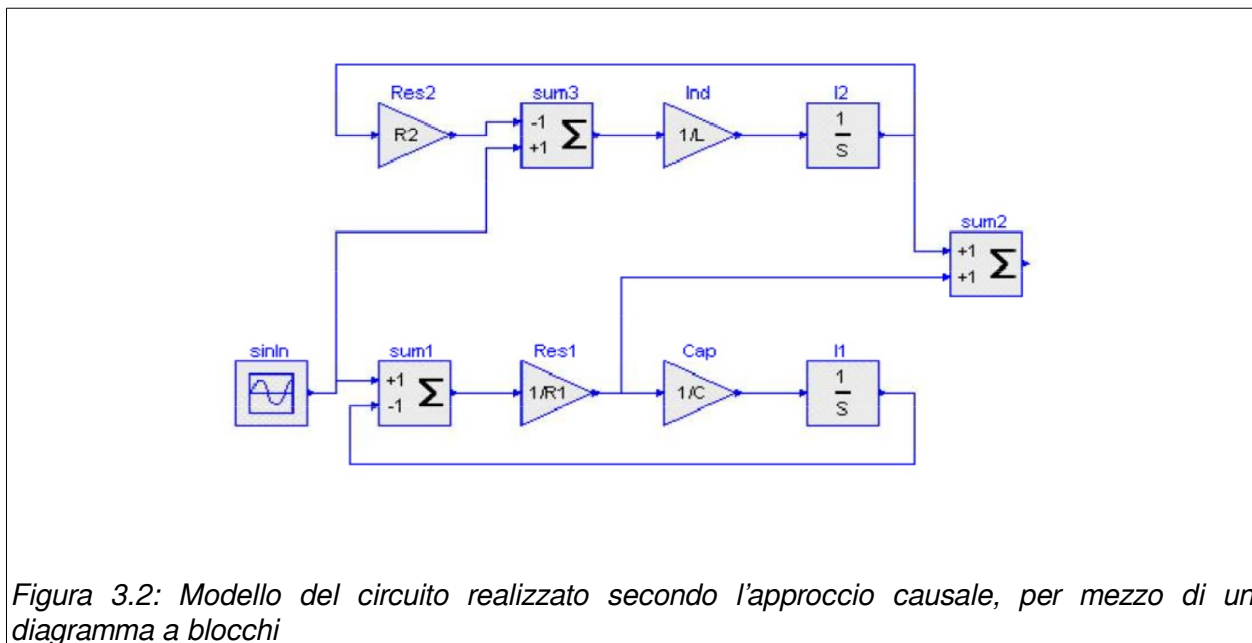


Figura 3.2: Modello del circuito realizzato secondo l'approccio causale, per mezzo di un diagramma a blocchi

3.3. Strumento di simulazione CFD: OpenFOAM

OpenFOAM (Open Field Operation and Manipulation) è uno strumento CFD (Computational Fluid Dynamics, fluidodinamica computazionale) open source, distribuito sotto licenza GNU General Public Licence [www.openfoam.org] da parte di OpenFOAM Foundation. OpenFOAM ha un vasto utilizzo in molti campi della scienza e dell'ingegneria ed è utilizzato sia da organizzazioni accademiche che commerciali.

Il suo scopo è quello di risolvere equazioni differenziali alle derivate parziali attraverso l'approccio a volumi finiti; è libreria software con potenzialità molto più ampie rispetto alla soluzione di problemi fluidodinamici, che restano comunque l'obiettivo principale della stragrande maggioranza dei solutori già pronti.

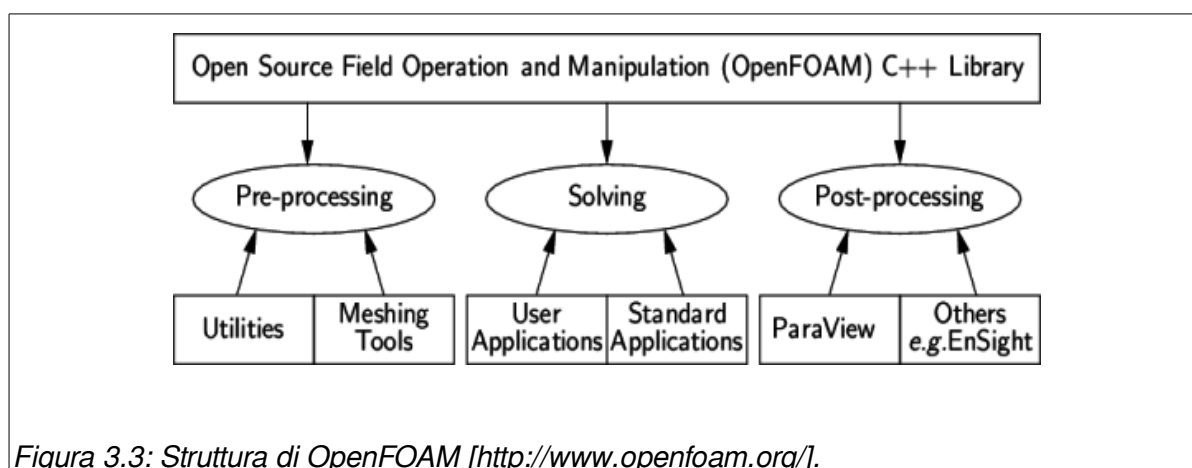
OpenFOAM è innanzitutto una libreria scritta in C++ che crea eseguibili chiamati *applications*.

Le *applications* si dividono in due categorie: *solvers*, ciascuno progettato per risolvere uno specifico problema nella meccanica del continuo, e *utilities* che sono progettate per eseguire compiti che comportano la manipolazione dei dati.

Uno dei punti di forza di OpenFOAM consiste nel fatto che gli utenti possono creare nuovi *solvers* e *utilities* ovviamente con i prerequisiti di conoscere la fisica del problema, la struttura e le regole di OpenFOAM e le tecniche di programmazione.

OpenFOAM è fornito completo di ambienti per la il pre- e il post-processing. Le interfacce per il pre- e il post-processing sono loro stesse *utilities*, il che assicura una gestione consistente dei dati attraverso tutto l'ambiente.

La struttura di OpenFOAM può essere riassunta attraverso lo schema mostrato in Figura 3.3.



OpenFOAM ha un esteso range di caratteristiche in grado di risolvere ogni problema riguardante flussi complessi di fluidi che includano anche reazioni chimiche, turbolenza e trasferimento di calore. Inoltre è in grado di risolvere problemi di dinamica dei solidi e elettromagnetismo.

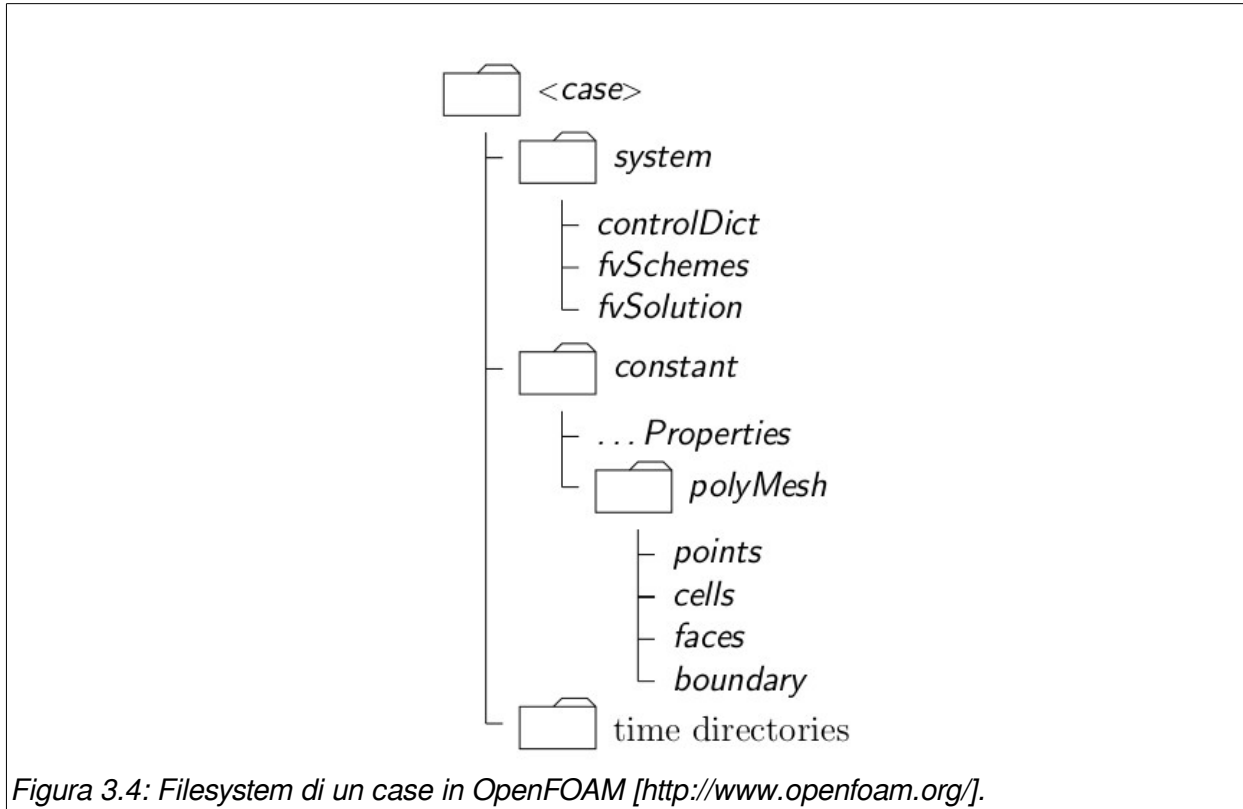
OpenFOAM è totalmente scritto in C++ e questo consente una programmazione definita object-oriented, che consente di dichiarare tutte le grandezze fisiche e le relative operazioni in un unico oggetto, chiamato classe, in modo tale che la scrittura delle equazioni da risolvere sia analoga al linguaggio verbale e matematico usato per descriverle. Per fare un esempio più concreto si pensi al campo di velocità, indicato con U .

La velocità è un campo vettoriale, definito da una classe *vectorField*, mentre il modulo della velocità è $\text{mag}(U)$.

Il fatto di avere oggetti nella programmazione che rappresentano oggetti fisici o entità astratte rende il codice molto più compatto e comprensibile, concentrando la dichiarazione delle classi in una parte limitata del codice ed evitando inutili ripetizioni. Inoltre risulta piuttosto facile modificare o creare nuove classi da quelle già esistenti.

Struttura di un case in OpenFOAM

In OpenFOAM ogni sistema fisico di cui eseguire la simulazione prende il nome di *case*. Tutti gli elementi necessari alla simulazione di un case sono residenti all'interno della stessa directory in una struttura ben precisa, come quella mostrata in Figura 3.4



All'interno della cartella *system* trovano spazio i dizionari che definiscono il problema da risolvere e i metodi numerici. I dizionari fondamentali sono il *controlDict*, *fvSchemes* e *fvSolution*. Spesso ne sono presenti molti altri, per il pre-processing o per altre operazioni intermedie sulla mesh di volume.

Il dizionario *controlDict* comprende la definizione dei parametri che andranno a controllare la simulazione come ad esempio il risolutore ed i parametri relativi alla parte temporale della simulazione (tempo iniziale, finale e l'incremento temporale). Seguono quindi le scelte dei parametri per il salvataggio delle soluzioni intermedie ed i dettagli di come devono essere salvati i risultati (formato binary o ascii, precisione e compressione).

Il dizionario *fvSchemes* comprende gli schemi per tutti gli operatori matematici delle equazioni risolte dal caso in esame. Si hanno perciò gli schemi dell'integrazione temporale, il calcolo dei gradienti e della divergenza, gli schemi per il laplaciano, quelli per l'interpolazione, lo schema per calcolare la componente del gradiente normale alla faccia di una cella e lo schema per calcolare i flussi. Per ciascuno di essi si può inoltre specificare un diverso schema a seconda della grandezza fisica a cui viene applicato l'operatore.

Il dizionario *fvSolution* serve a definire il tipo di solutore da usare nelle equazioni. In questo lavoro verrà utilizzato il risolutore *simpleFoam* (definito nel dizionario *controlDict*), relativo ad un fluido incomprimibile e newtoniano in un flusso laminare. Nel caso *simpleFoam*, si devono risolvere le equazioni nelle incognite velocità e pressione. Il solutore per la pressione è GAMG (Generalised geometric-algebraic multi-grid), mentre per la velocità è chiamato *smoothSolver*. Il sub-dizionario SIMPLE (Semi-Implicit Method for Pressure Linked Equations) è relativo al fatto che le equazioni vengono risolte con tecniche iterative e bisogna specificare il numero dei correttori da utilizzare.

La cartella *constant* contiene una sottocartella chiamata *polyMesh* e alcuni dizionari del tipo *Properties*. Nella cartella *polyMesh* sono contenuti i file di descrizione della mesh di volume (strutturata o meno), generati dalle utilities di pre-processing. Questi file contengono l'elenco di tutti i punti, le celle, le facce e i contorni del dominio di calcolo.

I dizionari, nel caso più generale di flusso turbolento, sono invece *transportProperties*, *turbulenceProperties* e *RASProperties*. In questo lavoro il flusso approssimato è ritenuto laminare e pertanto l'unico parametro definito è la viscosità cinematica:

```
nu [ 0 2 -1 0 0 0 0 ] 3.3e-06;
```

All'interno del file *RASProperties* è definito il tipo di flusso e quindi, in questo caso, laminare.

Le parentesi quadre servono a stabilire l'unità di misura della grandezza in questione: la prima cifra indica l'esponente della massa, la seconda indica l'esponente della lunghezza, la terza l'esponente del tempo e così via per le 7 unità di misura fondamentali del sistema internazionale.

I risultati della simulazione sono contenuti in una sottocartella del case, ciascuna nominata con il valore del time step corrispondente. Ogni cartella conterrà al proprio interno un numero di file corrispondente al numero delle variabili calcolate all'interno del dominio di interesse.

Ciascun file contiene i valori della variabile associata per ogni singola cella del dominio considerato a quel time step.

In particolare, per poter iniziare con la simulazione vanno definite le condizioni iniziali. Queste ultime sono definite all'interno della cartella "0".

Capitolo 4:

Modello matematico 0D

In questo capitolo vengono introdotti i modelli matematici che descrivono il comportamento fluidodinamico di un vaso sanguigno; è utilizzata l'analogia idraulica/elettrica per descrivere il sistema cardiocircolatorio in un modello a parametri concentrati.

E' importante definire innanzitutto un modello matematico per i vasi sanguigni. L'obiettivo consiste nel ricavare un modello elettrico equivalente a parametri concentrati dei un generico vaso in modo da poterlo utilizzare all'interno del modello di ogni singolo distretto del sistema circolatorio.

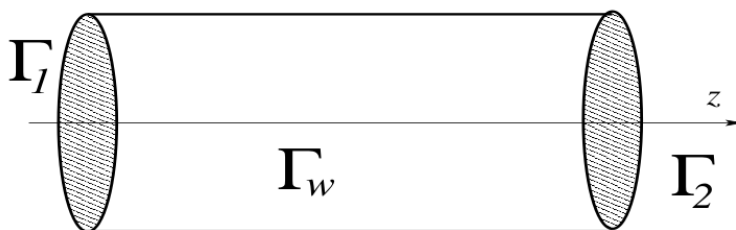


Figura 4.1: Modello di vaso standard [9].

Consideriamo un'arteria cilindrica Ω rappresentata in Figura 4.1. In seguito x denoterà il vettore delle variabili spaziali e $t > 0$ il tempo.

Il confine di Ω è composto dalla parete Γ_w e le sezioni Γ_1 e Γ_2 , le quali sono le interfacce verso il resto del sistema. Chiamiamo Γ_1 la sezione *prossimale*, ad esempio la sezione dalla quale il sangue arriva dal cuore; allo stesso modo chiamiamo Γ_2 la sezione *distale*, quella da cui il sangue lascia Ω per andare verso la periferia.

Associamo ad ogni distretto di questo tipo un asse orientato z . La posizione lungo il vaso sarà quindi identificata dalla media dell'ascissa associata all'asse, ancora chiamata z . In Ω abbiamo $0 \leq z \leq l$ (viene adottato il simbolo l al posto di L per indicare la lunghezza del vaso perché più avanti L indicherà l'induttanza). Per ogni z , la sezione assiale del vaso verrà indicata con $A(t, z)$. In seguito, $A(t, z)$ indicherà la misura di $A(t, z)$. Assumendo un dominio cilindrico circolare, avremo $A(t, z) = \pi R^2(t, z)$, dove $R(t, z)$ è il raggio della sezione in z . Infine $d\sigma$ indicherà l'infinitesima parte di $A(t, z)$ e il volume di Ω sarà indicato con V .

Il modello matematico per la descrizione del flusso sanguigno in Ω è ricavato dalle equazioni di Navier Stokes per un fluido incomprimibile associato ad una legge costitutiva di parete riguardante lo sforzo esercitato dal fluido sulla parete attraverso lo spostamento del tessuto vascolare.

Vengono introdotte delle ipotesi in modo da ottenere un modello semplificato di questo problema di interazione fluido-struttura.

1. *Simmetria assiale*: indipendenza di tutte le quantità interessate rispetto alla coordinata circonferenziale θ .
2. *Spostamento radiale*: ogni punto su Γ_w si muove solo in direzione radiale, in modo tale che se $\boldsymbol{\eta}$ è lo spostamento della parete, \boldsymbol{e}_r il vettore unità nella direzione radiale e

R_0 il raggio di riferimento, $\boldsymbol{\eta} = (R - R_0) \mathbf{e}_r$.

3. *Asse cilindrico fisso*: l'asse z è assunto come fisso nel tempo.
4. *Pressione costante su ogni sezione*: la pressione P è assunta dipendente solo dalla coordinata assiale z e dal tempo t .
5. *Assenza di forze specifiche di massa*: le forze specifiche di massa come ad esempio la gravità sono trascurate.
6. *Velocità assiale dominante*: le componenti del campo di velocità ortogonali all'asse z sono considerate trascurabili rispetto a quelle parallele all'asse z . La componente assiale della velocità verrà indicata con u_z .

Definiamo $Q(t, z) := \int_{A(t, z)} u_z d\sigma$ la portata volumetrica media attraverso la sezione A

e, in particolare:

(4.1)

$$Q_1(t) := Q(t, 0), \quad P_1(t) := P(t, 0), \quad Q_2(t) := Q(t, l), \quad P_2(t) := P(t, l)$$

Partendo dalle precedenti ipotesi, integriamo su ciascuna sezione $A(t, z)$ le equazioni di Navier-Stokes semplificate e otteniamo il seguente set di equazioni 1D per $z \in (0, L)$ e $t \in (0, T]$,

(4.2)

$$\begin{cases} \frac{\partial A}{\partial t} + \frac{\partial Q}{\partial z} = 0 \\ \frac{\partial Q}{\partial t} + \alpha \frac{\partial}{\partial z} \left(\frac{Q^2}{A} \right) + \frac{A}{\rho} \frac{\partial P}{\partial z} + K_R \frac{Q}{A} = 0 \end{cases}$$

completate da un adeguato set di condizioni al contorno e dati iniziali. Nelle (4.2), si assume che α (secondo coefficiente di Coriolis: coefficiente di ragguglio che dipende dal profilo di velocità considerato correzione della quantità di moto), ρ (densità del sangue) e K_R (parametro di frizione) siano costanti.

In particolare, K_R è correlato alla viscosità cinematica ν e la sua reale espressione dipende dal profilo di velocità. A titolo di esempio, se il profilo è di tipo parabolico (Poiseuille) $K_R = 8\pi\nu$. A , P e Q sono le incognite. Per il sistema, servono altre equazioni che sono date dalla legge costitutiva dei tessuti dei vasi. In altre parole introduciamo altre ipotesi.

7. *Lo spostamento della parete del vaso* η è legato alla pressione da una legge lineare algebrica. Assumiamo che:

$$(P - P_{ext}) = c(R - R_0) = \beta_0 \frac{\sqrt{A} - \sqrt{A_0}}{A_0} \quad (4.3)$$

dove P_{ext} e $A_0 = \pi R_0^2$ sono rispettivamente la pressione costante di riferimento e l'area costante di riferimento, mentre c è una costante legata alle proprietà fisiche dei tessuti vascolari e $\beta_0 = A_0 c / \sqrt{A_0}$. Più precisamente, se h indica lo spessore del tessuto vascolare, E il modulo di Young, ξ^2 il rapporto di Poisson (si imposta pari a $1/2$, come per un materiale incomprimibile), la legge costitutiva lineare elastica diventa:

$$c = \frac{hE}{R_0^2(1-\xi^2)} = \frac{4hE}{3R_0^2} \quad (4.4)$$

Ora osserviamo che:

$$\frac{\partial A}{\partial t} = 2\pi R \frac{d\eta}{dt} \approx \frac{\partial \eta}{\partial t} \quad (4.5)$$

assumiamo che :

$$\frac{\partial A}{\partial t} = \frac{3\pi R_0^3}{2Eh} \frac{\partial}{\partial t} P \quad (4.6)$$

Nel seguito, definiamo $k_1 = \frac{3\pi R_0^3}{2Eh}$

In modo da ricavare un descrizione a parametri concentrati del percorso del sangue in un intero distretto Ω è necessario effettuare delle medie delle equazioni (4.5) e (4.6) sulla coordinata assiale $z \in (0, l)$. Per questo scopo, è utile introdurre la seguente notazione. Definiamo come *la media volumetrica del flusso sull'intero distretto* la quantità:

$$\hat{Q} = \int_V u_z dV = \frac{1}{l} \int_0^l \int_{A(z)} u_z d\sigma dz = \frac{1}{l} \int_0^l Q dz \quad (4.7)$$

Analogamente, definiamo la *pressione media sull'intero compartimento* come:

$$\hat{p} = \frac{1}{l} \int_0^l P dz \quad (4.8)$$

Integriamo in $z \in (0, l)$ il sistema (4.2). Dalla prima equazione otteniamo:

$$\int_0^l \frac{\partial A}{\partial t} dz + \int_0^l \frac{\partial Q}{\partial z} dz = 0$$

Dalla (4.3), $\partial A/\partial t = k_1 \partial P/\partial t$ possiamo esplicitare la notazione introdotta nella (4.1). L'equazione precedente diventa:

$$k_1 l \frac{d\hat{p}}{dt} + Q_2 - Q_1 = 0 \quad (4.9)$$

Dall'integrazione della seconda equazione della (4.2) si ottiene:

$$\int_0^l \left[\frac{\partial Q}{\partial t} + \alpha \frac{\partial}{\partial z} \left(\frac{Q^2}{A} \right) + \frac{A}{\rho} \frac{\partial P}{\partial z} + K_R \frac{Q}{A} \right] dz =$$

$$l \frac{d\hat{Q}}{dt} + \alpha \left[\frac{Q_2^2}{A_2} - \frac{Q_1^2}{A_1} \right] + \int_0^l \left[\frac{A}{\rho} \frac{\partial P}{\partial z} + K_R \frac{Q}{A} \right] dz = 0$$

Per ottenere un *modello lineare*, introduciamo altre ipotesi chiamate:

8. *Il contributo dei termini convettivi può essere trascurato.* Assumiamo che la quantità

$$\left(\frac{Q_2^2}{A_2} - \frac{Q_1^2}{A_1} \right)$$

sia piccola rispetto agli altri termini e può essere trascurata. Questo è ammissibile per vasi di piccola lunghezza.

9. *La variazione di A rispetto alla lunghezza z sia piccola comparata a quella di P e Q ,* cosicché nell'ultimo integrale si ha:

$$\int_0^l \left[\frac{A}{\rho} \frac{\partial P}{\partial z} + K_R \frac{Q}{A} \right] dz \approx \int_0^l \left[\frac{A_0}{\rho} \frac{\partial P}{\partial z} + K_R \frac{Q}{A_0} \right] dz$$

Sfruttando queste ipotesi otteniamo, dopo alcune manipolazioni algebriche:

$$\rho \frac{l}{A_0} \frac{d\hat{Q}}{dt} + \frac{\rho K_R l}{A_0^2} \hat{Q} + P_2 + P_1 = 0 \quad (4.10)$$

Le equazioni (4.9) e (4.10) rappresentano una descrizione a parametri concentrati del flusso del sangue in un vaso cilindrico standard Ω , e coinvolgono i valori medi di pressione e portata lungo il dominio, oltre che i valori dei flussi e alle pressioni in ingresso e in uscita al vaso. Questo modello può essere considerato un *mattoncino elementare* per la descrizione di sistemi più complessi. In questa prospettiva è utile introdurre l'analogia di queste equazioni con quelle governanti una rete elettrica.

a) Analogia elettrica del modello a parametri concentrati

Nelle equazioni (4.9) e (4.10) ci sono alcuni coefficienti ottenuti da un processo di integrazione. Essi sono di fatto i parametri concentrati che riassumono le caratteristiche geometriche e fisiche del sistema dinamico formato dal flusso sanguigno e la parete vascolare. Qui di seguito vengono riassunti i significati:

R: Nell'equazione (4.10) definiamo $R := \frac{\rho K_R l}{A_0^2}$. Se assumiamo un profilo di velocità parabolico otteniamo:

$$R = \frac{8 \pi \rho \nu l}{\pi^2 R_0^4} = \frac{8 \mu l}{\pi R_0^4}$$

dove R rappresenta la *resistenza* applicata al flusso dalla viscosità del sangue. Una diversa espressione per R può essere ovviamente ottenuta per diversi profili di velocità o per un modello reologico non Newtoniano applicato al modello.

L: Nell'equazione (4.10) definiamo $L := \frac{\rho l}{A_0} = \frac{\rho l}{\pi R_0^2}$.

L rappresenta il termine inerziale nella legge di conservazione del momento e verrà chiamata *inerzia* del flusso.

C: Nell'equazione (4.9) definiamo $C := k_1 l = \frac{3 \pi R_0^3 l}{2 E h}$

C rappresenta il coefficiente del termine di accumulo di massa nella legge di conservazione della massa, dovuto alla *compliance* del vaso.

Con questa notazione, le equazioni (4.9) e (4.10) diventano:

$$\begin{cases} C \frac{d\hat{p}}{dt} + Q_2 - Q_1 = 0 \\ L \frac{d\hat{Q}}{dt} + R\hat{Q} + P_2 - P_1 = 0 \end{cases} \quad (4.11)$$

Ora supponiamo che alcuni dati a monte e a valle del vaso siano disponibili. Per esempio, supponiamo che siano dati Q_1 e P_2 . Quindi l'equazione (4.11) rappresenta un sistema di due equazioni in quattro incognite, \hat{Q} , \hat{p} , P_1 e Q_2 . Per chiudere matematicamente il problema ci servono ulteriori ipotesi. In particolare, la dinamica del sistema è rappresentata

da \hat{p} e \hat{Q} , le incognite che sono sottoposte alla derivata nel tempo (variabili di stato), quindi è ragionevole approssimare le incognite delle sezioni a monte e a valle con le variabili di stato:

$$\hat{p} \approx P_1, \hat{Q} \approx Q_2$$

Attraverso queste ulteriori ipotesi, che sono applicabili per tubi cilindrici di breve lunghezza, il modello a parametri concentrati diventa:

$$\begin{cases} C \frac{dP_1}{dt} + Q_2 = Q_1 \\ L \frac{dQ_2}{dt} + RQ_2 - P_1 = P_2 \end{cases} \quad (4.12)$$

dove i dati a monte e a valle sono stati inseriti sia a destra e a sinistra delle formule.

Questo sistema può essere schematizzato dalla rete elettrica \mathcal{L} -network mostrata in Figura 4.2 (sinistra). La conformità dell'accoppiamento è gestita alla sezione Γ_1 , il flusso è imposto e gli effetti inerziali sono allocati alla sezione Γ_2 dove è assegnata la pressione media.

Nella analogia elettrica, il flusso del sangue (portata) è associato alla corrente, mentre la pressione sanguigna corrisponde alla tensione (Tabella 2).

Tabella 2: Corrispondenza delle grandezze tra i modelli idraulico ed elettrico

Modello idraulico	Modello elettrico
Pressione	Tensione
Portata	Corrente
Viscosità del sangue	Resistenza R
Inerzia del sangue	Induttanza L
Capacitanza della parete	Capacità C

Analogamente a quanto descritto in precedenza, se la pressione P_1 e la portata Q_2 sono preimpostate, possiamo approssimare le grandezze incognite nelle sezioni a monte e a valle con le variabili di stato:

$$\hat{p} \approx P_2, \hat{Q} \approx Q_1$$

di conseguenza il sistema è:

$$\begin{cases} C \frac{dP_2}{dt} - Q_1 = Q_2 \\ L \frac{dQ_1}{dt} + RQ_1 + P_2 = P_1 \end{cases} \quad (4.13)$$

Il circuito analogico che rappresenta il sistema (4.13), chiamato \mathcal{L} -inverted network è mostrato in Figura 4.2 a destra.

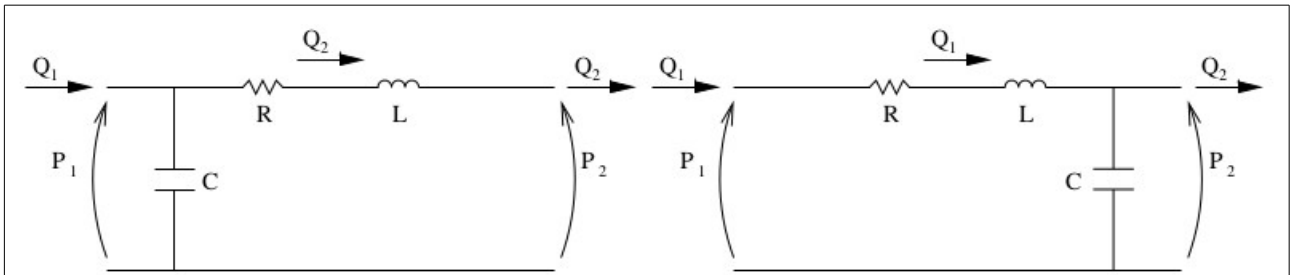


Figura 4.2: Reti equivalenti a par. concentrati: \mathcal{L} -network (sinistra) e \mathcal{L} -inverted network (destra)

Nel caso in cui siano assegnate le pressioni medie P_1 e P_2 , il sistema può essere modellato dalla connessione di circuiti \mathcal{L} -network e \mathcal{L} -inverted network in cascata. Più precisamente si può suddividere il vaso in due parti Ω_1 e Ω_2 di lunghezza $\frac{1}{2}$.

Nella prima parte Ω_1 si assume che P_1 e la portata \hat{Q} siano conosciute; nella seconda parte Ω_2 si assume che la portata \hat{Q} e P_2 siano conosciute. In questo modo l'intero vaso Ω è reinterpretato come la \mathcal{T} -network mostrata in Figura 4.3.

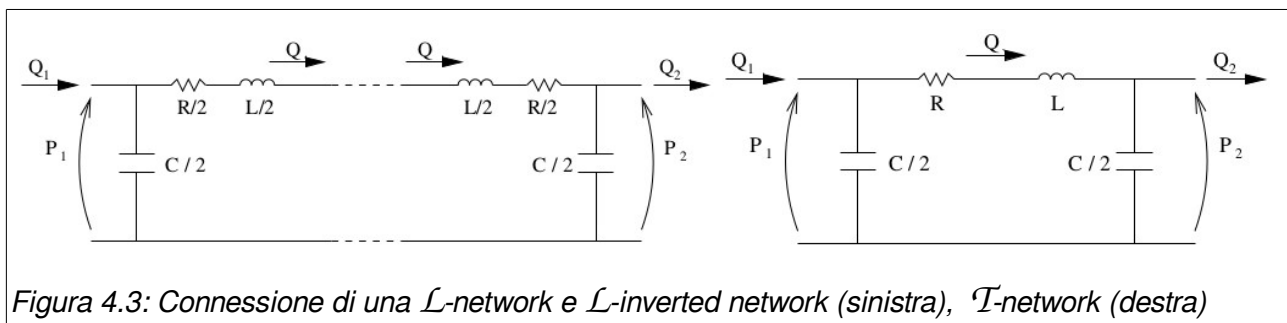


Figura 4.3: Connessione di una \mathcal{L} -network e \mathcal{L} -inverted network (sinistra), \mathcal{T} -network (destra)

4.2. Modello matematico del circuito sistemico e polmonare

Il circuito vascolare sistemico extra intestinale è stato rappresentato in quattro distinti compartimenti:

- circolazione nel cervello (b);
- circolazione nei muscoli scheletrici (m);
- circolazione coronarica (h);
- circolazione di tutti i rimanenti letti vascolari (e);

In Figura 4.4 è schematizzato il modello completo utilizzato e che è stato introdotto nel lavoro di M.Ursino [10]. In questo modello è rappresentato anche l'albero coronarico (grandezze con pedice hv e hp). Il modello che verrà presentato in seguito utilizza uno schema più dettagliato dell'albero coronarico.

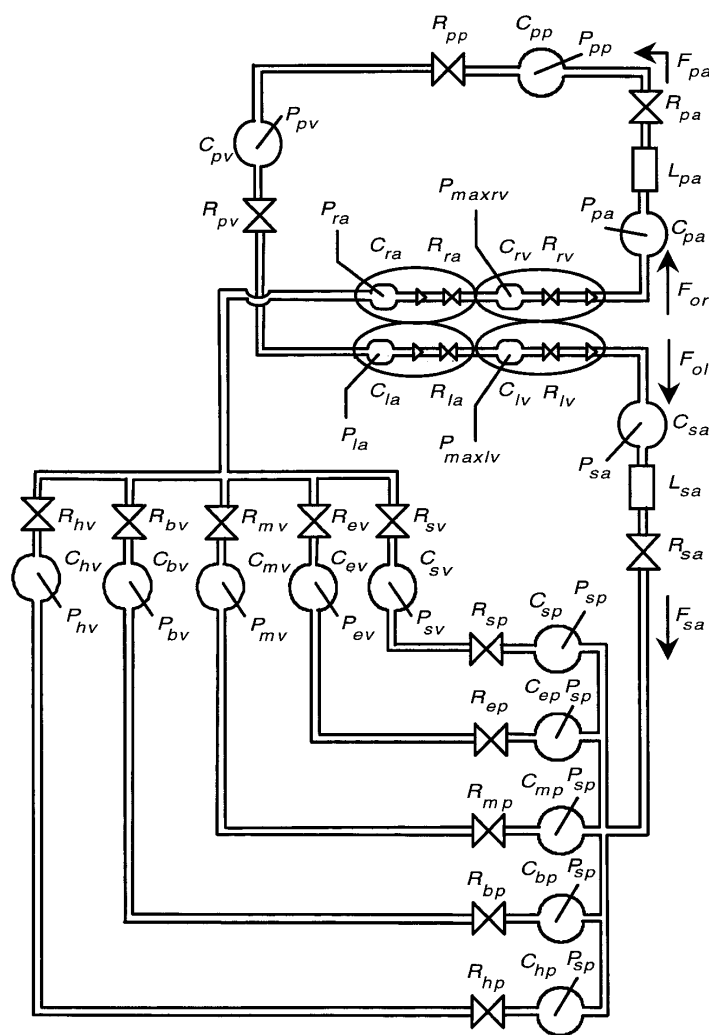


Figura 4.4: Analogo idraulico del sistema cardiovascolare. P , pressioni; R , resistenze idrauliche; C , capacità; L , inerzie; F , portate.

Le equazioni che seguono, esprimono i bilanci di massa per le capacità e gli equilibri delle quantità di moto alle induttanze.

Vengono indicate con P_j le pressioni intravascolari del j -esimo compartimento, V_{uj} è il corrispondente volume rilassato (definito come volume a pressione zero), Q_j è la portata volumetrica, mentre C_j , L_j e R_j sono rispettivamente le capacitance/capacità (inglese: compliance), induttanze/inertanze (inglese: inertance) e resistenze idrauliche. $Q_{o,r}$ e $Q_{o,l}$ sono la portata uscente dal ventricolo destro e sinistro.

Conservazione di massa calcolata alle arterie polmonari (C_{pa})

$$\frac{dP_{pa}}{dt} = \frac{1}{C_{pa}} \cdot (Q_{o,r} - Q_{pa}) \quad (4.14)$$

dove P_{pa} e Q_{pa} sono rispettivamente pressione e portata delle arterie polmonari.

Bilancio delle quantità di moto alle arterie polmonari (L_{pa})

$$\frac{dQ_{pa}}{dt} = \frac{1}{L_{pa}} \cdot (P_{pa} - P_{pp} - R_{pa} \cdot Q_{pa}) \quad (4.15)$$

dove P_{pp} è la pressione polmonare periferica e R_{pa} è la resistenza delle arterie polmonari.

Conservazione di massa calcolata alla circolazione polmonare periferica (C_{pp})

$$\frac{dP_{pp}}{dt} = \frac{1}{C_{pp}} \cdot \left(Q_{pa} - \frac{P_{pp} - P_{pv}}{R_{pp}} \right) \quad (4.16)$$

dove C_{pp} è la complianza della circolazione polmonare, P_{pv} è la pressione venosa polmonare e R_{pp} è la resistenza della circolazione polmonare.

Conservazione di massa calcolata alle vene polmonari (C_{pv})

$$\frac{dP_{pv}}{dt} = \frac{1}{C_{pv}} \cdot \left(\frac{P_{pp} - P_{pv}}{R_{pp}} - \frac{P_{pv} - P_{la}}{R_{pv}} \right) \quad (4.17)$$

dove R_{pv} è la resistenza del circuito venoso polmonare e P_{la} è la pressione atriale sinistra.

Conservazione di massa calcolata alle arterie del circuito sistemico (C_{sa})

$$\frac{dP_{sa}}{dt} = \frac{1}{C_{sa}} \cdot (Q_{o,l} - Q_{sa}) \quad (4.18)$$

dove P_{sa} è la pressione arteriale sistemica e Q_{sa} è la portata arteriale sistemica.

Bilancio delle quantità di moto alle arterie del circuito sistemico (L_{sa})

$$\frac{dQ_{sa}}{dt} = \frac{1}{L_{sa}} \cdot (P_{sa} - P_{sp} - R_{sa} \cdot Q_{sa}) \quad (4.19)$$

dove P_{sp} è la pressione periferica sistemica e R_{sa} è la resistenza arteriale sistemica.

Conservazione di massa calcolata alla circolazione periferica sistemica (C_{sp} , C_{ep} , C_{mp} , C_{bp} e C_{hp})

$$\frac{dP_{sp}}{dt} = \frac{1}{C_{sp} + C_{ep} + C_{mp} + C_{bp} + C_{hp}} \cdot \left(Q_{sa} - \frac{P_{sp} - P_{sv}}{R_{sp}} - \frac{P_{sp} - P_{ev}}{R_{ep}} - \frac{P_{sp} - P_{mv}}{R_{mp}} - \frac{P_{sp} - P_{mv}}{R_{mp}} - \frac{P_{sp} - P_{bv}}{R_{bp}} - \frac{P_{sp} - P_{hv}}{R_{hp}} \right) \quad (4.20)$$

dove C_{sp} , C_{ep} , C_{mp} , e C_{bp} sono rispettivamente la complianza del circuito vascolare intestinale, l'extra-intestinale, muscolo scheletrica e del cervello; R_{sp} , R_{ep} , R_{mp} , e R_{bp} sono rispettivamente la resistenza del circuito vascolare, l'extra-intestinale, muscolo scheletrica e del cervello; P_{sv} , P_{ev} , P_{mv} , e P_{bv} sono rispettivamente le pressioni del circuito venoso intestinale, l'extra-intestinale, muscolo scheletrica e del cervello;

Conservazione di massa calcolata al circuito venoso intestinale (C_{sv})

$$\frac{dP_{sv}}{dt} = \frac{1}{C_{sv}} \cdot \left(\frac{P_{sp} - P_{sv}}{R_{sp}} - \frac{P_{sv} - P_{ra}}{R_{sv}} - \frac{dV_{u,sv}}{dt} \right) \quad (4.21)$$

dove R_{sv} e R_{sp} sono rispettivamente le resistenze del circuito venoso intestinale e periferico. P_{ra} è la pressione atriale destra e $V_{u,sv}$ è il volume rilassato del del circuito venoso intestinale.

Conservazione di massa calcolata al circuito venoso del cervello (C_{bv})

$$\frac{dP_{bv}}{dt} = \frac{1}{C_{bv}} \cdot \left(\frac{P_{sp} - P_{bv}}{R_{bp}} - \frac{P_{bv} - P_{ra}}{R_{bv}} \right) \quad (4.22)$$

dove R_{bv} è la resistenza del circuito venoso del cervello.

In ultimo la pressione venosa nel compartimento extra-intestinale (P_{ev}) è calcolato assumendo che il volume sanguigno contenuto nel sistema circolatorio (V_{tot}) sia noto.

Quindi:

$$P_{ev} = \frac{1}{C_{ev}} \cdot [V_{tot} - C_{sa} \cdot P_{sa} - (C_{sp} + C_{ep} + C_{mp} + C_{bp}) P_{sp} - C_{sv} \cdot P_{sv} - C_{mv} \cdot P_{mv} - C_{bv} \cdot P_{bv} - C_{hv} \cdot P_{hv} - C_{ra} \cdot P_{ra} - V_{rv} - C_{pa} \cdot P_{pa} - C_{pp} \cdot P_{pp} - C_{pv} \cdot P_{pv} - C_{la} \cdot P_{la} - V_{lv} - V_u] \quad (4.23)$$

4.3. Modello matematico del cuore

Anche per quanto riguarda il modello matematico del cuore si può affrontare con lo stesso approccio utilizzato fin ora.

P_{la} Può essere calcolata applicando la conservazione di massa. Quindi:

$$\frac{dP_{la}}{dt} = \frac{1}{C_{la}} \cdot \left(\frac{P_{pv} - P_{la}}{R_{pv}} - Q_{i,l} \right) \quad (4.24)$$

dove il primo termine tra le parentesi è il ritorno venoso, mentre $Q_{i,l}$ rappresenta il flusso di sangue che entra nel ventricolo sinistro. Questo termine dipende dall'apertura della valvola atrioventricolare. Quindi:

$$Q_{i,l} = \begin{cases} 0 & \text{if } P_{la} \leq P_{max,lv} \\ \frac{P_{la} - P_{max,lv}}{R_{la}} & \text{if } P_{la} > P_{max,lv} \end{cases} \quad (4.25)$$

V_{lv} È calcolato applicando la seguente conservazione di massa:

$$\frac{dV_{lv}}{dt} = Q_{i,l} - Q_{o,l} \quad (4.26)$$

Dove $Q_{o,l}$ è la portata in uscita dal ventricolo sinistro. L'output cardiaco dipende dall'apertura della valvola aortica. Assumendo che quest'ultima sia ideale:

$$Q_{o,l} = \begin{cases} 0 & \text{if } P_{max,lv} \leq P_{sa} \\ \frac{P_{max,lv} - P_{sa}}{R_{lv}} & \text{if } P_{max,lv} > P_{sa} \end{cases} \quad (4.27)$$

dove $P_{max,lv}$ è la pressione isometrica del ventricolo sinistro. La resistenza viscosa del ventricolo sinistro (R_{lv}) è ipotizzata proporzionale alla pressione isometrica e, a conseguenza di ciò, si incrementa significativamente durante la contrazione. Si può quindi scrivere:

$$R_{lv} = k_{R,lv} - P_{max,lv} \quad (4.28)$$

Dove $k_{R,lv}$ è un parametro costante.

La pressione istantanea del ventricolo sinistro può essere ottenuta come differenza tra la pressione isometrica e le perdite viscosi. Quindi:

$$P_{lv} = P_{max,lv} - R_{lv} \cdot F_{o,l} \quad (4.29)$$

Si assume che il rapporto pressione isometrica / volume sia una funzione esponenziale nella fase di diastole, quando il ventricolo è rilassato, mentre lineare alla fine della sistole, quando il ventricolo si trova alla massima contrazione [11]:

$$P_{max,lv}(t) = \varphi(t) \cdot E_{max,lv} \cdot (V_{lv} - V_{u,lv}) + [1 - \varphi(t)] \cdot P_{0,lv} \cdot (e^{k_{E,lv} \cdot V_{lv}} - 1) \quad 0 \leq \varphi(t) \leq 1 \quad (4.30)$$

dove $k_{R,lv}$ è l'elastanza del ventricolo all'istante di massima contrazione. $V_{u,lv}$ È il corrispondente volume rilassato del ventricolo sinistro. $P_{0,lv}$ e $k_{E,lv}$ sono parametri costanti che caratterizzano la funzione esponenziale monotona in diastole. $\varphi(t)$ è la funzione di attivazione del ventricolo (con $\varphi(t)=1$ alla massima contrazione e $\varphi(t)=0$ al completo rilassamento). Viene adottata la seguente espressione [12]:

$$\varphi(t) = \begin{cases} \sin^2 \left[\frac{\pi \cdot T(t)}{T_{sys}(t)} \cdot u \right] & 0 \leq u \leq T_{sys}/T \\ 0 & T_{sys}/T \leq u \leq 1 \end{cases} \quad (4.31)$$

dove T è il periodo del ciclo cardiaco, T_{sys} è la durata della sistole e u è una variabile adimensionale, compresa tra 0 e 1, che rappresenta la frazione di ciclo cardiaco. Per convenzione il valore $u=0$ è assunto all'inizio della sistole. L'espressione adottata per $u(t)$ è ottenuta mediando l'espressione di un modello "integrate and fire":

$$u(t) = frac \left[\int_{t_0}^t \frac{1}{T(\tau)} d\tau + u(t_0) \right] \quad (4.32)$$

dove la funzione "parte frazionaria" ($frac$) azzera la variabile $u(t)$ a zero quando quest'ultima raggiunge il valore di +1.

T_{sys} decresce linearmente con la frequenza cardiaca.

$$T_{sys} = T_{sys,0} - k_{sys} \cdot \frac{1}{T} \quad (4.33)$$

dove $T_{sys,0}$ e k_{sys} sono parametri costanti.

Analogamente le equazioni (4.24) e (4.33) possono essere riscritte per il lato destro del cuore.

4.4. Modello matematico dell'albero coronarico

Le arterie coronarie vengono suddivise in 17 segmenti. Anche per questa parte vengono sfruttati gli analoghi elettrici di resistenza, compliance e elastanza.

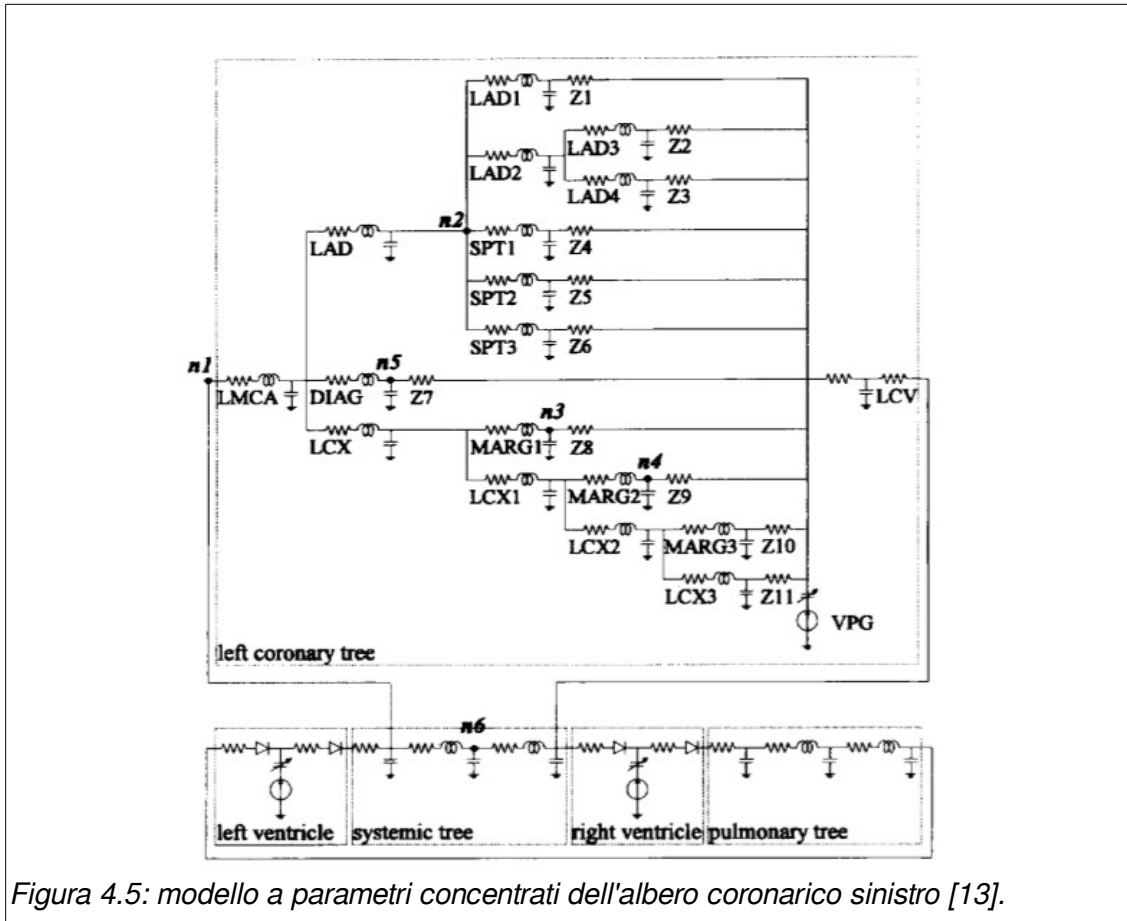


Figura 4.5: modello a parametri concentrati dell'albero coronarico sinistro [13].

In Figura 4.5 è mostrato il modello a parametri concentrati dell'albero coronarico. Tale modello è stato introdotto nel lavoro di R. Pietrabissa [13].

Si fa notare che, nella fase sistolica, la pressione che produce la perfusione del miocardio è la differenza di tra le pressioni aortica e coronarica, mentre nella fase diastolica il flusso di sangue è generato dalla differenza di pressione tra l'aorta e l'atrio destro.

Per prendere in considerazione questa caratteristica peculiare della circolazione coronarica viene introdotto un generatore di pressione (*VPG*), che simula la pressione all'interno del miocardio durante tutto il ciclo cardiaco. *VPG* fornisce una pressione che ha lo stesso valore e fase di quella del ventricolo sinistro [13].

I parametri che costituiscono le costanti delle rete elettrica dell'albero coronarico sono ricavati dalle seguenti formule:

$$R = \frac{128 \mu l}{\pi D} \qquad L = \frac{4 \rho l}{\pi D^2} \qquad C = \frac{\pi D^3 l}{4 E h}$$

dove $\mu = 4 \cdot 10^{-3} \text{ kg} \cdot \text{m}^{-1} \cdot \text{s}^{-1}$ è la viscosità del sangue; $\rho = 1 \cdot 10^3 \text{ kg} \cdot \text{m}^{-3}$ è la densità del sangue utilizzata nel modello; $E = 2 \cdot 10^5 \text{ Pa}$ è il modulo di Young del vaso; l , D e h espresse in *metri* sono rispettivamente la lunghezza, il diametro e lo spessore del vaso. L'ultimo parametro può essere stimato con la formula $h = 0.08 \cdot D$ [13].

Tabella 3: Valori dei parametri R, L, C e Z del modello dell'albero coronarico sinistro [13].

Ramo	Resistenza (R) (mmHg·s/ml)	Induttanza (L) (mmHg·s²/ml)	Capacità (C) (ml/mmHg)	Afterload (Z) (mmHg·s/ml)
LMCA	6.0	0.02	2.0×10^{-3}	
LAD	0.5	0.03	1.5×10^{-3}	
LAD1	3.1	0.10	1.3×10^{-3}	115.6
LAD2	0.9	0.04	1.2×10^{-3}	
LAD3	6.7	0.10	2.8×10^{-4}	425.0
LAD4	4.7	0.16	1.9×10^{-3}	55.2
DIAG	1.9	0.08	1.6×10^{-3}	55.1
LCX	0.3	0.02	1.1×10^{-3}	
MARG1	1.4	0.06	1.2×10^{-3}	115.0
LCX1	0.4	0.02	6.5×10^{-4}	
MARG2	2.2	0.07	7.7×10^{-4}	128.0
LCX2	0.8	0.04	7.5×10^{-4}	
MARG3	2.4	0.08	1.0×10^{-3}	122.0
LCX3	2.7	0.09	1.0×10^{-3}	130.1
SPT1	3.1	0.10	1.3×10^{-3}	115.6
SPT2	6.7	0.10	2.8×10^{-4}	327.5
SPT3	18.3	0.14	9.8×10^{-5}	327.0
Capillari	10.6			
Circuito venoso	3.0		6.0×10^{-2}	

Capitolo 5:

Implementazione tramite il linguaggio Modelica del modello 0D.

In questo capitolo viene descritta l'implementazione tramite il linguaggio Modelica del modello a parametri concentrati dell'intero sistema vascolare descritto nel capitolo precedente.

La formulazione computazionale dell'intero modello 0D è implementata tramite il linguaggio Modelica, in parte utilizzando il tool OMEdit incluso nell'ambiente OpenModelica, in parte utilizzando direttamente il linguaggio Modelica attraverso la scrittura diretta di file di testo con estensione *.mo.

5.1. Attivazione cardiaca

La funzione di attivazione descritta nel paragrafo 4.3 è sviluppata tramite blocchi matematici presenti nella libreria standard OpenModelica.

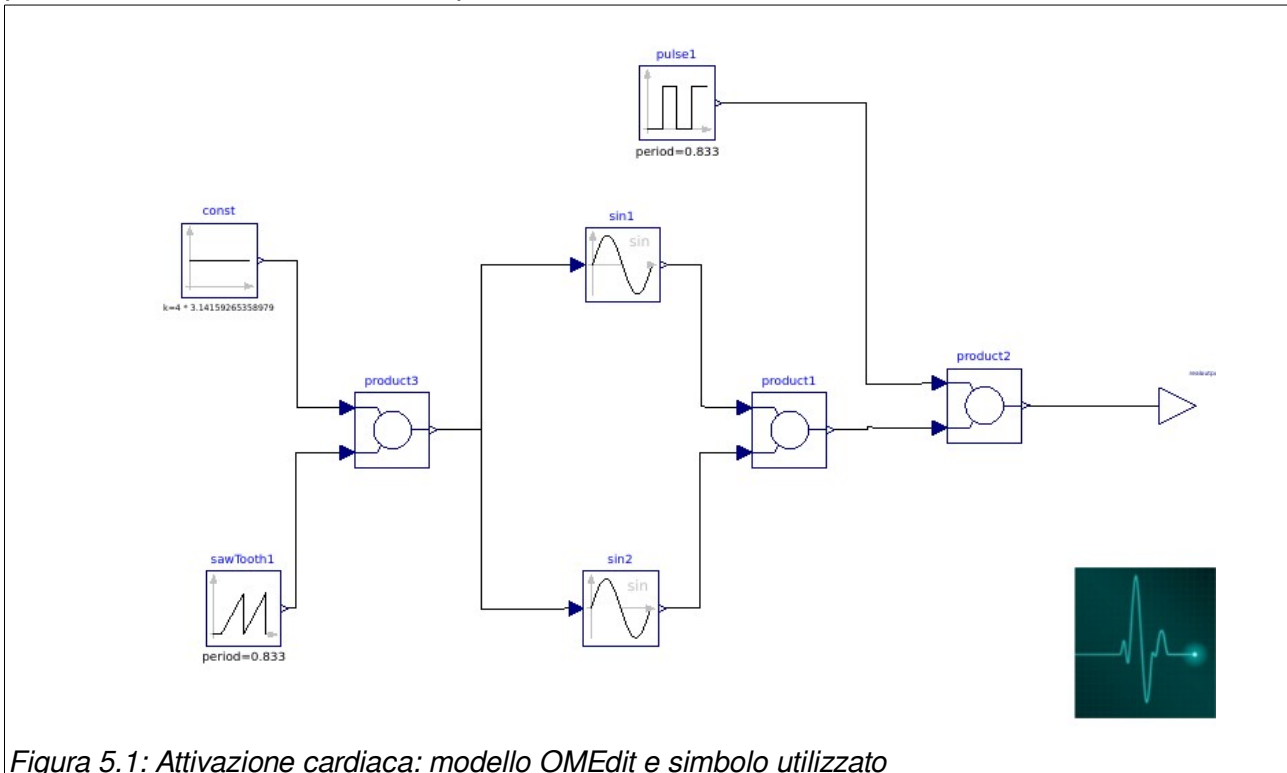


Figura 5.1: Attivazione cardiaca: modello OMEdit e simbolo utilizzato

Il modello grafico riconduce ad un file di testo di facile lettura, necessario e sufficiente per la compilazione del modello.

La frequenza cardiaca è impostata costante a 72 battiti al minuto mentre $T_{sys,0}$ e k_{sys} sono parametri costanti impostati rispettivamente a 0,5 e 0,075 [10]. La funzione di attivazione risulta quindi una funzione compresa tra 0 ed 1 periodica di periodo 0,833s.

L'andamento nel tempo della attivazione cardiaca è mostrato in Figura 5.1.

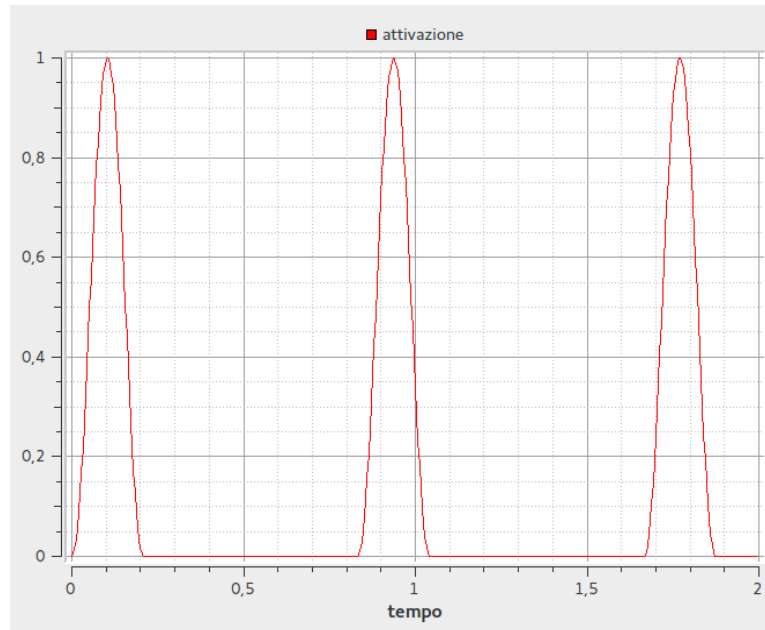


Figura 5.2: Funzione di attivazione. Andamento temporale.

5.2. Vasi arteriosi

In Figura 5.3 è mostrato l'equivalente elettrico di un vaso arterioso utilizzato nel modello 0D. A ciascun modello è collegato, sia in ingresso che in uscita, un blocco di interfaccia che ne permette il collegamento con gli altri blocchi.

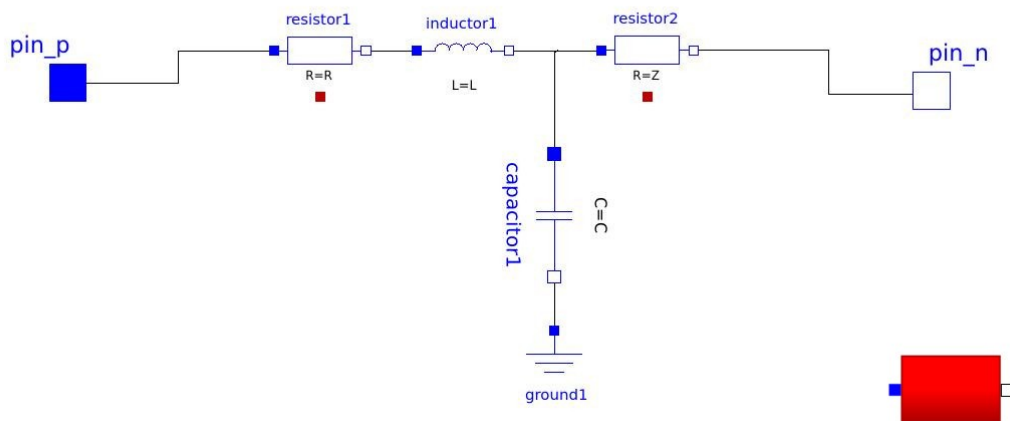


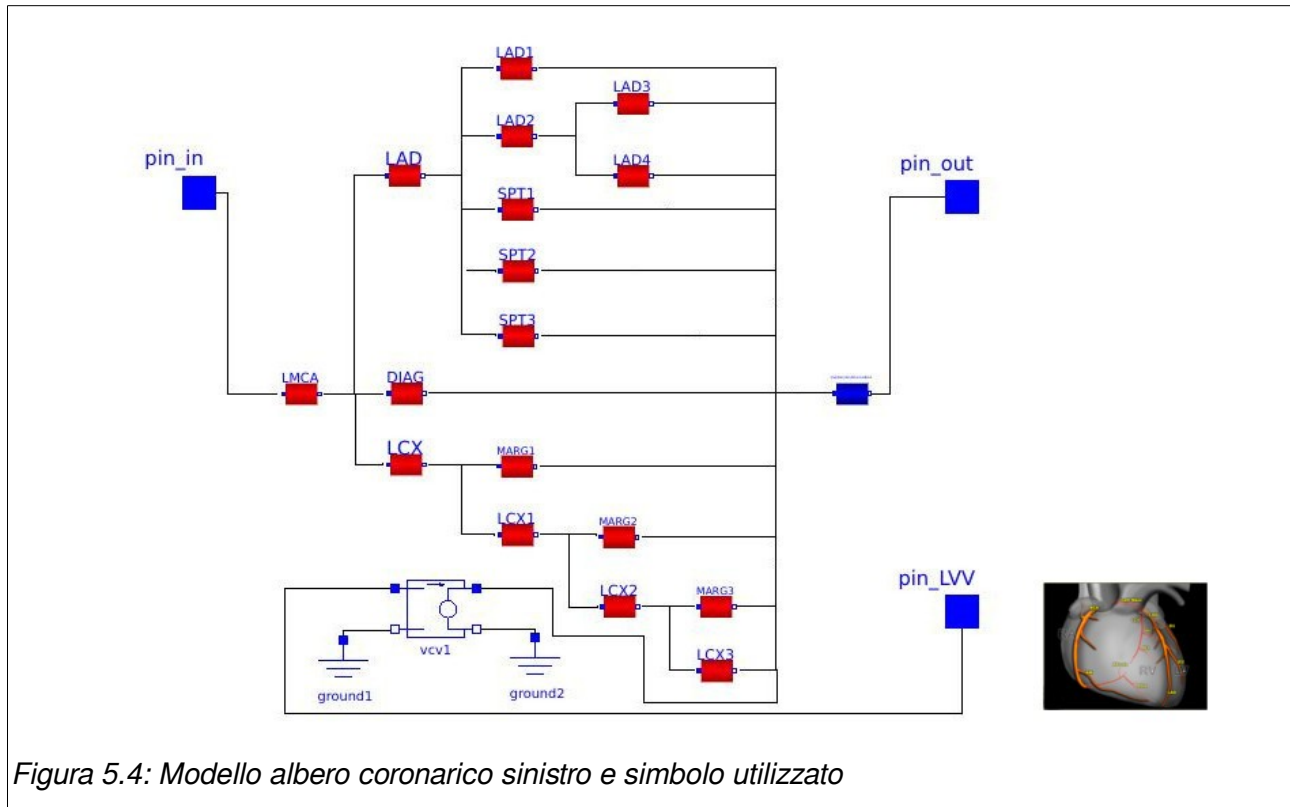
Figura 5.3: Equivalente elettrico di un vaso arterioso e simbolo utilizzato.

5.3. Circuito Sistemico e muscolo cardiaco

L'implementazione in linguaggio Modelica del modello di circuito sistemico e del modello del muscolo cardiaco descritti nei paragrafi 4.2 e 4.3, è avvenuta compilando direttamente il file

di testo con estensione “.mo”. Il vantaggio che comporta l'utilizzo di tale linguaggio ha fatto sì che la traduzione delle equazioni fosse estremamente semplice, di facile lettura e revisione. Il file risultato di questa implementazione è riportato in Appendice.

5.4. Albero coronarico (coronaria sinistra)



L'albero coronarico composto come in Figura 5.4 riproduce il modello introdotto al paragrafo 4.4. Sono stati utilizzati i modelli del vaso arterioso/venoso introdotti nel paragrafo 5.2. Questo blocco funzionale ha tre elementi di interfaccia. Il primo (indicato con `pin_in`) andrà collegato al blocco corrispondente all'aorta. Il secondo (indicato con `pin_out`) andrà collegato al ritorno venoso e il terzo (indicato con `pin_LVV`) verrà collegato al valore di pressione ventricolare sinistra al fine di simulare la caratteristica peculiare della fluidodinamica dell'albero coronarico descritta al paragrafo 4.4. Al fine di non introdurre un passaggio di corrente (e quindi una portata di sangue) tra il ventricolo sinistro e il lato venoso dell'albero coronarico si è applicato un generatore di tensione comandato in tensione al posto del collegamento diretto.

5.5. Sistema 0D completo

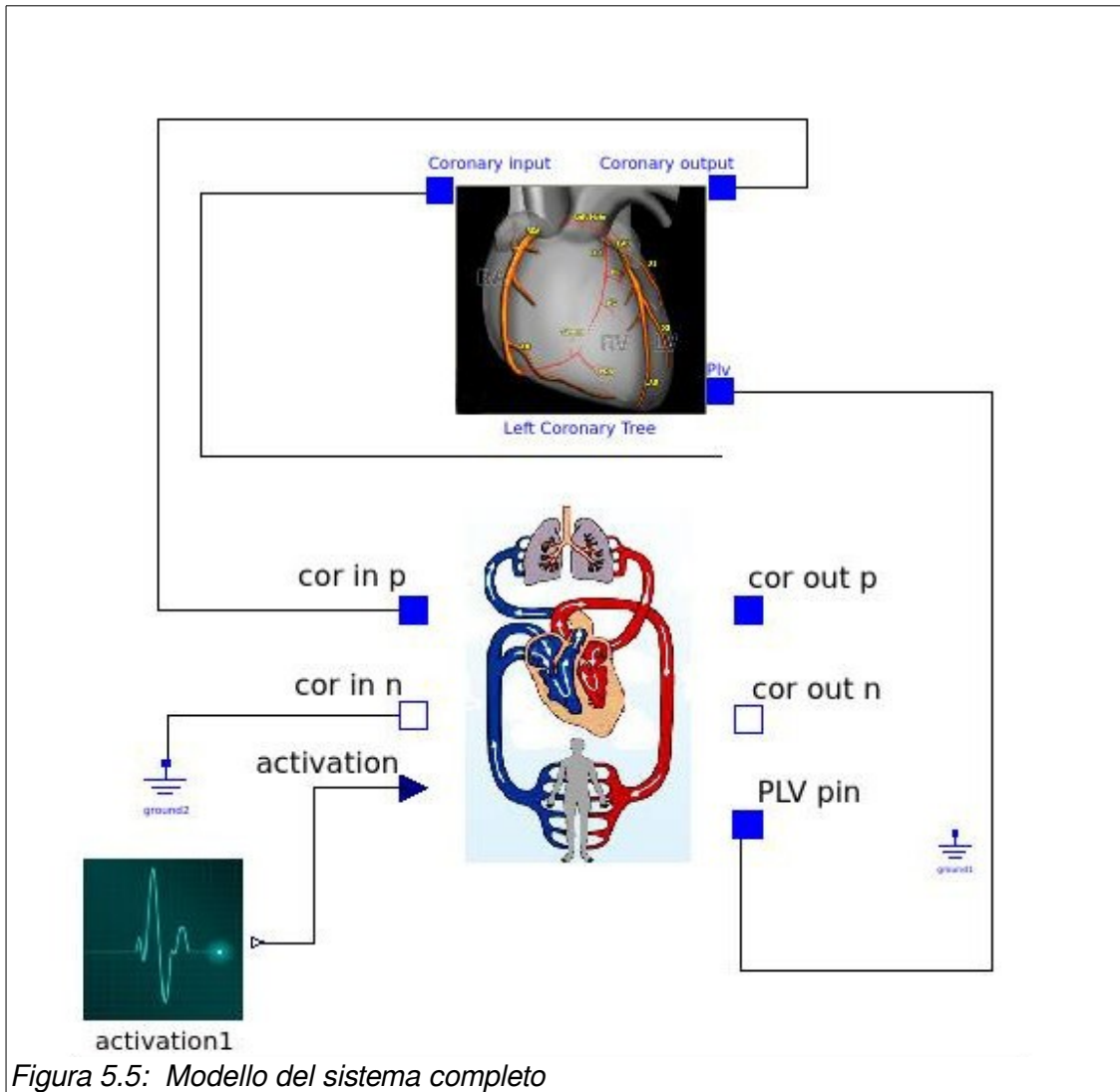


Figura 5.5: Modello del sistema completo

Ciascun modello possiede una rappresentazione compatta permettendo il suo utilizzo all'interno di un altro circuito e, di conseguenza, semplificando la visualizzazione del sistema globale.

Capitolo 6:

Modello fluidodinamico 3D

In questo capitolo viene descritta la progettazione del modello matematico tridimensionale delle due tipologie di vaso, fisiologico e stenotico, che si andranno a considerare. Vengono descritte inoltre le griglie di discretizzazione dei domini di interesse e l'impostazione dei casi di studio in OpenFOAM

Come già accennato in precedenza hanno una grande incidenza le patologie legate all'albero coronarico e in particolare all'occlusione di uno o più dei suoi vasi.

Una coronaria stenotica può essere descritta come un condotto altamente deformabile (con un comportamento non elastico) nel quale scorre un fluido dal comportamento non newtoniano (che potrebbe anche essere descritto come fluido bifase) in un regime di moto non stazionario e che presenta zone di ricircolo e fenomeni di turbolenza. Ovviamente indagare contemporaneamente tutte queste caratteristiche con un approccio di tipo computazionale risulta alquanto oneroso dal punto di vista delle risorse di calcolo richieste.

Un possibile approccio consiste nell'accettare delle semplificazioni che consentano di valutare l'effetto di una singola caratteristica per volta.

Esula dagli obiettivi di questo lavoro lo studio di una discretizzazione del dominio spaziale ad hoc e l'utilizzo di modelli matematici più sofisticati in grado di simulare, con buona approssimazione, il flusso di sangue all'interno di un vaso coronarico.

Per quanto riguarda la simulazione CFD del dominio tridimensionale, non vengono presi in considerazione modelli di turbolenza o multifase ma viene utilizzato il risolutore simpleFoam della libreria OpenFOAM relativo ad un calcolo incomprimibile non-stazionario di un flusso laminare per fluidi Newtoniani.

6.1. Definizione della geometria

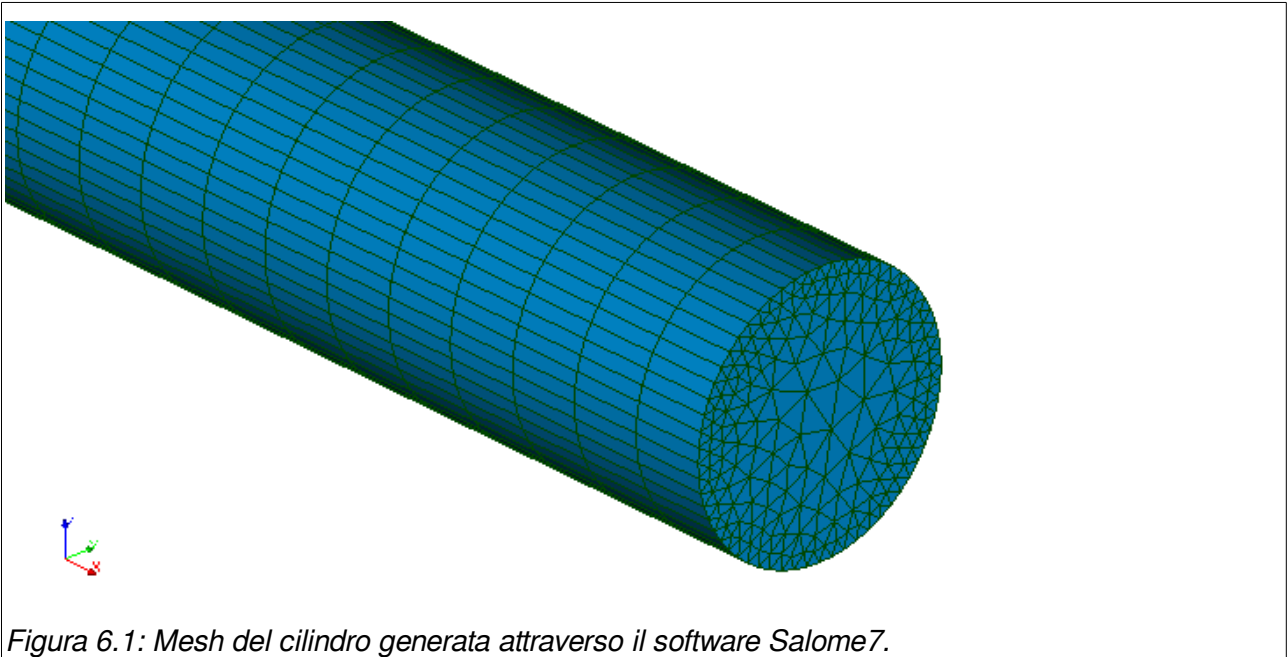
Le geometrie utilizzate rappresentano il ramo discendente anteriore interventricolare della coronaria sinistra (LAD) nel caso fisiologico e in presenza di stenosi.

Le due tipologie sono:

- Cilindrica che rappresenta il vaso in condizioni fisiologiche.
- Cilindrica con restringimento centrale che rappresenta il vaso in presenza di stenosi.

a) Geometria e mesh del vaso fisiologico.

Il modello di vaso coronarico fisiologico è stato scelto con l'obiettivo di ottenere dei valori globali di pressione e portata consistenti con il modello reale al fine di poter testare il sistema accoppiato 0D/3D. Lo studio fluidodinamico dettagliato di tale geometria non è obiettivo di questo lavoro di tesi, oltre a non essere molto significativo in senso assoluto a causa della sua semplicità.



In conseguenza di queste ipotesi è stata scelta la geometria più semplice possibile e quindi si è scelto un cilindro a base circolare di diametro costante di 3mm e di lunghezza 50mm. L'asse x è stato scelto come asse di simmetria.

Creazione della mesh

La mesh è costituita da volumi prismatici.

Le facce del cilindro sono state discretizzate attraverso l'algoritmo NETGEN2D, mentre in senso longitudinale è stata realizzata una discretizzazione quadrangolare regolare.

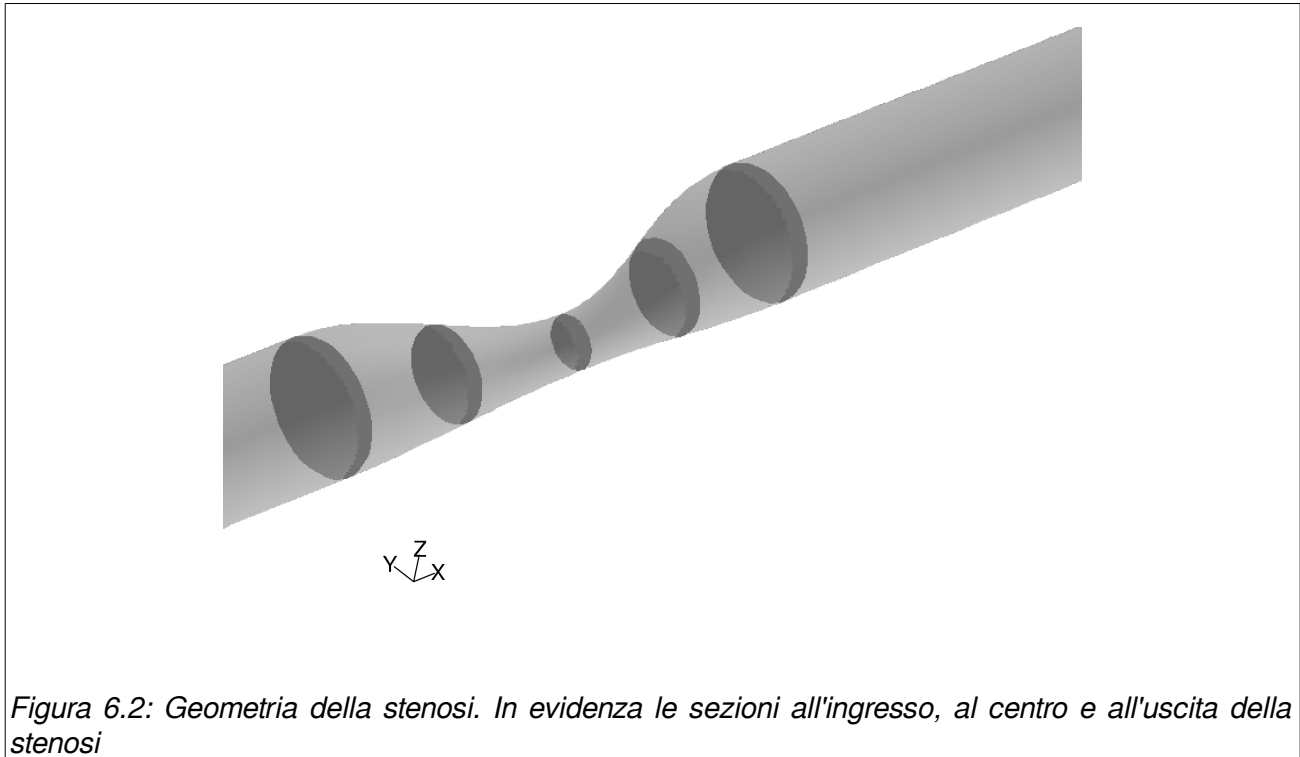
La mesh è risultata composta da 21840 elementi.

Vengono definite tre superfici di confine del dominio:

- inflow: interfaccia di ingresso del flusso
- outflow: interfaccia di uscita del flusso
- wall: parete del vaso.

b) Geometria e mesh del vaso patologico.

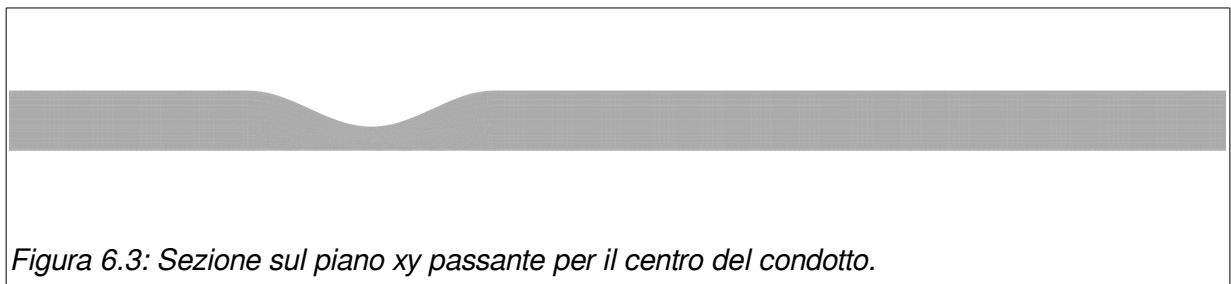
Come tipologia di stenosi è stata scelta una stenosi eccentrica dell'84%, poiché risulta essere la conformazione più diffusa. In particolare è stato creato un cilindro del diametro



interno di 3mm a rappresentare una coronaria sana; al centro di esso è stato creato un restringimento, in cui la sezione minima ha un diametro pari a 1,2mm atto a modellizzare la stenosi dell'84%.

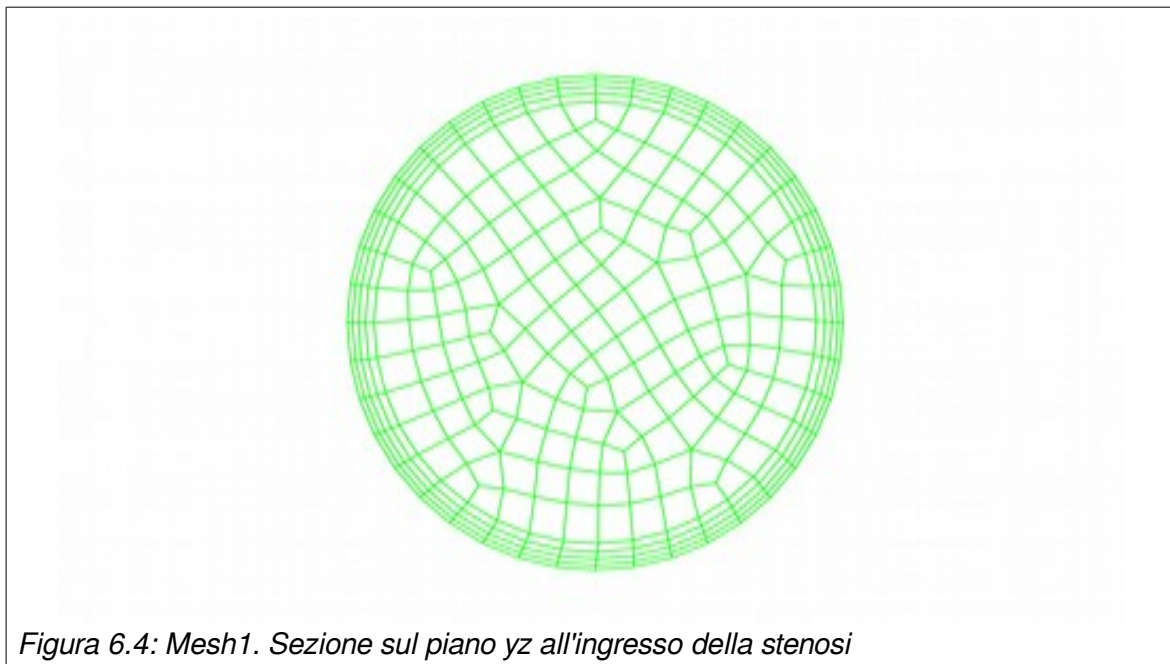
Per raccordare le due parti “sane” della stenosi si è scelto di utilizzare una funzione sinusoidale tale per cui non fossero presenti discontinuità all'interno del condotto.

In particolare selezionato il piano XY passante per il centro del condotto sono stati creati 5 punti tali per cui essi giacessero sulla sinusoide che congiunge le due parti della coronaria e che abbia un minimo al centro distante 1,2mm dall'altra parete del vaso (Figura 6.3).



Creazione della Mesh

La mesh, costituita da volumi esaedrici, è stata realizzata tramite il software commerciale GAMBIT (Ansys, Inc). E' stato creato un boundary layer costituito da 4 righe e caratterizzato da un rapporto di incremento pari a 1.2. Lungo la circonferenza che delimita la sezione del vaso sono stati creati 40 segmenti della stessa dimensione, a partire dai quali è stato costruito il boundary layer. La mesh relativa alla superficie di ingresso del vaso è stata completata con l'algoritmo di GAMBIT "PAVED", in modo da ottenere celle quadrangolari. Per costruire la mesh definitiva si è utilizzato l'algoritmo "COOPER", estraendo i volumi a partire dalle celle della sezione di ingresso. L'altezza di ogni volume è stata determinata attraverso la suddivisione di un segmento rettilineo posizionato sulla superficie laterale del condotto in direzione assiale. Per quanto riguarda il condotto a monte della stenosi questo è stato diviso assialmente in 80 sezioni caratterizzate da un rapporto last-first pari a 1,7. Questa scelta è dovuta al fatto che non si nutre molto interesse su quanto accada molto a monte della stenosi, ma la necessità di avere un profilo di velocità sviluppato ha reso necessario creare tale geometria.



La parte della stenosi è quella su cui si concentra l'attenzione e quindi si è scelto di avere una mesh più fitta. La parte immediatamente a monte della sezione minima del condotto è stata discretizzata con la creazione di 30 sezioni caratterizzate da un rapporto first-last pari a 1,7; in questo modo si è andati ad affinare la griglia progressivamente avvicinandosi alla zona di maggiore interesse.

La parte direttamente a valle della sezione minima è stata suddivisa in sezioni di spessore uguale e dimensione pari a 0,15mm, ovvero la dimensione della sezione immediatamente

precedente al massimo restringimento. La scelta di avere una griglia più fine a monte del restringimento è dovuta al fatto che ci si attendono maggiori disturbi al flusso in questa parte della stenosi. Il condotto cilindrico a valle della stenosi è stato invece discretizzato con sezioni di spessore da 0,2mm a 0,4mm (last-first ratio pari a 2). In questo modo la parte vicina alla stenosi mantiene comunque una sufficiente raffinatezza di griglia, sufficiente a cogliere eventuali effetti che si propagano a valle della stenosi, quali ad esempio getti. In totale quindi la mesh risulta composta da 161.821 volumi. Vengono definite tre superfici di confine del dominio:

- inflow: interfaccia di ingresso del flusso
- outflow: interfaccia di uscita del flusso
- wall: parete del vaso.

6.2. Modello fluidodinamico adottato

Per quanto riguarda le simulazioni fluidodinamiche descritte nel seguito, sono state adottate le seguenti ipotesi generali:

- fluido incomprimibile ($\rho = 1060 \text{ kg/m}^3$);
- fluido Newtoniano ($\mu = 0.0035 \text{ kg/ms}$);
- pareti rigide;
- condizioni di non scorrimento alla parete;
- regime di moto laminare;
- problema isoterma;
- assenza di effetti gravitazionali.

Viene calcolato il numero di Reynolds nella sezione dove si prevede possa essere massimo, al fine di trovare conferma del fatto che il flusso possa essere laminare.

Una ulteriore verifica preliminare verrà eseguita analizzando le curve di portata generate dal solo modello 0D.

Utilizziamo come sezione per il calcolo la sezione di massima stenosi posta alla coordinata $x=0$ e orientata in direzione dell'asse x . Risulta quindi:

$$\text{Re} = \frac{\rho D U}{\mu}$$

dove:

ρ : massa volumica;

D : diametro del condotto al massimo restringimento (1.2mm);

U : velocità media del fluido nella sezione;

μ : viscosità del fluido.

Consideriamo una velocità media sulla stenosi, calcolata al picco massimo di flusso pari a:

$$U = \frac{U_{in} \cdot S_{in}}{S}$$

dove:

U_{in} : velocità al picco del flusso in ingresso al condotto (0.23 m/s);

S_{in} : sezione di ingresso del condotto;

S : sezione della stenosi a $x=0$.

Il numero di Reynolds risulta $Re = 209$;

Considerando la direzione del flusso (x) il numero di punti di discretizzazione necessario risulta circa 54 in una distanza pari a $D = 1.2$ mm. In condizioni isotrope questo significherebbe che per una geometria relativamente semplice come quella trattata servirebbero in prima approssimazione più di 7 milioni di celle per applicare il metodo DNS e descrivere gli effetti di turbolenza qualora ci fossero.

a) Implementazione del modello in OpenFOAM

L'implementazione del modello in OpenFOAM implica la scrittura di tutti i file che definiscono i parametri, il risolutore e la mesh del case di cui si deve effettuare la simulazione come già descritto nel paragrafo 3.3.

La libreria OpenFOAM mette a disposizione dei case di esempio per ciascuno dei risolutori esistenti. E' buona norma quindi partire clonando uno di questi esempi e modificarlo in base alle esigenze del modello da simulare.

Le impostazioni legate ai risolutori vengono mantenute di default. Le parti che si andranno ad impostare sono le seguenti:

- **Risolutori (file `./system/fvSolution`)** Il solutore per la pressione è GAMG (Generalised geometric-algebraic multi-grid), mentre per la velocità è chiamato smoothSolver.
- **Mesh (cartella `./constants/polymesh`):** attraverso una utility presente in OpenFOAM (`gambitToFoam`) la mesh creata in precedenza attraverso il software GAMBIT può essere convertita per essere utilizzata con OpenFOAM. Analogamente

per la mesh creata con Salome7 (ideasUnvToFoam). Automaticamente vengono definite anche le superfici di confine (*boundary*).

- **Parametri trasporto (file `./constants/transportProperties`):** contiene in questo caso solo la definizione della viscosità cinematica: `nu [0 2 -1 0 0 0 0] 3.302e-6 ;`
- **Condizioni iniziali (files *U* e *p* nella cartella `./0`):** le condizioni iniziali in questo lavoro andranno scritte ad ogni timestep nella cartella corrispondente prelevandole opportunamente dal modello 0D.
- **Calcoli postprocessing (file `./system/controlDict`):** in questo lavoro i valori delle variabili calcolate ad ogni timestep dovranno essere scambiate con il modello 0D. A tal fine nel file *controlDict* vengono introdotte due funzioni per il calcolo della portata e della pressione media alle interfacce “inflow” e “outflow”.
- **Impostazione della simulazione (file `./system/controlDict`):** la durata dei time step è determinata da uno script esterno. Il formato con cui verranno scritti i risultati è scelto ASCII in modo che sia facilmente leggibile da uno script esterno.

massFlow

```
{
  type patchMassFlow;
  functionObjectLibs
  (
    "libsimpleFunctionObjects.so"
  );
  verbose true;
  patches
  (
    inflow
    outflow
  );
  factor 1;
  outputControl timeStep;
  outputInterval 1;
}
```

pressureAtInterface

```
{
  type patchAverage;
  functionObjectLibs
  (
    "libsimpleFunctionObjects.so"
  );
  verbose true;
  patches
  (
    inflow
    outflow
  );
  factor 1;
  fields
  (
    p
  );
  outputControl timeStep;
  outputInterval 1;
}
```


Capitolo 7:

Il problema accoppiato 0D/3D

In questo capitolo viene introdotto il problema accoppiato 0D e 3D e vengono mostrate due strategie di soluzione.

La buona posizione del problema accoppiato 3D/0D è stata dimostrata, nel caso di struttura indeformabile [14]. La suddivisione del problema nei due domini Ω_{3D} , Ω_{0D} può essere riformulata matematicamente grazie alla tecnica di decomposizione del dominio [15]. Se la soluzione esiste per il sistema accoppiato, allora può essere trovata risolvendo il problema separatamente nei due domini, e utilizzando la soluzione parziale ottenuta in ciascun dominio per calcolare le condizioni al bordo nell'altro, in una logica iterativa.

Il problema accoppiato va risolto secondo il seguente schema:

1. risolvere il problema fluidodinamico in Ω_{0D} con condizioni al contorno sull'interfaccia ricavate da una delle condizioni previste dal modello di interazione scelto e condizioni assorbenti sull'altro estremo del dominio.
2. risolvere il problema fluido-dinamico in Ω_{3D} con condizioni al contorno sull'interfaccia ricavate dalle rimanenti condizioni del modello di interazione e condizioni di Neumann o Dirichlet all'altro estremo del dominio.

Se il problema è matematicamente ben posto in ciascuno dei due domini, è possibile cercare la soluzione globale come l'insieme delle soluzioni dei due sotto-problemi, compatibili con le condizioni di interfaccia.

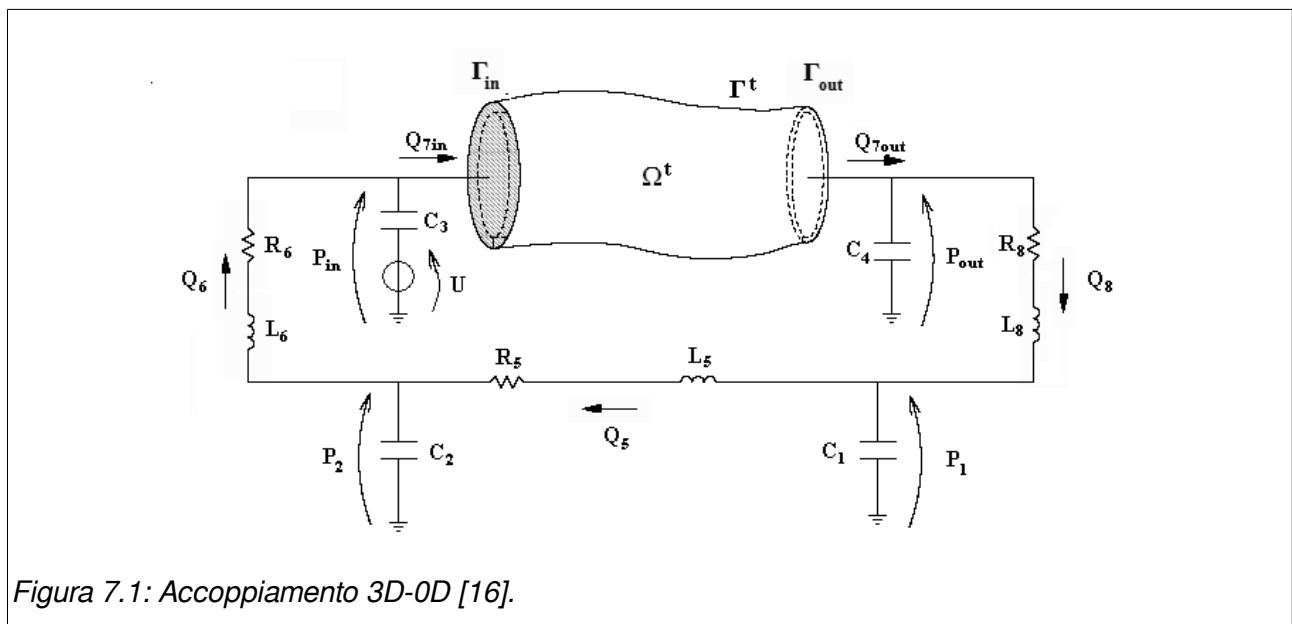


Figura 7.1: Accoppiamento 3D-0D [16].

E' necessario prestare particolare attenzione al trattamento dell'interfaccia tra i due modelli. Innanzitutto la connessione deve garantire la conservazione del flusso e del momento in corrispondenza delle interfacce e, come diretta conseguenza dell'analogia elettrica, le grandezze da monitorare sono pressione P e portata Q in corrispondenza delle interfacce. Nell'analogia con il circuito elettrico, queste relazioni di conservazione corrispondono all'applicazione delle leggi di Kirchhoff per i nodi (conservazione della corrente) e per le maglie (conservazione del potenziale).

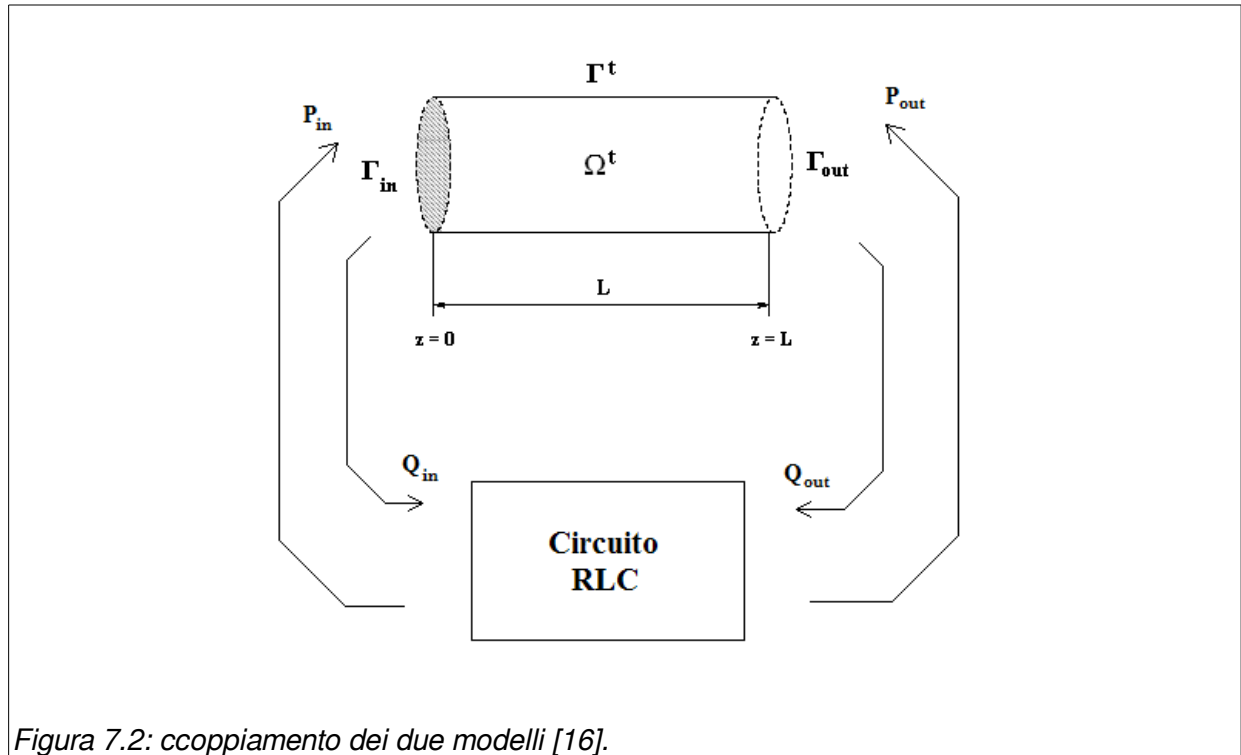
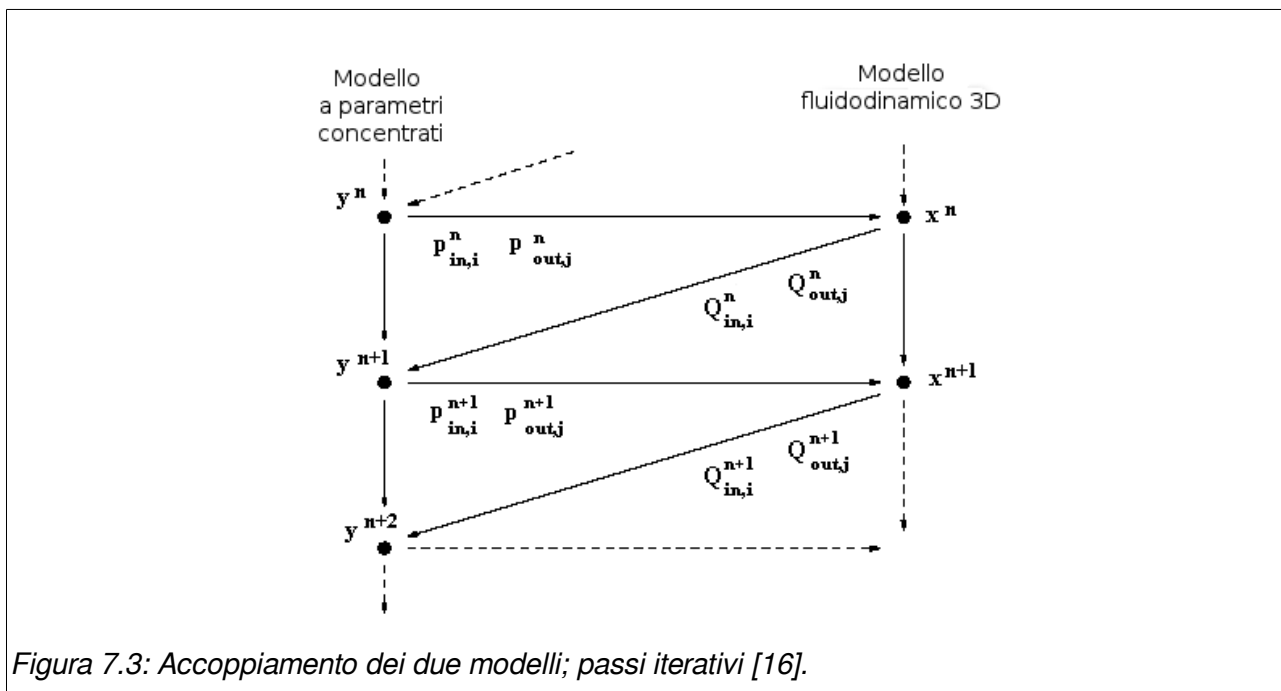


Figura 7.2: ccoppiamento dei due modelli [16].

Per risolvere numericamente il problema accoppiato, si è scelto un approccio iterativo basato sullo splitting del problema nei due sotto-problemi che lo costituiscono, il problema fluido-dinamico 3D ed il problema a parametri concentrati 0D.

Adottando un approccio iterativo, si devono risolvere, alternandoli, il problema fluido dinamico tridimensionale ed il modello a parametri concentrati, dove i rispettivi dati al bordo vengono forniti dall'iterazione precedente. Nel caso particolare in cui il modello a parametri concentrati viene risolto con il metodo di Eulero Esplicito, l'algoritmo si riduce alla soluzione del problema tridimensionale e della rete elettrica una sola volta ad ogni passo temporale.

Utilizzando un approccio multiscala si ottiene dunque una soluzione maggiormente accurata rispetto a quella calcolata con il semplice modello tridimensionale con condizioni al contorno assorbenti: inoltre tale guadagno in termini di accuratezza non comporta un incremento eccessivo del costo computazionale.



Vengono proposte due tipologie dei schema di accoppiamento: uno esplicito, senza sotto-iterazioni ad ogni passo temporale e un altro implicito che comprende delle sotto-iterazioni ad ogni passo temporale.

L'accoppiamento esplicito

Ad ogni passo temporale di simulazione (time step) t_n , l'accoppiamento esplicito, consiste nel passare al risolutore del modello 0D, la portata o la pressione calcolate dal modello 3D al passo precedente. Il risolutore del modello 0D, a sua volta, calcola la pressione o la portata che devono poi essere passate al modello 3D. Fatto questo, è possibile passare al passo temporale successivo. A causa della condizione per la stabilità CFL (Courant-Friedrichs-Lewy), il passo temporale della simulazione del risolutore 0D deve essere più piccolo di quello richiesto dal problema 3D. Quindi l'uso dello stesso time step sottopone entrambi i problemi all'applicazione della condizione CFL.

Tuttavia, siccome il problema 3D è più costoso in termini computazionali, questa strategia implica un inaccettabile aumento dei costi computazionali. Questo problema potrebbe essere affrontato considerando due diversi time step per ogni modello e avanzando con il modello col passo minore, tipicamente il modello 0D, finché non viene raggiunto il time step più elevato.

L'accoppiamento implicito

In modo da garantire la continuità della portata e della pressione totale all'interfaccia tra i due domini viene disposto un algoritmo di accoppiamento implicito basato su un punto fisso di iterazione. In questo caso l'accoppiamento avviene attraverso delle sotto-iterazioni ad ogni

time step attraverso una procedura di rilassamento. Questo è l'algoritmo che verrà implementato in questo lavoro di tesi.

Supponiamo di avere la soluzione del modello 3D e 0D ad ogni time step t_n . Quindi, attraverso un parametro di rilassamento $0 < \omega < 1$, l'accoppiamento implicito si formula come segue (pseudo-codice):

dati φ_{3D}^n e φ_{0D}^n si esegue:

I: $\varphi_{3D}^{(0)} = \varphi_{3D}^{(n)}$ e $\varphi_{0D}^{(0)} = \varphi_{0D}^{(n)}$;

II: for $k=1, \dots,$

1) set $\varphi_{0D}^{(k-1)} = \omega \varphi_{3D}^{(k-1)} + (1-\omega) \varphi_{0D}^{(k-1)}$

2) si risolve il problema 0D con i dati $\varphi_{0D}^{(k-1)}$.

3) trovata la soluzione 0D si risolve il problema 3D

4) verifica di convergenza:

- if : $\|\varphi_{3D}^{(k)} - \varphi_{0D}^{(k)}\| > tol$ si ritorna al passo 1

- else: si utilizza l'ultima soluzione trovata e si passa al time step successivo.

tol Rappresenta il valore di tolleranza accettato.

Ad ogni sotto-iterazione dell'accoppiamento implicito, gli unici dati aggiornati sono quelli delle condizioni al bordo del modello 0D sull'interfaccia. La soluzione completa è aggiornata alla fine di ogni time step.

Capitolo 8:

Implementazione software di gestione della simulazione CoSimO: COronary SIMulation Organizer

In questo capitolo viene descritta la strategia software di accoppiamento e gli algoritmi risolutivi della problema accoppiato 0D e 3D.

Viene introdotto il funzionamento del software CoSimO per la gestione della simulazione del sistema accoppiato

L'accoppiamento di un modello 0D e di un modello 3D è spesso realizzato a livello di codice all'interno di un unico tool di simulazione. Solitamente le parti che costituiscono questo codice sono difficilmente riutilizzabili come componenti di partenza per simulazioni anche se si discostano di poco. Ad esempio la simulazione di un altro tratto di coronaria o dello stesso vaso ma trattato con uno stent.

L'intento di questo lavoro è realizzare un sistema modulare che permetta l'interscambio dei dati tra un modello 0D e un modello 3D realizzati in modo indipendente con i tool OpenModelica e OpenFOAM.

Lo script cuore della procedura prende il nome di CoSimO: COronary SIMulation Organizer. Il nome deriva dal fatto che è stato concepito per la simulazione dei vasi appartenenti all'albero coronarico ma nulla vieta una sua estensione ad altre applicazioni.

Nel caso trattato il modello 3D è un vaso e quindi una struttura tubolare con un ingresso e un'uscita ma una estensione del sistema può portare all'accoppiamento di un modello 3D costituito da un unico o più ingressi e più uscite come ad esempio è il caso delle biforcazioni. Il linguaggio utilizzato per la programmazione dello script CoSimO è Python.

a) Introduzione a Python

Python è innanzitutto un linguaggio di script pseudocompilato. Questo significa che, similmente al Perl ed al Tcl/Tk, ogni programma sorgente deve essere pseudocompilato da un interprete. L'interprete è un normale programma che va installato sulla propria macchina, e si occuperà di interpretare il codice sorgente e di eseguirlo. Quindi, diversamente dal C++, non abbiamo un fase di compilazione - linking che trasforma il sorgente in eseguibile, ma avremo a disposizione solo il sorgente che viene eseguito dall'interprete.

Il principale vantaggio di questo sistema è la portabilità: lo stesso programma potrà essere eseguito su una piattaforma Linux, Mac o Windows purché vi sia installato l'interprete.

Python è un linguaggio orientato agli oggetti. Supporta le classi, l'ereditarietà e si caratterizza per il binding dinamico. Ragionando in termini di C++ si può dire che tutte le funzioni sono virtuali. La memoria viene gestita automaticamente e non esistono specifici costruttori o distruttori; inoltre esistono diversi costrutti per la gestione delle eccezioni.

Python possiede una struttura aperta del linguaggio, priva di dichiarazioni ridondanti e estremamente simile ad un linguaggio parlato. L'indentazione perde il suo ruolo inteso come stile di buona programmazione per facilitare la lettura del codice, per diventare parte integrante della programmazione che consente di suddividere il codice in blocchi logici [17].

b) Le librerie OMPython e PyFoam

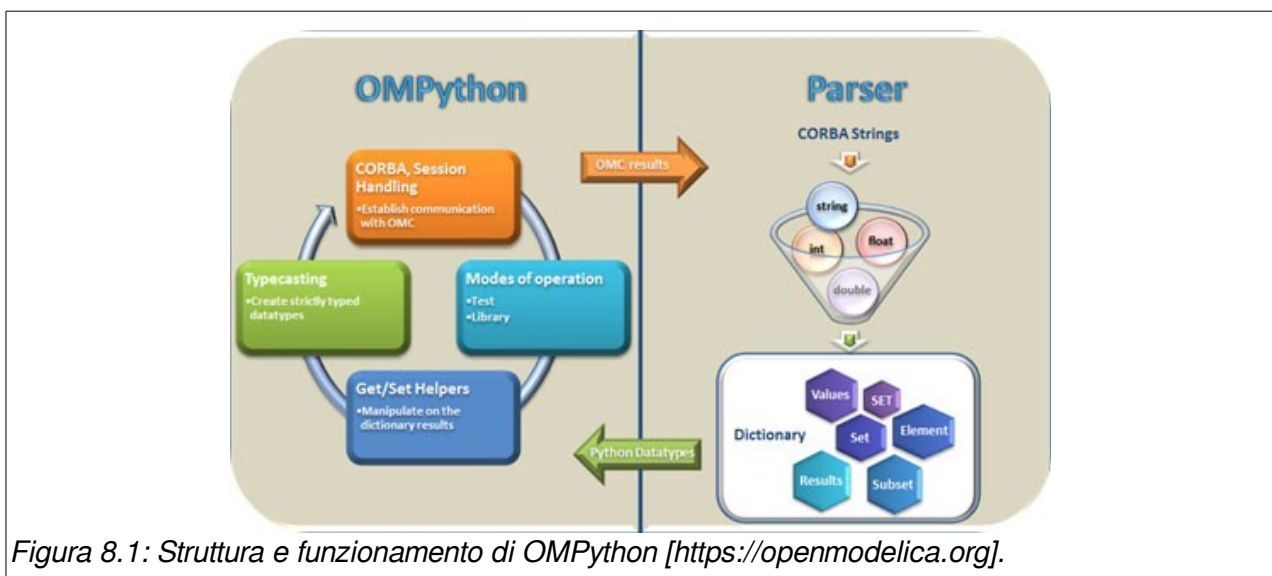
Le due piattaforme utilizzate OpenModelica e OpenFOAM hanno ciascuna un loro utilizzo preferenziale: OpenModelica possiede una serie di utilities e un tool grafico OMEdit, mentre OpenFOAM è utilizzata principalmente tramite riga di comando.

Esistono però, vista la natura Open Source di OpenModelica e OpenFOAM, delle librerie in grado di svolgere funzioni e manipolare i dati.

PyFOAM è una libreria Python in grado di controllare i processi di simulazione e di manipolare i dati di OpenFOAM, in particolare:

- analizzare i log prodotti da solvers di OpenFOAM;
- eseguire i solver e le utilities di OpenFOAM e, simultaneamente, analizzare il loro risultato;
- manipolare i file contenenti i parametri e le condizioni iniziali in maniera non distruttiva;
- stampare a video i residuals calcolati dai solvers durante la simulazione.

OMPython (OpenModelica Python Interface) è anch'essa una libreria Python capace di controllare i processi di simulazione di OpenModelica.



c) Funzionamento di CoSimO

CoSimO sfrutta le librerie sopracitate (PyFOAM e OMPython) per lo scambio di dati con i tools OpenFOAM e OpenModelica.

L'algorithmo utilizzato per l'accoppiamento dei due modelli è di tipo implicito simile a quello mostrato nel capitolo 7.

L'algorithmo è schematizzato in Figura 8.2:

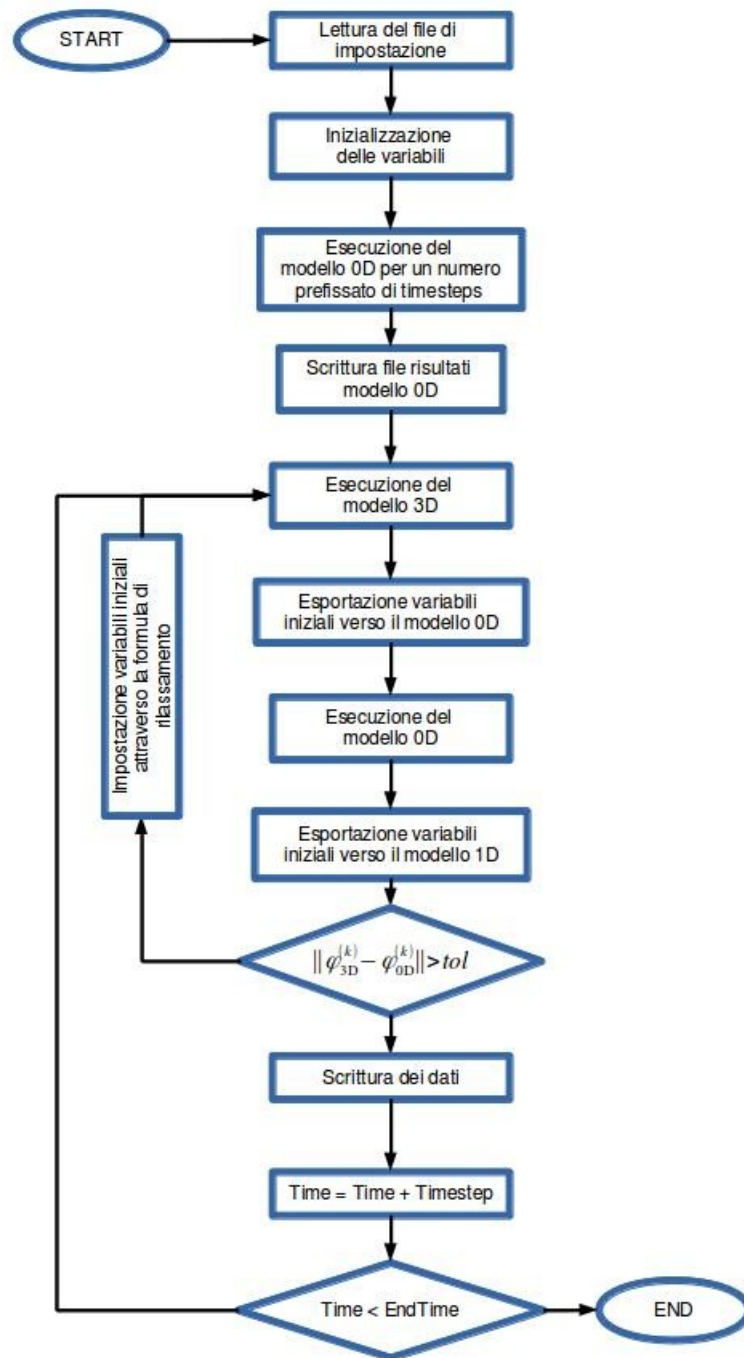


Figura 8.2: Schema di funzionamento dell' algoritmo base del software CoSimO

La maggiore differenza, tra l'algoritmo di accoppiamento implicito descritto in precedenza al capitolo 7 e quello effettivamente utilizzato all'interno delle simulazioni, consiste nel blocco di ritorno all'interno dell'iterazione del singolo time step.

All'interno della formula del rilassamento utilizzata si inserisce la variabile $\Delta^{(k)}$:

$$\Delta^{(k)} = \varphi_{1D}^{(k)} - \varphi_{3D}^{(k)}$$

L'algoritmo iterativo al singolo time step diventa (pseudo-codice):

dati φ_{3D}^n e φ_{0D}^n si esegue:

I: $\varphi_{3D}^{(0)} = \varphi_{3D}^{(n)}$, $\varphi_{0D}^{(0)} = \varphi_{0D}^{(n)}$ e $\omega = 0$;

II: for $k=1, \dots$,

1) set $\varphi_{0D}^{(k-1)} = \varphi_{0D}^{(k-1)} + \omega \Delta^{(k-1)}$

2) si risolve il problema 0D con i dati $\varphi_{0D}^{(k-1)}$.

3) trovata la soluzione 0D si risolve il problema 3D

4) verifica di convergenza:

- if : $\|\varphi_{3D}^{(k)} - \varphi_{0D}^{(k)}\| > tol$ si ritorna al passo 1 con

$$\Delta^{(k)} = \varphi_{3D}^{(k)} - \varphi_{0D}^{(k)}$$

- else: si utilizza l'ultima soluzione trovata e si passa al time step successivo.

d) Blocco di interscambio dati tra modello 0D e CoSimO

CoSimO inoltre si occupa dello scambio dei dati tra i vari ambienti e della gestione della tempistica della simulazione eseguendo a dovere i risolutori ad ogni timestep.

All'interno del modello 0D dell'albero coronarico, viene introdotto un blocco che permette di scollegare il vaso interessato alla simulazione 3D e collegare la relativa interfaccia verso CoSimO.

Il modello proposto in Figura 5.4 diventa come quello mostrato in Figura 8.3.

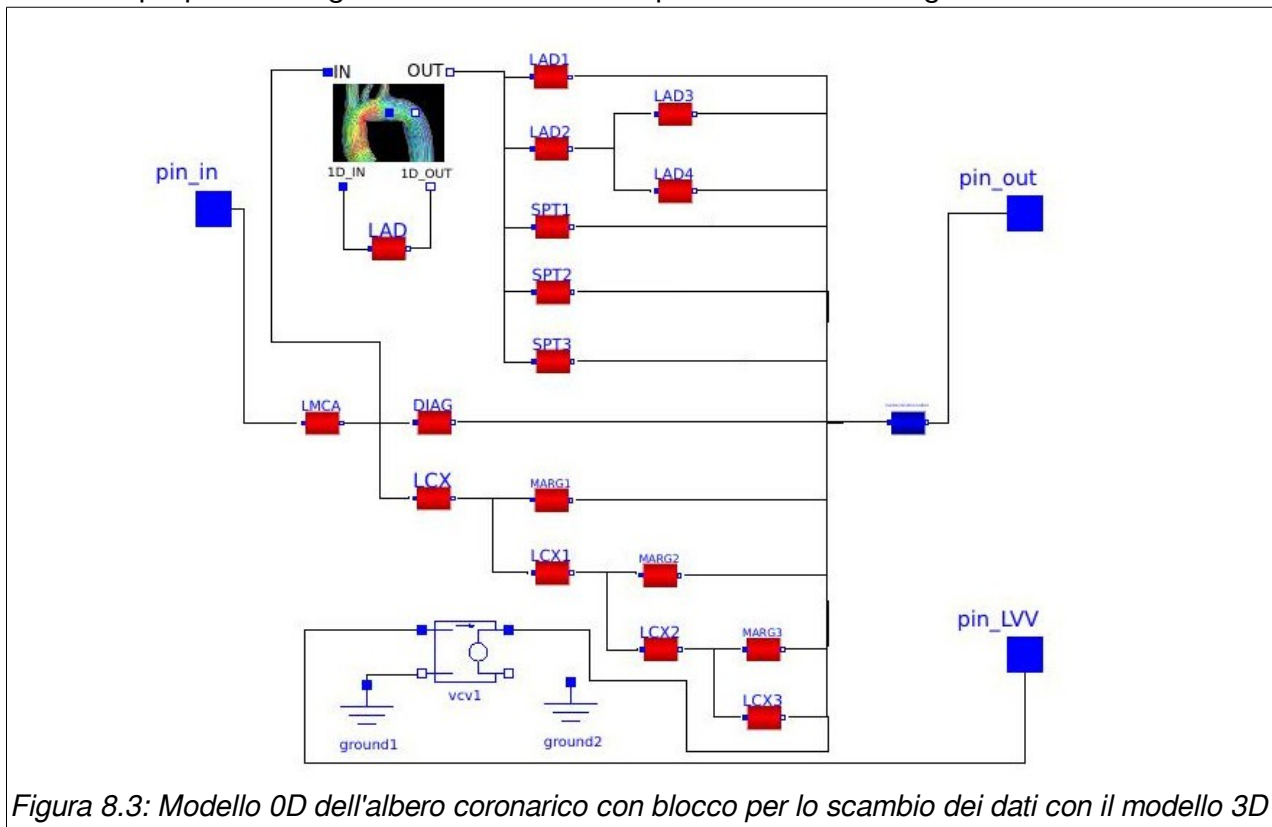


Figura 8.3: Modello 0D dell'albero coronarico con blocco per lo scambio dei dati con il modello 3D

Al suo interno il blocco di interscambio dati contiene un sistema di switch che, attivati dallo script CoSimO, abilitano lo scambio dei dati con lo script stesso, piuttosto che utilizzare il vaso a parametri concentrati.

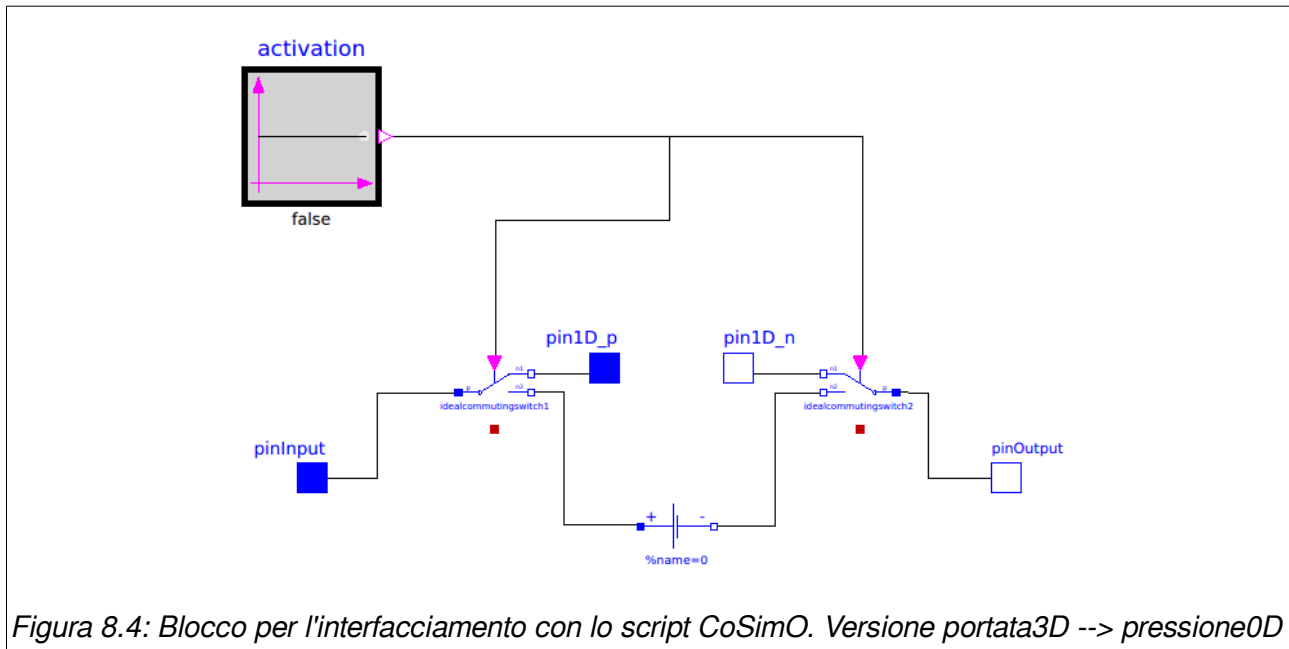


Figura 8.4: Blocco per l'interfacciamento con lo script CoSimO. Versione portata3D --> pressione0D

I blocchi di interscambio devono essere adeguati alle variabili scambiate tra il modello 0D e 3D all'interfaccia.

In questo lavoro ne vengono proposti di due tipologie:

1. Il primo tipo di blocco di interscambio è quello mostrato in Figura 8.4 chiamato per convenzione "*portata3D --> pressione0D*".

In questo caso, durante la singola iterazione, viene imposta una portata in ingresso e una pressione in uscita al dominio 3D.

Dopo il passo di calcolo il modello 3D ritorna a CoSimO la differenza di pressione ingresso-uscita calcolata. Tale valore va imposto al modello 0D.

2. Il secondo tipo di blocco di interscambio è quello mostrato in Figura 8.5, chiamato per convenzione "*pressione3D --> portata0D*" ed è il duale del precedente.

In questo caso, durante la singola iterazione, viene imposta una differenza di pressione al dominio 3D.

Dopo il passo di calcolo il modello 3D ritorna a CoSimO le portate in ingresso e uscita calcolate. Tale valore va imposto al modello 0D.

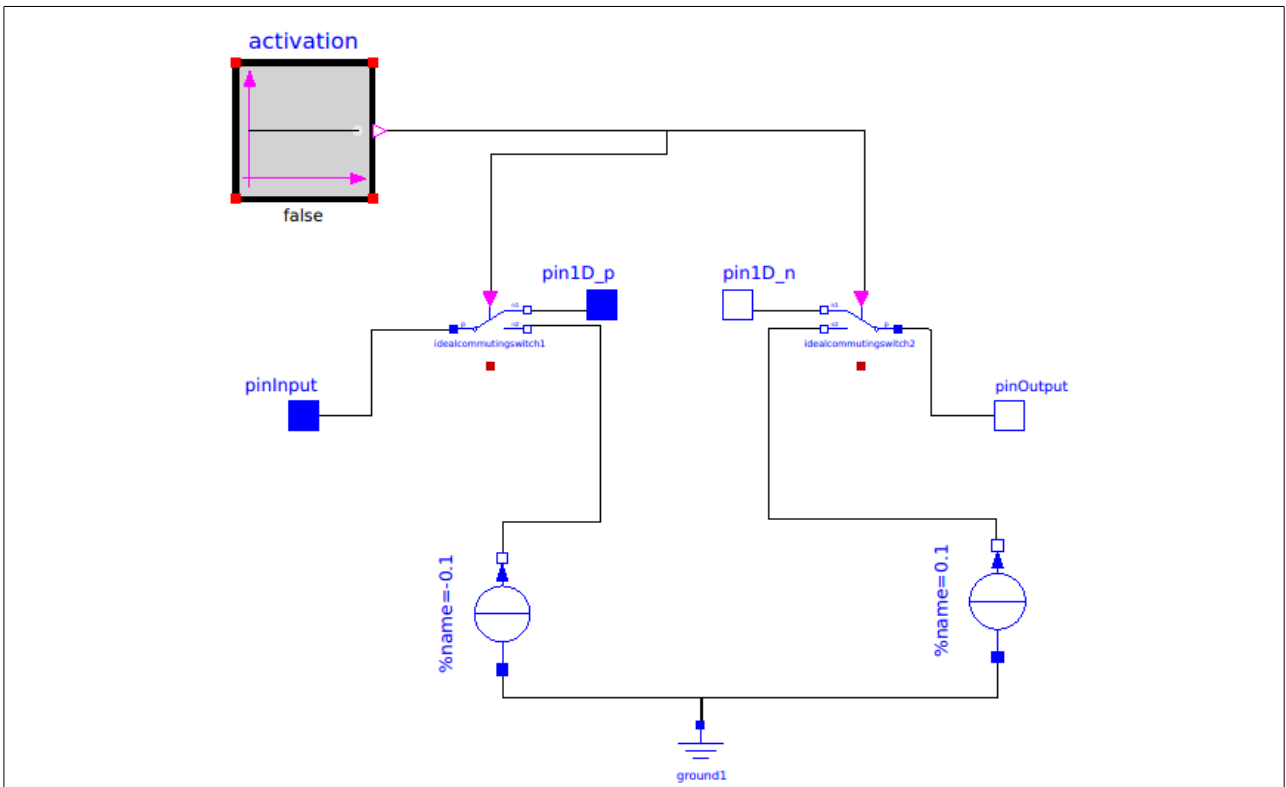


Figura 8.5: Blocco per l'interfacciamento con lo script CoSimO. Versione pressione3D --> portata0D

CoSimO gestisce anche le conversioni dimensionali tra i vari modelli utilizzando i coefficienti opportunamente imposti (anche con il giusto segno), di volta in volta, nel file di configurazione.

Questo meccanismo di interscambio, per come è stato concepito, può essere esteso a diverse tipologie di interfacciamento, a patto di comporre correttamente sia il blocco di interscambio all'interno di OpenModelica, che il file di configurazione di CoSimO.

Capitolo 9:

Risultati

In questo capitolo si illustrano i risultati delle simulazioni. Vengono presentati i risultati dell'esecuzione del solo modello 0D e, successivamente, dei modelli accoppiati 0D/3D.

Per entrambi vengono presentati due casi: coronaria fisiologica e coronaria stenotica.

I due principali obiettivi di questo lavoro di tesi sono stati:

- Realizzare un modello modulare 0D del sistema cardiovascolare che fosse attendibile e modulare. La modularità facilita implementazioni aggiuntive senza dover ridefinire quanto già implementato.
- Mettere a punto un sistema software in grado di accoppiare modelli 0D e 3D indipendenti tra loro, sfruttando codice open source.

Al fine di verificare gli obiettivi preposti sono state svolte simulazioni sia esclusivamente del modello 0D, sia dei modelli accoppiati 0D/3D per due casi studio.

Il **caso 1** è costituito dal sistema cardiovascolare, completo dell'albero coronarico, in condizioni fisiologiche attraverso i parametri ricavati dal lavoro di Pietrabissa [13] ed Ursino [10].

Il **caso 2** è costituito dal sistema cardiovascolare, completo dell'albero coronarico, in condizioni fisiologiche tranne che per il ramo discendente anteriore interventricolare della coronaria sinistra (LAD) per il quale è stato imposto un incremento della resistenza idraulica, al fine di simulare una condizione di stenosi.

9.1. Risultati modello 0D, caso 1

a) Impostazioni della simulazione

La simulazione è stata eseguita con i parametri indicati in Tabella 4. La resistenza idraulica è stata mantenuta come da parametri tabellati nel lavoro di R. Pietrabissa [13] :

$$R_{LAD} = 0.5 \frac{mmHg \cdot s}{ml}$$

Tabella 4: Parametri di simulazione test case 1, modello 0D

Parametri di simulazione con OpenModelica	
Istante iniziale:	0 s
Istante finale:	30 s
Passo di simulazione:	0.006 s
Algoritmo risolutore:	dassl
Tolleranza:	10e-6
Frequenza cardiaca:	72 bpm
Tempo di compilazione:	10s
Tempo esecuzione:	3s

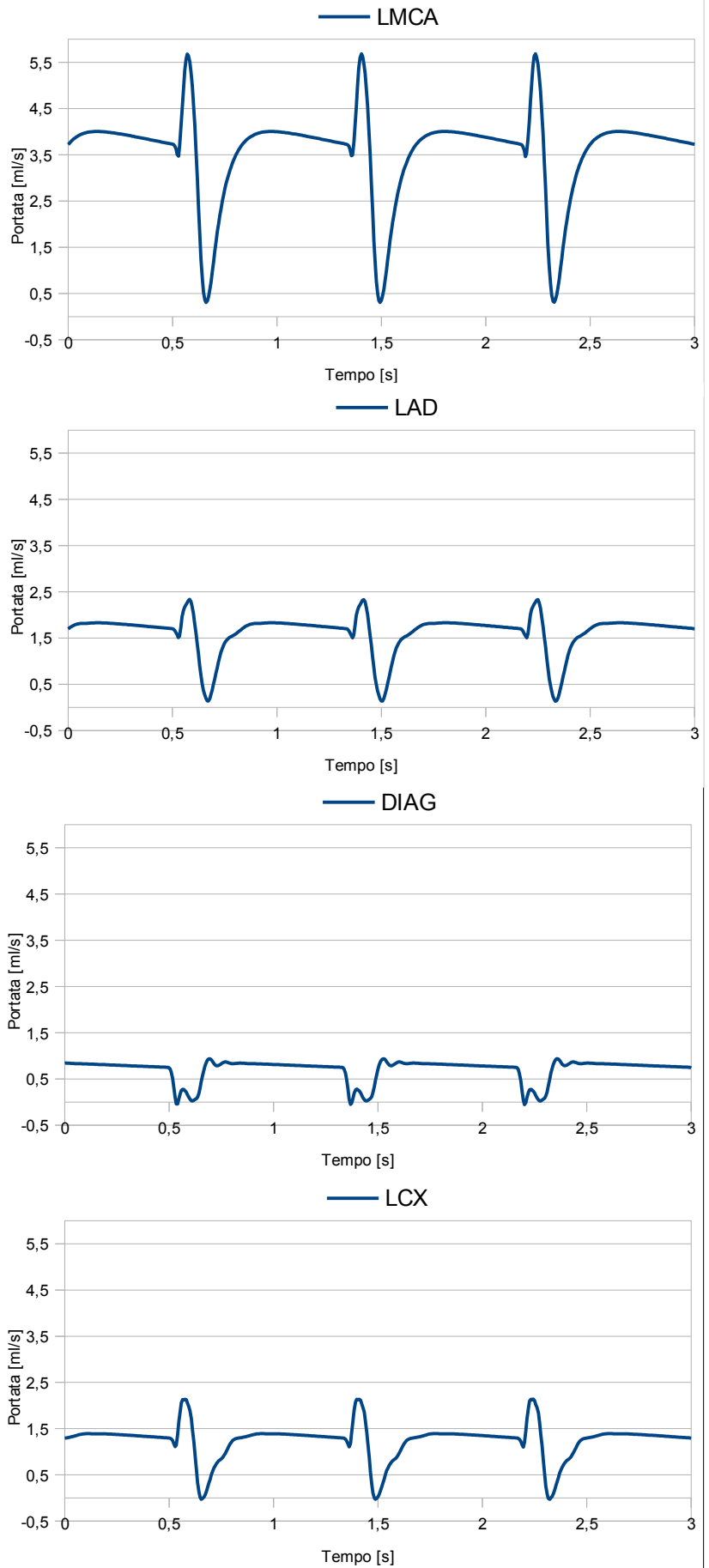
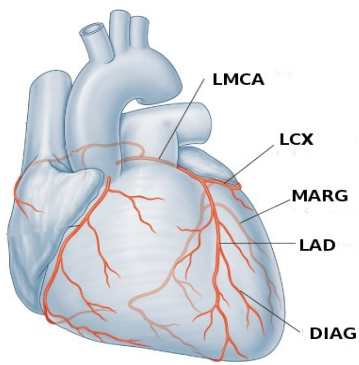


Figura 9.1: Rappresentazione degli andamenti temporali della portata nei vasi: LMCA, LAD, DIAG, LCX. Modello 0D, caso 1.

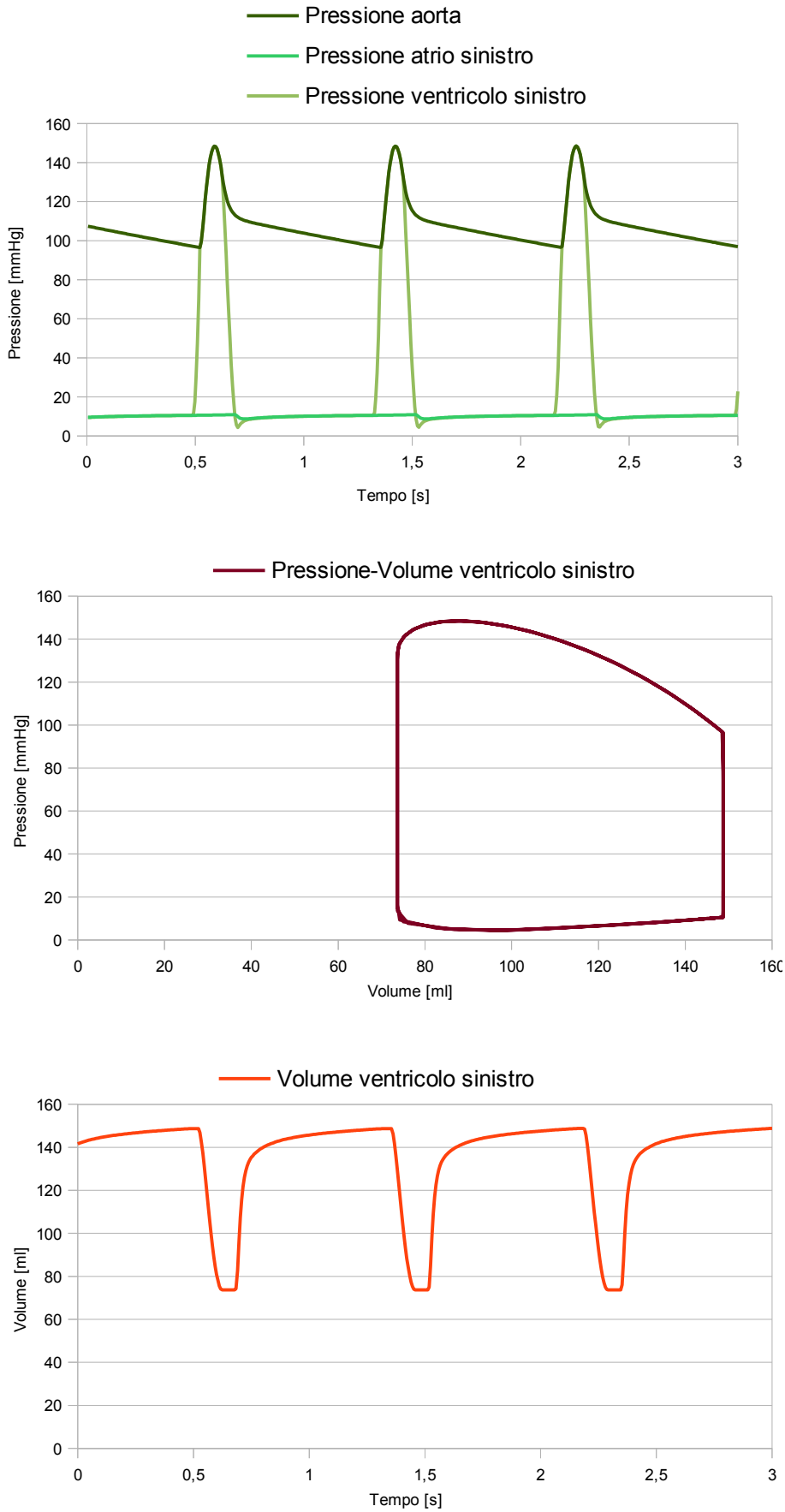
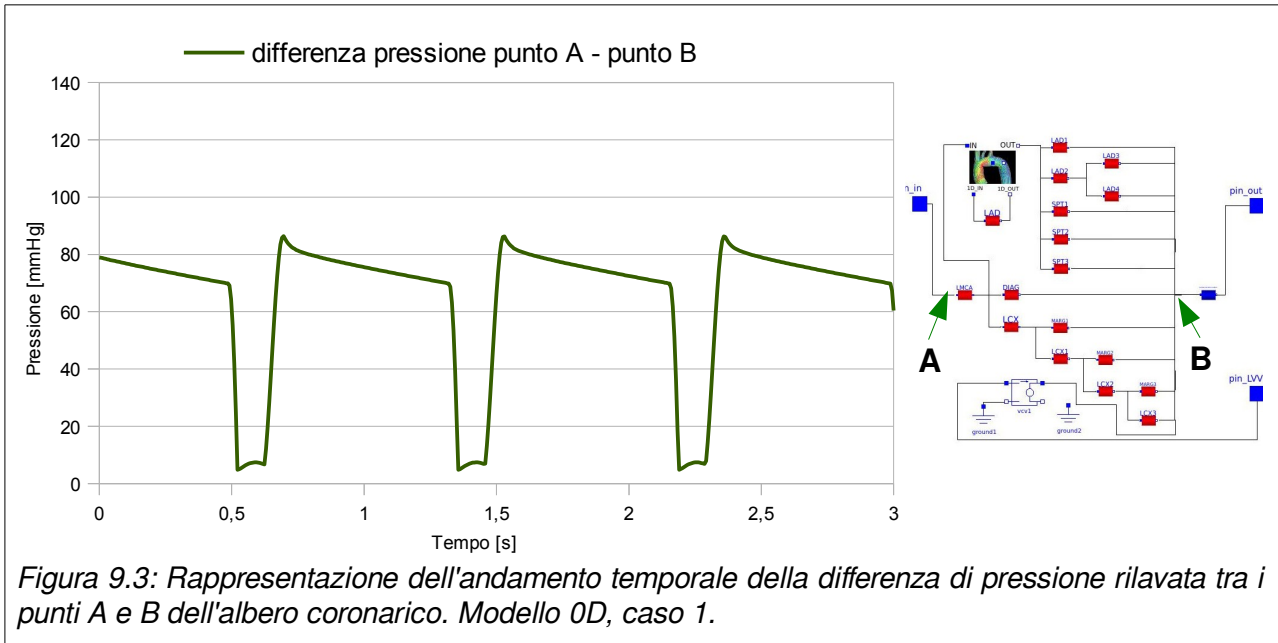


Figura 9.2: Rappresentazione degli andamenti temporali di: pressione ventricolare sinistra, pressione atriale sinistra, pressione aortica, volume ventricolare. Rappresentazione del ciclo pressione-volume ventricolare sinistro. Modello 0D, caso 1.



9.2. Risultati modello 0D, caso 2

La simulazione è stata eseguita con i parametri indicati in Tabella 5. La resistenza idraulica del vaso LAD è stata incrementata a:

$$R_{LAD} = 3.0 \frac{mmHg \cdot s}{ml}$$

Tabella 5: Parametri di simulazione test case 2, modello 0D

Parametri di simulazione con OpenModelica	
Istante iniziale:	0 s
Istante finale:	30 s
Passo di simulazione:	0.006 s
Algoritmo risolutore:	dassl
Tolleranza:	10e-6
Frequenza cardiaca:	72 bpm
Tempo di compilazione:	10s
Tempo esecuzione:	3s

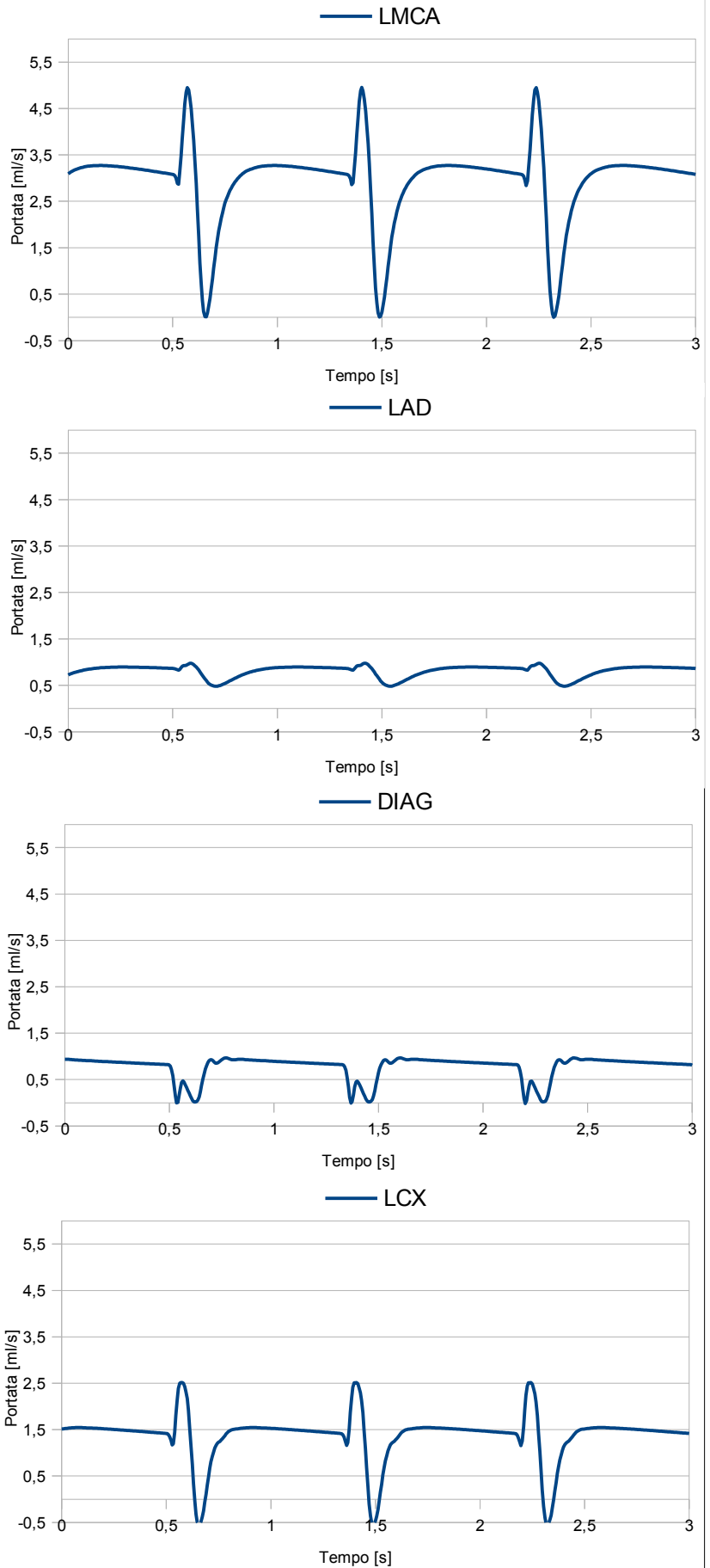
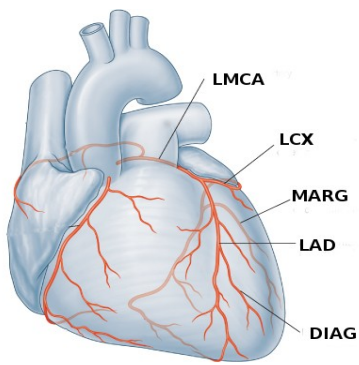


Figura 9.4: Rappresentazione degli andamenti temporali della portata nei vasi: LMCA, LAD, DIAG, LCX. Modello 0D, caso 2.

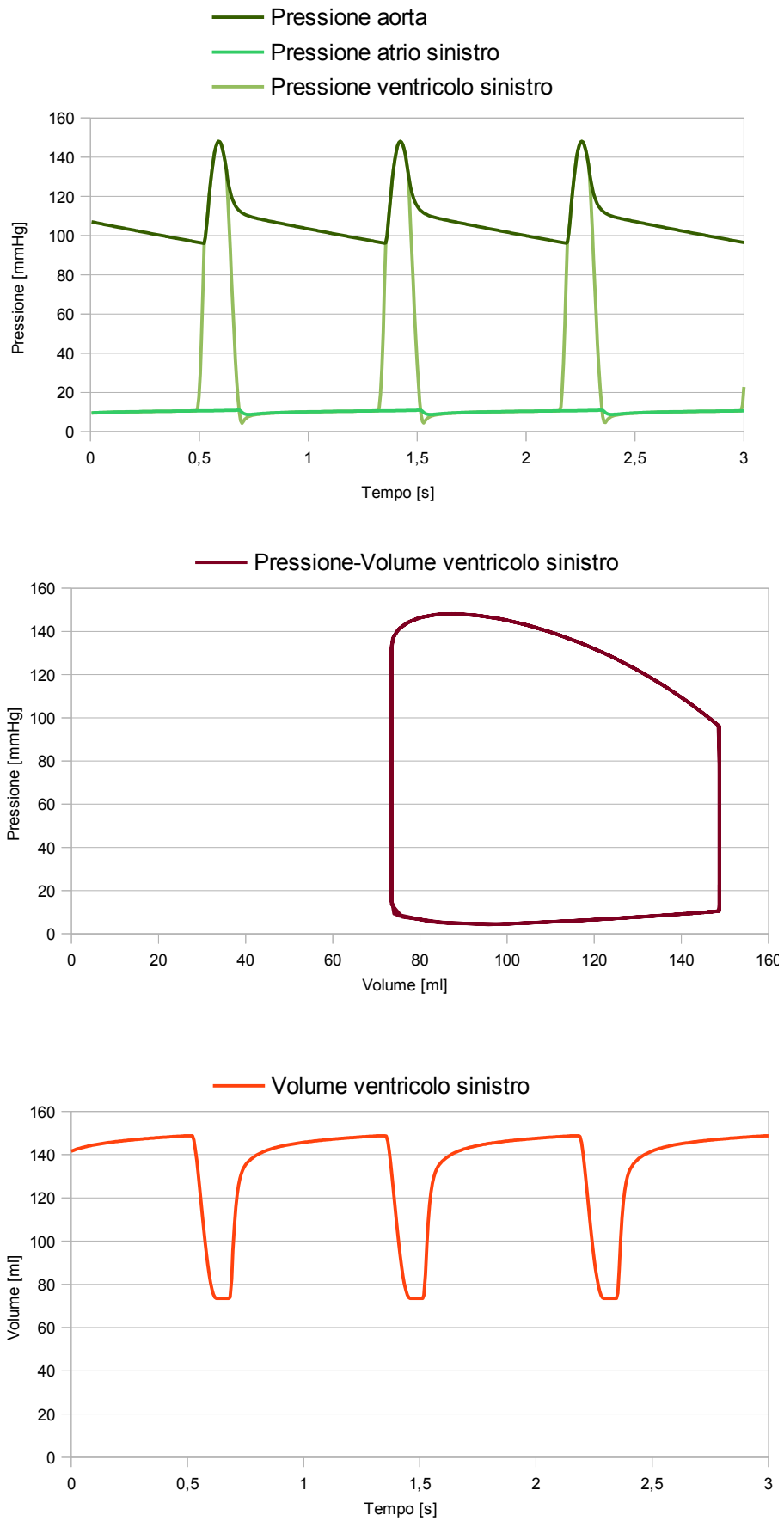


Figura 9.5: Rappresentazione degli andamenti temporali di: pressione ventricolare sinistra, pressione atriale sinistra, pressione aortica, volume ventricolare. Rappresentazione del ciclo pressione-volume ventricolare sinistro. Modello 0D, caso 2.

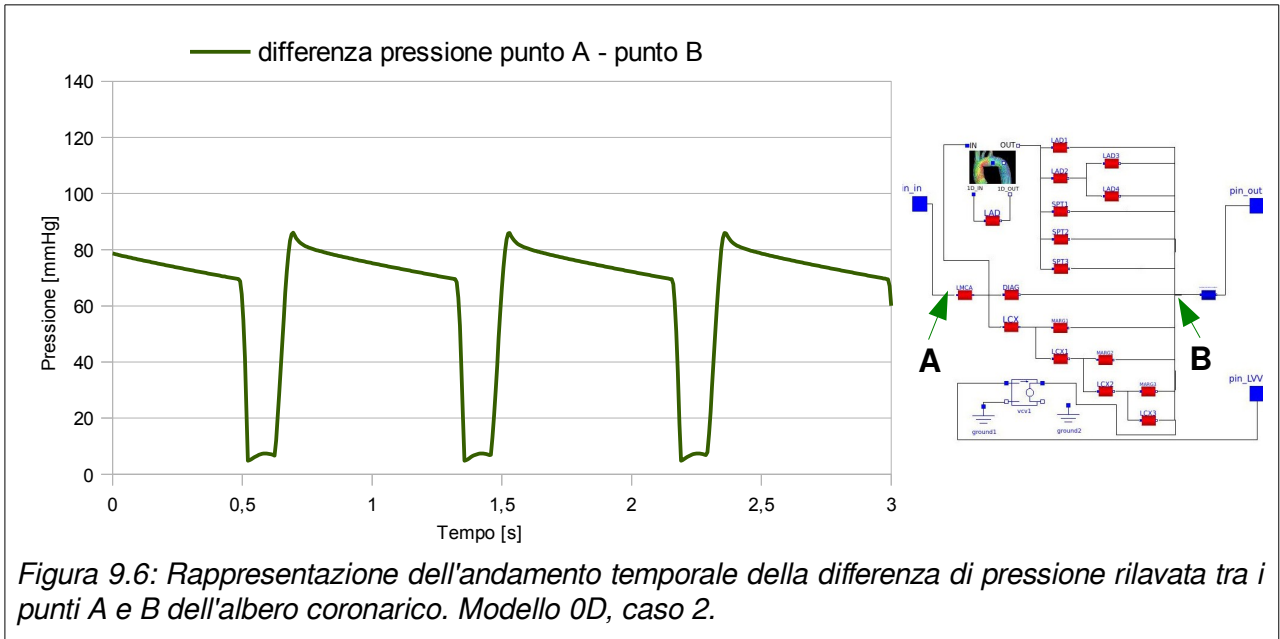


Figura 9.6: Rappresentazione dell'andamento temporale della differenza di pressione rilevata tra i punti A e B dell'albero coronarico. Modello 0D, caso 2.

9.3. Risultati modello accoppiato 0D/3D, caso 1

a) Impostazioni della simulazione

La simulazione è stata eseguita con i parametri indicati in Tabella 6. Il vaso LAD fisiologico è stato sostituito con il modello 3D di vaso fisiologico, descritto al paragrafo 6.1.

Le variabili imposte al sistema 3D sono state: Portata alla sezione inflow e pressione alla sezione outflow. Le variabili imposte al modello 1D sono state la differenza di pressione tra il pin_p e il pin_n del modulo di accoppiamento descritto al paragrafo 6.1.

Tabella 6: Parametri di simulazione test case 1, modello 0D/3D

Parametri di simulazione con CoSimO	
Istante iniziale:	0 s
Istante finale:	3 s
Passo di simulazione:	0.001 s
Risolutore modello 0D:	OpenModelica (dassl)
Risolutore modello 3D:	OpenFOAM
Tolleranza:	1e-3
Parametro rilassamento	0.3
Frequenza cardiaca:	72 bpm

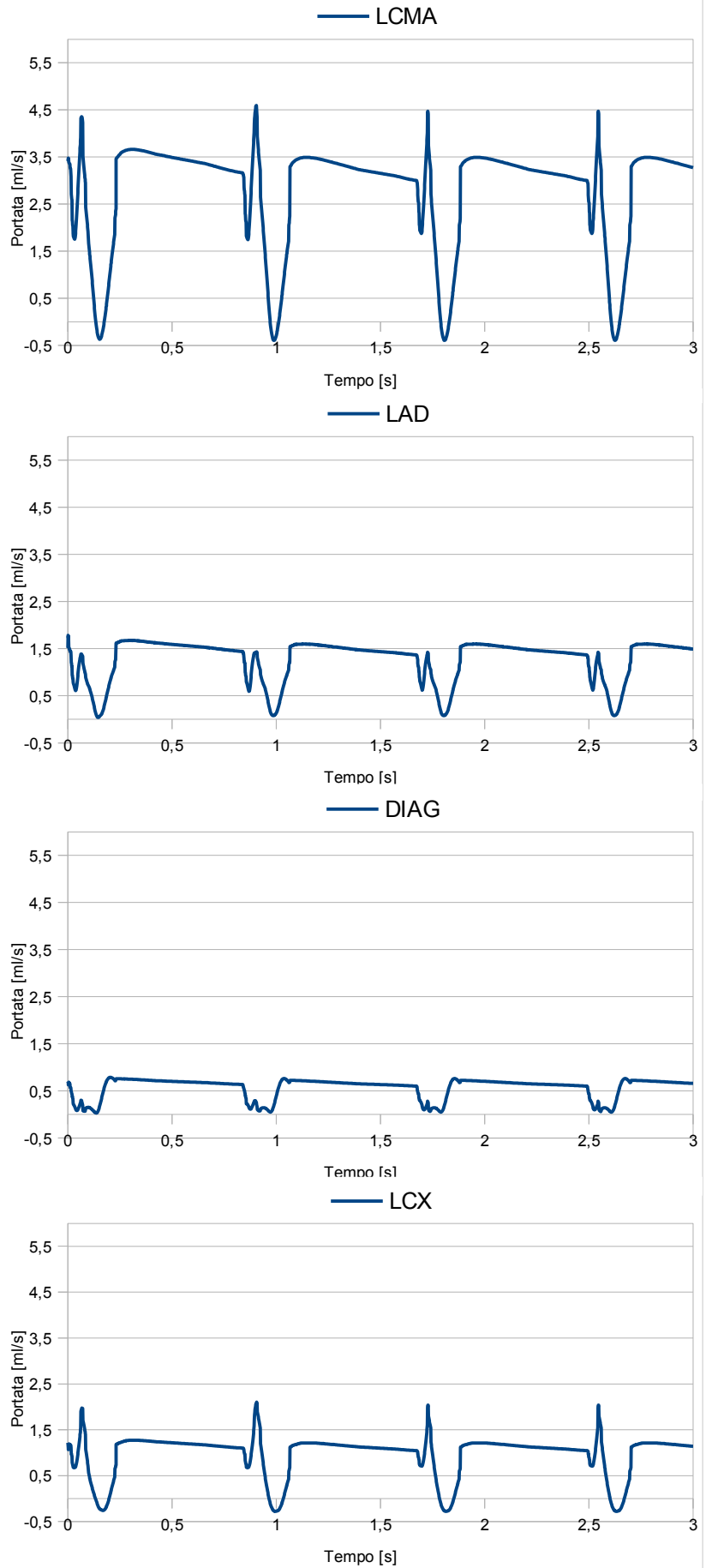
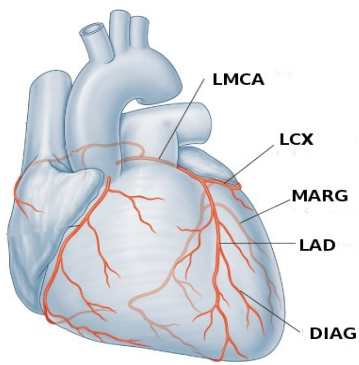


Figura 9.7: Rappresentazione degli andamenti temporali della portata nei vasi: LMCA, LAD, DIAG, LCX. Modello 0D/3D, caso 1.

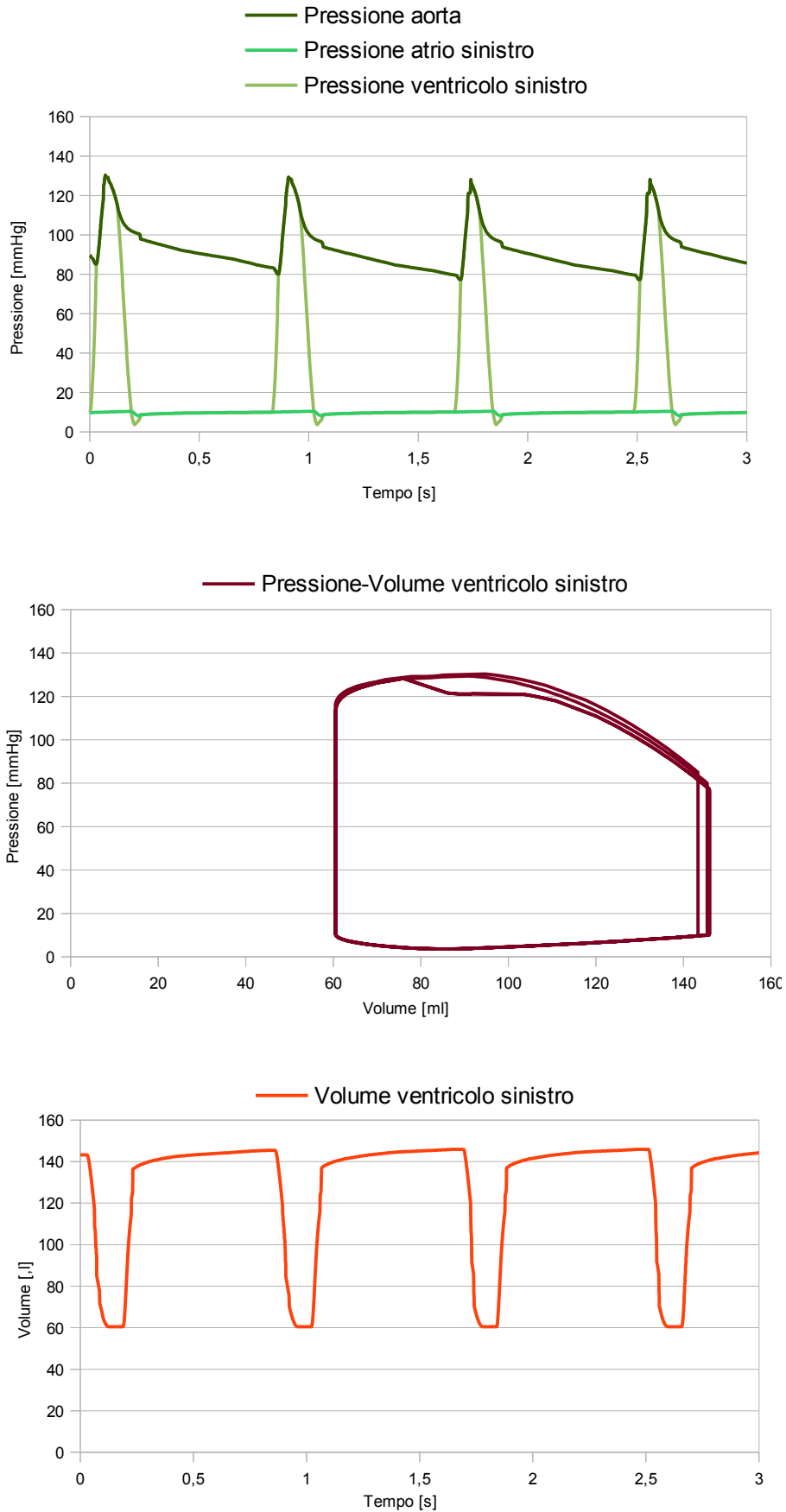


Figura 9.8: Rappresentazione degli andamenti temporali di: pressione ventricolare sinistra, pressione atriale sinistra, pressione aortica, volume ventricolare. Rappresentazione del ciclo pressione-volume ventricolare sinistro. Modello 0D/3D, caso 1.

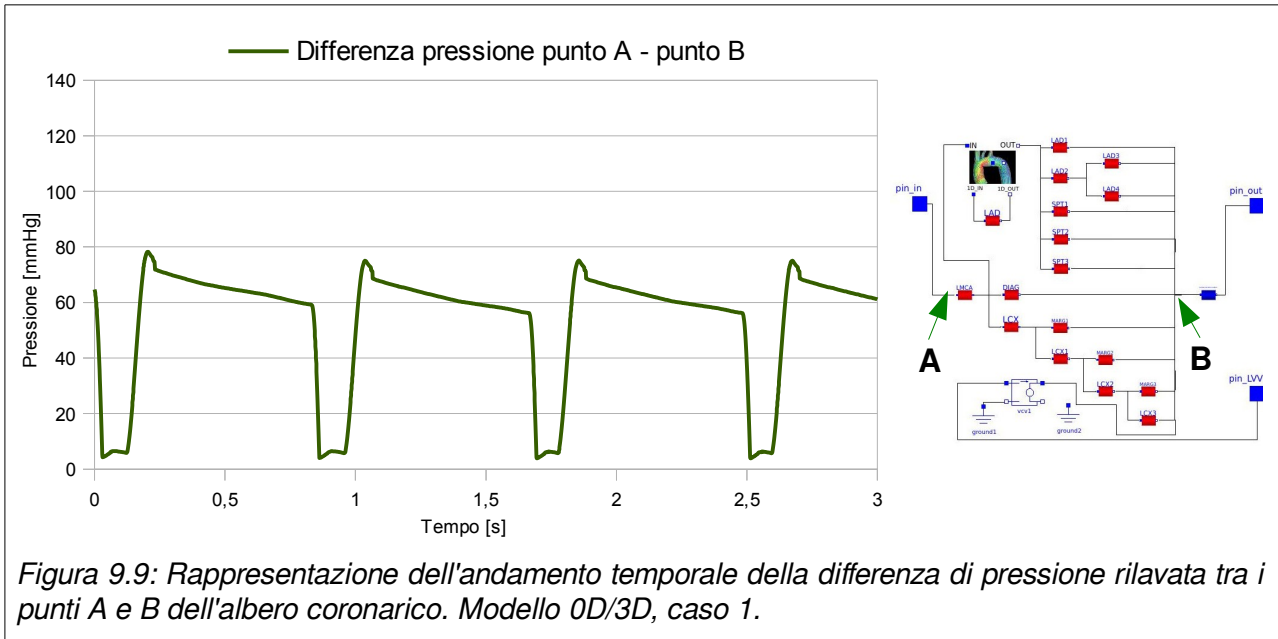


Figura 9.9: Rappresentazione dell'andamento temporale della differenza di pressione rilevata tra i punti A e B dell'albero coronarico. Modello 0D/3D, caso 1.

9.4. Risultati modello accoppiato 0D/3D, caso 2

a) Impostazioni della simulazione

La simulazione è stata eseguita con i parametri indicati in Tabella 7. Il vaso LAD fisiologico è stato sostituito con il modello 3D di vaso stenotico, descritto al paragrafo 6.1.

Le variabili imposte al sistema 3D sono state: Portata alla sezione inflow e pressione alla sezione outflow. Le variabili imposte al modello 1D sono state la differenza di pressione tra il pin_p e il pin_n del modulo di accoppiamento descritto al paragrafo 6.1.

Tabella 7: Parametri di simulazione test case 1, modello 0D/3D

Parametri di simulazione con CoSimO	
Istante iniziale:	0 s
Istante finale:	3 s
Passo di simulazione:	0.001 s
Risolutore modello 0D:	OpenModelica (dassl)
Risolutore modello 3D:	OpenFOAM
Tolleranza:	1e-3
Parametro rilassamento	0.3
Frequenza cardiaca:	72 bpm

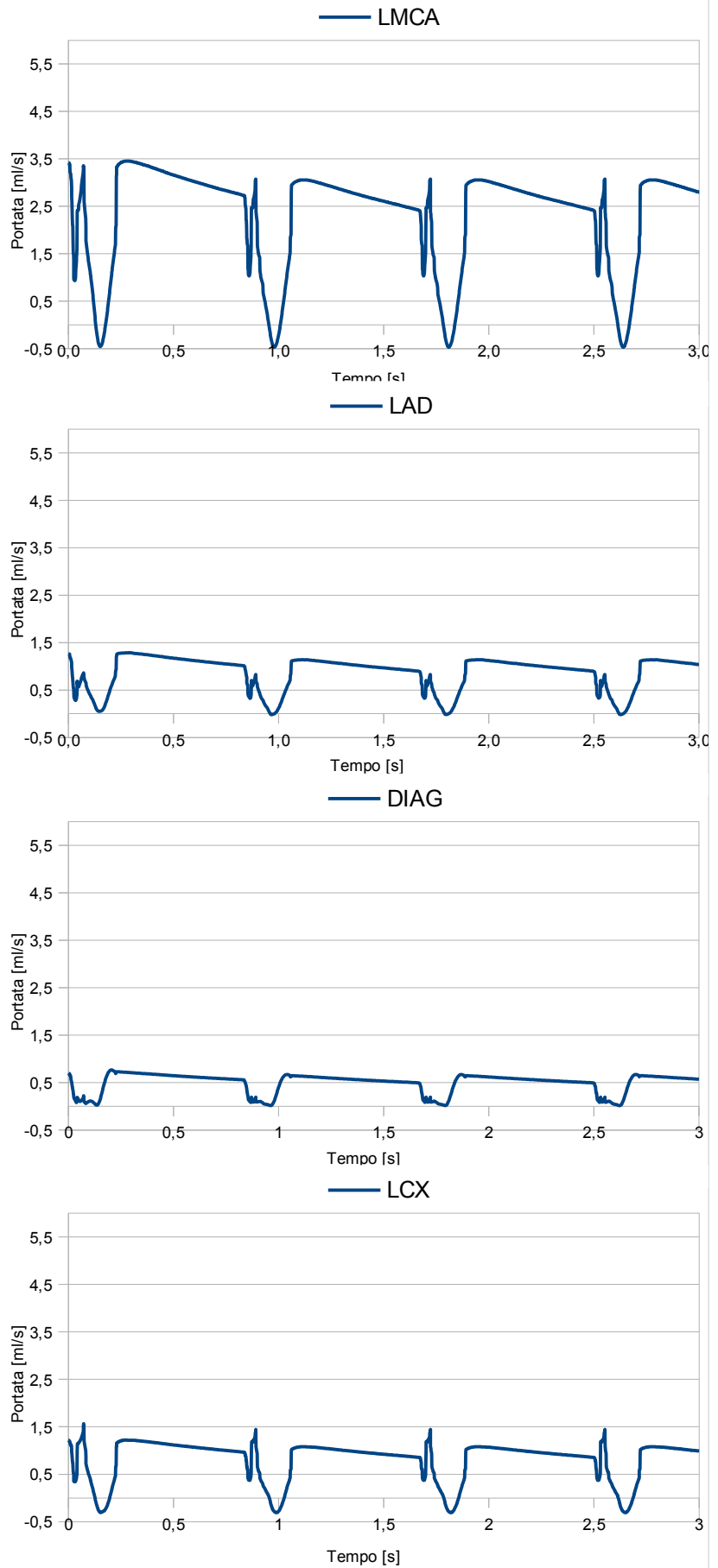
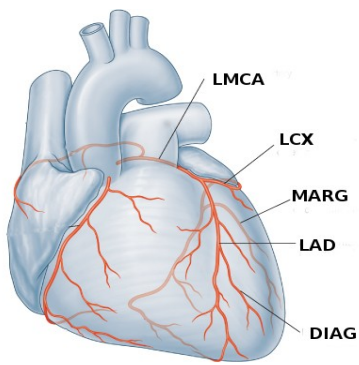


Figura 9.10: Rappresentazione degli andamenti temporali della portata nei vasi: LMCA, LAD, DIAG, LCX. Modello 0D/3D, caso 2.

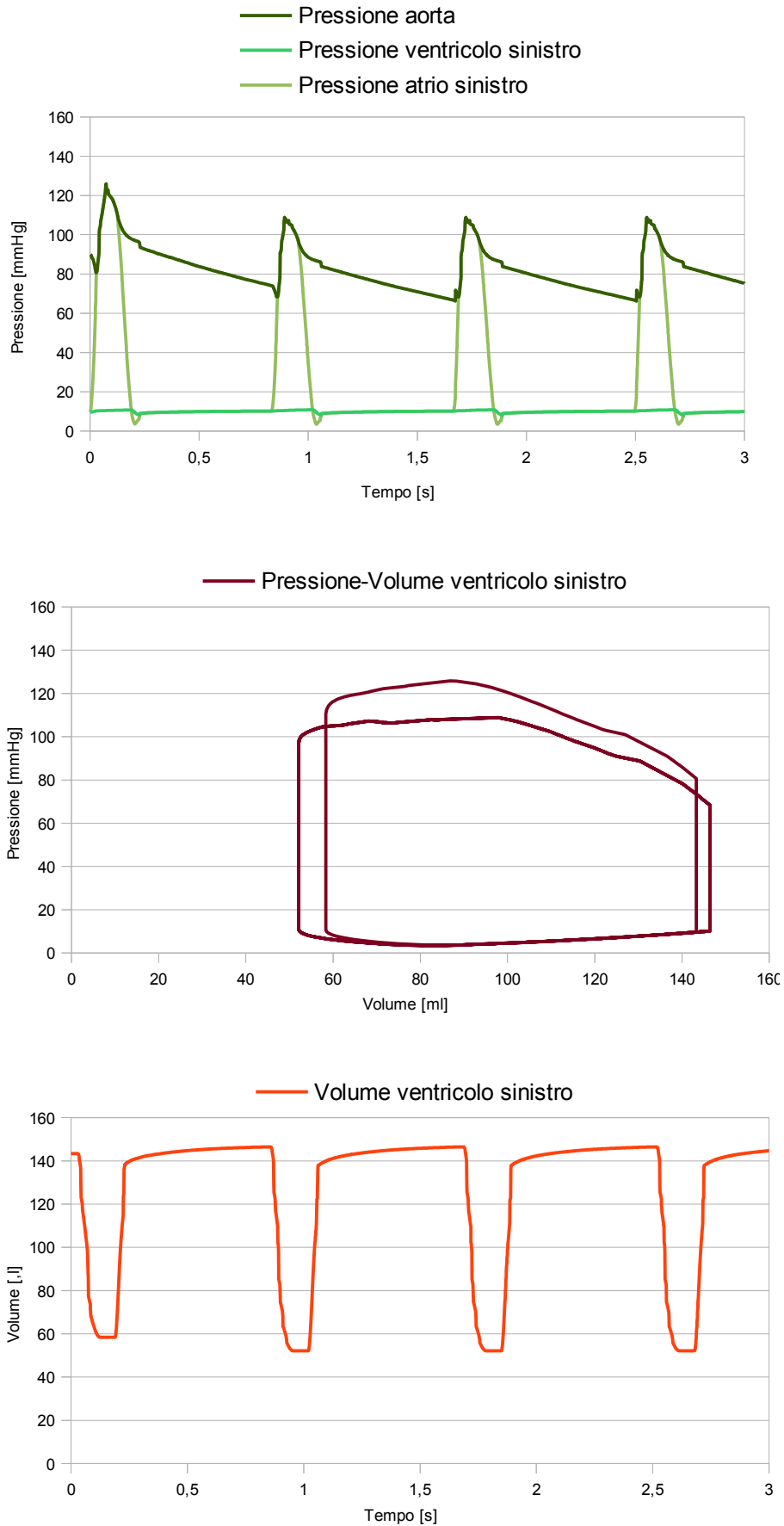


Figura 9.11: Rappresentazione degli andamenti temporali di: pressione ventricolare sinistra, pressione atriale sinistra, pressione aortica, volume ventricolare. Rappresentazione del ciclo pressione-volume ventricolare sinistro. Modello 0D/3D, caso 2.

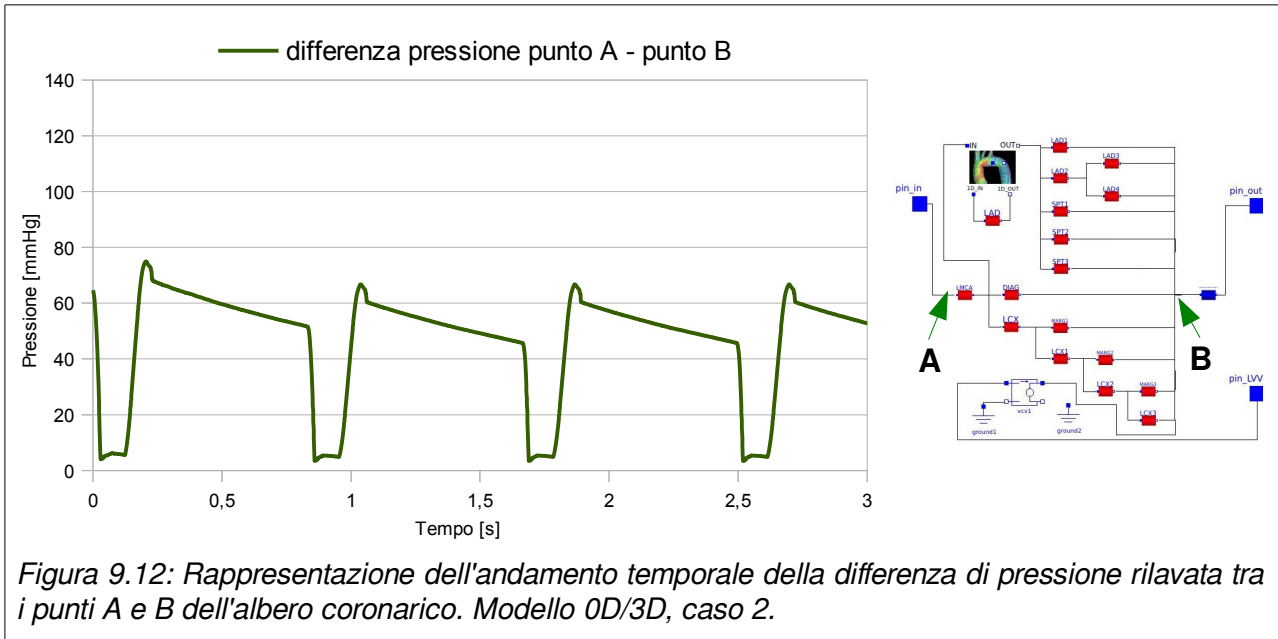


Figura 9.12: Rappresentazione dell'andamento temporale della differenza di pressione rilevata tra i punti A e B dell'albero coronarico. Modello 0D/3D, caso 2.

Per questo caso di studio, vengono mostrati, per completezza, attraverso l'utilizzo di mappe colorimetriche, anche gli andamenti di pressione (p), il modulo della velocità (U Magnitude) e lo sforzo tangenziale (shear stress) all'interno del vaso stenotico LAD (Figura 9.13, Figura 9.14 e Figura 9.15, Figura 9.16 con una scala colorimetrica diversa per rappresentare meglio l'andamento delle grandezze dove il vaso ha diametro maggiore).

Le grandezze di pressione e sforzo tangenziale sono espresse rapportate alla densità ρ :

$$p^* = \frac{p}{\rho} \quad \text{e} \quad \tau^* = \frac{\tau}{\rho}$$

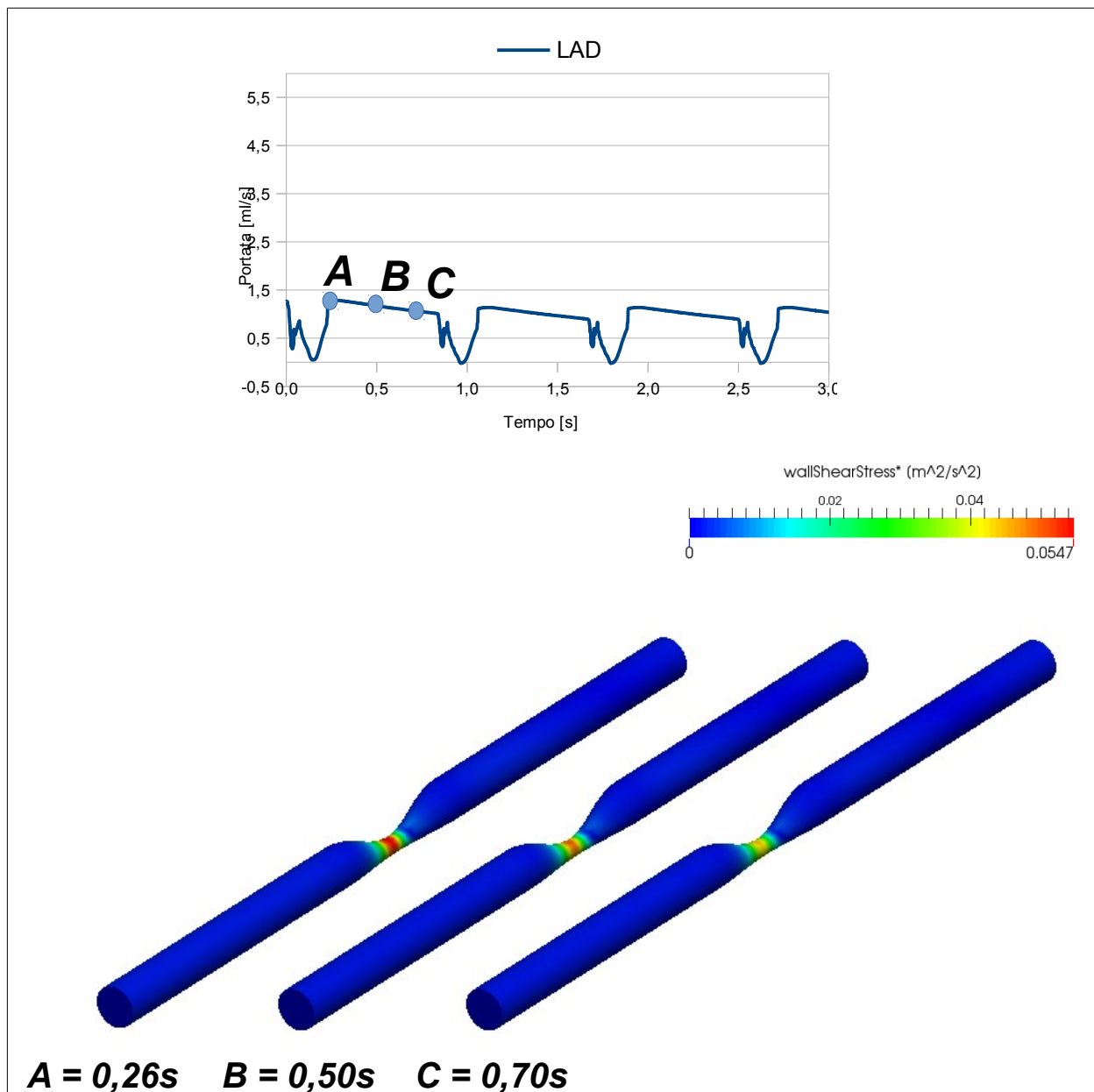


Figura 9.13: Rappresentazione colorimetrica, per tre valori temporali, degli andamenti dello sforzo tangenziale del vaso stenotico LAD. Al di fuori della stenosi lo sforzo tangenziale rimane massimo alla parete. Al fine di osservare le differenze di valori alla stenosi per i diversi time step, la scala scelta in questa rappresentazione non ne permette una visualizzazione dettagliata nella parte del vaso con diametro maggiore.

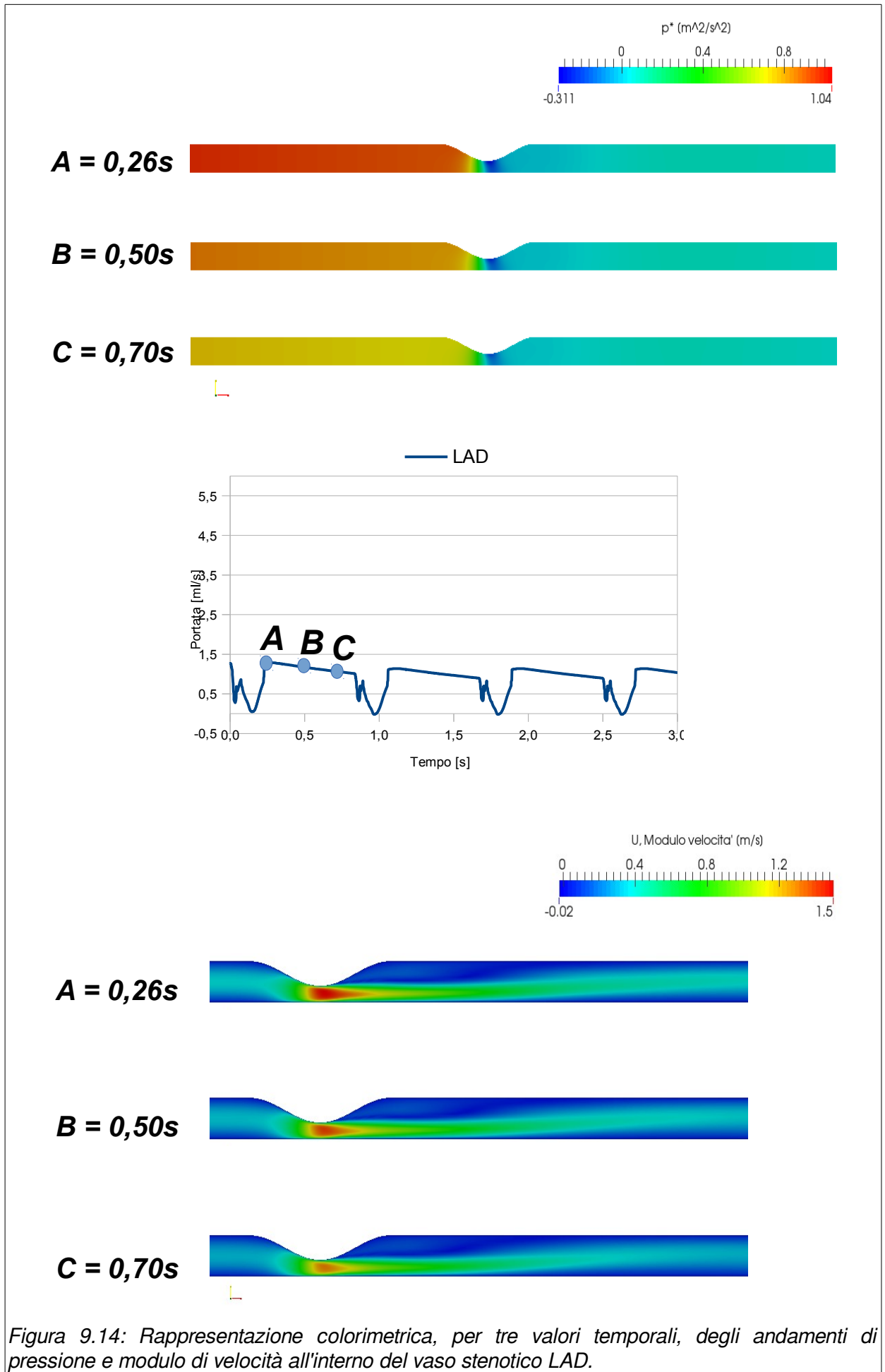
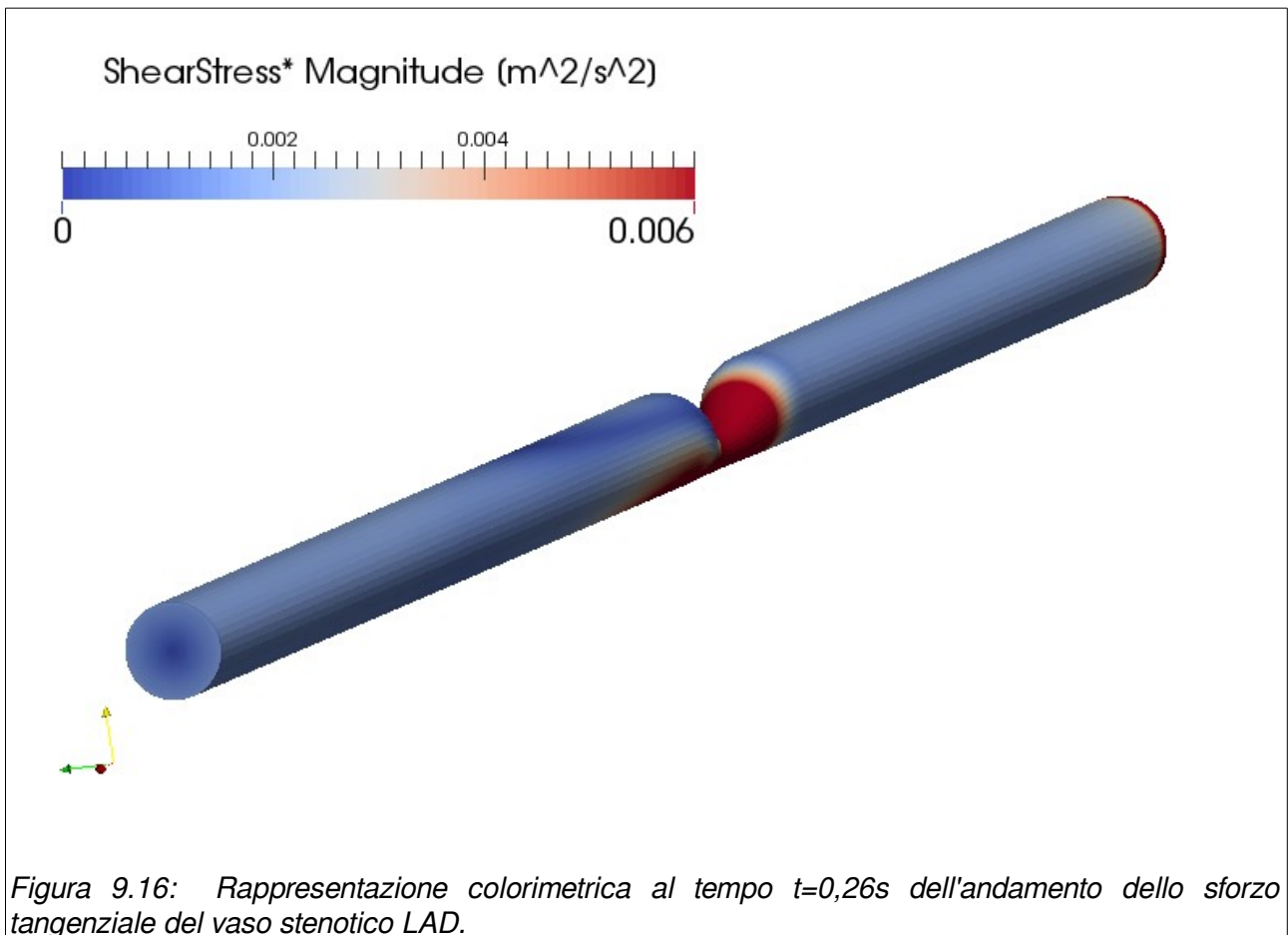
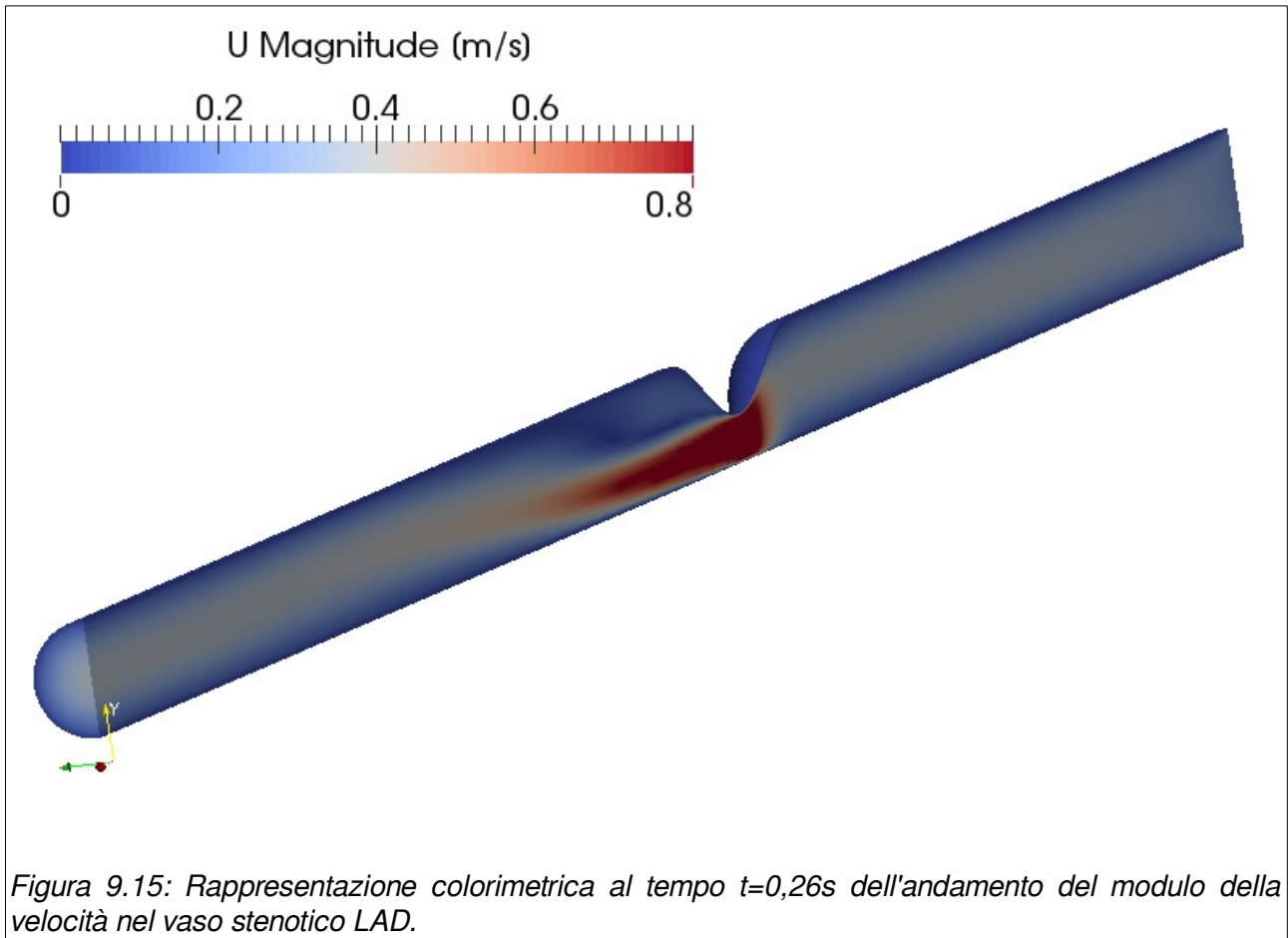


Figura 9.14: Rappresentazione colorimetrica, per tre valori temporali, degli andamenti di pressione e modulo di velocità all'interno del vaso stenotico LAD.



Capitolo 10:

Discussione, limitazioni e sviluppi futuri

In questo capitolo sono analizzati i risultati ottenuti nel presente lavoro di tesi. Sono riportate inoltre le limitazioni relative ai modelli numerici implementati e i possibili sviluppi futuri.

10.1. Obiettivo 1

Il primo obiettivo di questo lavoro è stato realizzare un modello modulare 0D del sistema cardiovascolare che fosse attendibile e modulare. La modularità ne facilita implementazioni aggiuntive senza dover ridefinire quanto già implementato.

Si prendono a confronto gli andamenti proposti nei lavori di R. Pietrabissa [13] (Figura 10.1).

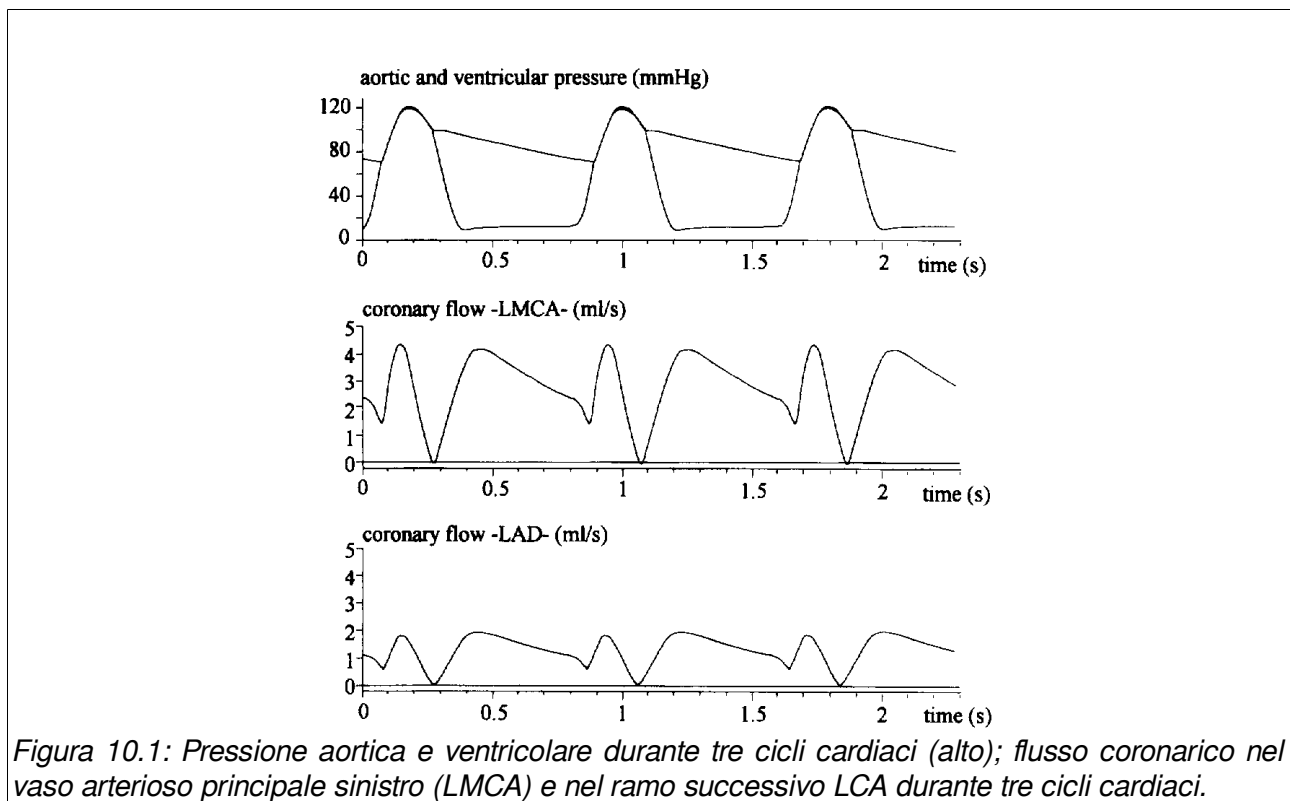


Figura 10.1: Pressione aortica e ventricolare durante tre cicli cardiaci (alto); flusso coronarico nel vaso arterioso principale sinistro (LMCA) e nel ramo successivo LCA durante tre cicli cardiaci.

I risultati ottenuti dalle simulazioni effettuate riproducono con buona approssimazione gli andamenti qui indicati. Lavori futuri potranno verificare attraverso lo studio di casi clinici la bontà del modello.

Confrontando i risultati tra il caso 1 e il caso 2 (arteria LAD fisiologica e stenotica) si evidenzia, come previsto, una diminuzione della portata nel vaso stenotico, mentre la riduzione della portata del vaso a monte è meno significativa perché vi è un incremento di flusso sanguigno negli altri rami dell'albero coronarico. Questa conseguenza trova un riscontro clinico come descritto nel lavoro di R. Pietrabissa [13]. In Figura 10.2 è rappresentata la portata di sangue nei quattro rami maggiori dell'albero coronarico al variare del grado di stenosi nell'arteria LAD. L'effetto qui descritto si evidenzia maggiormente dopo un grado di stenosi superiore all'80%.

Sempre confrontando i risultati tra il caso 1 e il caso 2 si può notare come non ci siano differenze di pressione per quanto riguarda l'atrio sinistro, ventricolo sinistro e aorta. Questo può essere imputato al fatto che il modello non è in grado di simulare la variazione di contrattilità del muscolo cardiaco dovuta ad un diverso apporto di sangue nelle coronarie. Non esiste infatti alcuna retroazione che modifichi il comportamento del muscolo cardiaco.

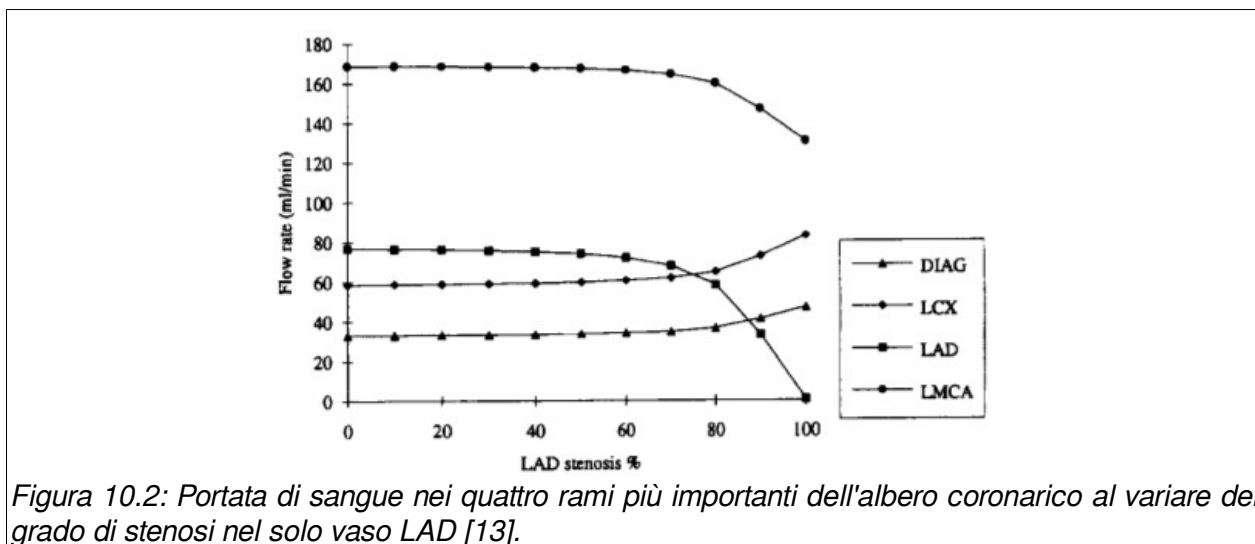


Figura 10.2: Portata di sangue nei quattro rami più importanti dell'albero coronarico al variare del grado di stenosi nel solo vaso LAD [13].

Considerata la notevole velocità di esecuzione del modello compilato (Tabella 4), lavori futuri potrebbero studiarne l'utilizzo all'interno di un sistema di diagnostica e di simulazione per valutare gli effetti di un intervento di angioplastica a partire da una determinata situazione pre-intervento dalla quale ricavare i parametri del modello.

Il modello proposto simula il comportamento del solo ramo coronarico sinistro essendo questo legato alla perfusione del lato sinistro del muscolo cardiaco che si trova in situazioni più critiche in termini di pressione rispetto al lato destro. Lavori futuri potranno estendere quanto descritto per questo distretto anche al distretto destro.

10.2. Obiettivo 2

Il secondo obiettivo di questo lavoro è stato mettere a punto un sistema software in grado di accoppiare modelli 0D e 3D indipendenti tra loro, il tutto sfruttando codice open source.

Analizzando i risultati ottenuti e confrontandoli con quelli del solo modello a parametri concentrati si notano alcune differenze: le portate nei vasi coronarici mantengono gli stessi valori medi ma gli andamenti si scostano leggermente. Questa conseguenza si suppone sia da imputare al modello tridimensionale utilizzato che non descrive nel modo migliore il vaso coronarico. Il prototipo di arteria utilizzato nel modello 0D include il parametro di complianza che non trova riscontro nel modello 3D a causa dell'ipotesi adottata di vaso a parete indeformabile.

Non sono presenti notevoli differenze tra i due modelli per quanto riguarda il comportamento del cuore.

Lavori futuri potranno testare meglio l'intero sistema (OpenModelica, CoSimO, OpenFOAM) applicandolo a casi studio noti al fine di determinarne l'accuratezza e l'efficienza.

Poter eseguire in modo indipendente il codice a parametri concentrati offre la possibilità di stimare i valori di pressione e portata che verranno imposti durante la simulazione accoppiata. E' molto importante stimare a priori l'entità queste variabili in modo da utilizzare il modello migliore per la simulazione del dominio tridimensionale. Ad esempio prevedere se è possibile la presenza del fenomeno della turbolenza permette di prendere decisioni che andranno ad influenzare la costruzione della griglia di calcolo e il tipo di risolutore da impiegare.

La modularità del sistema di accoppiamento è un notevole vantaggio perché permette il riutilizzo del sistema 0D e l'accoppiamento di domini tridimensionali in qualsiasi punto del circuito. Per ora è stato implementato solo l'accoppiamento con modelli che abbiano una sola interfaccia in ingresso e una sola in uscita. In questo modo non è possibile simulare domini tridimensionali che abbiano più porte di ingresso o uscita (es biforcazioni). Questo limite è comunque superabile replicando quanto già implementato e estendendolo a sistemi con più interfacce.

Appendice

Implementazione in codice Modelica del sistema circolatorio e del muscolo cardiaco

`class` `cardio.SistemaCircolatorio`

```
parameter Real Csa = 0.28;
parameter Real Csp = 2.05;
parameter Real Cep = 0.668;
parameter Real Csv = 61.11;
parameter Real Cmp = 0.525;
parameter Real Cmv = 15.71;
parameter Real Cbv = 10.71;
parameter Real Cbp = 0.358;
parameter Real Rmp = 4.48;
parameter Real Rbp = 6.57;
parameter Real Cev = 20.0;
parameter Real Cpa = 0.76;
parameter Real Cpp = 5.8;
parameter Real Cpv = 25.37;
parameter Real Cla = 19.23;
parameter Real Vula = 25.0;
parameter Real Ria = 0.0025;
parameter Real P0lv = 1.5;
parameter Real KElv = 0.014;
parameter Real Vulv = 16.77;
parameter Real Emaxlv = 2.95;
parameter Real Krlv = 0.000375;
parameter Real Cra = 31.25;
parameter Real Vura = 25.0;
parameter Real Rra = 0.0025;
parameter Real P0rv = 1.5;
parameter Real KErv = 0.011;
parameter Real Vurv = 40.8;
parameter Real Emaxrv = 1.75;
parameter Real Krrv = 0.00014;
parameter Real Rsa = 0.06;
parameter Real Rsp = 3.307;
parameter Real Rep = 3.52;
parameter Real Rsv = 0.038;
parameter Real Rev = 0.04;
parameter Real Rpa = 0.023;
parameter Real Rpp = 0.08939999999999999;
parameter Real Rpv = 0.0056;
parameter Real Rmv = 0.05;
parameter Real Rbv = 0.075;
parameter Real Lsa = 0.00022;
parameter Real Lpa = 0.00018;
parameter Real Vusa = 0.0;
parameter Real Vusp = 274.4;
parameter Real Vuep = 134.64;
parameter Real Vusv = 1121.0;
parameter Real Vuev = 550.0;
parameter Real Vupa = 0.0;
```

```

parameter Real Vupp = 123.0;
parameter Real Vupv = 120.0;
parameter Real Vump = 105.8;
parameter Real Vubp = 72.13;
parameter Real Vuhp = 24.0;
parameter Real Vumv = 432.14;
parameter Real Vubv = 294.64;
parameter Real Vuhv = 98.20999999999999;
parameter Real Vt = 5300.0;
Real Ppa(start = 16.0);
Real For;
Real Fpa;
Real Ppp(start = 15.8);
Real Pmp;
Real Pbp;
Real Ppv(start = 11.1);
Real Pla(start = 11.8);
Real Psa(start = 99.0);
Real Fol;
Real Fil;
Real Fsa;
Real Psp(start = 99.0);
Real Psv(start = 5.75);
Real Pmv(start = 5.75);
Real Pbv(start = 5.75);
Real Pev(start = 5.75);
Real Pep;
Real Pra(start = 4.9);
Real Plv;
Real Pmaxlv;
Real Pmaxrv;
Real Vlv;
Real Vu;
Real Vrv;
Real Rlv;
Real Rrv;
Real Fir;
Real Prv;
Real Pho;
Real Phi;
Real Fho;
Real Fhi;
input Real At;
Real corOut_p.v(quantity = "ElectricPotential", unit = "V") "Potential at the pin";
Real corOut_p.i(quantity = "ElectricCurrent", unit = "A") "Current flowing into the pin";
Real corIn_p.v(quantity = "ElectricPotential", unit = "V") "Potential at the pin";
Real corIn_p.i(quantity = "ElectricCurrent", unit = "A") "Current flowing into the pin";
Real corOut_n.v(quantity = "ElectricPotential", unit = "V") "Potential at the pin";
Real corOut_n.i(quantity = "ElectricCurrent", unit = "A") "Current flowing into the pin";
Real corIn_n.v(quantity = "ElectricPotential", unit = "V") "Potential at the pin";
Real corIn_n.i(quantity = "ElectricCurrent", unit = "A") "Current flowing into the pin";
Real pin_plv.v(quantity = "ElectricPotential", unit = "V") "Potential at the pin";

```

```

Real pin_plv.i(quantity = "ElectricCurrent", unit = "A") "Current flowing into the pin";
equation
pin_plv.v = Plv;
Phi = corIn_p.v - corIn_n.v;
0.0 = corIn_p.i + corIn_n.i;
Fhi = corIn_p.i;
Pho = corOut_p.v - corOut_n.v;
0.0 = corOut_p.i + corOut_n.i;
Fho = corOut_p.i;
Psp = Pho;
Phi = Pra;
Psp = Pep;
Psp = Pmp;
Psp = Pbp;
der(Ppa) * Cpa = For - Fpa;
der(Fpa) * Lpa = Ppa + (-Ppp) - Rpa * Fpa;
der(Ppp) * Cpp = Fpa + (Ppv - Ppp) / Rpp;
der(Ppv) * Cpv = (Ppp - Ppv) / Rpp + (Pla - Ppv) / Rpv;
der(Psa) * Csa = Fol - Fsa;
der(Fsa) * Lsa = Psa + (-Psp) - Rsa * Fsa;
der(Psp) * (Csp + Cep + Cmp + Cbp) = Fsa + (Psv - Psp) / Rsp + (Pev - Psp) / Rep + (Pmv - Psp) / Rmp + (Pbv - Psp)
/ Rbp - Fho;
der(Psv) * Csv = (Psp - Psv) / Rsp + (Pra - Psv) / Rsv;
der(Pmv) * Cmv = (Psp - Pmv) / Rmp + (Pra - Pmv) / Rmv;
der(Pbv) * Cbv = (Psp - Pbv) / Rbp + (Pra - Pbv) / Rbv;
Pev * Cev = Vt + (-Csa) * Psa + ((-Csp) - Cep - Cmp - Cbp) * Psp + (-Csv) * Psv + (-Cmv) * Pmv + (-Cbv) * Pbv +
(-Cra) * Pra + (-Cpa) * Ppa + (-Cpv) * Ppp + (-Cla) * Pla + (-Vlv) - Vrv - Cpv * Ppv - Vu;
Vu = Vusa + Vusp + Vuep + Vump + Vubp + Vusv + Vuev + Vumv + Vubv + Vura + Vupa + Vupp + Vupv + Vula;
Cla * der(Pla) = (Ppv - Pla) / Rpv - Fil;
Fil = if Pla > Plv then (Pla - Plv) / Rla else 0.0;
der(Vlv) = Fil - Fol;
Fol = if Pmaxlv > Psa then (Pmaxlv - Psa) / Rlv else 0.0;
Rlv = Krlv * Pmaxlv;
Plv = Pmaxlv - Rlv * Fol;
Pmaxlv = At * Emalv * (Vlv - Vulv) + (1.0 - At) * P0lv * (-1.0 + exp(KElv * Vlv));
Cra * der(Pra) = (Psv - Pra) / Rsv + (Pev - Pra) / Rev + (Pmv - Pra) / Rmv + Fhi + (Pbv - Pra) / Rbv - Fir;
Fir = if Pra > Prv then (Pra - Prv) / Rra else 0.0;
der(Vrv) = Fir - For;
For = if Pmaxrv > Ppa then (Pmaxrv - Ppa) / Rrv else 0.0;
Rrv = Krrv * Pmaxrv;
Prv = Pmaxrv - Rrv * For;
Pmaxrv = At * Emarv * (Vrv - Vurv) + (1.0 - At) * P0rv * (-1.0 + exp(KErv * Vrv));
corOut_p.i = 0.0;
corIn_p.i = 0.0;
corOut_n.i = 0.0;
corIn_n.i = 0.0;
pin_plv.i = 0.0;
end cardio.SistemaCircolatorio;

```


Bibliografia

- [1] Katia Laganà, Rossella Balossino, Francesco Migliavacca, Giancarlo Pennati, Edward L. Bove, Marc R. de Leval, Gabriele Dubini. *Multiscale modeling of the cardiovascular system: application to the study of pulmonary and coronary perfusions in the univentricular circulation*, 2005
- [2] Smuclovisky, Claudio, *Coronary Artery CTAA Case-Based Atlas*, 2010
- [3] *European Cardiovascular Disease Statistics, 2012 edition*, 2012
- [4] H. J. Kim , I. E. Vignon -Clementel , J. S. Coogan , C. A. Figueroa , K. E. J Ansen , and C. A. Taylor. *Patient-Specific Modeling of Blood Flow and Pressure in Human Coronary Arteries.*,2010
- [5] Raffaele Ponzini, Massimo Lemma, Umberto Morbiducci, Franco M. Montevicchi, Alberto Redaelli. *Doppler derived quantitative flow estimate in coronary artery bypass graft: A computational multiscale model for the evaluation of the current clinical procedure*, 2008
- [6] Claudio Gheller, *Le risorse Open Source per la ricerca in ambito scientifico*, 2007
- [7] Roberto Ponciroli. *Modellistica orientata agli oggetti del reattore Triga Mark II dell'Università di Pavia*, 2010
- [8] Peter Fritzon, Peter Bunus. *Modelica – A General Object-Oriented Language for Continuous and Discrete-Event System Modeling and Simulation*, 2000
- [9] L. Formaggia and A. Veneziani. *Reduced and multiscale models for the human cardiovascular system.*, 2003
- [10] M. Ursino, E. Magosso. *Acute cardiovascular response to isocapnic hypoxia*, 2000
- [11] Sagawa K, L. Maughan, H. Suga, K. Sunagawa.. *Cardiac Contraction and the Pressure-Volume Relationship*, 1998
- [12] Piene H. *Impedance matching between ventricle and load*, 1984
- [13] *A lumped parameter model to evaluate the fluid dynamics of different coronary bypasses.*, 1995
- [14] Quarteroni A., Veneziani A.. *Analysis of a geometrical multiscale model based on the coupling of ODE's and PDE's for blood flow simulations*, 2003
- [15] A. Quarteroni and A. Valli. *Domain Decomposition Methods for Partial Differential Equations*, 1999
- [16] Susanna Carcano, Maurizio Mansueti Cogliati. *Modelli ed Algoritmi per lo studio dell'interazione fluido-struttura nell'ambito dell'emodinamica*, 2007
- [17] Breve introduzione a Python, , <http://www.python.it/doc/intro/>