

UNIVERSITÀ DEGLI STUDI DI ROMA TORVERGATA
MACROAREA DI INGEGNERIA



CORSO DI LAUREA MAGISTRALE: INGEGNERIA MECCANICA

**Personalizzazione di Human Body Models
attraverso mesh morphing avanzato:
THUMS parametrico**

Relatore:

Prof Marco Evangelos Biancolini

Laureando:

Emanuele Lombardi

Correlatore:

Ing Emanuele Di Meo

ANNO ACCADEMICO: 2022/2023

Indice

1	Introduzione	6
2	Nozioni preliminari	9
2.1	THUMS	9
2.1.1	Panoramica generale	9
2.1.2	THUMS Version 7	10
2.1.3	THUMS AM50 e THUMS AM95	13
2.2	Metodo degli elementi finiti	15
2.3	Analisi transitoria esplicita	16
2.3.1	Metodo delle differenze centrali	16
2.4	Radial Basis Function	18
2.4.1	Mesh morphing basato sulle RBF	19
3	Strumenti e software	21
3.1	Ansys Workbench	21
3.2	Ansys LS-DYNA	21
3.2.1	Ansys LS-Prepost	21
3.3	RBF Morph	22
3.3.1	RBF Morph Structures	22
3.3.2	Estensione RBF_FGP.dll	22
3.4	Linguaggi di programmazione	23
4	Criterio di verifica geometrica	24
4.1	Minima distanza media	24
4.2	Massima distanza minima	25
4.3	THUMS AM50 e AM95 a confronto	25
5	Criterio di mesh morphing	28
5.1	Definizione del mesh morphing da AM50 ad AM95	28
5.2	Parametrizzazione	34
6	Mesh morphing semplificato	35
6.1	THUMS semplificato	36
6.2	Implementazione del mesh morphing	36
6.2.1	Importazione di THUMS AM50s e THUMS AM95s	37
6.2.2	Export dei source points	37
6.2.3	Calcolo degli spostamenti	37
6.2.4	RBF Morph Set Up	39
6.3	Esecuzione del mesh morphing	42
7	Mesh morphing completo	44
7.1	Preparazione del K-FILE	45

7.2	Automatizzazione del mesh morphing	47
7.2.1	Definizione del mesh morphing	47
7.2.2	Esecuzione del mesh morphing	49
7.3	Automatizzazione del confronto tra mesh	50
7.4	THUMS parametrico	50
8	Simulazione	52
8.1	Frontal Sled Test	52
8.1.1	Configurazioni di simulazione	54
8.1.2	Estrazione dei risultati	55
9	Risultati	58
9.1	Confronto geometrico	58
9.2	Cinematica nell'impatto frontale	64
9.2.1	Influenza del mesh morphing	64
9.2.2	THUMS AM50m95 vs THUMS AM95	66
9.2.3	Considerazioni	70
	Conclusioni	72
	Appendice	73
9.3	Grafici	73
9.3.1	Source points	73
9.3.2	Confronto geometrico: THUMS AM50 vs THUMS AM95	77
9.3.3	Confronto geometrico: THUMS AM50m vs THUMS AM95	80
9.3.4	Confronto geometrico: THUMS AM50 vs THUMS AM50m	83
9.3.5	Cinematica dei THUMS morphati	86
9.3.6	Evoluzione della coordinata x	92
9.3.7	Evoluzione della coordinata y	98
9.3.8	Evoluzione della coordinata z	104
9.4	Codici	110
9.4.1	Generazione_PTS.py	110
9.4.2	Mesh_Morphing.py	115
9.4.3	Confronto.py	119

Indice delle figure

1.1	Confronto tra un ATD (a sinistra) e un FE HBM (a destra)	6
1.2	Confronto THUMS AM50 e THUMS AM95 in vista laterale	8
2.1	Tipologie di THUMS Version7. Da sinistra verso destra: AF05, AM50 e AM95	10
2.2	Inclinazioni dello schienale implementate in THUMS Version7. Da sinistra verso destra: angolo di inclinazione di 25 deg, 35 deg e 45 deg	10
2.3	Organi interni nelle viste laterale e frontale	11
2.4	Parti scheletriche afferenti al torso nelle viste laterale e frontale	11

2.5	Organi interni meshati attraverso elementi solidi tetraedrici ed ossa, carne, legamenti, cartilagini ecc. della gamba meshati attraverso elementi solidi esaedrici	12
2.6	Pelle di busto e bacino meshata attraverso elementi shell	12
2.7	Muscoli e tendini come elementi <i>asta</i> e <i>seatbelt</i>	13
2.8	Confronto tra THUMS AM50 e THUMS AM95: forma interna	14
2.9	Confronto tra THUMS AM50 e THUMS AM95: forma esterna	14
3.1	Architettura del PTS-file identificativo del mesh morphing	22
4.1	Stessa geometria discretizzata diversamente: numero di nodi della mesh2 (in azzurro) maggiore di quello della mesh1 (in rosso)	25
4.2	Valori della MDA nel confronto delle zone della pelle di THUMS AM50 e THUMS AM95	26
4.3	Valori della MDM nel confronto delle zone della pelle di THUMS AM50 e THUMS AM95	27
4.4	Valori del rapporto MDA/MDM nel confronto delle zone della pelle di THUMS AM50 e THUMS AM95	27
5.1	Coppie di alcuni edges omologhi in evidenza	28
5.2	Definizione dei 22 edges corrispondenti ai source points: rappresentati solo quelli relativi alla parte destra	29
5.3	Identificazione dei 22 edges omologhi: rappresentati solo quelli relativi alla parte destra	29
5.4	Confronto tra edges prima e dopo il morphing	31
5.5	Valori della <i>MDA</i> a confronto	32
5.6	Valori della <i>MDM</i> a confronto	33
5.7	Valori del rapporto <i>MDA/MDM</i> a confronto	33
6.1	THUMS AM50s	35
6.2	K-FILE di THUMS AM50s	36
6.3	Struttura del modello <i>Workbench</i>	37
6.4	Definizione delle <i>path</i> corrispondenti agli edges d'interesse	38
6.5	RBF Morph Set Up	39
6.6	Funzioni di <i>Scaling</i> e <i>Translation</i>	40
6.7	Riferimento locale baricentrico dell'edge <i>spalla-busto destra</i>	40
6.8	Preview del campo di spostamenti alle RBF per <i>spalla-busto destra</i>	41
6.9	Preview del campo di spostamenti alle RBF complessivo	41
6.10	Interpolazione ai target points del campo di spostamenti definito per i source points	42
6.11	Esportazione del mesh morphing come PTS-FILE	42
6.12	Visualizzazione grafica del confronto tra AM50 vs AM95 e AM50m vs AM95 in termini di forma interna (pelle)	43
6.13	Visualizzazione grafica del confronto tra AM50 vs AM95 e AM50m vs AM95 in termini di forma interna (scheletro)	43
7.1	Creazione della <i>Entity Keyword</i> denominata <i>collo_50</i>	45
7.2	Selezione dell'edge corrispondente a <i>collo_50</i>	46
7.3	Identificazione di <i>collo_50</i> nel K-FILE di simulazione	46
7.4	Identificazione della numerazione di un set di nodi nel K-FILE	47
7.5	Identificazione delle coordinate di un set di nodi nel K-FILE	48
7.6	Sezione <i>NODE</i> del K-FILE	48
7.7	PTS-FILE	48
7.8	Prime righe della sezione <i>NODE</i> del K-FILE di THUMS AM50	49
7.9	Prime righe della sezione <i>NODE</i> del K-FILE di THUMS AM50m95	49
7.10	Funzione che implementa il calcolo di <i>MDA</i> e <i>MDM</i>	50
7.11	Parametrazioni di THUMS: 35°, 50°, 75° e 95° percentile	51
7.12	Confronto scheletri di AM50m95 e AM50m95b	51

8.1	Posizionamento di THUMS all'istante iniziale di simulazione nel <i>frontal sled test</i>	53
8.2	Curva d'accelerazione	53
8.3	Effetti della collisione visualizzati al termine della simulazione	54
8.4	<i>Configurazione50m95</i> all'istante iniziale di simulazione	55
8.5	Selezione dei risultati sui nodi del set di source points	56
8.6	Traiettorie dei nodi del set nella coordinata selezionata	57
8.7	Traiettoria della coordinata baricentrica	57
9.1	Zone della pelle migliore e peggiore in merito alla <i>MDA</i> . AM50m95 è abbreviato ad AM50m in figura.	59
9.2	Valori di <i>MDA</i> per le zone della pelle	61
9.3	Valori di <i>MDA</i> e <i>MDM</i> per cassa toracica ed ossa pelviche	61
9.4	AM50 vs AM50m95, AM50m95 vs AM95 e AM50 vs AM50m95: forma esterna e interna	63
9.5	Massimo e minimo S_{meanx} riscontrati	64
9.6	Migliore e peggior fit del comportamento target in merito ai punti di controllo	67
9.7	Evoluzione cinematica di <i>busto-spalla destra</i> e <i>mano destra</i>	68
9.8	Posizionamento iniziale di THUMS AM50m95 e THUMS AM95 rispetto ai vincoli	70
9.9	Posizionamento di THUMS AM50m95 e THUMS AM95 rispetto ai vincoli a $t = 0.075s$	71
9.10	Posizionamento di THUMS AM50m95 e THUMS AM95 a fine simulazione	71

Sommario

I Finite Elements Human Body Models (FE HBMs) rappresentano uno strumento prezioso per la ricerca e la progettazione nell'ambito della sicurezza veicolare. La potenzialità dello strumento, tuttavia, risulta in un certo modo limitata dal piccolo numero di forme nel quale è reso disponibile. Da questo presupposto nasce il filone di ricerca riguardante la personalizzazione di FE HBMs nel quale si inquadra il presente lavoro di tesi. Modulando le differenze di forma tra THUMS AM50 e THUMS AM95 attraverso il mesh morphing guidato da radial basis functions (RBF), reso automatico in *Python* in quanto ad implementazione, viene proposta una parametrizzazione di THUMS in base al percentile antropometrico, verificata geometricamente e sperimentata in termini cinematici nell'ambito del *frontal sled test*.

Capitolo 1

Introduzione

In un contesto sempre più complesso e tecnologicamente avanzato, le potenzialità della simulazione numerica nel campo dell'ingegneria sono diventate strumentali per progettare e analizzare sistemi con precisione e flessibilità senza precedenti. La capacità di simulare virtualmente il comportamento di un sistema in una vasta gamma di scenari offre infatti vantaggi significativi in termini di costi, tempi di sviluppo e ottimizzazione del design. In ambito automobilistico e, precisamente, della sicurezza veicolare, la simulazione numerica ha permesso di ridurre il ricorso ai test sperimentali in varie fasi del processo di sviluppo, quali: progettazione e ottimizzazione delle zone a deformazione controllata; progettazione e ottimizzazione dei sistemi di restrizione e infine analisi del rischio di lesioni. Uno strumento di grande rilevanza in tal senso è rappresentato dai modelli del corpo umano agli elementi finiti, ovvero i FE HBMs (Finite Elements Human Body Models).

Introdotti a supporto dei tradizionali manichini per i crash tests, ovvero, gli ATDs (Antropomorphic Test Devices), i FE HBMs permettono di simulare in modo più dettagliato il comportamento del corpo umano durante impatti e collisioni, offrendo un maggior livello di fedeltà biologica. In Figura 1.1 sono messi a confronto un ATD e un FE HBM, il THUMS [23]: mentre l'ATD è un assieme di parti in plastica e metallo, il FE HBM è dotato di uno scheletro dettagliato, organi interni e tessuti molli; mediante il FE HBM risulta possibile una previsione infortunistica al livello dei tessuti mentre con l'ATD, evidentemente no.



Figura 1.1: Confronto tra un ATD (a sinistra) e un FE HBM (a destra)

L'utilizzo di FE HBM nello studio sugli impatti biomeccanici ha una storia di circa 50 anni. Solo

nell'ultimo ventennio, tuttavia, il raggiungimento delle potenze di calcolo sufficienti e l'accumulo dei dati necessari a validare i modelli, hanno reso i FE HBMs una risorsa preziosa per la ricerca e la progettazione. Un punto di svolta è stato segnato dalla realizzazione di FE HBMs completi (dotati di tutto il corpo), come THUMS (*Iwamoto et al.* [11]) e HUMOS (*Robin et al.* [18]), rappresentativi di un maschio adulto di taglia media. Da allora il numero di FE HBMs disponibili, sviluppati prevalentemente ad opera di case produttrici e fornitori del settore automobilistico e costantemente aggiornati e migliorati, è cresciuto notevolmente e così il loro impiego nella progettazione di dispositivi di sicurezza e in studi disparati di assoluto rilievo. A titolo di esempio, si riportano a seguire alcuni casi di letteratura. In *Östh et al.* [13], utilizzando il THUMS come FE HBM e implementando un sistema di feedback controller viene simulata e validata l'attivazione muscolare dei muscoli di tronco e collo di un occupante durante un intervento di frenata autonoma. Lo studio mette inoltre in evidenza gli effetti benefici delle moderne cinture di sicurezza con appoggio a tre punti in situazioni di guida normali, come appunto la semplice frenata, e non soltanto in caso di incidente. In *Kumar et al.* [3], utilizzando THUMS come FE HBM vengono analizzati 5 scenari di impatto tra motocicletta e pedone evidenziando come in 4 dei 5 casi in esame sia possibile predire con precisione l'insorgenza e la localizzazione di fratture nelle ossa lunghe. Per finire, in *Holdgrün et al.* [7], viene condotto uno studio sulle dinamiche precollisione che occorrono negli incidenti tra autoveicoli e pedoni. Utilizzando il GHBMC-PS [5] come FE HBM vengono confrontati gli effetti che ha sull'impatto la postura del pedone statisticamente riscontrata e la postura standard SAE [9].

Quanto riportato, unitamente alle numerose applicazioni nelle quali i FE HBMs sono impiegati alla stregua di ATDs, mette ulteriormente in evidenza le grandi potenzialità dello strumento. I FE HBMs attualmente disponibili, tuttavia, rappresentano un numero limitato di dimensioni corporee, ovvero, tipicamente, le stesse poche dimensioni nelle quali sono realizzati gli ATDs: si fa notare che fino a tempi piuttosto recenti gli unici ATDs impiegati nei crash tests sono stati il maschio adulto di taglia media (50° percentile antropomorfo) e la femmina adulta di taglia piccola (5° percentile antropomorfo). Studi come *Ridella et al.* [17], *Viano et al.* [29] e *Rupp et al.* [20] evidenziano come negli incidenti il rischio di fatalità sia significativamente variabile con l'età, l'occorrenza di determinate patologie, l'obesità e in determinate condizioni in base al genere. In sintesi, tutto ciò che è causa di modifiche nella geometria, nella composizione e nelle proprietà dei materiali che costituiscono le ossa e i tessuti di un soggetto, si riflette nell'entità del grado di rischio associato a un incidente. Alla luce di queste conoscenze e della grande variabilità antropometrica esistente tra gli esseri umani appare chiaro dunque come i FE HBMs in relazione al loro potenziale, siano inadeguatamente rappresentativi. È sulla scia di tali considerazioni che si inquadra il filone di ricerca riguardante la personalizzazione dei FE HBM. Si riportano a seguire alcuni dei più autorevoli studi in merito. In *Ito et al.* [10], a partire dall'H-model (*Haug et al.* [6]) vengono modellate le differenze in termini di forma delle ossa, spessore osseo corticale e proprietà dei materiali costituenti, introdotte sulla cassa toracica di un maschio di taglia media per effetto dell'età. In *El-Jawahri et al.* [4], viene condotto uno studio analogo usando tuttavia come FE HBM base il Ford model (*Ruan et al.* [19]). In *Schoell et al.* [21], viene sviluppato un FE HBM di un maschio di taglia media di 65 anni modificando il GHBMC. Per modellare l'invecchiamento vengono ridefinite le geometrie di cervello, testa, cassa toracica, bacino, femore e tibia sulla base di modelli geometrici statistici e riconfigurate le proprietà dei materiali di testa, torace, bacino e arti inferiori sulla base di test condotti sui tessuti di PMHSs (Post Mortem Human Subjects). In *Shi et al.* [22], per studiare gli effetti dell'obesità sul rischio di lesioni negli impatti frontali, vengono sviluppati 4 FE HBMs a partire dal THUMS rappresentativi di 4 diversi livelli di BMI (body mass index), introducendo per la prima volta il discorso degli effetti legati alla forma del corpo nella previsione infortunistica attraverso i FE HBMs. Per concludere in *Hwang et al.* [8], viene realizzato un S-THUMS (Subject-specific THUMS), ovvero un THUMS modellato sulle specifiche antropometriche di un PMHS e validato in merito ai test di impatto laterale. Attraverso il mesh morphing del THUMS di base rappresentativo del maschio adulto di taglia media, avendo come target l'S-THUMS, viene successivamente ottenuto il P-THUMS (Parametric THUMS) e validato in base all'S-THUMS, realizzando il primo esempio di FE HBM parametrico.

Nel presente lavoro di tesi si concentra l'attenzione sul THUMS. Come si legge in [24] il THUMS,

acronimo di Total HUMAN Model for Safety, sviluppato da TOYOTA a partire dal 1997, costituisce uno dei più avanzati FE HBMs attualmente disponibili e a partire dal 2021 la casa automobilistica giapponese ne ha concesso la licenza a titolo gratuito. Nella versione più recente di THUMS (THUMS Version 7), per quanto riguarda i FE HBMs maschili, i modelli disponibili sono il THUMS AM50, ovvero il maschio adulto al 50° percentile antropometrico statistico e il THUMS AM95, ovvero il maschio adulto al 95° percentile (riportati a confronto in Figura 1.2). L'obiettivo della tesi è quello di coprire con continuità lo spazio antropometrico statistico esistente tra i 2 THUMS introdotti. Sfruttando il software *RBF Morph* [15] che gestisce il mesh morphing attraverso l'uso di funzioni di base radiale, verrà messa a punto una procedura completamente automatizzata in *Python* per generare, entro un certo grado di approssimazione, qualsivoglia THUMS tra il 50° ed il 95° percentile antropometrico statistico a partire dal THUMS AM50. In altri termini, si realizzerà una parametrizzazione di THUMS in base al percentile statistico antropometrico. La bontà del FE HBM parametrico verrà infine valutata assumendo il THUMS AM95 come metro di confronto in quanto a rispondenza delle geometrie di pelle, cassa toracica e ossa pelviche a seguito del mesh morphing e comportamento cinematico nel test di impatto frontale.

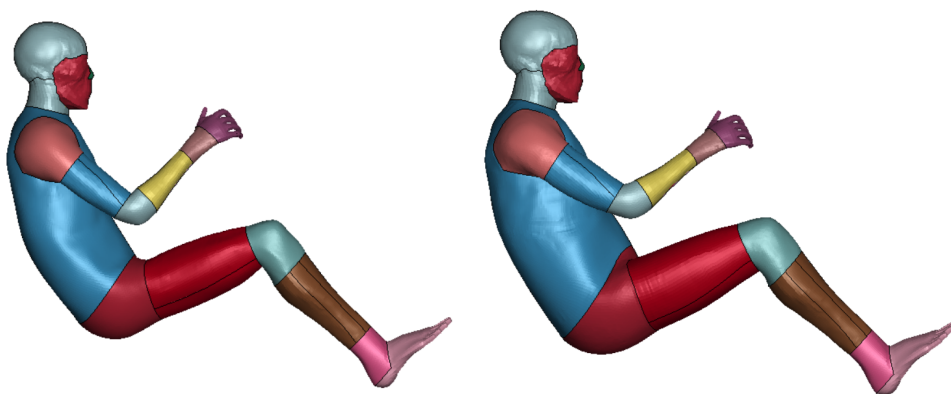


Figura 1.2: Confronto THUMS AM50 e THUMS AM95 in vista laterale

Capitolo 2

Nozioni preliminari

2.1 THUMS

Il presente paragrafo è interamente dedicato all'approfondimento di THUMS, con l'intento di fornire i concetti sufficienti ad una completa comprensione del lavoro di tesi. Per ulteriori dettagli e approfondimenti, si consiglia di fare riferimento ai manuali ufficiali, come [26] e [27], forniti contestualmente al download di THUMS [25].

2.1.1 Panoramica generale

Total Human Model for Safety (THUMS) è il FE HBM sviluppato congiuntamente da TOYOTA MOTOR CORPORATION e TOYOTA CENTRAL R&D LABS., INC con l'obiettivo di simulare la risposta del corpo umano in termini cinematici e infortunistici, nell'ambito degli incidenti stradali. La storia di THUMS ha inizio negli anni 2000 con lo sviluppo di THUMS Version1. In THUMS Version1 la modellazione si focalizzava sulle ossa principali e i legamenti, mentre cervello ed organi interni venivano semplificati come parti solide. Il risultato era un modello di circa 80.000 elementi con una dimensione media della mesh di 15mm che ciononostante ben coglieva l'occorrenza di fratture ossee e rotture di legamenti nelle simulazioni di crash. Nel 2004 fu realizzato THUMS Version2, migliorando la modellazione delle ossa craniche. Introducendo un modello dettagliato del cervello, il 2008 fu l'anno di THUMS Version3 che con 130.000 elementi totali risultava in grado di simulare le lesioni cerebrali. Il numero di elementi totali crebbe fino a 2.000.000 con THUMS Version4, nel 2010, che finalmente incorporava le parti relative agli organi interni. THUMS Version 5 (2015) e Version 6 (2019) furono aggiornamenti, rispettivamente, di THUMS Version 3 e Version 4 in termini di modellazione muscolare.

La più recente versione di THUMS, nonché oggetto di questa trattazione è THUMS Version 7 (2.100.000 elementi circa). Basata su THUMS Version 6, con il quale condivide la modellazione anatomica del corpo umano in tutte le sue parti, essa vanta 3 tipologie di FE HBMs di occupante¹ (Figura 2.1), ovvero, la femmina adulta di taglia piccola (AF05), il maschio adulto di taglia media (AM50) e il maschio adulto di taglia grande (AM95), ciascuno dei quali può essere ulteriormente caratterizzato in base all'inclinazione dello schienale come si vede in Figura 2.2. Le versioni di THUMS attualmente disponibili nella piattaforma di riferimento [24] sono quelle da Version 4 a seguire. Accettando la *THUMS USER POLICY* e previa sottoscrizione è possibile effettuare il download in forma di codice sorgente di simulazione per *Ansys LS-DYNA* [2].

¹Tra i FE HBMs si distinguono generalmente quelli in posizione eretta, i pedestrini, da quelli in posizione di guida, gli occupanti

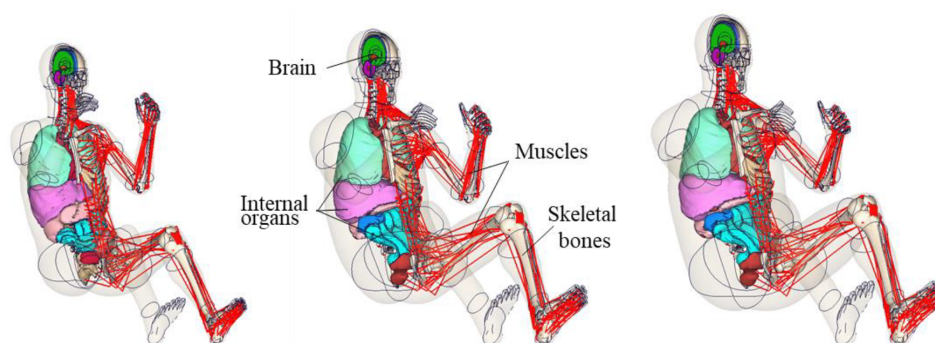


Figura 2.1: Tipologie di THUMS Version7. Da sinistra verso destra: AF05, AM50 e AM95

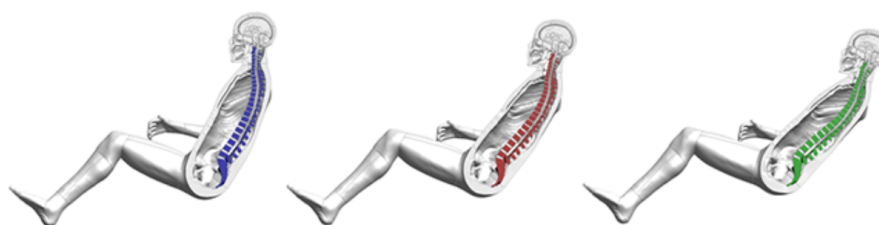


Figura 2.2: Inclinazioni dello schienale implementate in THUMS Version7. Da sinistra verso destra: angolo di inclinazione di 25 deg, 35 deg e 45 deg

2.1.2 THUMS Version 7

Il razionale alla base di THUMS è quello di rappresentare il corpo umano senza introdurre alcune semplificazioni morfologiche, modellando le sue parti con mesh agli elementi finiti e i materiali attraverso leggi costitutive. Questo risulta chiaramente in THUMS Version 7 e nelle righe a seguire sono evidenziate alcune delle linee guida adottate per la realizzazione dei modelli appartenenti alla serie.

- **Anatomia macroscopica**

Per realizzare l'anatomia macroscopica² dei vari THUMS vengono adottate tecniche specifiche in base alla natura dei tessuti in esame. Le geometrie di organi interni (cuore, polmoni, fegato, reni, intestino ecc.), principali vene e arterie, scheletro e tessuti di tipo membrana (pelle, diaframma, diaframma, pleure, ecc.) afferenti al torso, sono ottenute con un processo di mascheratura di immagini di CT scan ad alta risoluzione di PMHSs fornite dalla University of Michigan. Le geometrie dei muscoli sono determinate a partire dalla letteratura dedicata e infine le geometrie di arti e testa vengono ottenute raffinando le stesse impiegate in THUMS Version 3. A titolo esemplificativo di quanto detto si mostra in Figura 2.3 la geometria degli organi interni e in Figura 2.4 quella delle parti scheletriche afferenti al torso.

²Si distingue l'anatomia macroscopica, riguardante le strutture visibili ad occhio nudo da quella microscopica, riguardante i tessuti a livello cellulare

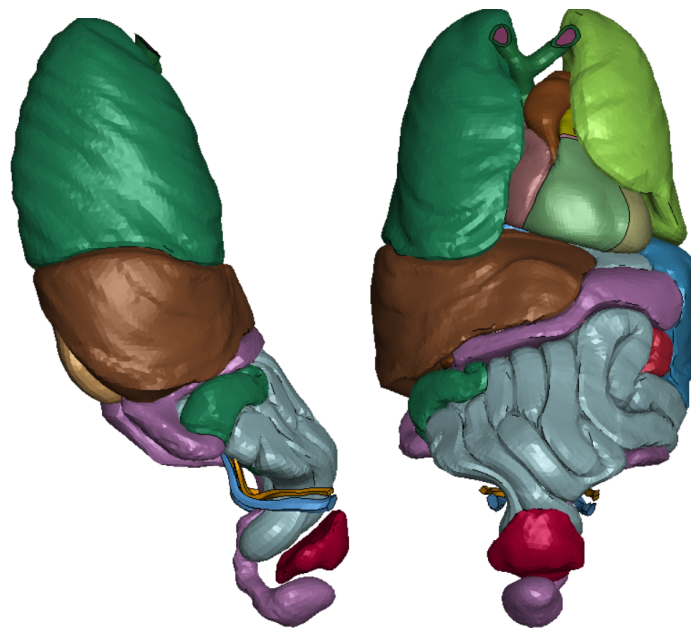


Figura 2.3: Organi interni nelle viste laterale e frontale

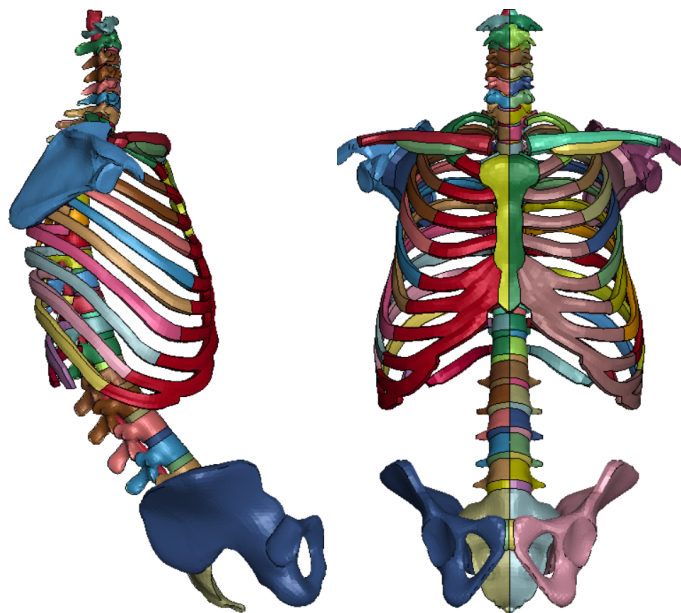


Figura 2.4: Parti scheletriche afferenti al torso nelle viste laterale e frontale

- **Mesh**

Per generare le mesh sulle geometrie ottenute viene seguito il razionale di impiegare elementi *solidi* ove possibile e in tal senso la maggior parte di organi, ossa, legamenti (in particolare i più spessi), tessuto adiposo/connettivo e volumi muscolari sono così meshati (Figura 2.6). Quelle parti assimilabili a gusci e membrane come ad esempio talune ossa corticali, il diaframma, i legamenti più sottili e banalmente, la pelle, vengono invece modellate con elementi *shell* (Figura 2.6).

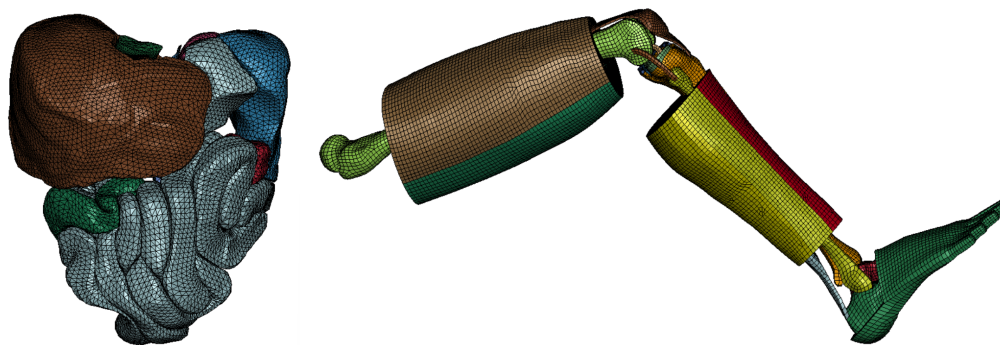


Figura 2.5: Organi interni meshati attraverso elementi solidi tetraedrici ed ossa, carne, legamenti, cartilagini ecc. della gamba meshati attraverso elementi solidi esaedrici

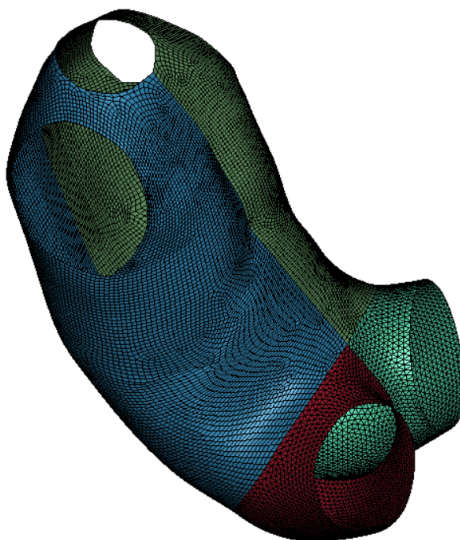


Figura 2.6: Pelle di busto e bacino meshata attraverso elementi shell

- **Controllo qualità della mesh**

Nel contesto del controllo qualità della mesh vengono definiti i seguenti valori di riferimento: *warpage* di massimo 50 deg, *aspect ratio* pari massimo a 5, *skew* di massimo 60 deg e infine *Jacobiano* pari almeno a 0.3. Viene inoltre stabilito un'ordine di preferenza nell'utilizzo di elementi solidi, prediligendo gli elementi *Hexaedron* ai *Tetrahedron*, comunque impiegati in quelle parti come organi interni ed ossa corticali difficili da meshare nel rispetto dei valori di qualità prefissati.

- **Modellazione dei materiali**

Per modellare le proprietà meccaniche dei materiali sono stati impiegati dati tratti dalla letteratura di settore. In generale, alle parti scheletriche vengono assegnati legami costitutivi elasto-plastici; a legamenti, tendini, pelle e altri tessuti molli vengono assegnati legami costitutivi iperelastici e infine gli organi interni vengono modellati come incompressibili (ovvero in grado di subire deformazioni di forma ma non volumetriche).

- **Modellazione dei muscoli**

Nel corso di questa discussione sulle linee guida adottate nella modellazione di THUMS è già stato fatto riferimento ai muscoli nel contesto di come vengono meshati i volumi muscolari. È importante tuttavia precisare che ai fini di una corretta rappresentazione della risposta del corpo umano bisogna distinguere l'ingombro associato alla massa muscolare dalla funzione da essi esplicata. In merito a questo secondo aspetto si fa notare che il sistema muscoli-tendini è modellato in THUMS impiegando per le fibre muscolari elementi *asta* a cui si associa il materiale *hill type muscle material* e per i tendini elementi *seatbelt*, ovvero elementi resistenti a sola trazione. In Figura 2.7 sono rappresentati tali elementi in relazione alla geometria dello scheletro.

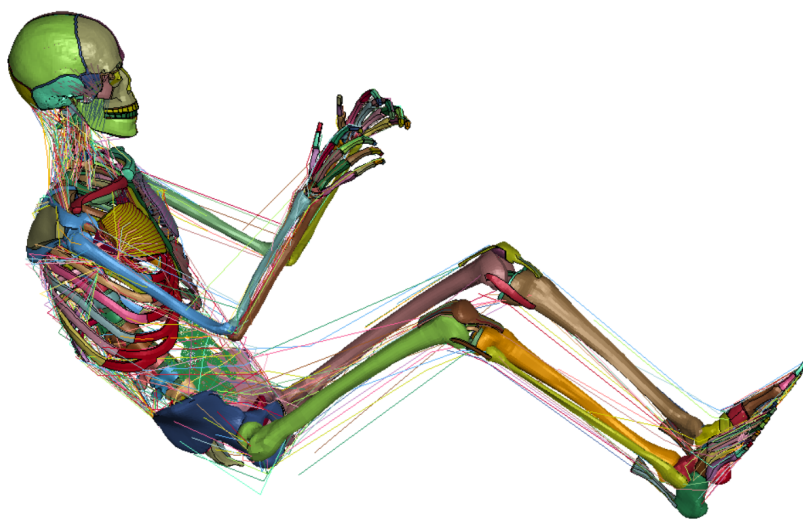


Figura 2.7: Muscoli e tendini come elementi *asta* e *seatbelt*

2.1.3 THUMS AM50 e THUMS AM95

I FE HBMs maschili della serie Version 7 sono THUMS AM50 e THUMS AM95, corrispondenti rispettivamente al maschio di taglia media e al maschio di taglia grande, ovvero 50° e 95° percentile antropometrico statistico, entrambi in posizione di occupante. Per un efficace confronto visivo tra le geometrie dei 2 modelli sono state processate in *Python* le rispettive mesh, isolando per ciascuno dei 2 THUMS un insieme di nodi afferenti alla pelle e uno alle ossa. Sovrapponendo in *Matlab* le coppie di insiemi omologhi a parità di baricentro della distribuzione sono stati ottenuti i grafici in Figura 2.8 e Figura 2.9, rappresentativi delle differenze tra THUMS AM50 e THUMS AM95 in termini di forma esterna del corpo (pelle) e interna (scheletro). Un confronto maggiormente dettagliato è riportato in Appendice, al Paragrafo 9.3.2.

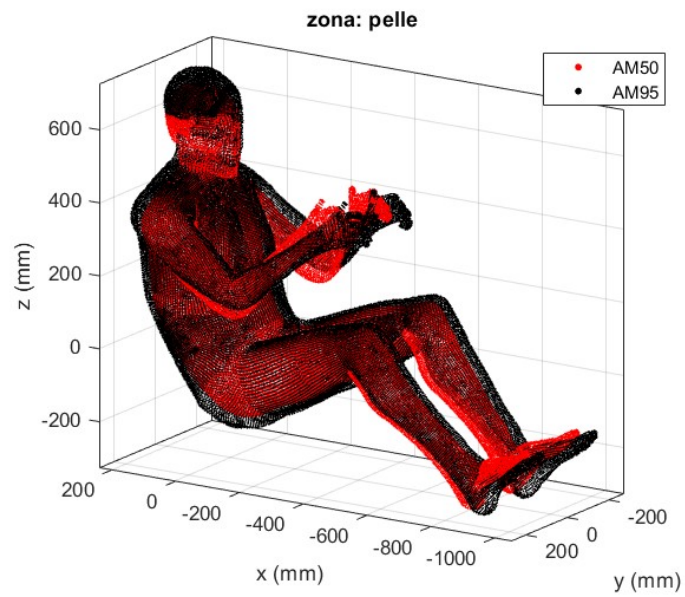


Figura 2.8: Confronto tra THUMS AM50 e THUMS AM95: forma interna

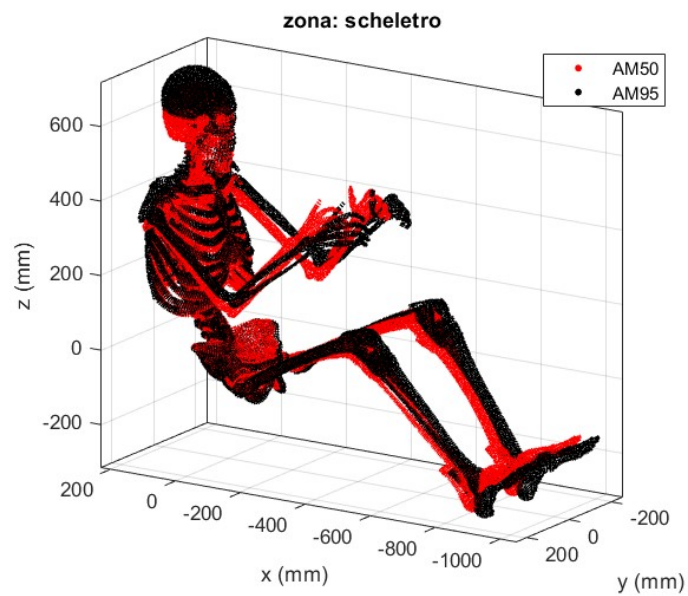


Figura 2.9: Confronto tra THUMS AM50 e THUMS AM95: forma esterna

2.2 Metodo degli elementi finiti

Il Metodo degli Elementi Finiti (FEM) è una tecnica numerica per la risoluzione di equazioni differenziali alle derivate parziali (PDEs) che basa il suo funzionamento sulla discretizzazione del dominio di calcolo in un insieme di più piccoli sotto-domini rappresentati dagli elementi finiti. Con il metodo degli elementi, si rinuncia a determinare la soluzione al continuo che soddisfi le PDEs, cercando invece la soluzione approssimata su un numero limitato di punti del dominio, ovvero i nodi degli elementi finiti. Nota la soluzione in tali punti, essa viene estesa infine a tutto il dominio colto dalla discretizzazione mediante funzioni approssimanti caratteristiche della tipologia di elementi finiti impiegati.

In questo paragrafo, procedendo per passi logici consecutivi, viene riportata qualitativamente la formulazione generale del metodo FEM applicato alla dinamica strutturale, d'interesse in questa trattazione, con l'obbiettivo di fornire un'idea del procedimento dietro la scrittura delle equazioni di governo.

- **Formulazione dell'elemento**

Viene definita la tipologia di elemento da adottare per realizzare la discretizzazione, ovvero il numero di nodi da cui è costituito, la posizione di tali nodi e i gradi di libertà associati ad essi.

- **Definizione delle funzioni approssimanti degli spostamenti interni**

Per ogni elemento vengono definite le funzioni la cui combinazione lineare sia in grado di esprimere (in maniera approssimata) il campo di spostamenti interno al dominio dell'elemento

- **Definizione del legame tra spostamenti interni e spostamenti nodali**

Per ogni elemento viene imposto il passaggio delle funzioni approssimanti gli spostamenti interni nei punti corrispondenti alle posizioni nodali e in tal modo definite le cosiddette "funzioni di forma", ovvero le funzioni che legano spostamenti nodali a spostamenti interni.

- **Definizione del legame deformazioni-spostamenti nodali**

Per ogni elemento viene definito il legame tra deformazioni dell'elemento e spostamenti nodali mediante operazioni di natura differenziale rappresentative delle equazioni di congruenza del problema strutturale.

- **Definizione del legame tensioni-spostamenti nodali**

Per ogni elemento viene definito il legame tra tensioni e deformazioni imponendo il legame costitutivo e di conseguenza, essendo già stato determinato il legame deformazioni-spostamenti nodali si ottiene il legame tensioni-spostamenti nodali.

- **Calcolo della matrice di rigidezza** Per ogni elemento, imponendo un principio variazionale, come ad esempio il *principio dei lavori virtuali*, rappresentativo dell'equilibrio³ integrale dell'elemento, si calcolano \mathbf{K}_{elem} e \mathbf{M}_{elem} , ovvero, rispettivamente, la matrice di rigidezza e la matrice di massa dell'elemento.

Assemblando le matrici di massa e rigidezza dei vari elementi nelle matrici di massa e rigidezza della struttura, indicate \mathbf{M} e \mathbf{K} , si ottiene l'equazione di governo come:

$$\mathbf{M}\ddot{\mathbf{x}} + \mathbf{C}\dot{\mathbf{x}} + \mathbf{K}\mathbf{x} = \mathbf{f} \quad (2.1)$$

dove \mathbf{f} rappresenta un vettore di carichi nodali e \mathbf{C} la matrice degli smorzamenti, per la quale determinazione non esiste un valido modello fisico e che pertanto viene spesso assunta proporzionale, attraverso coefficienti empirici, ad una combinazione di matrice di massa e rigidezza.

³Si parla di equilibrio statico quando si tratta di determinare la matrice di rigidezza e dinamico per la matrice delle masse

2.3 Analisi transitoria esplicita

I problemi della dinamica agli elementi finiti possono essere divisi in due categorie principali. La prima, corrispondente alle analisi dinamiche lineari, ha come obiettivo quello di determinare il comportamento in frequenza del sistema e in tal senso trae beneficio nel caratterizzare lo stesso attraverso le frequenze proprie di vibrazione e le corrispondenti forme modali. La seconda, corrispondente alle analisi transitorie, ha invece come obiettivo quello di determinare come il sistema si comporta nel tempo durante l'applicazione di un carico⁴ tale da indurre spiccate non linearità. È sull'analisi transitoria che si concentra l'attenzione in questo lavoro e in particolare su quella esplicita, di cui si discuterà a breve.

L'analisi transitoria può essere distinta in esplicita ed implicita in base agli algoritmi di calcolo che impiega nell'integrazione al passo. Entrambe le due tipologie presuppongono l'introduzione di schemi di discretizzazione temporale per integrare l'equazione di governo del sistema rispetto al time step Δt , con la differenza che:

- nei metodi espliciti la soluzione al tempo $t_{t+\Delta t}$ è ottenuta imponendo l'equazione di governo al tempo t e quindi estrapolata rispetto al time step senza dover introdurre processi iterativi di calcolo.
- nei metodi impliciti la soluzione al tempo $t_{t+\Delta t}$ è ottenuta imponendo l'equazione di governo al tempo $t_{t+\Delta t}$ stesso, il che rende necessario ricorrere a processi di convergenza iterativi quali *Newton-Raphson*.

L'una e l'altra tipologia di analisi presentano vantaggi e svantaggi che le rendono più o meno adatte in base alla fenomenologia degli eventi in esame. Le analisi esplicite, sebbene non necessitino di calcolo iterativo, pagano il basso onere computazionale della soluzione ad ogni istante con l'elevato numero di time step di simulazione richiesti: sono infatti necessari time step sufficientemente piccoli per limitare l'entità dell'approssimazione introdotta dallo schema di discretizzazione temporale e rispettare i requisiti di stabilità⁵. Le analisi implicite al contrario richiedono procedimenti di calcolo iterativi molto dispendiosi a livello computazione ma risultano incondizionatamente stabili, ovvero non pongono vincoli in merito al time step che almeno teoricamente⁶ può essere preso di qualsivoglia dimensione. Le caratteristiche introdotte rendono di fatto le analisi implicite preferibili in quegli eventi in cui le non linearità del sistema si sviluppano in tempi sufficientemente estesi e senza modificare troppo repentinamente la caratteristica del sistema (come prove di trazione quasi statiche e prove di creep). Le analisi esplicite viceversa trovano applicazione in eventi di breve durata (talvolta nell'ordine dei millisecondi) che raccontano una storia fortemente non lineare, come esplosioni, test balistici e urti. Il frontal sled test, ovvero il test di impatto frontale condotto nell'ambito di questo lavoro si inquadra certamente in tal senso.

2.3.1 Metodo delle differenze centrali

Il metodo delle differenze centrale costituisce uno degli schemi di integrazione esplicita maggiormente impiegati nei solutori commerciali. Si riporta a seguire una formalizzazione matematica del metodo a completamento delle informazioni sulle analisi transitorie esplicite. Sia l'equazione di governo del sistema data come:

$$\mathbf{M}_t \ddot{\mathbf{x}}_t + \mathbf{C}_t \dot{\mathbf{x}}_t + \mathbf{K}_t \mathbf{x}_t = \mathbf{f}(t) \quad (2.2)$$

con \mathbf{M}_t , \mathbf{C}_t e \mathbf{K}_t le matrici di massa, smorzamento e rigidezza; $\ddot{\mathbf{x}}_t$, $\dot{\mathbf{x}}_t$ e \mathbf{x}_t le accelerazioni, le velocità e gli spostamenti nodali; $\mathbf{f}(t)$ il vettore dei carichi agenti e il pedice t ad indicare la corrispondenza

⁴Il carico è generalmente variabile nel tempo e può essere inteso sia in termini di forze che di spostamenti

⁵Come sarà chiarito più avanti i metodi espliciti sono condizionatamente stabili, ovvero stabili solamente nel rispetto di un certo vincolo sul time step Δt di simulazione

⁶Time step troppo grandi sono comunque da evitare per questioni legate all'accuratezza della soluzione e alla convergenza: in presenza di forti non linearità che si sviluppano in tempi relativamente brevi il metodo di *Newton-Raphson* può entrare in crisi

delle grandezze introdotte con l'istante generico t . Assumendo costante il time step Δt , il metodo delle differenze centrali approssima accelerazioni e velocità all'istante considerato come

$$\ddot{\mathbf{x}}_t = \frac{1}{\Delta t^2} (\mathbf{x}_{t+\Delta t} - 2\mathbf{x}_t + \mathbf{x}_{t-\Delta t}) \quad (2.3)$$

e

$$\dot{\mathbf{x}}_t = \frac{1}{2\Delta t} (\mathbf{x}_{t+\Delta t} - \mathbf{x}_{t-\Delta t}) \quad (2.4)$$

che sostituite nella (2.2) producono l'equazione:

$$\frac{1}{\Delta t^2} \mathbf{M}_t (\mathbf{x}_{t+\Delta t} - 2\mathbf{x}_t + \mathbf{x}_{t-\Delta t}) + \frac{1}{2\Delta t} \mathbf{C}_t (\mathbf{x}_{t+\Delta t} - \mathbf{x}_{t-\Delta t}) + \mathbf{K}_t \mathbf{x}_t = \mathbf{f}(t) \quad (2.5)$$

Dalla (2.5) è possibile infine isolare il termine $\mathbf{x}_{t+\Delta t}$ come:

$$\mathbf{x}_{t+\Delta t} = \left(\frac{1}{\Delta t} \mathbf{M}_t + \frac{1}{2\Delta t} \mathbf{C}_t \right)^{-1} \left[\mathbf{f}(t) - \left(\mathbf{K}_t - \frac{1}{\Delta t^2} \right) \mathbf{x}_t - \left(\frac{1}{\Delta t} \mathbf{M}_t + \frac{1}{2\Delta t} \mathbf{C}_t \right) \mathbf{x}_t \right] \quad (2.6)$$

che inserito a sua volta nelle equazioni (2.3) e (2.4) permette di determinare completamente lo stato del sistema al tempo $t + \Delta t$. A beneficio del discorso sul basso onere computazionale legato al calcolo della soluzione associato agli schemi espliciti, si fa notare che adottando matrici di massa e smorzamento *lumped* (e quindi diagonali) l'inversione che appare nella 2.6 risulta piuttosto agevole e che in problemi che non prevedono l'erosione della mesh durante la simulazione, essa va eseguita una sola volta. In merito al time step Δt di simulazione si sottolinea infine che il metodo delle differenze centrali risulta condizionatamente stabile e in particolare, la soluzione risulta stabile solo se

$$\Delta t \leq \Delta t_{critico} = \frac{1}{\pi \nu_{max}} \quad (2.7)$$

dove ν_{max} rappresenta la più elevata frequenza propria del sistema. Ciò conduce a un'ultima ulteriore considerazione, ovvero che tanto più piccola è la dimensione degli elementi della mesh e quindi più accurata è la discretizzazione, tanto più elevata sarà la ν_{max} e quindi più piccolo il time step richiesto, con un aggravio significativo a livello computazione.

2.4 Radial Basis Function

Le radial basis function (RBF) sono una classe di funzioni a valori reali, con dominio in R^n , la cui forma generale può essere espressa come:

$$f(\mathbf{x}) = \varphi(\|\mathbf{x} - \mathbf{x}_k\|) \quad (2.8)$$

Dove \mathbf{x}_k è un punto noto del dominio della funzione detto centro della RBF, o source point, ed \mathbf{x} è il generico punto del dominio della RBF, o target point.

L'uso di funzioni di base radiale trova spazio in vari settori delle scienze applicate. In questa trattazione è tuttavia d'interesse il loro impiego nell'interpolazione di generiche funzioni a partire da un insieme di valori discreti. Sia $g : R^n \rightarrow R$ una funzione sufficientemente regolare di cui sono noti, nella distanza euclidea da N punti sorgente, un certo numero di valori discreti, è possibile costruire una funzione interpolante (tali valori) come combinazione lineare di N funzioni di base radiale e un termine polinomiale $h(\mathbf{x})$. Detta $s : R^n \rightarrow R$ la funzione interpolante suddetta, essa potrà quindi esprimersi come:

$$s(\mathbf{x}) = \sum_{i=1}^N \gamma_i \varphi(\|\mathbf{x} - \mathbf{x}_{ki}\|) + h(\mathbf{x}) \quad (2.9)$$

Si può dimostrare che se φ è condizionatamente definita positiva di ordine $m \leq 2$, allora la funzione interpolante esiste ed il termine polinomiale $h(\mathbf{x})$ è lineare. Scegliendo la tipologia di RBF in tal senso e supponendo $R^n = R^3$, coerentemente con le applicazioni di fit nello spazio d'interesse nell'ambito del mesh morphing, il termine polinomiale $h(\mathbf{x})$ risulterà:

$$h(\mathbf{x}) = \beta_0 + \beta_1 x + \beta_2 y + \beta_3 z \quad (2.10)$$

e l'equazione (2.9), di rimando:

$$s(\mathbf{x}) = \sum_{i=1}^N \gamma_i \varphi(\|\mathbf{x} - \mathbf{x}_{ki}\|) + \beta_0 + \beta_1 x + \beta_2 y + \beta_3 z \quad (2.11)$$

La funzione interpolante potrà pertanto essere determinata risolvendo un sistema lineare avente per $N + 4$ incognite i coefficienti γ delle N RBF e i coefficienti β del termine polinomiale. Le equazioni necessarie a costruire tale sistema risolvente possono essere derivate imponendo dapprima la condizione di passaggio della funzione interpolante per i source points, vale a dire:

$$s(\mathbf{x}_{ki}) = g_i \quad per \quad i = 1, \dots, N \quad (2.12)$$

e successivamente, imponendo le condizioni di ortogonalità dei contributi polinomiali. Il polinomio $h(\mathbf{x})$, infatti, non è legato all'esistenza della funzione interpolante sebbene, una volta determinato, ne garantisca l'unicità. La condizione appena introdotta può essere formulata come nella (2.13).

$$\sum_{i=1}^N \gamma_i (\beta_0 + \beta_1 x_i + \beta_2 y_i + \beta_3 z_i) = 0 \quad (2.13)$$

Chiamando:

$$[M]_{N \times N} = \begin{bmatrix} \varphi(\|\mathbf{x}_{k1} - \mathbf{x}_{k1}\|) & \cdots & \varphi(\|\mathbf{x}_{k1} - \mathbf{x}_{kN}\|) \\ \vdots & \ddots & \vdots \\ \varphi(\|\mathbf{x}_{k1} - \mathbf{x}_{kN}\|) & \cdots & \varphi(\|\mathbf{x}_{kN} - \mathbf{x}_{kN}\|) \end{bmatrix} \quad (2.14)$$

la matrice di interpolazione;

$$[P]_{N \times 4} = \begin{bmatrix} 1 & x_{k1} & y_{k1} & z_{k1} \\ 1 & x_{k2} & y_{k2} & z_{k2} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_{kN} & y_{kN} & z_{kN} \end{bmatrix} \quad (2.15)$$

la matrice derivata dai contributi del polinomio $h(\mathbf{x})$;

$$(\beta)_{1 \times 4}^T = (\beta_0 \quad \beta_1 \quad \beta_2 \quad \beta_3) \quad (2.16)$$

il vettore trasposto dei coefficienti del polinomio;

$$(\gamma)_{1 \times N}^T = (\gamma_1 \quad \gamma_2 \quad \cdots \quad \gamma_N) \quad (2.17)$$

il vettore trasposto dei pesi delle funzioni di base radiale e infine

$$(g)_{1 \times N}^T = (g_1 \quad g_2 \quad \cdots \quad g_N) \quad (2.18)$$

il vettore trasposto dei valori da interpolare, per l'equazione (2.12) sarà valida la scrittura in forma matriciale come:

$$[M \quad P] \begin{pmatrix} \gamma \\ \beta \end{pmatrix} = (g) \quad (2.19)$$

mentre l'equazione (2.13) risulterà verificata per

$$[P] (\beta) = (0) \quad (2.20)$$

e ancora, per

$$[P]^T (\gamma) = (0) \quad (2.21)$$

Combinando le equazioni (2.19), (2.20) e (2.21) è possibile ottenere in forma compatta il sistema risolvente come

$$\begin{bmatrix} M & P \\ P^T & 0 \end{bmatrix} \begin{pmatrix} \gamma \\ \beta \end{pmatrix} = \begin{pmatrix} g \\ 0 \end{pmatrix} \quad (2.22)$$

2.4.1 Mesh morphing basato sulle RBF

Nel mesh morphing basato sulle RBF, i source points sono l'insieme di nodi della griglia di calcolo ai quali viene imposto, secondo le esigenze specifiche, un campo di spostamenti nello spazio tridimensionale, mentre i target points sono i nodi di un insieme prescelto, che in virtù dello spostamento anzidetto, adatteranno il loro posizionamento. Ricorrendo alla teoria introdotta, è pertanto possibile risolvere il sistema (2.22) per:

$$\begin{cases} (\gamma)^x = (\gamma_1^x & \gamma_2^x & \cdots & \gamma_N^x) \\ (g)^x = (g_1^x & g_2^x & \cdots & g_N^x) \\ (\beta)^x = (\beta_0^x & \beta_1^x & \beta_2^x & \beta_3^x) \end{cases} \quad (2.23)$$

$$\begin{cases} (\gamma)^y = (\gamma_1^y & \gamma_2^y & \cdots & \gamma_N^y) \\ (g)^y = (g_1^y & g_2^y & \cdots & g_N^y) \\ (\beta)^z = (\beta_0^z & \beta_1^z & \beta_2^z & \beta_3^z) \end{cases} \quad (2.24)$$

$$\begin{cases} (\gamma)^z = (\gamma_1^z & \gamma_2^z & \cdots & \gamma_N^z) \\ (g)^z = (g_1^z & g_2^z & \cdots & g_N^z) \\ (\beta)^z = (\beta_0^z & \beta_1^z & \beta_2^z & \beta_3^z) \end{cases} \quad (2.25)$$

dove gli apici x , y e z sono identificativi degli assi del riferimento adottato per descrivere lo spazio tridimensionale e la componente i -esima del vettore $(g)^k$ per $k = x, y, z$, è pari allo spostamento assegnato all' i -esimo source point, valutato lungo la direzione k . Si ottiene in questo modo il campo vettoriale che definisce complessivamente la trasformazione della porzione di griglia di calcolo desiderata come:

$$\begin{pmatrix} s_x(\mathbf{x}) \\ s_y(\mathbf{x}) \\ s_z(\mathbf{x}) \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^N \gamma_i^x \varphi(\|\mathbf{x} - \mathbf{x}_{ki}\|) + \beta_0^x + \beta_1^x x + \beta_2^x y + \beta_3^x z \\ \sum_{i=1}^N \gamma_i^y \varphi(\|\mathbf{x} - \mathbf{x}_{ki}\|) + \beta_0^y + \beta_1^y x + \beta_2^y y + \beta_3^y z \\ \sum_{i=1}^N \gamma_i^z \varphi(\|\mathbf{x} - \mathbf{x}_{ki}\|) + \beta_0^z + \beta_1^z x + \beta_2^z y + \beta_3^z z \end{pmatrix} \quad (2.26)$$

Per completare questa introduzione all'uso delle funzioni di base radiale ai fini indicati, si riporta la Tabella 2.1 delle più comuni tipologie di RBF che soddisfano le ipotesi sotto le quali è stata determinata la (2.22). Per $r = \|\mathbf{x} - \mathbf{x}_k\|$:

Spline type	r^n , n <i>dispari</i>
Thin plate spline	$r^n \log(r)$, n <i>pari</i>
Multiquadratic (MQ)	$\sqrt{1 + r^2}$
Inverse multiquadratic (IMQ)	$\frac{1}{\sqrt{1+r^2}}$
Inverse quadratic (IQ)	$\frac{1}{1+r^2}$
Gaussian (GS)	e^{-r^2}

Tabella 2.1: funzioni di base radiali comuni

Capitolo 3

Strumenti e software

3.1 Ansys Workbench

Ansys Workbench [1] è la piattaforma sviluppata da *Ansys* per mettere in comunicazione i vari software di modellazione e analisi numerica che rientrano nel suo panorama applicativo, facilitando la gestione del progetto che ivi si articola attraverso vari step di lavoro. La sua caratteristica fondamentale è quella di consentire il trasferimento e la condivisione dei dati di input e output non solo tra i numerosi strumenti di pre e post processing disponibili, ma anche tra più modelli di simulazione (il che consente la facile implementazione di sofisticate simulazioni multi-fisiche). Se ne segnala l'utilizzo in questo lavoro come ambiente di supporto per *RBF Morph Structures*.

3.2 Ansys LS-DYNA

LS-DYNA [2] è un software di simulazione dinamica avanzata agli elementi finiti sviluppato dalla *Livermore Software Technology Corporation* (LSTC) di cui *Ansys* è proprietaria a partire dal 2019. *LS-DYNA* costituisce un leader nel settore dei solutori espliciti e il suo utilizzo trova ampiamente spazio in svariate tipologie di applicazioni, quali test di caduta, impatti, esplosioni e caratterizzazione di materiali non lineari in scenari di carico complessi. Esso costituisce inoltre l'ambiente di sviluppo di THUMS, che è distribuito nei canali ufficiali proprio sotto forma di codice sorgente di simulazione *LS-DYNA*, con estensione ".k"¹. Si segnala l'utilizzo di *LS-DYNA* in questo lavoro per quanto riguarda la simulazione di impatto frontale eseguita sulle varie configurazioni, morphate e non, di THUMS.

3.2.1 Ansys LS-Prepost

LS-PrePost è il software di pre e post-processing specifico delle simulazioni *LS-DYNA*. In merito al pre-processing, esso mette a disposizione molteplici strumenti che facilitano la creazione, la gestione, l'analisi e la modifica di complessi K-FILE di simulazione e in quest'ultima accezione, è stato utilizzato per predisporre il K-FILE di THUMS AM50 all'implementazione di una procedura automatizzata di mesh morphing. In merito al post-processing è stato invece impiegato per definire i risultati necessari a valutare la cinematica dell'occupante nella simulazione di impatto.

¹Si farà spesso riferimento al codice sorgente di simulazione in quanto a K-FILE

3.3 RBF Morph

RBF Morph è il software che permette di automatizzare la procedura di mesh morphing esposta nel paragrafo 2.4. Esso gestisce la matematica associata al processo di mesh morphing basato sulle RBF mediante algoritmi avanzati che ottimizzano i tempi di calcolo e la qualità dei risultati attendibili. Si segnala l'utilizzo di *RBF Morph* in questa trattazione nella versione *RBF Morph Structures* [16] supportata da *ANSYS Workbench* e sotto forma di libreria a collegamento dinamico nell'estensione *RBF_FGP.dll* (implementata in un codice *Python*).

3.3.1 RBF Morph Structures

RBF Morph Structures è l'estensione ACT di *RBF Morph*. Esso può essere installato in *Ansys Workbench*, dove presenterà la stessa struttura interattiva e logica di *Ansys Mechanical*: per impostare il problema alle RBF, l'utente crea all'interno dell'albero di analisi di Mechanical una struttura gerarchica di oggetti *parents* e *children*, contenenti le istruzioni per effettuare il morphing. In questo senso, la selezione dei nodi costituenti i target points e i source points è agevolata dalla possibilità di selezionare direttamente sulla mesh i nodi di interesse e la definizione del campo di spostamenti da interpolare con le RBF è automatizzata attraverso funzioni predefinite che l'utente può selezionare e compilare in base alla necessità. Una volta lanciata la simulazione, il mesh morphing viene eseguito prima del calcolo agli elementi finiti che risulta in tal modo effettuato sulla nuova configurazione morphata. Si fa notare che *RBF Morph Structures* può essere impiegato per la risoluzione del problema alle RBF anche indipendentemente dal *run* della simulazione. È possibile infatti esportare il mesh morphing risolto come file di estensione ".pts"² come nell'esempio in Figura 3.1.

```

95952
-849.19546614 70.39494648 -294.90052375 0.0 0.0 0.0 81000774 0 s p
-845.62186307 71.42938429 -299.46462542 0.0 0.0 0.0 81000775 0 s p
-842.2974642 73.22029407 -303.81782932 0.0 0.0 0.0 81000776 0 s p
-860.49343611 70.17685811 -280.8718095 0.0 0.0 0.0 81000782 0 s p
.....
..... righe omesse .....
.....
-45.08681934 -73.84500733 -53.71215484 -13.51345722 9.90261812 -1.467123371 83000163 0 s p
-32.26530084 -69.38350938 -43.24389832 -14.66478716 9.30795653 -3.433540426 83000164 0 s p
-30.00356969 -76.86993929 -63.03508162 -14.86788314 10.30580352 0.2841484078 83000168 0 s p
-37.24805478 -75.4725273 -65.45558356 -14.21735234 10.11954607 0.73882930 83000170 0 s p

```

Figura 3.1: Architettura del PTS-file identificativo del mesh morphing

La prima riga del PTS-file indica il numero di nodi coinvolti nel mesh morphing, ovvero quelli selezionati come target points; le righe successive identificano uno ad uno tali nodi attraverso posizione (prime 3 colonne) e numerazione utente (colonna 7), definendo infine il campo di spostamenti risultante dall'operazione di mesh morphing in maniera incrementale nelle 3 direzioni dello spazio (colonne 4, 5 e 6). Leggendo l'esempio: la prima riga indica che si ha a che fare con il mesh morphing di 95952 nodi. la seconda riga indica che al netto dell'operazione di mesh morphing il nodo 81000774 viene spostato dalle sue coordinate originarie di un valore pari a 0.0 lungo x , 0.0 lungo y e 0.0 lungo z ; l'ultima riga indica che al netto del mesh morphing il nodo 83000170 viene spostato di $-14.2173... mm$ lungo x , $10.1195... mm$ lungo y e $0.7388... mm$ lungo z .

3.3.2 Estensione RBF_FGP.dll

L'utilizzo di *RBF Morph* nell'estensione *RBF_FGP.dll* è analogo in quanto a definizione del problema alle RBF sebbene non benefici dell'interattività della versione *Structures* nelle varie fasi di impostazione. In generale l'utente fornisce in input un PTS-FILE contenente le informazioni relative

²In questo lavoro si farà spesso riferimento ai file ".pts" come PTS-file

ai source points (coordinate, numerazione, campo di spostamenti imposto) e definisce in coordinate l'insieme dei target points rispetto ai quali effettuare l'interpolazione alle RBF. Operando su questi dati *RBF_FGP.dll* esegue il mesh morphing fornendo in output i valori degli spostamenti incrementali che sommati alle coordinate dei target points produrranno la deformazione della mesh desiderata.

3.4 Linguaggi di programmazione

I linguaggi di programmazione impiegati in questo lavoro sono *Matlab* [12] e *Python* [14]. Sia *Matlab* che *Python* sono linguaggi ad alto livello ampiamente utilizzati in ambito scientifico e sebbene simili in molti aspetti hanno caratteristiche distintive. *Matlab* presenta una logica orientata al calcolo matriciale e in tal senso risulta molto efficace per gestire, organizzare e visualizzare dataset di grandi dimensioni. *Python* presenta invece una logica orientata agli oggetti e pur non essendo specializzato come *Matlab* vanta un ricco ecosistema di librerie in grado di fornire funzionalità avanzate in molteplici operazioni. In merito al ruolo ricoperto da entrambi in questo lavoro, l'uso di *Python* è stato necessario per automatizzare la procedura di mesh morphing di THUMS. In tal senso è stato preferito a *Matlab* per la maggior semplicità con la quale consente l'interazione rispetto al K-FILE di simulazione *LS-DYNA* e alla libreria *RBF_FGP.dll*. L'uso di *Matlab* è stato invece prezioso nell'ambito dell'elaborazione dei risultati e relativa visualizzazione grafica.

Capitolo 4

Criterio di verifica geometrica

Il presente lavoro ha come obiettivo quello di parametrizzare THUMS rispetto al percentile antropometrico statistico sfruttando il mesh morphing di THUMS AM50. Per verificare l'efficacia del mesh morphing è stato preso come riferimento THUMS AM95, ovvero la configurazione più critica raggiungibile in quanto limite superiore del dominio di parametrizzazione, e rispetto ad esso, effettuato un confronto geometrico. In tal senso è risultato necessario poter quantificare gli scostamenti tra i 2 THUMS e in particolare essere in grado di cogliere "in numeri" il confronto tra zone omologhe di mesh distinte.

4.1 Minima distanza media

Il primo dei 2 parametri introdotti per effettuare il confronto tra 2 mesh è la *minima distanza media*, abbreviando *MDA*, che può essere definita quale *la distanza media ottenuta rispetto alle distanze che intercorrono tra ciascun nodo della prima mesh e il più vicino dei nodi della seconda* e che serve a fornire una misura del grado di scostamento medio tra due geometrie: esso sarà tanto maggiore quanto più grande risulta la *MDA*. Formalizzando la definizione matematicamente, siano:

- $\mathbf{A} = \{\mathbf{x}_{A1}, \mathbf{x}_{A2}, \dots, \mathbf{x}_{An}\}$ una mesh di n nodi nota in coordinate fisiche
- $\mathbf{B} = \{\mathbf{x}_{B1}, \mathbf{x}_{B2}, \dots, \mathbf{x}_{Bm}\}$ una mesh di m nodi nota in coordinate fisiche
- $d_{min,i} = \min \{\|\mathbf{x}_{Ai} - \mathbf{x}_{Bj}\|\}$ la distanza minima misurabile tra il generico $\mathbf{x}_{Ai} \in \mathbf{A}$ e il generico $\mathbf{x}_{Bj} \in \mathbf{B}$

si definisce la *MDA* come:

$$MDA = \sum_1^n \frac{d_{min,i}}{n} \quad (4.1)$$

È importante far notare che in assenza di isotopologia tra le 2 mesh, la definizione appena fornita della *MDA* conduce a valori diversi in base alla scelta di quale delle due sia \mathbf{A} e quale \mathbf{B} . In questi casi è buona norma scegliere come mesh \mathbf{A} quella col minor numero di nodi. Per dare un esempio efficace di quanto asserito si considerino le mesh in Figura 4.1, ovvero 2 mesh rappresentanti la stessa geometria planare costruite in maniera tale che per ogni elemento della mesh1 (in rosso) corrispondano 4 elementi identici della mesh2 (in azzurro). Si prendano ora in esame le due seguenti casistiche:

- Caso1: mesh1 = mesh \mathbf{A} e mesh2 = mesh \mathbf{B}
- Caso2: mesh1 = mesh \mathbf{B} e mesh2 = mesh \mathbf{A}

Calcolando la MDA nel Caso1 si ottiene un valore pari esattamente all'offset tra le due geometria, mentre nel Caso2, un valore certamente maggiore. Immaginando ora di ridurre l'offset esistente tra le due geometrie fino a far coincidere la mesh2 sulla mesh1: nel Caso1 l' $MDA = 0$ coglierà esattamente la perfetta aderenza tra le due mesh mentre nel Caso2 valori dell' $MDA > 0$ per quanto piccoli, saranno meno efficaci nel quantificare il confronto.

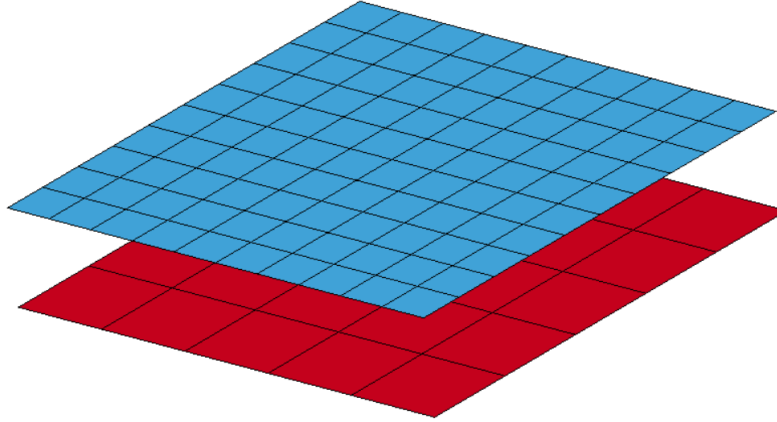


Figura 4.1: Stessa geometria discretizzata diversamente: numero di nodi della mesh2 (in azzurro) maggiore di quello della mesh1 (in rosso)

4.2 Massima distanza minima

Il secondo dei 2 parametri introdotti per effettuare il confronto tra 2 mesh è la *minima distanza media*, abbreviando MDM , che può essere definita quale *la distanza massima ottenuta rispetto alle distanze che intercorrono tra ciascun nodo della prima mesh e il più vicino dei nodi della seconda* e che serve a fornire una misura locale del grado di scostamento massimo tra due geometrie nella zona maggiormente critica: esso sarà tanto maggiore quanto più grande risulta la MDM . Formalizzando la definizione matematicamente, siano:

- $\mathbf{A} = \{\mathbf{x}_{A1}, \mathbf{x}_{A2}, \dots, \mathbf{x}_{An}\}$ una mesh di n nodi nota in coordinate fisiche
- $\mathbf{B} = \{\mathbf{x}_{B1}, \mathbf{x}_{B2}, \dots, \mathbf{x}_{Bm}\}$ una mesh di m nodi nota in coordinate fisiche
- $d_{min,i} = \min \{\|\mathbf{x}_{Ai} - \mathbf{x}_{Bj}\|\}$ la distanza minima misurabile tra il generico $\mathbf{x}_{Ai} \in \mathbf{A}$ e il generico $\mathbf{x}_{Bj} \in \mathbf{B}$

si definisce la MDM come:

$$MDM = \max \{d_{min,i}\} \quad (4.2)$$

Riguardo il calcolo della MDM valgono le stesse considerazioni in merito alla scelta di mesh \mathbf{A} e mesh \mathbf{B} nel calcolo della MDA .

4.3 THUMS AM50 e AM95 a confronto

Per le mesh relative alla pelle di THUMS AM50 e THUMS AM95, identificative della forma esterna dei due modelli, sono state distinte 14 zone omologhe, contrassegnate come: bacino, busto, viso,

spalla, braccio, gomito, avambraccio, mano, coscia, ginocchio, stinco, piede. Un esempio di confronto geometrico attraverso la *MDA* e la *MDM* è presto dato in merito alle zone in questione dei due THUMS. In Figura 4.2 sono riportati i valori della *MDA*. Si noti come essa non risulti mai inferiore a circa 1.5 *cm* e sia al più pari a circa 3 *cm* nella zona del bacino. In Figura 4.3 sono riportati invece valori della *MDM*, tutti piuttosto elevati. Rapportando *MDA* su *MDM* come in Figura 4.4, è possibile infine evidenziare come considerate tutte le zone della pelle il valore della *MDA*, o in altre parole lo scostamento medio tra le 2 geometrie, sia sempre almeno il 30% dello scostamento massimo rappresentato dalla *MDM*. Mediando i valori mostrati tra tutte le zone in esame è possibile infine ottenere quanto in evidenza in Tabella 4.1. Interpretando i valori: Lo scostamento medio tra la forma esterna di THUMS AM50 e THUMS AM95 è di circa 2 *cm* e come indicato dal valore di *MDA/MDM* pari a circa il 47%, esso è piuttosto diffuso globalmente.

	<i>MDA</i> [mm]	<i>MDM</i> [mm]	<i>MDA/MDM</i> [%]
bacino	30.74	43.40	70.81
busto	24.67	45.99	53.63
viso	14.73	44.06	33.44
...
...
stinco	14.88	30.47	48.81
caviglia	17.54	35.99	48.72
piede	14.30	39.68	36.02
media	19.75	42.45	46.94

Tabella 4.1: Forma esterna di THUMS AM50 e THUMS AM95 a confronto: *MDA*, *MDM* e *MDA/MDM*

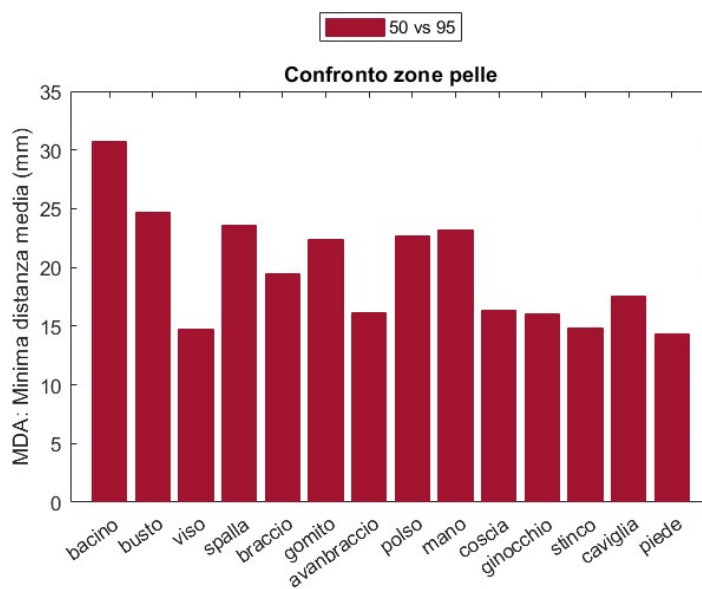


Figura 4.2: Valori della *MDA* nel confronto delle zone della pelle di THUMS AM50 e THUMS AM95

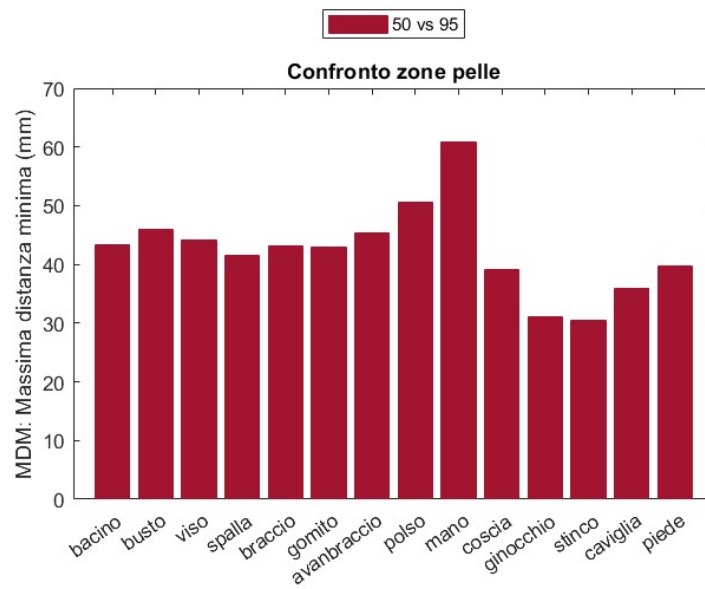


Figura 4.3: Valori della MDM nel confronto delle zone della pelle di THUMS AM50 e THUMS AM95

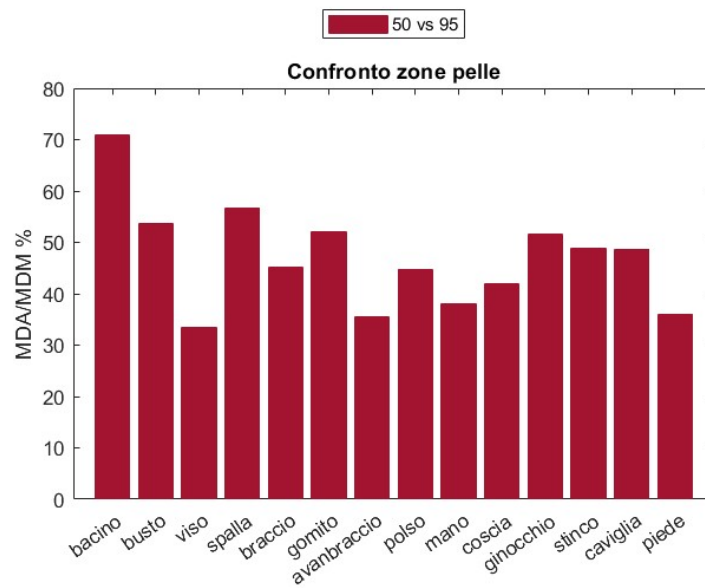


Figura 4.4: Valori del rapporto MDA/MDM nel confronto delle zone della pelle di THUMS AM50 e THUMS AM95

Capitolo 5

Criterio di mesh morphing

Il criterio di mesh morphing adottato in questo lavoro nasce dall'osservazione che sulla forma esterna di THUMS sono univocamente identificabili una serie di geometrie che complessivamente interpretano la forma del FE HBM in base al percentile antropometrico statistico. Si fa riferimento in particolare agli edges di connessione tra le varie zone della pelle, per i quali sono individuabili coppie di omologhi in THUMS AM50 e THUMS AM95 (esempio in Figura 5.1).

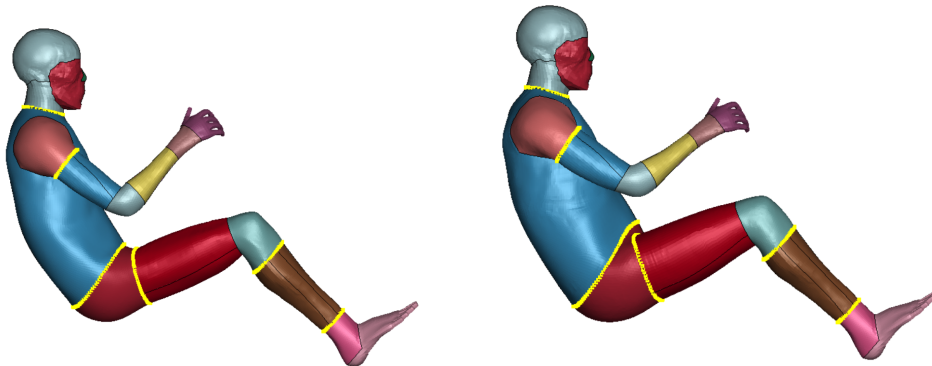


Figura 5.1: Coppie di alcuni edges omologhi in evidenza

Il criterio di mesh morphing adottato per parametrizzare THUMS nel percentile antropometrico statistico a partire da THUMS AM50 si basa sulla scelta dei nodi appartenenti ai suoi edges come source points del problema alle RBF e si articola in due passi:

1. Definire il mesh morphing di THUMS AM50 in THUMS AM95, ovvero il mesh morphing in grado di generare THUMS AM50m95 (da leggere "AM50 morphato al 95° percentile")
2. Ottenere la parametrizzazione di THUMS in base al percentile antropometrico statistico modulando il campo di spostamenti rappresentativo del mesh morphing tra AM50 e AM95.

5.1 Definizione del mesh morphing da AM50 ad AM95

Per definire il mesh morphing in grado di generare THUMS AM50m95 vengono innanzitutto scelti come source points i nodi corrispondenti a 22 edges di THUMS AM50, come in Figura 5.2. Gli edges omologhi vengono successivamente identificati in THUMS AM95 (Figura 5.3).

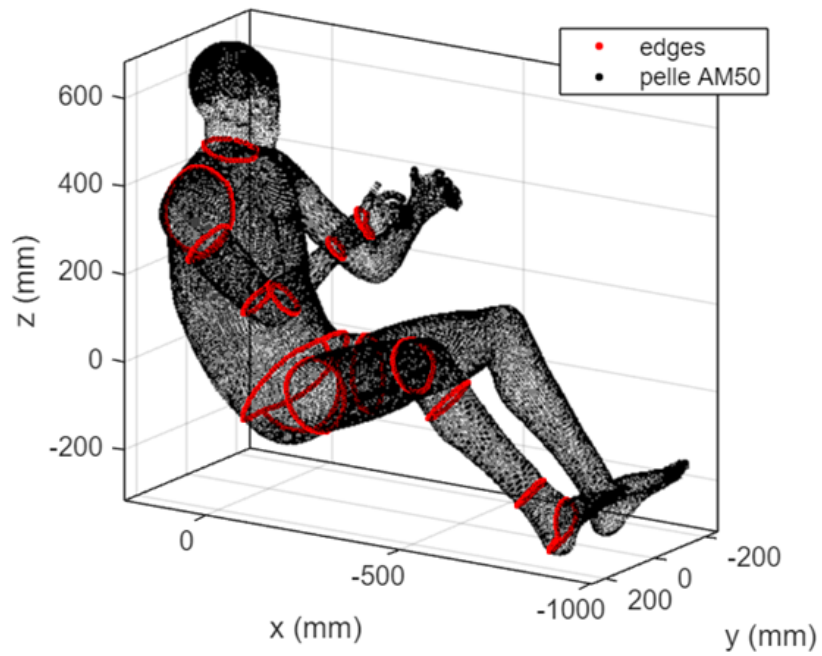


Figura 5.2: Definizione dei 22 edges corrispondenti ai source points: rappresentati solo quelli relativi alla parte destra

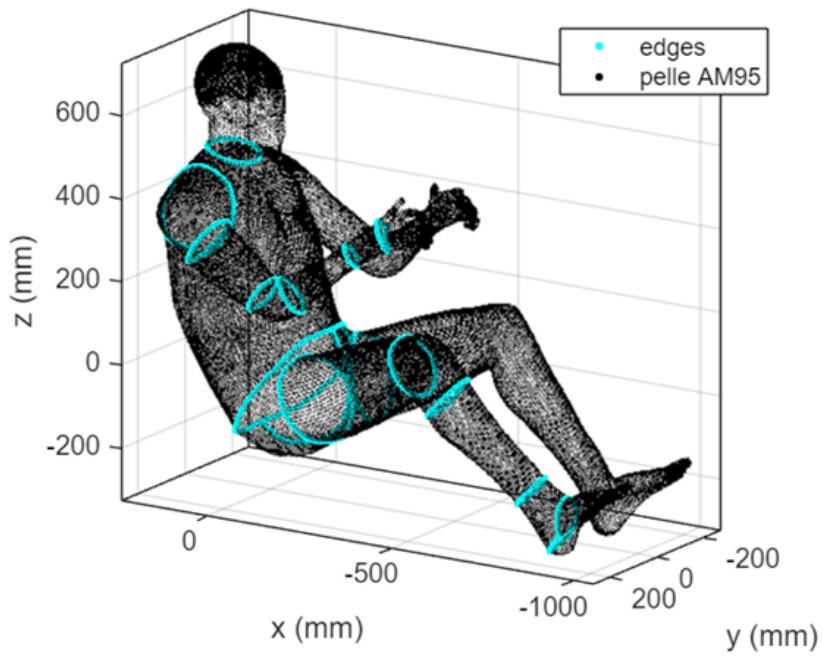


Figura 5.3: Identificazione dei 22 edges omologhi: rappresentati solo quelli relativi alla parte destra

Confrontando a coppie di omologhi ciascuno degli edges selezionati viene infine calcolato il campo di spostamenti da interpolare nel mesh morphing alle RBF, ovvero il campo di spostamenti da assegnare a ciascun source point. In merito a tale calcolo, siano:

- $\mathbf{x}_{50} = \{x_{50,1}, \dots, x_{50,n}\}$, $\mathbf{y}_{50} = \{y_{50,1}, \dots, y_{50,n}\}$ e $\mathbf{z}_{50} = \{z_{50,1}, \dots, z_{50,n}\}$ i vettori delle coordinate dei nodi appartenenti al generico edge di THUMS AM50, ovvero le coordinate del generico set di source points
- $\mathbf{x}_{95} = \{x_{95,1}, \dots, x_{95,n}\}$, $\mathbf{y}_{95} = \{y_{95,1}, \dots, y_{95,n}\}$ e $\mathbf{z}_{95} = \{z_{95,1}, \dots, z_{95,n}\}$ le coordinate dei nodi appartenenti all'omologo edge di THUMS AM95
- $\mathbf{s}_{RBF,i} \forall i = 1, \dots, n$ lo spostamento incognito da assegnare al generico source point

La prima operazione da effettuare consiste nella traslazione dei source points affinché il baricentro della distribuzione che essi costituiscono corrisponda al baricentro dell'omologo edge di THUMS AM95. Si calcolano in tal senso i delta di traslazione come:

$$\Delta_x = \text{mean} \{ \mathbf{x}_{95} \} - \text{mean} \{ \mathbf{x}_{50} \} \quad (5.1)$$

$$\Delta_y = \text{mean} \{ \mathbf{y}_{95} \} - \text{mean} \{ \mathbf{y}_{50} \} \quad (5.2)$$

$$\Delta_z = \text{mean} \{ \mathbf{z}_{95} \} - \text{mean} \{ \mathbf{z}_{50} \} \quad (5.3)$$

La seconda operazione da effettuare consiste nello scaling dei source points. È importante che lo scaling non venga effettuato nel riferimento globale ma in quello locale, centrato nel baricentro della distribuzione dei source points. A tal proposito, dette:

$$x_{50,c} = \text{mean} \{ \mathbf{x}_{50} \} \quad (5.4)$$

$$y_{50,c} = \text{mean} \{ \mathbf{y}_{50} \} \quad (5.5)$$

$$z_{50,c} = \text{mean} \{ \mathbf{z}_{50} \} \quad (5.6)$$

le coordinate del baricentro dei source points nel riferimento locale, i vettori delle coordinate dei source points nel riferimento locale baricentrico risulteranno:

$$\mathbf{x}_{50,loc} = \{x_{50,1} - x_{50,c}, \dots, x_{50,n} - x_{50,c}\} \quad (5.7)$$

$$\mathbf{y}_{50,loc} = \{y_{50,1} - y_{50,c}, \dots, y_{50,n} - y_{50,c}\} \quad (5.8)$$

$$\mathbf{z}_{50,loc} = \{z_{50,1} - z_{50,c}, \dots, z_{50,n} - z_{50,c}\} \quad (5.9)$$

A questo punto, una volta calcolati i fattori di scaling¹ come:

$$S_x = \frac{\max \{ \mathbf{x}_{95} \} - \min \{ \mathbf{x}_{95} \}}{\max \{ \mathbf{x}_{50} \} - \min \{ \mathbf{x}_{50} \}} \quad (5.10)$$

$$S_y = \frac{\max \{ \mathbf{y}_{95} \} - \min \{ \mathbf{y}_{95} \}}{\max \{ \mathbf{y}_{50} \} - \min \{ \mathbf{y}_{50} \}} \quad (5.11)$$

$$S_z = \frac{\max \{ \mathbf{z}_{95} \} - \min \{ \mathbf{z}_{95} \}}{\max \{ \mathbf{z}_{50} \} - \min \{ \mathbf{z}_{50} \}} \quad (5.12)$$

sarà infine possibile calcolare lo spostamento da assegnare al generico source point come somma di due contributi, uno legato alla traslazione ed uno allo scaling:

$$\mathbf{s}_{RBF,i} = \begin{pmatrix} x_{50,i,loc} (S_x - 1) + \Delta_x \\ y_{50,i,loc} (S_y - 1) + \Delta_y \\ z_{50,i,loc} (S_z - 1) + \Delta_z \end{pmatrix} \forall i = 1, \dots, n \quad (5.13)$$

¹Sebbene lo scaling sia da effettuare nel riferimento locale, i fattori di scaling possono essere comunque determinati in base alle coordinate nel riferimento globale

Sebbene apparentemente complicato, il calcolo dei source points segue un principio ben preciso, che può essere definito quale *uguaglianza del minimo inviluppo*. Sia infatti $I50_i$ il più piccolo parallelepipedo² in grado di circoscrivere l' i -esimo edge (nonché insieme di source points) di THUMS AM50 e sia $I95_i$ il più piccolo parallelepipedo in grado di circoscrivere l'omologo edge di THUMS AM95: gli spostamenti del problema alle RBF sono calcolati $\forall i$ come gli spostamenti in grado di traslare e scalare $I50_i$ fino a farlo coincidere con $I95_i$.

Definire il campo di spostamenti da assegnare ai source points è il fulcro del mesh morphing mediante RBF. I source points sono infatti gli unici nodi sui quali si ha direttamente controllo nel processo di mesh morphing, essendo che il campo di spostamenti loro assegnato è l'unico direttamente definito e controllato dall'utente³. Lo spostamento dei target points, di cui i source points sono un sottoinsieme, costituisce infatti l'interpolazione operata per mezzo delle RBF e risulta variabile in base alla tipologia di RBF adottata per tutti i nodi della mesh eccetto i source points, che si sposteranno per l'appunto come prefissato. Se, come nel caso specifico, il campo di spostamenti assegnato ai source points, ovvero gli edges di THUMS AM50, deve essere tale da garantire la miglior approssimazione possibile di una determinata geometria target, ovvero gli edges di THUMS AM95, è importante che ciò sia verificato. In Figura 5.4 sono mostrati a titolo di esempio, alcuni degli edges di THUMS AM50 selezionati come source points, gli omologhi di THUMS AM95 e come risulteranno gli edges di THUMS AM50m95 (abbreviato AM50m) una volta eseguito il mesh morphing. Si riporta il confronto completo in Appendice, Paragrafo 9.3.1.

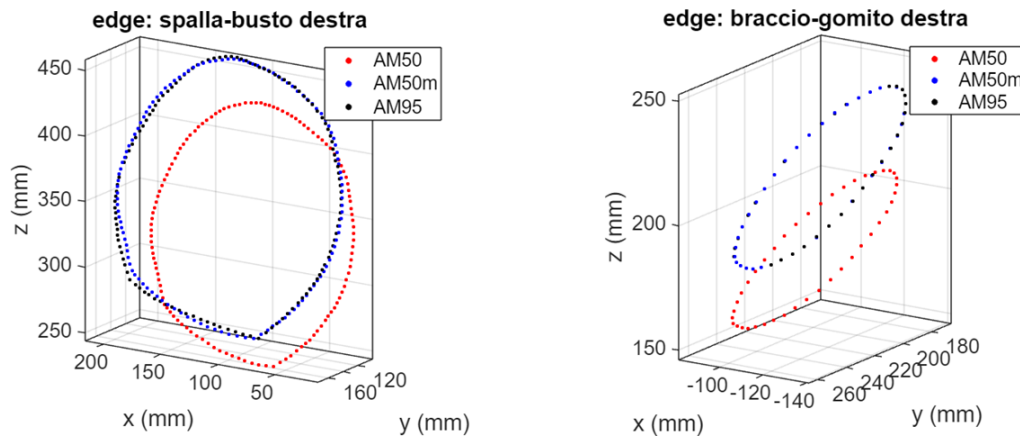


Figura 5.4: Confronto tra edges prima e dopo il morphing

Sempre in relazione agli edges di THUMS AM50, AM95 e AM50m si riportano i valori di MDA (Figura 5.5), MDM (Figura 5.6) e MDA/MDM (Figura 5.7) risultanti dal confronto tra AM50m e AM95, tra AM50 e AM95 e tra AM50 e AM50m. Si noti che nel mesh morphing, i valori dell' MDA passano dall'ordine del centimetro (AM50 vs AM95) a quello del millimetro (AM50m vs AM95) per quanto riguarda tutti gli edges eccetto quello contrassegnato *bacino*. Sempre in merito alla MDA risultano interessanti i valori del confronto AM50 vs AM50m, che decisamente più simili a quelli del confronto AM50 vs AM95 piuttosto che all'AM50m vs AM95, indicano quanto mediamente lo scostamento tra la geometria originaria e la geometria target sia molto simile allo scostamento tra la geometria originaria e quella morphata. I valori della MDM dopo il morphing (AM50m vs AM95) sono anch'essi molto bassi in merito a quasi tutti gli edges. Fa ancora una volta eccezione il *bacino* che sebbene presenti valori elevati della MDM trova un certo sostegno dai valori del rapporto MDM/MDA . Esso indica

²Per non creare ambiguità: si intende il parallelepipedo le cui facce siano orientate secondo gli assi del riferimento globale adottato

³Fanno in realtà eccezione a questo discorso alcune tipologie di mesh morphing alle RBF guidato da algoritmi evolutivi come il BGM, tuttavia non d'interesse in questa trattazione

che dopo il mesh morphing lo scostamento medio dalla geometria target dell'edge *bacino* è il più basso di tutti in relazione allo scostamento massimo, ovvero l'edge *bacino* è quello in cui la discordanza geometrica è meno diffusa. Concentrando l'attenzione solo sul confronto tra AM50 ed AM95, si noti infine come mediando i valori di MDA , MDM ed MDA/MDM su tutti gli edges esaminati si ottengano i valori in Tabella 5.1: lo scostamento medio, mediamente registrato nei source points tra la geometria morphata e la geometria target è di circa 2.3 mm , a fronte di uno scostamento massimo mediamente pari a 5.5 mm . È corretto osservare come tali valori siano davvero esigui in relazione alla scala dimensionale complessiva del modello.

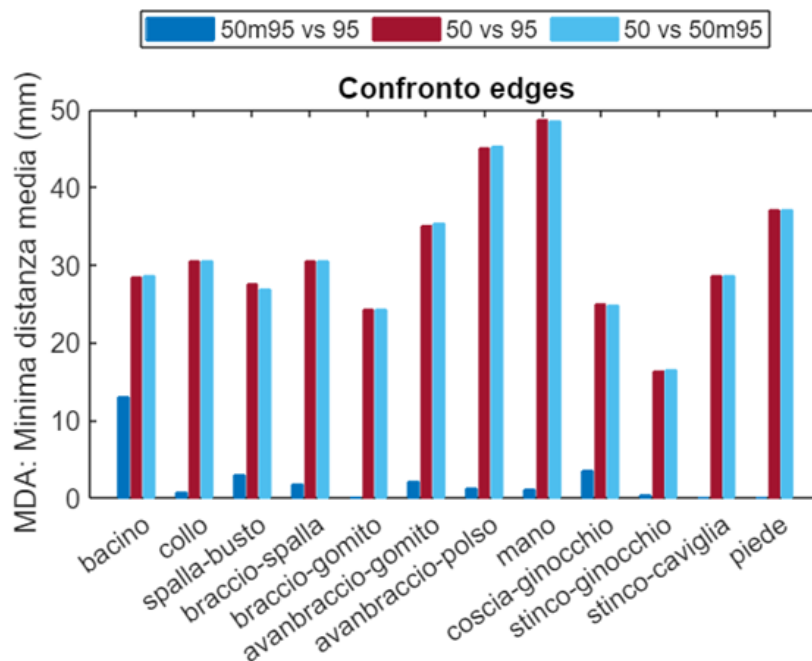


Figura 5.5: Valori della MDA a confronto

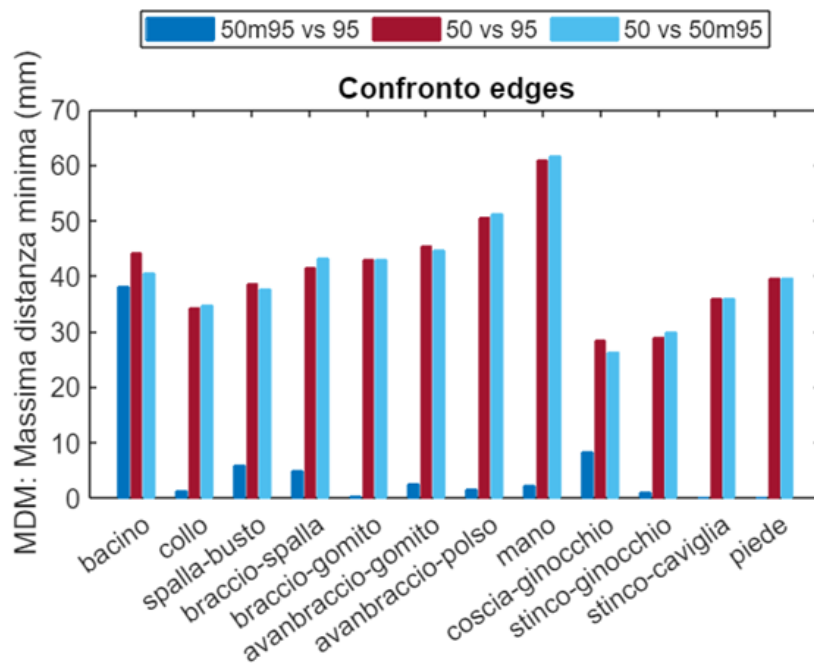


Figura 5.6: Valori della *MDM* a confronto

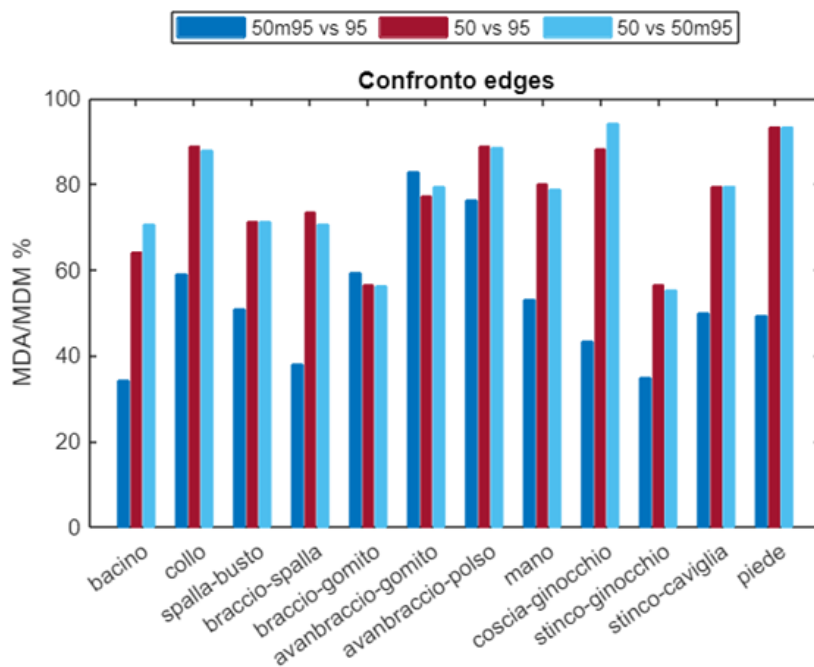


Figura 5.7: Valori del rapporto *MDA/MDM* a confronto

AM50m vs AM95			
	<i>MDA</i> [mm]	<i>MDM</i> [mm]	<i>MDA/MDM</i> [mm]
bacino	13.04	38.09	34.24
collo	0.83	1.42	59.07
spalla-busto	3.03	5.97	50.81
...
...
stinco-ginocchio	0.36	1.05	34.80
stinco-caviglia	$7.09e - 5$	$1.42e - 4$	49.97
piede	$7.43e - 5$	$1.50e - 5$	49.41
media	2.30	5.55	52.63

Tabella 5.1: Valori medi di *MDA*, *MDM* e *MDA/MDM* nel confronto tra AM50m ed AM95

5.2 Parametrizzazione

Per parametrizzare THUMS in base al percentile antropometrico statistico si parametrizza a sua volta il mesh morphing alle RBF da applicare a THUMS AM50. Se infatti il campo di spostamenti calcolato al paragrafo precedente è il campo di spostamenti da applicare ai source points per determinare il mesh morphing dal 50° al 95° percentile, allora esso può altrettanto essere inteso quale lo scostamento esistente tra il 50° e il 95° %ile antropometrico valutato nei source points. Posta questa assunzione, sarà chiaro che prendendo in considerazione un percentile antropometrico diverso dal 95°, ad esempio il 75°, l'entità dello scostamento dal 50° percentile sarà inferiore. Se si considera poi il 65° percentile sarà ancora più piccola, per il 55° sempre più piccola e al limite nulla se si considera lo scostamento tra il 50° e il 50° stesso. Per determinare qualsiasi percentile antropometrico statistico tra il 50° e il 95° si è quindi scelto di interpolare linearmente tra i due valori il parametro δ , detto di modulazione, tale che posti:

- $p_{50} = 50$ e $p_{95} = 95$ il 50° e il 95° percentile antropometrico statistico
- $s_{RBF,95}$ il campo di spostamenti complessivamente assegnato ai source points per determinare il mesh morphing dal 50° al 95° %ile
- p il generico percentile
- $s_{RBF,p}$ il campo di spostamenti complessivamente assegnato ai source points per determinare il mesh morphing dal 50° al percentile p

si ottenga

$$s_{RBF,p} = \delta \cdot s_{RBF,95} \quad (5.14)$$

con

$$\delta = \frac{p - p_{50}}{p_{95} - p_{50}} \quad (5.15)$$

Capitolo 6

Mesh morphing semplificato

Il presente capitolo è interamente dedicato all'implementazione del mesh morphing definito in precedenza su un modello semplificato di THUMS AM50, ovvero THUMS AM50s (da "semplificato") costituito solamente da pelle e ossa (Figura 6.1).

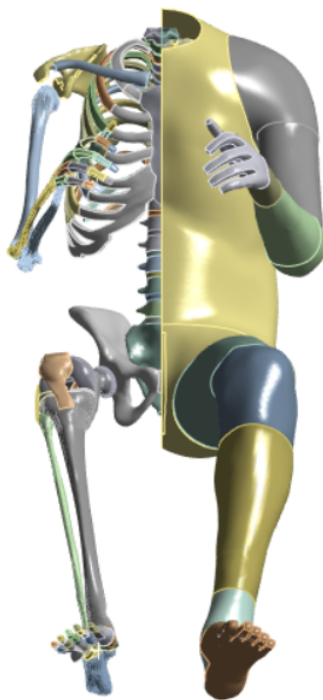


Figura 6.1: THUMS AM50s

Lavorare su un modello semplificato, ha consentito di implementare il primo tentativo di mesh morphing in ambiente *Ansys Workbench*, dove è stato possibile beneficiare in tal senso degli strumenti di selezione e visualizzazione forniti da *RBF Morph Structures*. Eseguendo il mesh morphing, è stato inoltre possibile avere una prima stima, sebbene non quantitativa, di cosa succede internamente a THUMS, ovvero allo scheletro, nel contesto di un mesh morphing guidato da source points posti esternamente, ovvero sulla pelle.

6.1 THUMS semplificato

Per realizzare THUMS AM50s è stato necessario modificare il K-FILE di simulazione attraverso il quale viene reso disponibile THUMS AM50, in modo tale da renderlo compatibile in termini di dimensioni in memoria con *Ansys Workbench*. Sfruttando l'interfaccia di pre e post processing associata ad *LS-DYNA*, vale a dire *LS Prepost* si è dunque agito in tal senso, eliminando dal K-FILE:

- Tutti gli elementi *asta* e *seatbelt* rappresentativi del sistema sistema muscoli-tendini
- Tutti gli elementi *solid* ad eccezione di quelli costituenti le ossa
- Tutti gli elementi shell ad eccezione di quelli costituenti la pelle

Sfruttando un generico visualizzatore di file testo sono state successivamente rimosse dal K-FILE tutte le Keyword (e relative *cards*) eccetto quelle strettamente necessarie a definire il collegamento tra i nodi degli elementi *solid*, il collegamento tra i nodi degli elementi shell, i nodi stessi (numerazione e coordinate) e i set di nodi che identificano le varie parti del modello (senza le quali non sarebbe possibile oltretutto riconoscere alcun edge sulla mesh della pelle). Al termine delle precedenti operazioni, è stato ottenuto il K-FILE di THUMS AM50s come in Figura 6.2.

```

$# LS-DYNA Keyword file created by LS-PrePost(R) V4.10.3-21Apr2023
$# Created on Dec-18-2023 (00:39:33)
*KEYWORD
*ELEMENT_SOLID
$#   eid      pid      n1      n2      n3      n4      n5      n6      n7      n8
81000001810000018106962381069008810685848106946181069461810694618106946181069461
..... righe omesse .....

*ELEMENT_SHELL
$#   eid      pid      n1      n2      n3      n4      n5      n6      n7      n8
810728608120000181017000810202808101576581023067      0      0      0      0
..... righe omesse .....

*NODE
$#   nid      x      y      z      tc      rc
86009170      -323.3849      -142.3567      318.8443      0      0
..... righe omesse .....

*PART_AVERAGED
$HNAME COMPS 7121100M_Psoas_1_R
$PR_PART_COL 7121100 f800556f
$#
$#
M_Psoas_1_R
$#   pid      secid      mid      eosid      hgid      grav      adpopt      tmid
      7121100      7121100      7121100      0      0      0      0      0
..... righe omesse .....
*END

```

Figura 6.2: K-FILE di THUMS AM50s

Operando allo stesso modo è stato ottenuto THUMS AM95s, necessario ai fini del mesh morphing nei termini mostrati al precedente capitolo. Per dare una misura delle semplificazioni apportate ai 2 modelli di THUMS si fa presente che le dimensioni dei K-FILE passano da circa 240MB a 40MB.

6.2 Implementazione del mesh morphing

Nel presente capitolo, si espone il procedimento adottato nell'implementazione del mesh morphing su THUMS AM50s. A partire dall'importazione dei modelli semplificati su *Ansys Workbench* fino al completamento del Set up di *RBF Morph Structures*, si illustrerà tutto il percorso seguito.

6.2.1 Importazione di THUMS AM50s e THUMS AM95s

Per importare THUMS AM50s e THUMS AM95s in Workbench è stato prima di tutto selezionato dalla finestra dei *Toolbox* l'oggetto *External Model*. Dall'interfaccia di *External Model* sono stati poi definiti i percorsi-file ai K-FILE di THUMS AM50s e THUMS AM95s e impostate le unità di misura rispetto alle quali effettuare l'import. Abbinando infine l'oggetto *Mechanical Model* all'*External Model* come mostrato in Figura 6.3 è stata infine completata l'importazione dei 2 modelli semplificati.

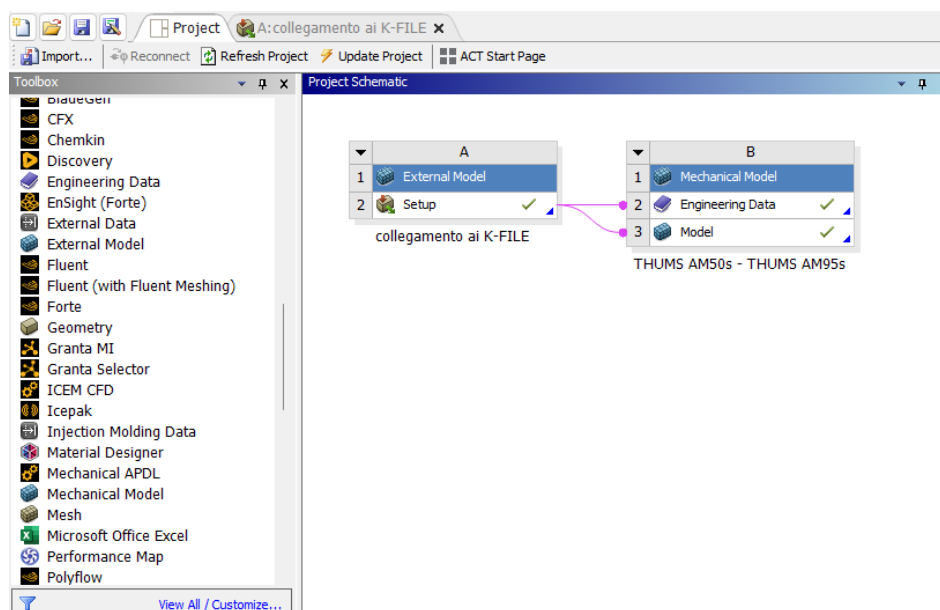


Figura 6.3: Struttura del modello *Workbench*

6.2.2 Export dei source points

Come già discusso al Paragrafo 5, per determinare il campo di spostamenti da assegnare ai source points è necessario poter disporre delle coordinate dei nodi selezionati come tali nell'AM50 e degli omologhi in AM95. Per fare ciò sono state create in ambiente *Model*, alla voce *Construction geometry*, tante *path* quanti sono gli edges coinvolti nella definizione del mesh morphing (Figura 6.4). Una ad una, ciascuna *path* è stata infine esportata come file di testo contenente informazioni circa le coordinate di ciascun nodo intercettato dalla *path*.

6.2.3 Calcolo degli spostamenti

Procedendo come mostrato al Paragrafo 5.1, vengono calcolati gli spostamenti da assegnare ai source points nel problema alle RBF. Come sarà chiaro a breve, RBF Morph Structures implementa come predefinite, da compilare, le funzioni *Translation* e *Scaling*, ovvero le operazioni sufficienti ad imporre il criterio di mesh morphing. Poiché per settare la funzione *Scaling* sono necessari 3 fattori di scala e un sistema di riferimento e per settare la funzione *Translation* sono necessari 3 delta di traslazione, il calcolo degli spostamenti in esame si è limitato, per ogni set di source points, al calcolo di Δ_x , Δ_y , Δ_z , S_x , S_y e S_z , rispettivamente: equazioni (5.1),(5.2),(5.3),(5.10),(5.11) e (5.12). Si riportano in Tabella 6.1 i valori dei delta di traslazione e dei fattori di scala calcolati (per tutti quei source points anatomicamente simmetrici, i.e mano, piede, ecc., si riporta soltanto la parte destra del corpo).

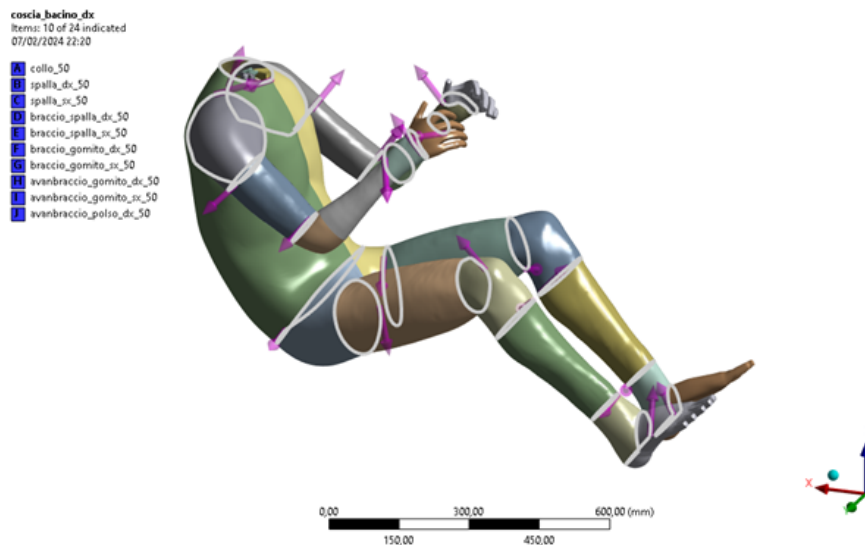


Figura 6.4: Definizione delle *path* corrispondenti agli edges d'interesse

	Δ_x [mm]	Δ_y [mm]	Δ_z [mm]	S_x [-]	S_y [-]	S_z [-]
bacino	-4.0721	0.3843	-0.4119	1.1598	1.2021	1.3214
collo	2.8212	-0.0183	30.4231	1.0535	1.0679	1.2263
spalla-busto	13.4026	10.7607	27.8519	1.112	1.1046	1.0502
braccio-spalla	8.619	14.4623	31.6232	1.0509	1.0874	1.2142
braccio-gomito	-12.346	15.0458	34.253	0.9814	1.0905	1.1416
avanbraccio-gomito	-23.8521	13.771	29.0997	0.8738	1.0292	1.1885
avanbraccio-polso	-46.489	6.9955	12.7241	0.8842	1.0493	1.1787
mano	-58.5998	3.2587	4.1245	0.9125	1.0781	1.1797
coscia-ginocchio	-17.1353	8.8417	13.2061	1.3429	1.1056	1.1081
stinco-ginocchio	-20.5143	-8.7046	13.5035	1.0613	1.070	1.0623
stinco-caviglia	-34.043	6.2158	0.4994	1.070	1.0699	1.0699
piede	-39.1372	5.0658	-4.449	1.07	1.07	1.07

Tabella 6.1: Delta di traslazione e fattori di scaling

6.2.4 RBF Morph Set Up

Per definire il mesh morphing attraverso *RBF Morph Structures* è stato creato l'*RBF Morph Set Up* come mostrato in Figura 6.5. L'*RBF Morph Set Up* è composto da una struttura gerarchica di oggetti *parents* e *children* legati secondo la logica che i nodi dei *children* corrispondano ai source points del problema alle RBF avente come target points i nodi dei *parents*. Nel caso specifico è stato definito un primo oggetto *parents* corrispondente a tutti i nodi di THUMS AM50s. Per tale oggetto, chiamato *morphing_completo*, sono stati creati 22 *children*, ovvero tanti quanti sono gli edge costituenti i source points. Per ognuno di questi *children*, infine, è stato creato un ulteriore *child*, ottenendo una struttura parentale complessiva a 3 livelli.

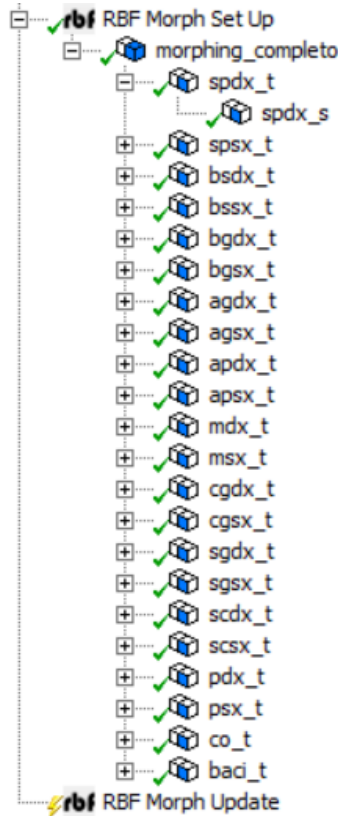


Figura 6.5: RBF Morph Set Up

Entrando nel merito delle funzioni svolte da ciascun livello, il terzo livello definisce lo *Scaling* dei source points ed ha come target points i source points stessi. Il secondo livello definisce la *Translation* dei source points, che per come è impostata la gerarchia di mesh morphing, viene effettuata sugli stessi già scalati. Il primo livello, infine, interpola su tutti i nodi del modello gli spostamenti complessivamente definiti ai livelli sottostanti. Si considerino a titolo di esempio i source points appartenenti all'edge *spalla-busto destra*. Per definire lo *Scaling* dell'edge, che si ricorda essere previsto nel riferimento locale baricentrico della distribuzione di source points (Figura 6.7), viene utilizzata l'apposita funzione (Figura 6.6), compilata secondo i valori in Tabella 6.1 precedentemente mostrati. Per definire la funzione *Translation* (Figura 6.6) si procede analogamente. Definendo lo *Scaling* quale *child* della *Translation*, quest'ultima viene effettuata sui source points già scalati, come in Figura 6.8: la combinazione di *Scaling* e *Translation* mostrata, costituisce il contributo di *spalla-busto destra* agli spostamenti del problema alle RBF. Estendendo il discorso a tutti i rimanenti edges - source points, si definisce infine il campo di spostamenti complessivo da interpolare alle RBF (Figura 6.9).

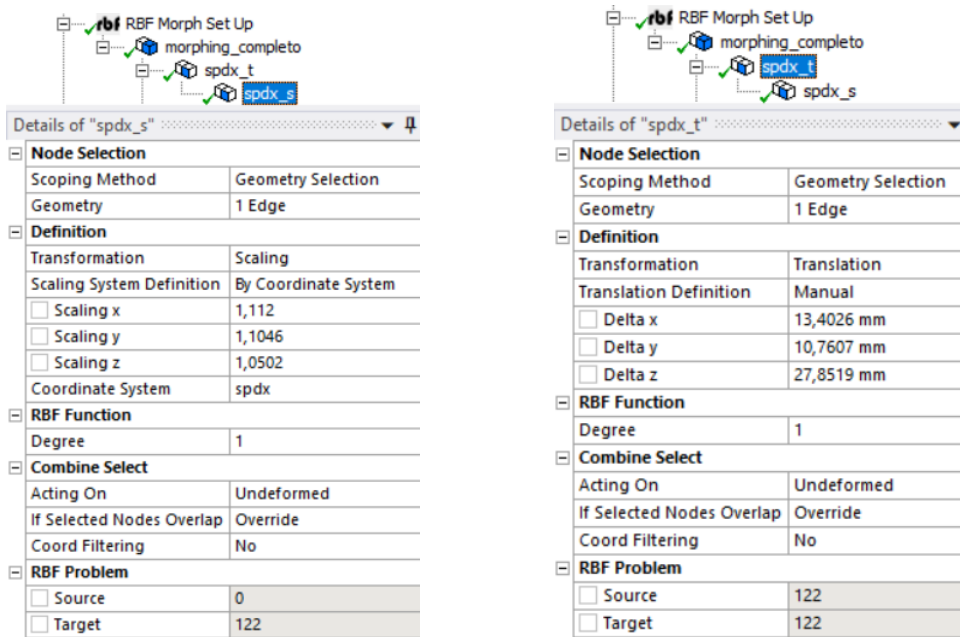


Figura 6.6: Funzioni di *Scaling* e *Translation*

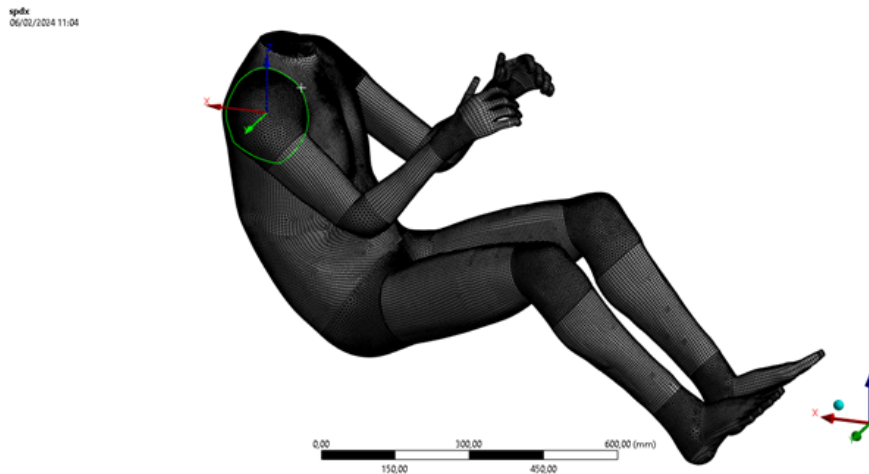


Figura 6.7: Riferimento locale baricentrico dell'edge *spalla-busto destra*

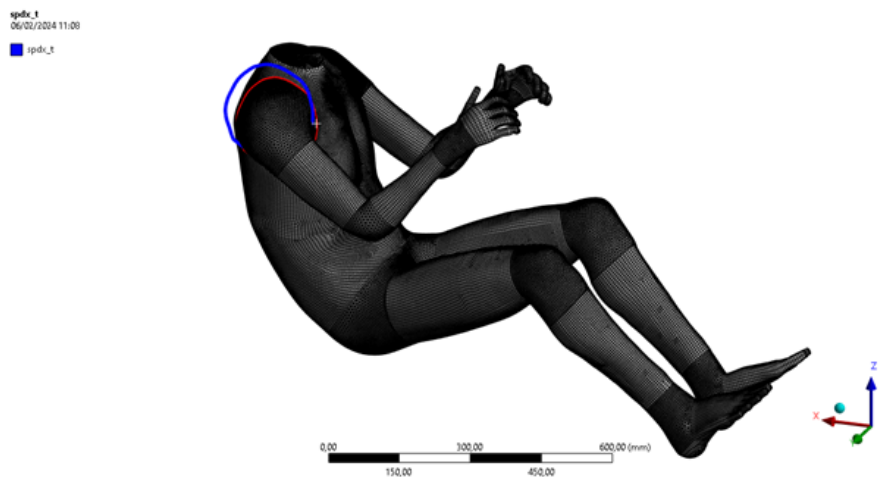


Figura 6.8: Preview del campo di spostamenti alle RBF per *spalla-busto destra*

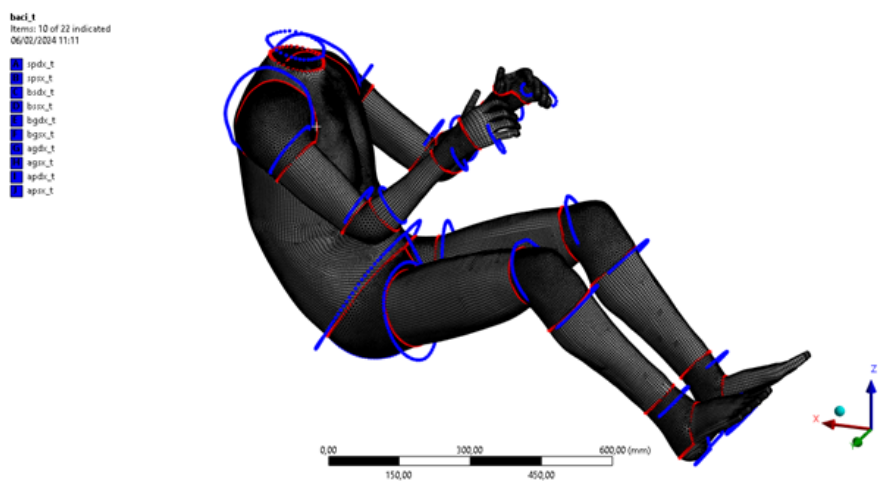


Figura 6.9: Preview del campo di spostamenti alle RBF complessivo

6.3 Esecuzione del mesh morphing

Una volta definito completamente il mesh morphing è stato possibile eseguirlo, generando, come in Figura 6.10, l'interpolazione ai target points del campo di spostamenti definito per i source points e in sintesi, THUMS AM50m95s. Procedendo come in Figura 6.11, il mesh morphing è stato infine esportato in qualità di PTS-FILE, per avere una stima, in termini di visualizzazione grafica, del confronto tra THUMS AM50s e THUMS AM50m95s.

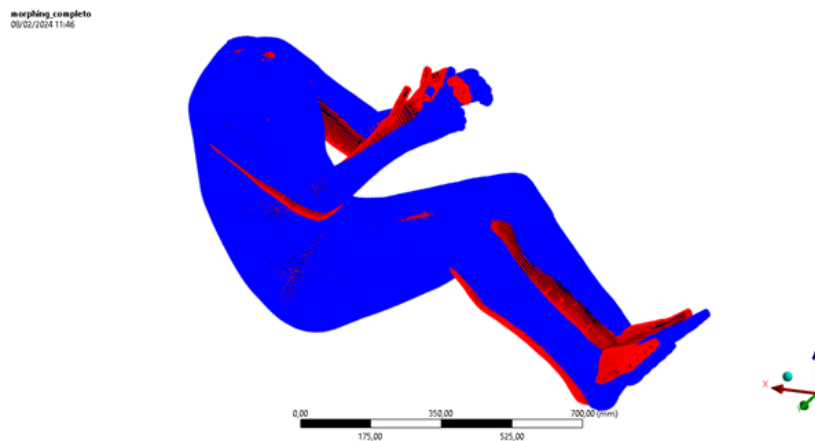


Figura 6.10: Interpolazione ai target points del campo di spostamenti definito per i source points

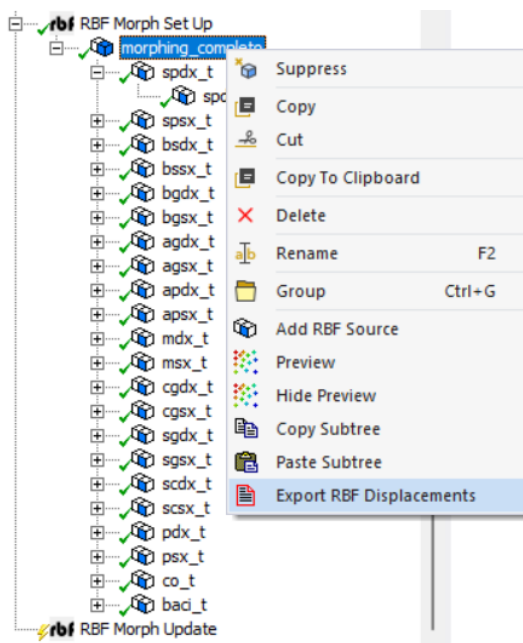


Figura 6.11: Esportazione del mesh morphing come PTS-FILE

Come si vede in Figura 6.12, la forma esterna di THUMS AM50¹ morphato al 95° percentile

¹Ai fini del confronto non ha senso specificare di avere a che fare col THUMS semplificato, essendo il mesh morphing di pelle e scheletro di THUMS AM50s identico a quello che può essere effettuato su THUMS AM50 "completo".

presenta una forte somiglianza con il THUMS AM95 originale, e lo stesso può dirsi in merito alla forma interna (Figura 6.12), rappresentata dallo scheletro, in tutti i punti eccetto forse la zona delle ossa pelviche.

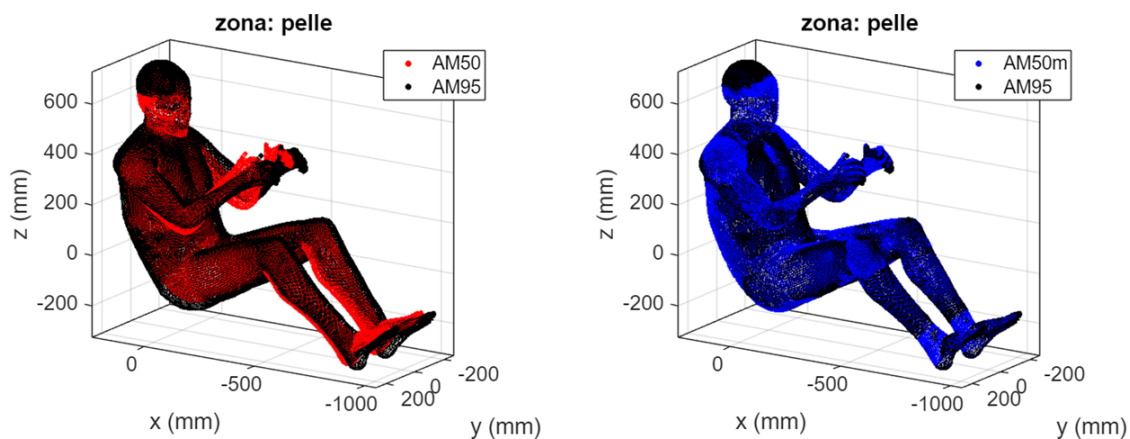


Figura 6.12: Visualizzazione grafica del confronto tra AM50 vs AM95 e AM50m vs AM95 in termini di forma interna (pelle)

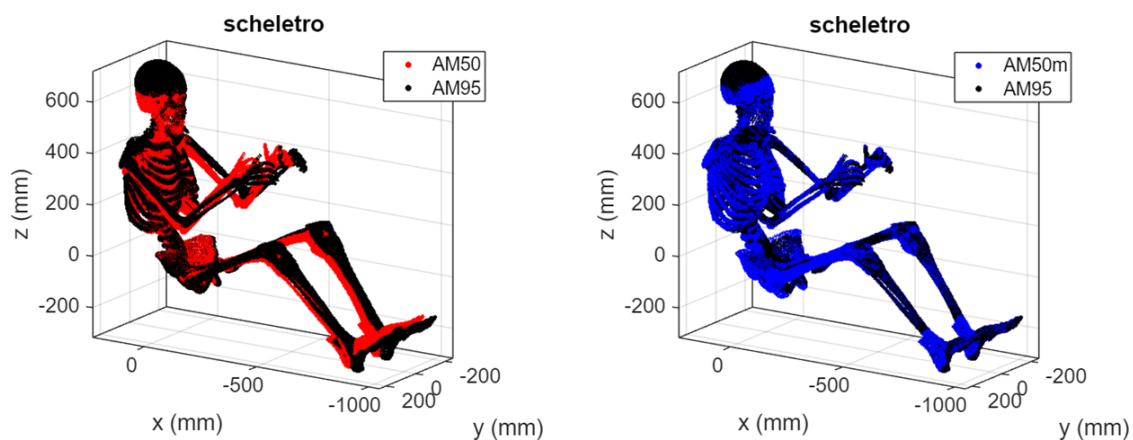


Figura 6.13: Visualizzazione grafica del confronto tra AM50 vs AM95 e AM50m vs AM95 in termini di forma interna (scheletro)

Capitolo 7

Mesh morphing completo

Realizzando il mesh morphing di THUMSs, viene in qualche modo giustificato il criterio di mesh morphing adottato. Si mostra in particolare come pur affidando a modifiche della forma prettamente esterna la trasformazione dal 50° al 95° percentile, il mesh morphing sia in grado di trascinare la forma interna con una logica non troppo lontana da quella insita nella variabilità antropometrica. D'altra parte, tuttavia, rimangono aperte alcune questioni. La prima, è che al netto del mesh morphing si vuole disporre di un modello fruibile nelle classiche applicazioni dei FE HBMs e in tal senso il THUMSs, che di THUMS ha solamente la geometria di pochissime delle sue parti, certamente non lo è. La seconda è che implementare il mesh morphing nella maniera mostrata, nonostante le agevolazioni fornite dalle funzionalità predefinite di *RBF Morph Structures* è un processo lungo e articolato, che richiede di base:

- importare i modelli in *Workbench*
- selezionare ed esportare una ad una le coordinate di ogni set di source points e omologhi, 44 in totale;
- accedere ad un ambiente di programmazione ed importare le coordinate precedentemente esportate per calcolare 3 delta di traslazione e 3 fattori di scaling per ogni set di source points, 66 in totale;
- tornare su *Workbench* e definire per ogni set di source points il sistema di riferimento locale rispetto al quale effettuare lo scaling, 22 in totale;
- definire 44 oggetti nell'albero dell' *RBF Morph Set Up* rispetto ai quali inserire manualmente i 66 valori rispettivi ai delta di traslazione ed ai fattori di scaling calcolati.

In questo capitolo si mostra la strategia adottata per superare le problematiche evidenziate. Bypassando completamente il ricorso ad *Ansys Workbench* viene strutturata in codice *Python* una procedura in grado di automatizzare il processo di mesh morphing, permettendo di:

- rintracciare i source points nel K-FILE di simulazione completo e ricavare le informazioni necessarie a calcolare il campo di spostamenti del problema alle RBF da assegnare loro;
- effettuare il mesh morphing di THUMS AM50 in base al percentile antropometrico desiderato mediante l'estensione *RBF_FGP.dll* di *RBF Morph*;
- confrontare geometricamente le mesh del THUMS ottenuto al netto del morphing e i THUMS originali.

7.1 Preparazione del K-FILE

Come primo passo verso l'automatizzazione del processo di implementazione, gestione e verifica del mesh morphing, è stato necessario predisporre i K-FILE di simulazione di AM50 ed AM95 affinché contenessero le informazioni sufficienti a poter identificare una serie di entità geometriche nel modello: i source points di AM50, gli omologhi edges di AM95 e le zone di interesse nel confronto geometrico post morphing. Sfruttando l'interfaccia di *LS Prepost* sono state quindi definite le *Entity Keywords* corrispondenti con la funzione *SET_NODE*. Si mostra a seguire un esempio in merito all'edge denominato "collo_50". Con riferimento alla Figura 7.1, aprendo il menù *Entity Creation* è possibile selezionare l'azione *Create* e definire la funzione *SET_NODE*. Si individua a questo punto l'edge corrispondente a "collo_50" come in Figura 7.2 e si intitola il set di nodi in maniera omonima, dopodiché "Apply".

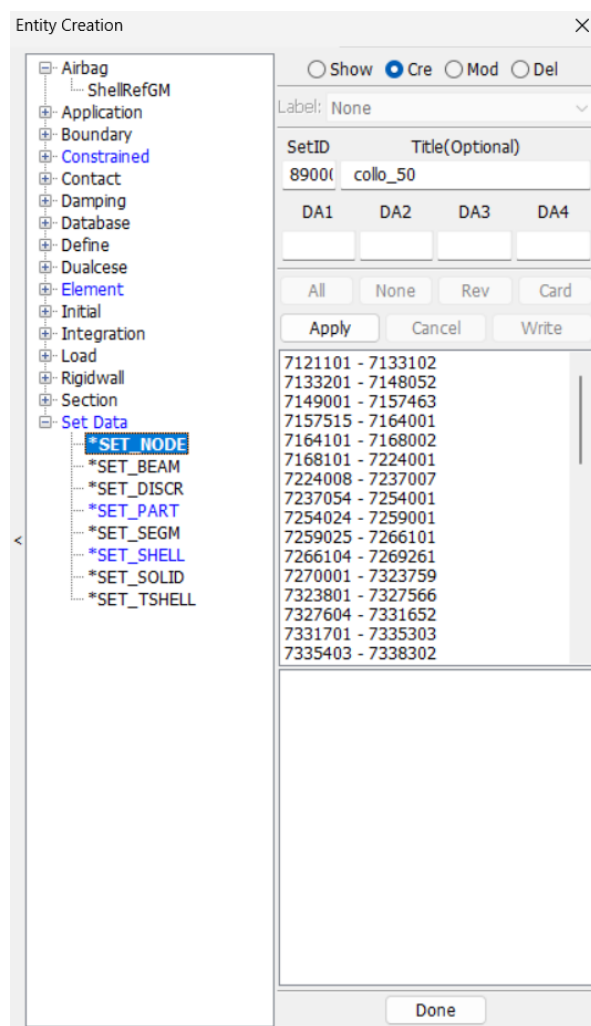


Figura 7.1: Creazione della *Entity Keyword* denominata *collo_50*

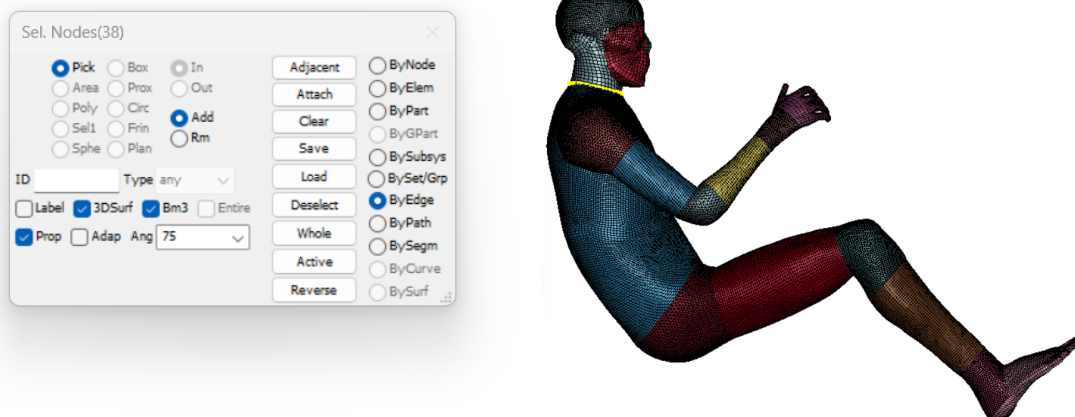


Figura 7.2: Selezione dell'edge corrispondente a *collo_50*

Aprendo il K-FILE di simulazione con un generico visualizzatore di testo sar  possibile verificare la presenza di *collo_50* tra le *Entity Keywords* di tipo *SET_NODE*, che risulter  identificato attraverso la numerazione di ciascuno dei nodi appartenenti ad esso, come in Figura 7.3.

```

.....
..... righe omesse .....
.....
*SET_NODE_LIST_TITLE
collo_50
$#   sid      da1      da2      da3      da4  solver      its      -
89000001  0.0      0.0      0.0      0.0MECH  1
$#   nid1     nid2     nid3     nid4     nid5     nid6     nid7     nid8
89500743 89500071 89500065 89500070 89000069 89000070 89000065 89000071
89000743 89000044 89000883 89000742 89000885 89000888 89000741 89000890
89000740 89000893 89000892 89000738 89000059 89000066 89000047 89000067
89500047 89500066 89500059 89500738 89500892 89500893 89500740 89500890
89500741 89500888 89500885 89500742 89500883 89500044          0          0
.....
..... righe omesse .....
.....

```

Figura 7.3: Identificazione di *collo_50* nel K-FILE di simulazione

7.2 Automatizzazione del mesh morphing

Nel processo di mesh morphing possono essere distinte due fasi fondamentali, vale a dire quella di *definizione del mesh morphing* e quella di *esecuzione del mesh morphing*. Nel paragrafo corrente viene mostrata la strategia impiegata per automatizzare ambedue le fasi, ovvero, si illustra il funzionamento degli script *Generazione_PTS.py* e *Mesh_Morphing.py* creati a tal proposito. Per una visione approfondita dell'architettura relativa ai due codici *Python* si rimanda in Appendice, ai Paragrafi 9.4.1 e 9.4.2, dove sono riportati per esteso.

7.2.1 Definizione del mesh morphing

La definizione del mesh morphing è gestita interamente dallo script *Generazione_PTS.py* che produce in output il PTS-FILE contenente il campo di spostamenti del problema alle RBF da assegnare ai source points. Lavorando su un set di source points alla volta, *Generazione_PTS.py* esegue 3 step operativi:

1. Ricorrendo alla funzione *numerazione_cercata(nome_file, keyword)* in Figura 7.4, legge il K-FILE di simulazione di THUMS AM50 e salva in un vettore la numerazione necessaria ad identificare i source points del set. Allo stesso modo identifica la numerazione dei nodi appartenenti all'edge omologo di THUMS AM95.

```
# nome_file=inserire il percorso file al K-FILE
# keyword=inserire il nome del node_set desiderato
def numerazione_cercata(nome_file,keyword):
    elenco=[]
    keyword1='*SET_PART_LIST'
    keyword2='*SET_NODE_LIST_TITLE'
    sezione_corretta=False
    with open(nome_file,'r') as file:
        for riga in file:
            if riga.startswith(keyword):
                sezione_corretta=True
            if sezione_corretta:
                try:
                    numeri =list(map(float,riga.split()))
                    elenco.append(numeri)
                except ValueError:
                    a=riga.startswith(keyword1)
                    b=riga.startswith(keyword2)
                    if a or b:
                        break
    matrice_numerazione=np.array(elenco)
    numerazione=matrice_numerazione.flatten()
    return numerazione
```

Figura 7.4: Identificazione della numerazione di un set di nodi nel K-FILE

2. Ricorrendo alla funzione *individua_path(nome_file, vettore)* in Figura 7.5, legge nuovamente il K-FILE di simulazione di THUMS AM50 e cerca nella sezione *NODE* (Figura 7.6) le coordinate dei source points del set, di cui è nota la numerazione grazie allo step precedente. Stessa operazione viene effettuata rispetto a THUMS AM95.
3. Viene calcolato il campo di spostamenti da assegnare ai source points con le modalità già descritte al Paragrafo 5.1.

Eseguiti i 3 step illustrati, per finire, *Generazione_PTS.py* assembla e scrive il PTS-FILE (Figura 7.7) corrispondente al campo di spostamenti del problema alle RBF nel mesh morphing di THUMS AM50 dal 50° al 95° percentile. Il tempo di esecuzione complessivo risulta pari a 274.99 s, 4 minuti e mezzo circa.


```

# nome_file=inserire il percorso file al K-FILE
# kword=inserire il vettore con la numerazione dei nodi del node Set desiderato
def individua_path(nome_file,vettore):
    elenco=[]
    kword1='*NODE'
    kword2='*PART_AVERAGED'
    sezione_corretta=False
    with open(nome_file,'r') as file:
        for riga in file:
            if riga.startswith(kword1):
                sezione_corretta=True
            if sezione_corretta:
                try:
                    numeri =list(map(float,riga.split()))
                    elenco.append(numeri)
                except ValueError:
                    a=riga.startswith(kword2)
                    if a:
                        break
    coordinate=np.array(elenco)
    path=coordinate[np.isin(coordinate[:,0],vettore)]
    return path

```

Figura 7.5: Identificazione delle coordinate di un set di nodi nel K-FILE

```

*NODE
$#  nid          x          y          z          tc          rc
7121101      132.9127      17.49752      150.8569          0          0
7121109      -163.0202      96.61806      -48.35065          0          0
7121152       118.2001       21.74364       156.744          0          0
.....
..... righe omesse .....
.....
89607442       45.44971       38.44339       369.5782          0          0
89607443       32.61136       44.40474       373.0219          0          0
89607444       108.5965       132.0475       94.84207          0          0

```

Figura 7.6: Sezione *NODE* del K-FILE

```

1522
-811.018 67.19753 -288.0066 -38.177466141229516 3.1974164850510567 -6.893923740617389 81000774 0 s p
-807.6782 68.16429 -292.2721 -37.94366307172289 3.2650942954424296 -7.192525414021546 81000775 0 s p
-804.5713 69.83803 -296.3405 -37.72616419255343 3.3822640773577533 -7.4773293169819155 81000776 0 s p
.....
..... righe omesse .....
.....
109.8944 100.7405 423.7275 13.755605995039172 8.026604098779977 32.901360113398766 89589508 0 s p
150.0275 106.0441 408.1986 18.249523776210857 8.581565927666018 32.121853233288185 89589563 0 s p
27.22313 111.8411 378.0515 4.498461885936079 9.188156492952311 30.608554038536354 89589795 0 s p

```

Figura 7.7: PTS-FILE

7.2.2 Esecuzione del mesh morphing

L'esecuzione del mesh morphing è gestita interamente dallo script *Mesh_Morphing.py* che produce in output il K-FILE di simulazione di THUMS AM50 morphato al percentile desiderato. Avviato il *run* di *Mesh_Morphing.py*, vengono eseguite le seguenti operazioni in sequenza:

1. Viene richiamata l'estensione *RBF_FGP.dll* di *RBF Morph* e verificata la validità della licenza d'utilizzo
2. Viene letto il FILE-PTS prodotto da *Generazione_PTS.py* salvando coordinate e spostamenti in una struttura di tipo *ctype*¹;
3. Gli spostamenti del problema alle RBF vengono modulati come indicato al Paragrafo 5.2 in base ad un valore in input del percentile antropometrico desiderato nel mesh morphing. Si fa notare che viene preferito effettuare la suddetta modulazione in questo stadio della procedura complessiva per ottimizzare i tempi di esecuzione;
4. Viene letto il K-FILE di simulazione di THUMS AM50 estrapolando le coordinate dei target points, salvandole ovvero, in una struttura di tipo *ctype*;
5. Viene eseguito il mesh morphing ad opera di *RBF Morph* e scritto un nuovo K-FILE di simulazione contenente le coordinate nodali di THUMS morphato anziché l'originale. Si riportano a titolo d'esempio le prime tre righe della sezione *NODE* del K-FILE di THUMS AM50 (Figura 7.8) e le omologhe tratte dal K-FILE di THUMS AM50m95 (Figura 7.9).

```
*NODE
7121101      132.9127      17.49752      150.8569      0      0
7121109     -163.0202      96.61806     -48.35065      0      0
7121152      118.2001      21.74364      156.744      0      0
.....
..... righe omesse.....
.....
```

Figura 7.8: Prime righe della sezione *NODE* del K-FILE di THUMS AM50

```
*NODE
7121101     150.37798102     19.41371560     166.42936123      0      0
7121109    -171.12580417     119.26328861    -44.33608968      0      0
7121152     133.95643805      24.15646037     173.18246829      0      0
.....
..... righe omesse.....
.....
```

Figura 7.9: Prime righe della sezione *NODE* del K-FILE di THUMS AM50m95

È interessante far notare che il tempo di esecuzione registrato per *Mesh_Morphing.py* è di 10.44 *sec*. Tempi così brevi sono possibili in primo luogo grazie all'efficienza di *RBF Morph* nella risoluzione del problema alle RBF e in secondo luogo per via della scelta dei source points operata. La taglia del problema alle RBF è infatti proporzionale al numero di source points impiegati. In questo senso aver identificato i source points negli edges del modello, ovvero un insieme di geometrie costituito da un piccolo numero di nodi ma molto rappresentativo dell'HBM oltre che ben distribuito su di esso, si è rivelata essere una scelta molto efficiente.

¹L'estensione *RBF_FGP.dll* utilizza come sintassi il linguaggio *C* e in tal senso viene impiegato il modulo *ctype* di *Python* per rendere compatibili i dati trasmessi

7.3 Automatizzazione del confronto tra mesh

Come ultimo passo della procedura per automatizzare il processo di mesh morphing viene realizzato lo script *Confronto.py*, avente la funzione di effettuare il confronto tra la mesh dei THUMS morphati e i THUMS originali in merito a quelle zone selezionate come di interesse. Per ognuna di queste zone *Confronto.py* effettua le seguenti operazioni:

1. ricorrendo alle funzioni già mostrate per *Generazione_PTS.py* estrae le coordinate rispettive nei K-FILE di THUMS AM50, THUMS AM95 e THUMS AM50m (da intendere come la configurazione morphata al percentile generico di THUMS AM50).
2. ricorrendo alla funzione *distanza(x1,y1,z1,x2,y2,z2)* in Figura 7.10 calcola i valori di *MDM* e *MDA* nel confronto tra AM50 e AM95, AM50m e AM95 e infine AM50 e AM50m

```
def distanza(x1,y1,z1,x2,y2,z2):
    min_rad=[]
    for i in range(len(x1)):
        r=[]
        for j in range(len(x1)):
            dx=x2[j]-x1[i]
            if abs(dx)>100:
                continue
            dy=y2[j]-y1[i]
            dz=z2[j]-z1[i]
            v=np.array([dx,dy,dz])
            r.append(np.linalg.norm(v))
        vec=np.array(r)
        min_rad.append(min(vec))
    a=np.mean(np.array(min_rad))
    b=np.min(np.array(min_rad))
    c=np.max(np.array(min_rad))
    return a,b,c
```

Figura 7.10: Funzione che implementa il calcolo di *MDA* e *MDM*

7.4 THUMS parametrico

Grazie agli strumenti presentati nei paragrafi precedenti è stata esplorata la parametrizzazione di THUMS in base al percentile antropometrico in 4 declinazioni che saranno oggetto di discussione nei risultati: THUMS AM50m95 (Figura), AM50m75 (Figura), THUMS AM50m35 (Figura) e THUMS AM50m95b. Per quanto riguarda le prime tre configurazioni (Figura 7.11), esse, come suggerito dalla nomenclatura, rappresentano il THUMS AM50 morphato al 95°, al 75° e al 35° percentile. THUMS AM50m95b, in cui la "b" indica "bones", rappresenta invece una versione di THUMS morphato al 95° percentile nella quale anche parte delle ossa è impiegata come source points. Sebbene una volta realizzato THUMS AM50m95 si fosse subito riscontrata una buona aderenza con il THUMS AM95 originale, era anche evidente come alcune zone dello scheletro, in particolare *cassa toracica* e *ossa pelviche*, avevano un certo margine di miglioramento in quanto a copiare più fedelmente la geometria target rappresentata da THUMS AM95. Si è deciso quindi di effettuare un secondo mesh morphing a partire da THUMS AM50m95 in cui tuttavia i source points del problema fossero *pelle*, *cassa toracica* e *ossa pelviche*: alla *pelle* è stato assegnato un campo di spostamenti nullo, affinché la forma esterna di THUMS AM50m95b risultasse pari alla forma esterna di THUMS AM50m95; a *cassa toracica* e *ossa pelviche* è stato assegnato il campo di spostamento calcolato secondo le modalità ormai note. Si mostrano in Figura 7.12 gli scheletri di THUMS AM50m95 e AM50m95b in relazione a quello di AM95.

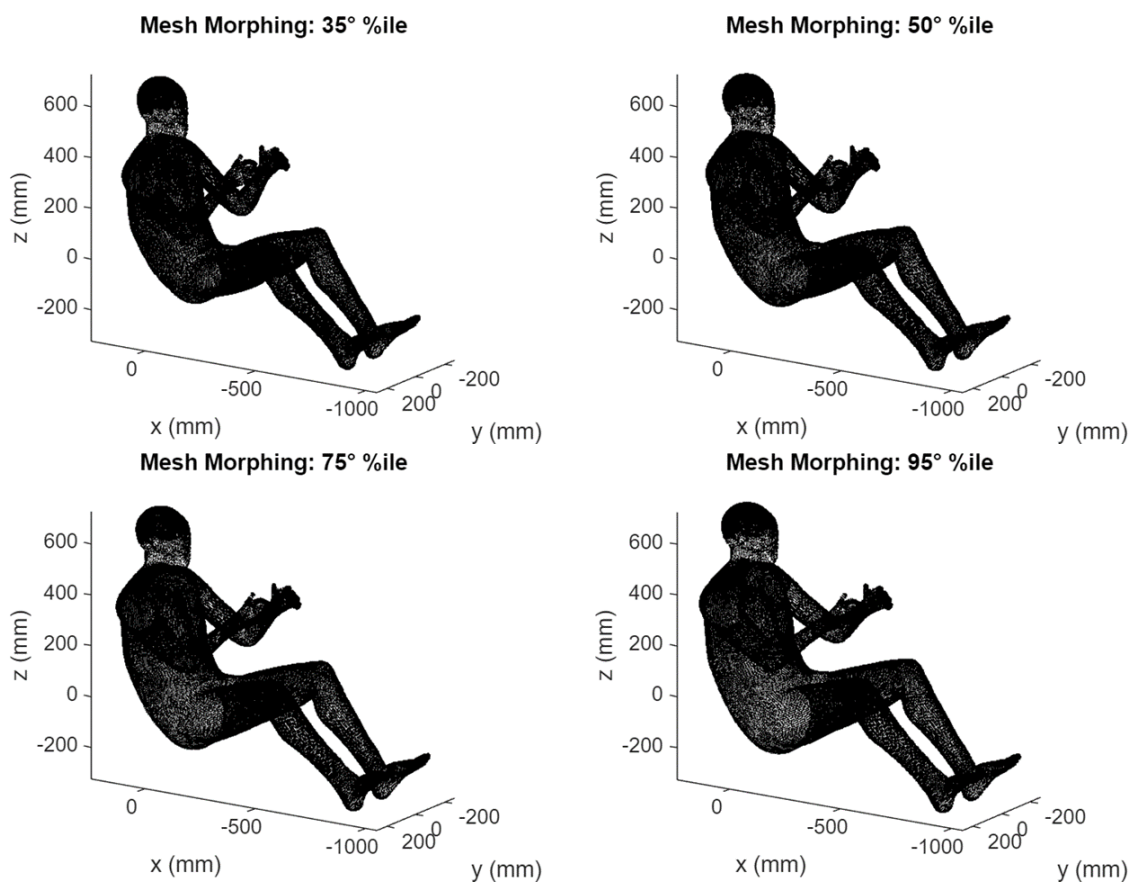


Figura 7.11: Parametrizzazioni di THUMS: 35°, 50°, 75° e 95° percentile

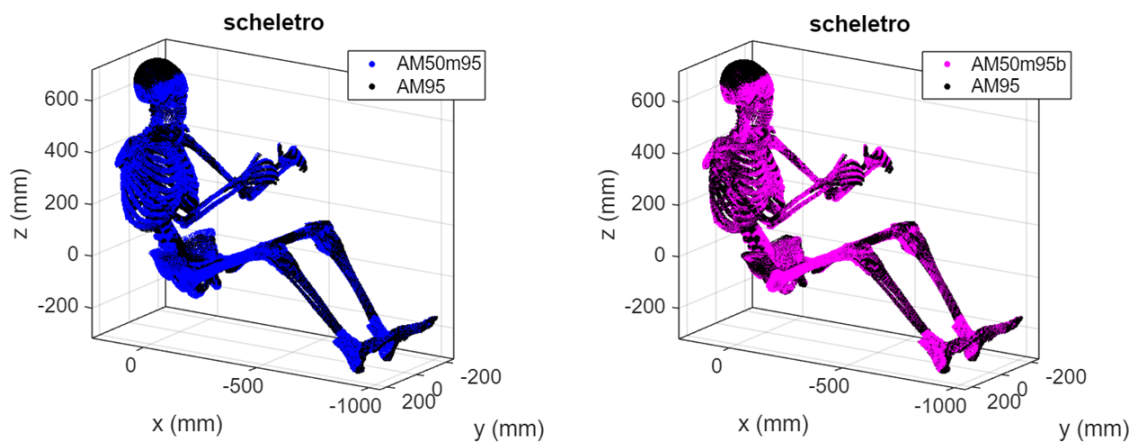


Figura 7.12: Confronto scheletri di AM50m95 e AM50m95b

Capitolo 8

Simulazione

Il focus della trattazione fino ad ora è stato determinare ed implementare nella maniera più efficiente possibile il mesh morphing di THUMS AM50 in funzione del percentile antropometrico statistico. Per valutare l'efficacia della modellazione operata in tal senso sono stati quindi introdotti una serie di strumenti di verifica geometrica. La geometria di THUMS tuttavia, è un aspetto che, da solo, racconta solo una piccola parte della storia sugli effetti introdotti dal mesh morphing, lasciando un interrogativo fondamentale: il mesh morphing di THUMS ha effetto sul suo comportamento? e se sì, ha il "giusto" effetto? Per rispondere alla prima domanda, le varie parametrizzazioni di THUMS ottenute sono state impiegate in una simulazione di impatto frontale, ovvero il *frontal sled test*. Il comportamento ad urto è stato quindi analizzato dal punto di vista cinematico, valutando e confrontando le traiettorie di punti omologhi nelle varie configurazioni antropometriche. Per rispondere alla seconda domanda, sempre nell'ambito della simulazione di impatto frontale, è stato invece approfondito il confronto tra la cinematica di THUMS AM95 e le due versioni di THUMS AM50 morphato al 95° percentile, ovvero THUMS AM50m95 e AM50m95b, assumendo la logica semplice ma chiara che: tanto più il comportamento dei THUMS ottenuti come mesh morphing al 95° percentile è simile al comportamento del THUMS "originale" al 95° percentile, tanto più è corretto l'effetto introdotto dal mesh morphing e in senso più ampio le modalità con le quali è stato parametrizzato THUMS.

8.1 Frontal Sled Test

Il *frontal sled test*, o tradotto, test della slitta frontale, è uno dei test impiegati nell'ambito della sicurezza veicolare per replicare gli effetti su un occupante, rappresentato fisicamente da ATDs o virtualmente da FE HBMs, di un impatto frontale. Nel *frontal sled test*, l'occupante viene posto su un sedile rigido ed opportunamente vincolato ad esso. Il sedile, libero di "slittare" su appositi binari viene repentinamente accelerato nella direzione opposta a quella identificata come "frontale" per l'occupante, che subirà gli effetti della brusca variazione rispetto al suo stato inerziale alla stregua di un impatto automobilistico. Il set up di simulazione del *frontal sled test* impiegato in questo lavoro, a cui si farà riferimento d'ora in avanti, è quello impiegato in *Vezin et al.* [28] e disponibile al download di THUMS nel corrispettivo "Validation Set", ovvero il set di simulazioni con le quali il FE HBMs viene validato rispetto a test condotti su PMHSs. Con riferimento alla Figura 8.1, per effettuare il *Frontal Sled Test*, THUMS viene posto su una struttura di supporto mobile formata da corpi rigidi tra loro solidali corrispondenti a seduta, schienale e pavimento del veicolo. Per vincolarlo alla struttura suddetta viene infine impiegata una cintura di sicurezza "a tre punti" modellata mediante elementi deformabili. Avviata la simulazione, la struttura di supporto mobile viene accelerata secondo la curva di accelerazione¹ mostrata in Figura 8.2, fino alla velocità di 50 *km/h*, realizzando così gli effetti della collisione (Figura 8.3).

¹Si noti che l'accelerazione in questione si esplica in un lasso di tempo pari neanche a 20 millesimi di secondo

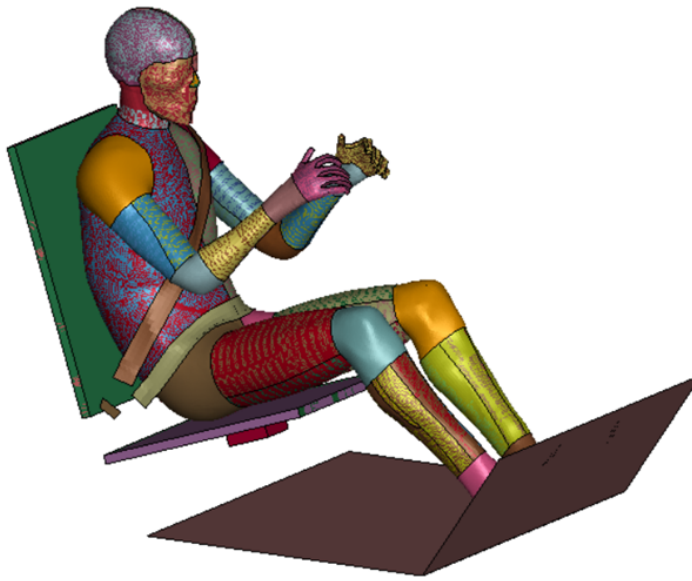


Figura 8.1: Posizionamento di THUMS all'istante iniziale di simulazione nel *frontal sled test*

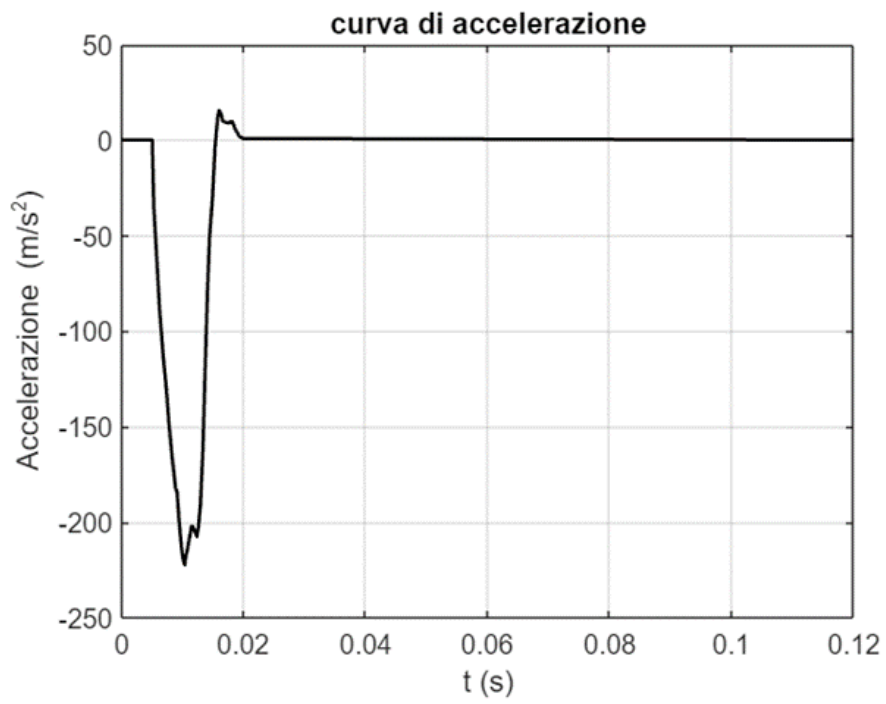


Figura 8.2: Curva d'accelerazione

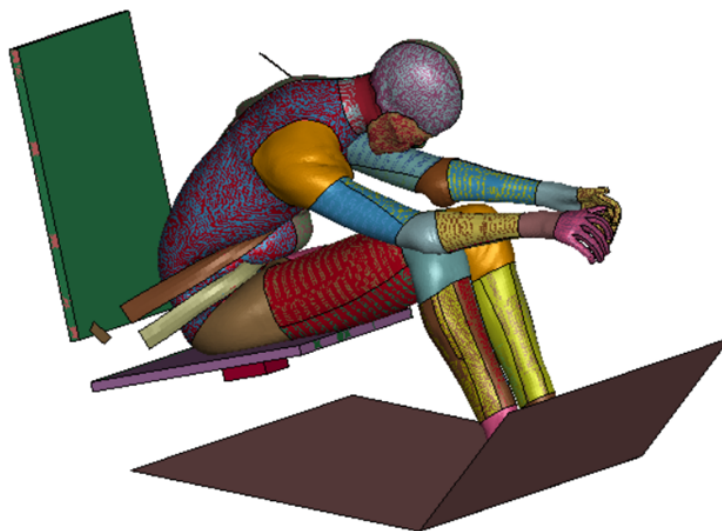


Figura 8.3: Effetti della collisione visualizzati al termine della simulazione

Per dare un'idea delle dimensioni della simulazione, si fa presente che un computer con 128 MB di RAM impiega circa 2 giorni per l'esecuzione completa.

8.1.1 Configurazioni di simulazione

Con il termine configurazioni di simulazione si fa riferimento alle simulazioni eseguite su l'uno o l'altro THUMS preso in esame. La *configurazione50* intenderà ad esempio la simulazione effettuata su THUMS AM50, la *configurazione95* quella effettuata su THUMS AM95, la *configurazione50m95* quella effettuata su THUMS AM50 morphato al 95° percentile e così via. Mentre per quanto riguarda *configurazione50* e *configurazione95* il K-FILE di simulazione del *frontal sled test* è scaricabile già "preconfezionato", per ottenere le configurazioni di simulazione di tipo *configurazione50mP*, dove "P" rappresenta il generico percentile, si è agito applicando il mesh morphing come automatizzato al Paragrafo 7.2, direttamente sul K-FILE di simulazione del *frontal sled test* dell'AM50. Procedendo in questo modo, il mesh morphing ha effetto anche sui vincoli di simulazione, ovvero struttura di supporto mobile e cintura di sicurezza, che diventando target points del problema alle RBF, modificano la loro forma compatibilmente alle modifiche di forma introdotte sull'occupante. Ciò è visibile in Figura 8.4 dove è presentata la *configurazione50m95* nelle condizioni iniziali di simulazione. Si noti come sedile, schienale e pavimento del veicolo presentino un certo grado di deformazione rispetto alle geometrie originali. Questi difetti, tuttavia, considerato che 1) sono di lieve entità e 2) riguardano corpi rigidi la cui interazione con l'occupante è relativamente marginale durante l'impatto, sono stati ritenuti secondari, dando più importanza ad aspetti quali la buona aderenza della cintura di sicurezza all'occupante.



Figura 8.4: *Configurazione50m95* all'istante iniziale di simulazione

8.1.2 Estrazione dei risultati

Per valutare la cinematica dell'occupante nel *frontal sled test* è stato necessario stabilire a quali entità geometriche assegnare il compito di "punti di controllo", ovvero punti di cui tracciare l'evoluzione spazio-temporale durante la storia di crash. Sono stati scelti in tal senso i 22 baricentri dei 22 set di source points (o omologhi). Le ragioni sono essenzialmente 2:

1. per come sono stati scelti i source points (e omologhi) essi risultano ben distribuiti su tutto il THUMS;
2. nel confronto tra THUMS AM95 e THUMS AM50m95, i baricentri dei source points sono gli unici punti che all'istante iniziale della simulazione risultano coincidenti nell'una e l'altra configurazione.

Si fa presente tuttavia che i baricentri dei source points, o meglio, le coordinate rispettive, sono grandezze derivate del problema e pertanto non disponibili quali entità geometriche da selezionare direttamente in *LS Prepost* nella definizione di un *History Plot*. Per estrarre i risultati è stata dunque ripetuta la seguente procedura, coadiuvata dagli strumenti di post processing disponibili in *LS Prepost*:

- abilitando la specifica *nodal* sono stati selezionati tutti i nodi di ciascun set di source points o omologhi quali entità di cui tracciare l'evoluzione temporale in coordinate (esempio in Figura 8.5);
- comandando il *Plot* della selezione sono state ottenute le traiettorie di ciascun nodo del set nella coordinata selezionata (Figura 8.6)
- Applicando l'operazione *average_curves* è stata infine ottenuta la traiettoria del baricentro dei source points come curva media.

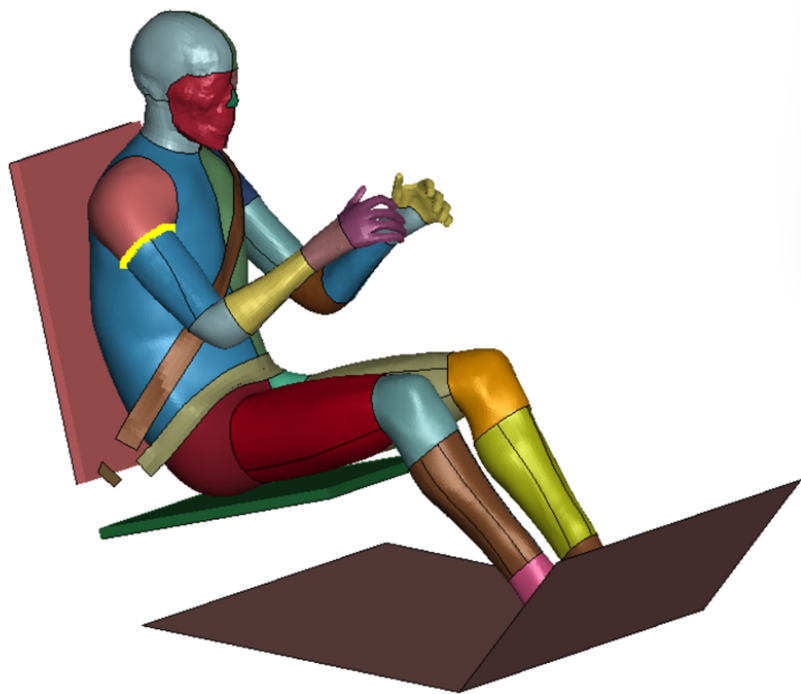
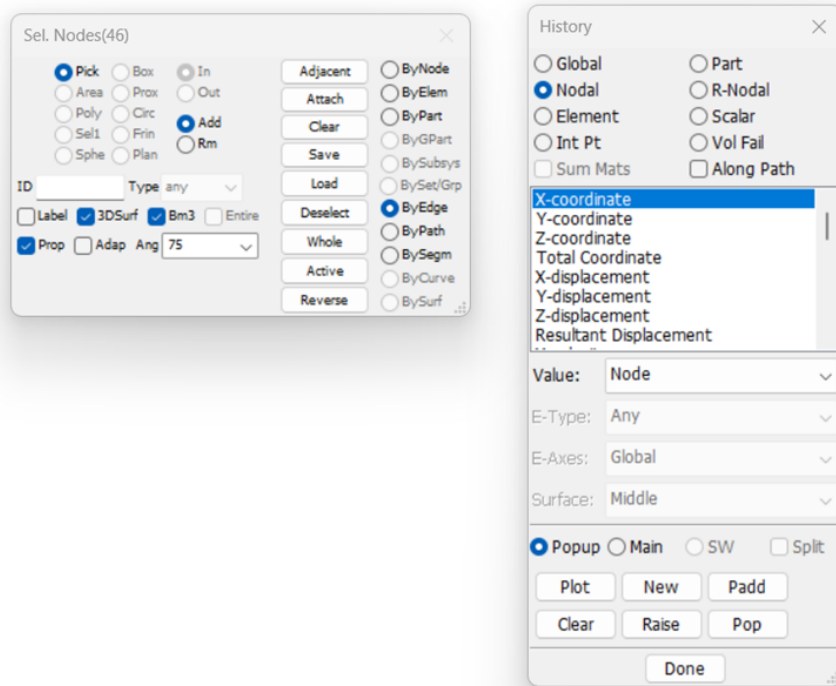


Figura 8.5: Selezione dei risultati sui nodi del set di source points

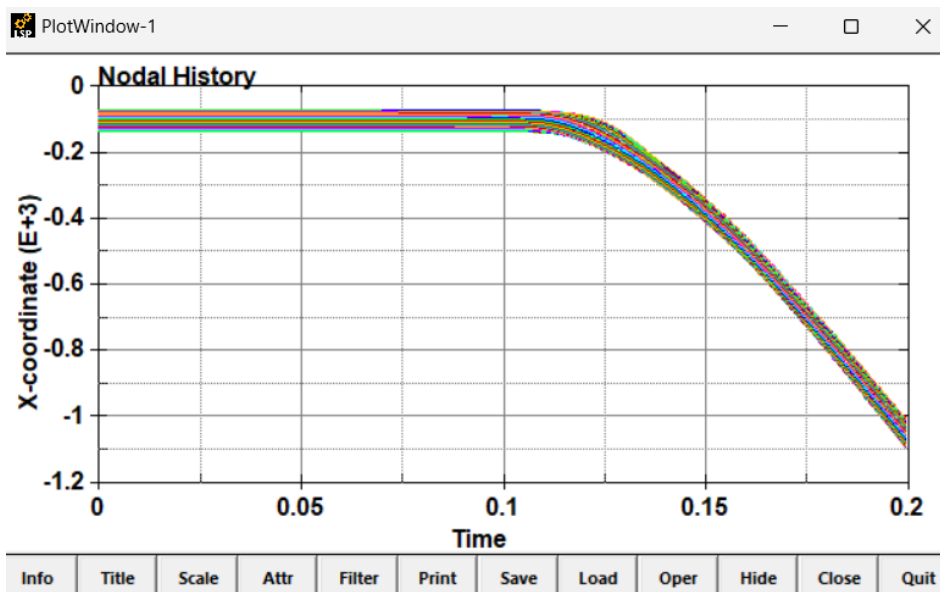


Figura 8.6: Traiettorie dei nodi del set nella coordinata selezionata

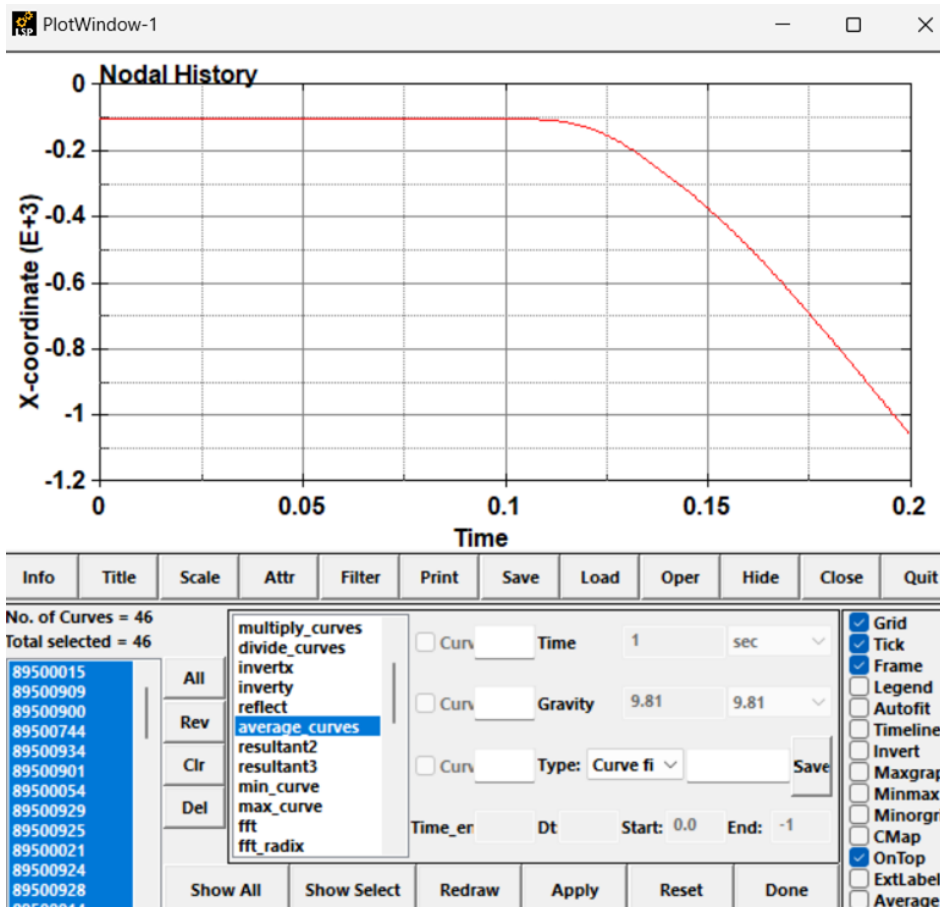


Figura 8.7: Traiettorie della coordinata baricentrica

Capitolo 9

Risultati

Nei Capitoli 4, 6, 7 è stato mostrato come definire, implementare ed automatizzare il mesh morphing di THUMS AM50 negli altri percentili antropometrici statistici. Nel Capitolo 8 è stato invece presentato il *frontal sled test* ovvero la simulazione di impatto frontale eseguita in merito alle varie parametrizzazioni di THUMS realizzate, nel dettaglio AM50m35, AM50m75, AM50m95 e AM50m95b. Nel presente capitolo, dedicato alla discussione dei risultati, viene valutata l'efficacia della strategia impiegata per parametrizzare THUMS.

La prima parte dei risultati riguarderà il confronto tra le geometrie di THUMS AM50m95 e THUMS AM95: poiché in questo lavoro la legge di variabilità antropometrica esistente tra il generico percentile statistico ed il 50° è stata identificata nell'opportuna modulazione delle differenze di forma esistenti tra THUMS AM50 ed AM95, è importante verificare che al limite del dominio di parametrizzazione, ovvero al 95° percentile, la forma di THUMS morphato in tal senso sia quanto più simile alla forma di THUMS AM95 originale.

La seconda parte dei risultati prenderà invece in esame la cinematica a crash di THUMS nel test di impatto frontale. Si confronteranno diverse configurazioni di THUMS per determinare se e quanto varia il comportamento cinematico a crash in base alla parametrizzazione introdotta e più in dettaglio se tali variazioni riflettono correttamente le modifiche di forma di THUMS. Per far ciò si approfondirà ancora una volta il confronto tra THUMS AM95 e il corrispettivo parametrico THUMS AM50m95.

9.1 Confronto geometrico

Per effettuare il confronto geometrico tra THUMS AM50m95 e THUMS AM95 originale sono stati impiegati gli strumenti ormai noti, ovvero la *MDA* e la *MDM*, prendendo in esame le seguenti zone del corpo: bacino, busto, viso, spalla, braccio, gomito, avambraccio, polso, mano, coscia, ginocchio, stinco, caviglia, piede per quanto riguarda la pelle; cassa toracica e ossa pelviche per quanto riguarda invece lo scheletro. Focalizzando l'attenzione sulla pelle, che identifica la forma esterna di THUMS, si riportano in Tabella 9.1 i valori di *MDA*, *MDM* e *MDA/MDM* nel confronto tra AM50m95 e AM95. In merito alla *MDA* il miglior risultato è rappresentato dalla zona della caviglia con 0.24 mm , mentre il peggiore da quella del bacino con 7.90 mm (Figura 9.1). In merito alla *MDM*, allo stesso modo, caviglia e bacino risultano la migliore (0.57 mm) e la peggiore casistica (38.1 mm). Sebbene una *MDM* di quasi 4 cm , registrata nella zona del bacino, sia certamente rilevante, si fa notare tuttavia che essa esprime una caratteristica prettamente locale del confronto tra le geometrie di AM50m95 e AM95, che risulta poco diffusa sulla stessa zona del bacino (*MDA/MDM* pari al 20.7%) e tanto meno sulla totalità della pelle. Mediando i valori della *MDA* tra le varie zone della pelle si ottiene una stima dello scostamento medio tra la forma esterna di THUMS AM50m95 e quella di THUMS AM95, che risulta pari a circa 3.5 mm . Questo conferma certamente la forte somiglianza tra i due THUMS in termini di aspetto e ingombri. A supporto di ciò si fa oltretutto notare che morphando THUMS dal 50° al 95° percentile,

il peso del FE HBM passa da 75 *kg* a 100 *kg*, scostandosi del solo 3% dal peso di THUMS AM95 pari a 103 *kg*.

AM50m95 vs AM95: forma esterna			
	<i>MDA</i>	<i>MDM</i>	<i>MDA/MDM</i>
	[<i>mm</i>]	[<i>mm</i>]	%
bacino	7.8996	38.0980	20.7350
busto	7.1033	24.3698	29.1481
viso	4.0516	11.4597	35.3553
spalla	3.4297	9.0678	37.8228
braccio	2.5492	7.1905	35.4528
gomito	2.5139	5.6771	44.2819
avanbraccio	1.9085	3.7900	50.3577
polso	1.5008	2.3003	65.2437
mano	3.6584	10.5185	34.7802
coscia	5.6064	17.3734	32.2700
ginocchio	2.7345	8.3414	32.7823
stinco	1.6803	3.1413	53.4916
caviglia	0.2359	0.5736	41.1221
piede	1.7281	7.0430	24.5362
media	3.3286	10.6389	38.3843

Tabella 9.1: Valori di *MDA*, *MDM* e *MDM/MDA* nel confronto tra le zone della pelle di AM50m95 e AM95

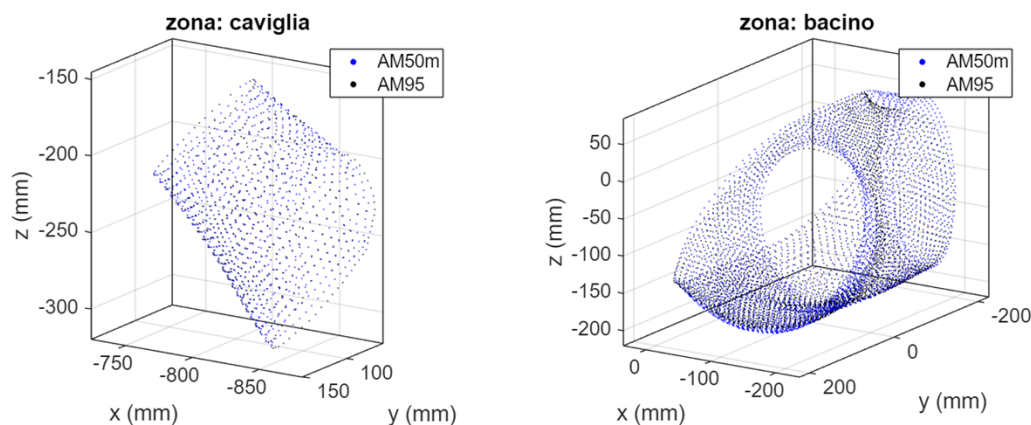


Figura 9.1: Zone della pelle migliore e peggiore in merito alla *MDA*. AM50m95 è abbreviato ad AM50m in figura.

Focalizzando invece l'attenzione sullo scheletro, e in particolare sulle geometrie, ritenute più critiche, di cassa toracica e ossa pelviche, si riportano in Tabella 9.2 i valori di *MDA*, *MDM* e *MDA/MDM* nel confronto tra AM50m95 ed AM95. Anche in questo caso, i valori della *MDA* risultano essere nell'ordine della frazione di centimetro, con 3.7 *mm* per la cassa toracica e 7.83 *mm* per le ossa pelviche, tuttavia i valori piuttosto elevati della *MDM* (quasi 3 *cm* per le ossa pelviche) hanno sollevato una riflessione su quanto fosse corretto ritenere la *MDA* una misura efficace dello scostamento medio per il caso in esame. Le mesh di cassa toracica e ossa pelviche, diversamente dalle mesh delle varie zone della pelle, sono mesh di elementi *solid* e non *shell*. In questo senso nel calcolo della *MDA* sono presi in considerazione

anche nodi appartenenti al volume interno della geometria e non solo quelli appartenenti alla superficie esterna che ne definisce la forma. Nelle regioni in cui, sovrapposte le geometrie di AM50m95 e AM95, i volumi rispettivi si intersecano, è lecito allora credere che risulteranno molto piccole le minime distanze calcolabili, influenzando la media di queste minime distanze, la MDA , in maniera scorretta. Sulla MDM , sebbene abbia valore locale, si può invece avere la certezza che essa rappresenti correttamente uno scostamento geometrico come lo si vuole intendere.

AM50m95 vs AM95: forma interna			
	MDA	MDM	MDA/MDM
	[mm]	[mm]	%
cassa toracica	3.7665	10.8088	0.3485
ossa pelviche	7.8291	29.6483	0.2641
media	5.7978	20.2286	0.3063

Tabella 9.2: Valori di MDA , MDM e MDM/MDA nel confronto tra le zone critiche dello scheletro di AM50m95 ed AM95

In virtù di queste considerazioni è stato realizzato THUMS AM50m95b come indicato a Paragrafo 7.4. A parità di forma esterna, l'AM50m95b presenta valori della MDM decisamente migliori rispetto all'AM50m95 nel confronto con l'AM95. Come si legge in Tabella 9.3 la MDM risulta infatti pari a 6.3 mm per la cassa toracica e 7.5 mm per le ossa pelviche, ovvero, rispettivamente, più piccola del 41% e del 75% in confronto all'AM50m95.

AM50m95b vs AM95: forma interna			
	MDA	MDM	MDA/MDM
	[mm]	[mm]	%
cassa toracica	1.9739	6.3110	0.3128
ossa pelviche	2.4811	7.5290	0.3295
media	2.2275	6.9200	0.3212

Tabella 9.3: Valori di MDA , MDM e MDM/MDA nel confronto tra le zone critiche dello scheletro di AM50m95b ed AM95

Per completare il discorso sul confronto geometrico, si riporta in Figura 9.2, il grafico dei valori della MDA nel confronto tra zone della pelle di THUMS AM50 ed AM95, THUMS AM50m95 ed AM95 e THUMS AM50 ed AM50m95; in Figura 9.3 i valori di MDA e MDM nel confronto tra le zone critiche dello scheletro di THUMS AM50 ed AM95, THUMS AM50m95b ed AM95 e THUMS AM50 ed AM50m95b. È possibile notare come nel mesh morphing dal 50° al 95° percentile, tutte le zone finora evidenziate mostrino drastiche riduzioni in quanto a MDA e MDM e forse ancor più importante come la forma di THUMS AM50m95¹ sia decisamente più vicina a THUMS AM95 rispetto a quanto lo sia a se stessa prima del morphing, ovvero a THUMS AM50. Facendo riferimento in particolare ai valori medi calcolati in Tabella 9.4, si noti 1) come nel mesh morphing dal 50° al 95° percentile lo scostamento da AM95 si riduca da 17.56 mm a 3.65 mm , ovvero il 79% 2) come lo scostamento di AM50 da AM50m95 differisca del solo 1.6% da quello con AM95. Si riporta infine per un confronto grafico la Figura 9.4. Per ulteriori approfondimenti di carattere grafico sul confronto geometrico si rimanda in Appendice, ai Paragrafi 9.3.2, 9.3.3 e 9.3.4.

¹Da intendere AM50m95b quando si considera la forma interna

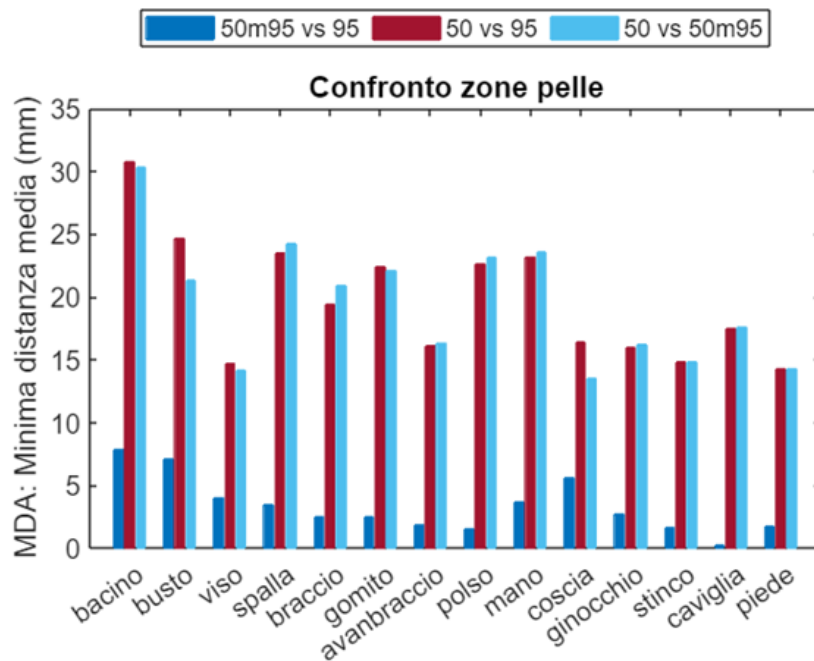


Figura 9.2: Valori di *MDA* per le zone della pelle

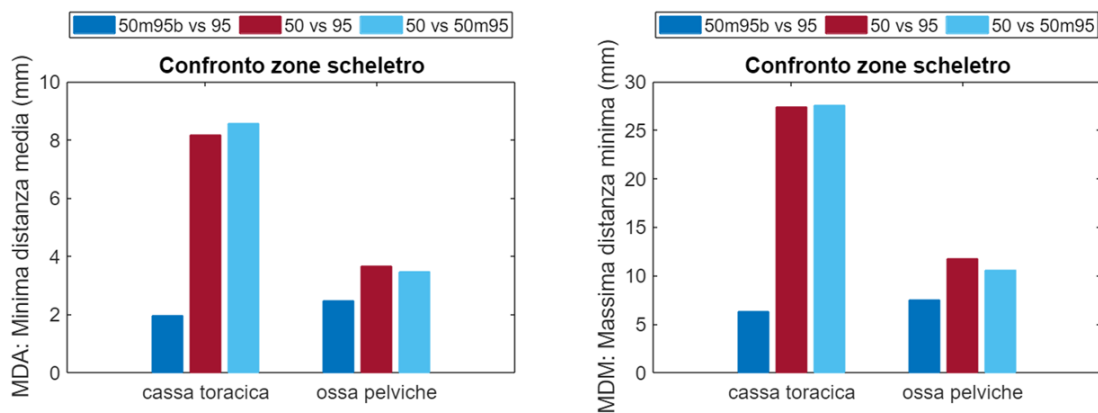


Figura 9.3: Valori di *MDA* e *MDM* per cassa toracica ed ossa pelviche

<i>MDA</i>			
	AM50 vs AM95	AM50m95 vs AM95	AM50 vs AM50m95
bacino	30.73601868	7.89961672	30.33149127
busto	24.67036177	7.10332986	21.30148135
viso	14.73757744	4.05160144	14.15879474
spalla	23.54336026	3.42970108	24.29498432
braccio	19.43713893	2.54923837	20.93198878
gomito	22.37176885	2.51390704	22.12322782
avanbraccio	16.16007953	1.90854884	16.34995647
polso	22.67779366	1.50079616	23.18019942
mano	23.20602721	3.65836255	23.55486203
coscia	16.38087976	5.60639369	13.54732149
ginocchio	16.00464924	2.7344848	16.26262522
stinco	14.87600467	1.68032126	14.86920897
caviglia	17.53889549	0.235875	17.63545228
piede	14.29612409	1.72807802	14.27663312
cassa toracica	1.97394784	8.17844736	8.57286371
ossa pelviche	2.48107774	3.65516682	3.48276479
media	17.56823157	3.652116813	17.80461599

Tabella 9.4: Valori della *MDA* nei confronti AM50 vs AM50m95 vs AM95

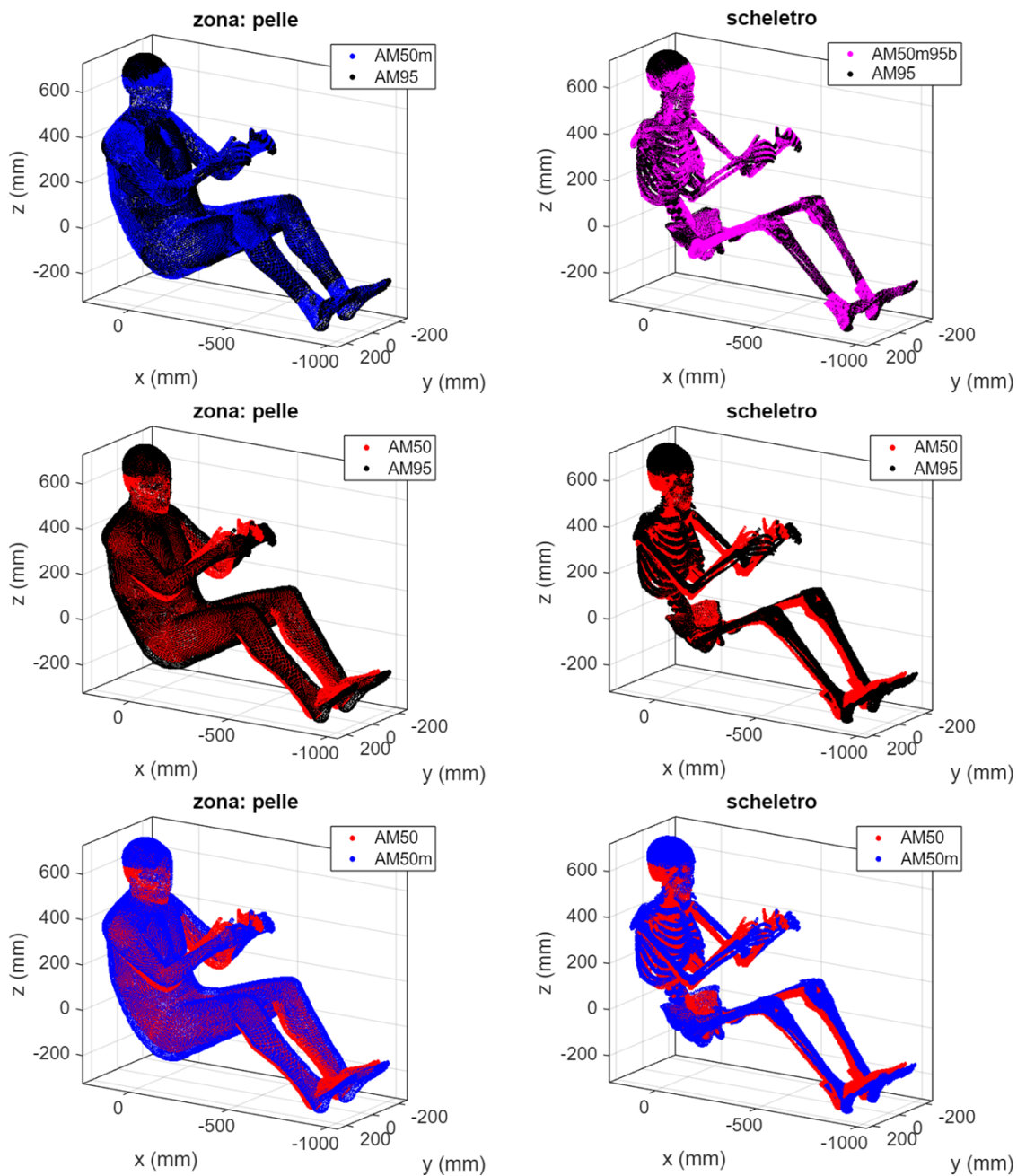


Figura 9.4: AM50 vs AM50m95, AM50m95 vs AM95 e AM50 vs AM50m95: forma esterna e interna

9.2 Cinematica nell'impatto frontale

Per analizzare la cinematica di THUMS a crash frontale sono state analizzate, per coordinate, le traiettorie dei cosiddetti punti di controllo, scelti quali i baricentri di ciascun set di source points (o edge omologhi) e contrassegnati di conseguenza. Confrontando opportunamente diverse configurazioni di simulazione in tal senso, è stato possibile giudicare 1) come influisce il mesh morphing sulla cinematica di THUMS e 2) se le modalità con cui influisce sono corrette.

9.2.1 Influenza del mesh morphing

Per studiare gli effetti del mesh morphing sulla cinematica di THUMS, sono state prese in considerazione le parametrizzazioni AMm35, AMm75 e AMm95 e confrontate rispetto a THUMS AM50 originale nel *frontal sled test*. In Tabella 9.5, per ciascun punto di controllo e per ciascuna delle suddette parametrizzazioni, sono riportati i valori dell' $S_{mean,x}$, ovvero lo scostamento mediamente riscontrato rispetto ad AM50 nell'evoluzione della coordinata x . La coordinata x rappresenta infatti la coordinata nella direzione frontale, dove si sviluppa maggiormente il moto dell'occupante durante il *frontal sled test*. Per tutte le parametrizzazioni, il massimo $S_{mean,x}$ si registra in corrispondenza di *busto-spalla destra* (al limite 65.38 mm per l'AMm95) ed il minimo in corrispondenza di *mano destra* (al limite 1.30 mm per AMm35), vedasi a tal proposito Figura 9.5.

Mediando i valori dell' $S_{mean,x}$ tra i vari punti di controllo è stata ottenuta una stima dello scostamento medio generale indotto dal mesh morphing ai vari percentili. È interessante notare che per ciascuna delle parametrizzazioni realizzate, rapportando i valori di questo scostamento (11.75 mm per l'AMm35, 19.99 mm per l'AMm75 e 35.86 mm per l'AMm95) all'escursione che esse rappresentano nei confronti del 50° percentile (15pt% per l'AMm35, 25pt% per l'AMm75 e 45pt% per l'AMm95), si ottengono i valori: 0.783, 0.799 e 0.796, tutti molto simili tra loro. Ciò significa che lo scostamento medio dal comportamento di AM50 è in buona approssimazione lineare rispetto al percentile antropometrico raggiunto col mesh morphing.

Per quanto riguarda l' $S_{max,x}$, ovvero il massimo scostamento rispetto all'AM50 nell'evoluzione della coordinata x , si riportano i valori calcolati² in Tabella 9.6, per i quali valgono le stesse considerazioni effettuate in merito all' $S_{mean,x}$. Per un approfondimento sulle traiettorie seguite da ciascun punto di controllo si rimanda al Paragrafo 9.3.5 in Appendice.

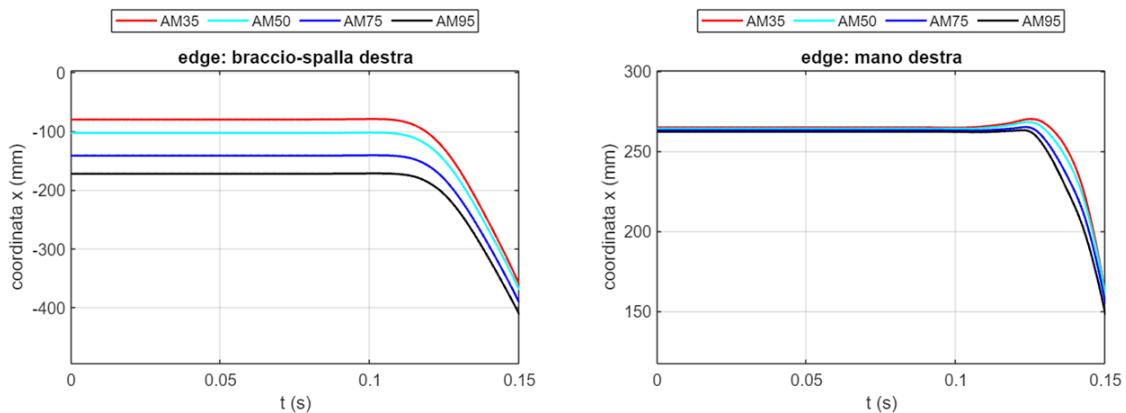


Figura 9.5: Massimo e minimo $S_{mean,x}$ riscontrati

²Sono riportate solo alcuni punti di controllo per alleggerire la trattazione, in Appendice la tabella completa

S_{meanx} [mm]			
	AMm35 vs AM50	AMm75 vs AM50	AMm95 vs AM50
bacino	14.5692	27.2591	48.1655
collo	19.9055	32.9198	59.6511
busto-spalla destra	23.0925	39.2095	69.8830
busto-spalla sinistra	19.9685	33.5836	60.7864
braccio-spalla destra	21.3802	36.1754	65.3824
braccio-spalla sinistra	18.7527	31.2618	56.5893
braccio-gomito destra	14.8410	25.4716	46.0103
braccio-gomito sinistra	13.6006	22.2288	40.0445
avanbraccio-gomito destra	11.1429	19.3081	34.9461
avanbraccio-gomito sinistra	10.9458	17.5526	31.2481
avanbraccio-polso destra	4.7046	8.2295	15.0021
avanbraccio-polso	5.3787	8.2168	14.4528
mano destra	1.3058	2.3655	4.4146
mano sinistra	2.4625	3.3628	5.6969
coscia-ginocchio destra	11.4918	21.8664	38.1416
coscia-ginocchio sinistra	12.1709	19.7262	35.7029
stinco-ginocchio destra	10.4424	19.3079	33.8270
stinco-ginocchio sinistra	10.8263	17.9033	32.3466
stinco-caviglia destra	8.3111	14.3203	25.3796
stinco-caviglia sinistra	8.1007	13.9041	25.1061
piede destra	7.6485	12.8227	23.0303
piede sinistra	7.6041	12.7698	23.0461
media	11.7566	19.9894	35.8570

Tabella 9.5: Scostamento medio rispetto ad AM50 nella coordinata x

S_{maxx} [mm]			
	AMm35 vs AM50	AMm75 vs AM50	AMm95 vs AM50
bacino	18.8517	31.4213	56.5584
collo	22.1792	36.9652	66.5374
busto-spalla destra	24.7456	41.6535	73.8678
...
...
stinco-caviglia sinistra	9.1129	15.2134	27.3584
piede destra	7.8998	13.3264	23.7672
piede sinistra	7.8395	12.9646	23.8993
media	15.7722	25.0007	44.8944

Tabella 9.6: Errore massimo rispetto ad AM50 nella coordinata x

9.2.2 THUMS AM50m95 vs THUMS AM95

Il paragrafo precedente ha mostrato l'influenza del mesh morphing di THUMS sul comportamento cinematico del FE HBM nel *frontal sled test*. In questo paragrafo si cerca di capire se tale influenza è ben contestualizzata, ovvero se morphando THUMS AM50 al 95° percentile si ottenga un comportamento a crash confrontabile con quello di THUMS AM95 preso come riferimento. Per far ciò sono stati calcolati i valori di S_{meanx} , S_{maxx} , S_{meany} , S_{maxy} , S_{meanz} ed S_{maxz} per ciascun punto di controllo nei confronti: AM50 vs AM95, AM50m95 vs AM95 e AM50m95b vs AM95. Prima di entrare nel dettaglio, si fa notare che a ciascun confronto sono associate precise funzionalità: AM50 vs AM95 costituisce il punto di partenza di questo lavoro e quindi il riferimento che permette di cogliere le caratteristiche introdotte dal mesh morphing; AM50m95 vs AM95 e AM50m95b vs AM95 costituiscono invece il punto di arrivo, e quindi il riferimento per valutare l'efficacia del mesh morphing.

Analizzando i valori dell' S_{meanx} in Tabella 9.7 è possibile subito notare come per quasi tutti i punti di controllo lo scostamento medio dal comportamento dell' AM95 si riduca drasticamente con il morphing. Fanno tuttavia eccezione in tal senso i punti di controllo da stinco-caviglia destra a scendere. In merito agli estremi del confronto, il miglior fit del comportamento target è rappresentato da *avanbraccio polso sinistra* di AM50m95b, con S_{meanx} pari a 0.62 mm mentre il peggiore (19.62 mm) da *piede destra* dello stesso AM50m95b (Figura 9.6).

Mediando i valori di S_{meanx} su tutti i punti di controllo è stata ottenuta una misura generale dello scostamento medio dal comportamento target e in tal senso i risultati sono molto positivi. Con il mesh morphing dal 50° al 95° percentile lo scostamento dall'AM95 originale si riduce del 77% passando da un valore di 34.43 mm a 7.91 mm per quanto riguarda AM50m95 e 7.85 mm per quanto riguarda AM50m95b. Se si fa riferimento alla Tabella 9.5 al paragrafo precedente, si può inoltre verificare come lo scostamento medio di AM50 da AM95 sia solo il 4% diverso dallo scostamento medio di AM50 da AM50m95.

L'entusiasmo per i miglioramenti mediamente ottenuti è tuttavia mitigato dai valori riscontrati in termini di S_{maxx} . Come si evince osservando la Tabella 9.8, valori dello scostamento medio globale inferiori al centimetro non danno traccia di scostamenti massimi, sebbene localizzati su taluni punti di controllo, che raggiungono anche i 6/7 cm, come ad esempio *spalla-busto destra* e *mano-destra* in Figura 9.7. Ragionando in termini globali, lo scostamento massimo riscontrato dal comportamento di AM95, sia per AM50m95 che per AM50m95b si attesta intorno ai 4 cm che sebbene per la scala del problema siano ampiamente ammissibili non rappresentano più un sostanziale miglioramento rispetto al punto di partenza, che risulta essere appena il 22% a fronte del 77% di prima. È importante sottolineare comunque che, il massimo scostamento globale tra AM50m95 (e/o AM50m95b) e AM95 risulta comunque inferiore rispetto a quello registrato tra AM50 e AM50m95 in Tabella 9.6.

In merito a S_{meany} , S_{maxy} , S_{meanz} ed S_{maxz} valgono considerazioni analoghe e per brevità non si riportano i valori rispettivi ai punti di controllo, privilegiando quelli mediati globalmente. Si faccia riferimento a tal proposito a Tabella 9.9, Tabella 9.10, Tabella 9.11 e Tabella 9.12. Le tabelle complete, insieme al grafico di tutte le traiettorie dei punti di controllo sono riportati in Appendice, ai Paragrafi 9.3.6, 9.3.7, 9.3.8.

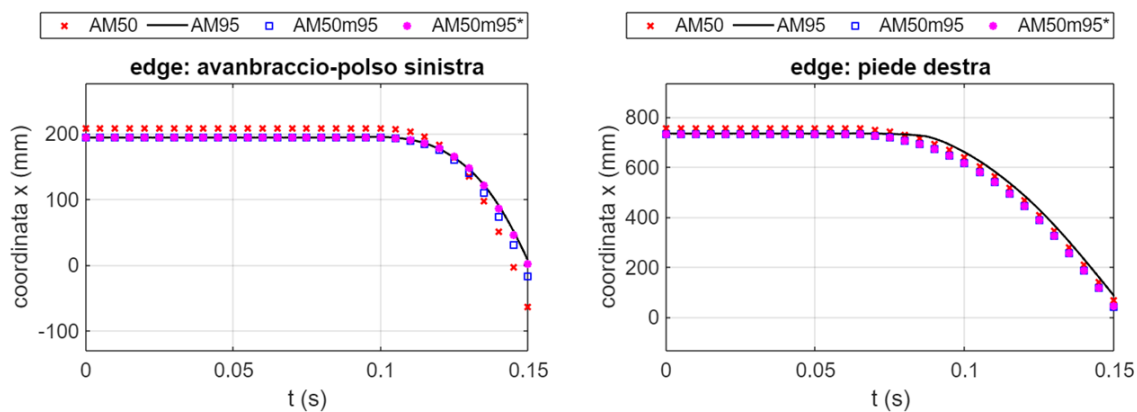
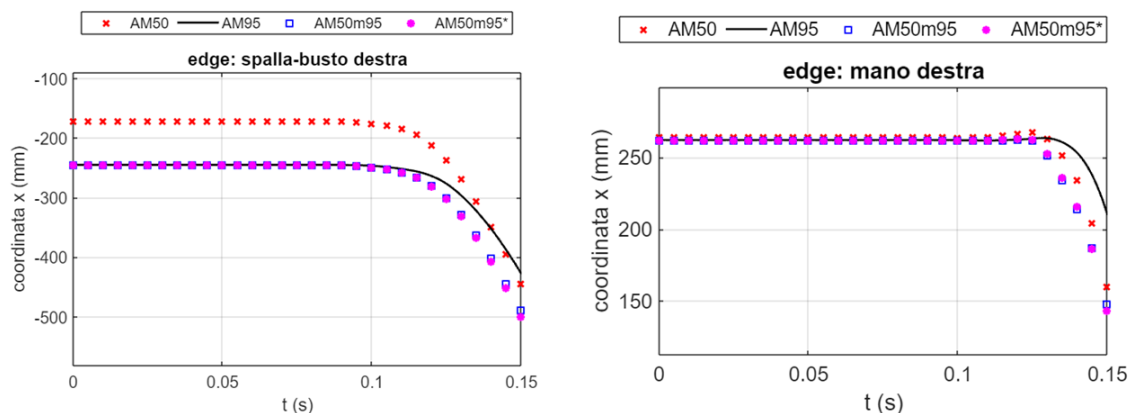


Figura 9.6: Migliore e peggior fit del comportamento target in merito ai punti di controllo

	S_{meanx} [mm]		
	AM50 vs AM95	AMm95 vs AM95	AMm95b vs AM95
bacino	55.8978	8.5703	8.6528
collo	54.7108	6.6763	8.7003
busto-spalla destra	61.7294	9.0008	9.8750
busto-spalla sinistra	58.3604	5.6298	4.3405
braccio-spalla destra	58.4107	8.5050	8.7849
braccio-spalla sinistra	56.1065	3.8125	1.6364
braccio-gomito destra	42.0352	6.9175	6.3161
braccio-gomito sinistra	40.9808	3.5321	1.0465
avanbraccio-gomito destra	33.1151	7.0263	6.5018
avanbraccio-gomito sinistra	33.1595	3.3562	0.9831
avanbraccio-polso destra	14.5444	5.8647	5.6235
avanbraccio-polso sinistra	16.4536	2.2589	0.6620
mano destra	4.8533	5.4218	5.3774
mano sinistra	8.1954	2.5092	1.3628
coscia-ginocchio destra	40.5625	7.4457	11.0736
coscia-ginocchio sinistra	38.1619	7.6255	8.2826
stinco-ginocchio destra	34.5765	6.0999	9.2807
stinco-ginocchio sinistra	32.4107	5.4842	6.3898
stinco-caviglia destra	17.3104	14.4397	13.9189
stinco-caviglia sinistra	17.8489	14.7097	14.5714
piede destra	18.9744	19.6219	19.6555
piede sinistra	18.9943	19.6291	19.6121
media	34.4269	7.9153	7.8476

Tabella 9.7: Scostamento medio rispetto ad AM95 nella coordinata x

Figura 9.7: Evoluzione cinematica di *busto-spalla destra* e *mano destra*

S_{maxx} [mm]	S_{maxx} [mm]		
	AM50 vs AM95	AMm95 vs AM95	AMm95b vs AM95
bacino	66.0608	48.3838	46.9607
collo	66.5374	48.9306	58.8649
busto-spalla destra	73.4974	60.6840	71.5630
busto-spalla sinistra	73.4961	30.2828	19.4385
braccio-spalla destra	69.6227	59.1353	66.9830
braccio-spalla sinistra	69.2116	23.3649	7.9769
braccio-gomito destra	48.2600	60.3276	62.9140
braccio-gomito sinistra	49.5796	22.4685	4.9736
avanbraccio-gomito destra	42.5880	60.4150	63.6591
avanbraccio-gomito sinistra	63.5258	23.0813	6.2558
avanbraccio-polso destra	46.7413	61.4218	65.6783
avanbraccio-polso sinistra	67.1806	21.7618	4.1335
mano destra	50.1796	62.1707	66.5442
mano sinistra	71.1755	22.3159	5.1407
coscia-ginocchio destra	50.8966	26.4992	53.5750
coscia-ginocchio sinistra	43.4564	37.5773	36.9529
stinco-ginocchio destra	40.0763	20.4830	42.5380
stinco-ginocchio sinistra	40.0763	23.7260	25.4500
stinco-caviglia destra	27.3434	38.9268	38.9879
stinco-caviglia sinistra	27.3531	38.8835	39.1511
piede destra sinistra	23.7706	41.9874	41.9361
piede sinistra	23.7843	42.0085	42.0591
media	51.5643	39.7653	39.6244

Tabella 9.8: Scostamento massimo rispetto ad AM95 nella coordinata x

$S_{mean y}$ [mm]			
	AM50 vs AM95	AMm95 vs AM95	AMm95b vs AM95
bacino	1.2098	0.6931	1.6445
...
media	8.8722	2.4004	2.2915

Tabella 9.9: Scostamento medio rispetto ad AM95 nella coordinata y

$S_{max y}$ [mm]			
	AM50 vs AM95	AMm95 vs AM95	AMm95b vs AM95
bacino	10.9768	6.0695	13.1258
...
media	16.7762	13.8212	13.3922

Tabella 9.10: Scostamento massimo rispetto ad AM95 nella coordinata y

$S_{mean z}$ [mm]			
	AM50 vs AM95	AMm95 vs AM95	AMm95b vs AM95
bacino	19.4174	7.1945	5.5078
...
media	17.1049	8.7038	8.4699

Tabella 9.11: Scostamento medio rispetto ad AM95 nella coordinata z

$S_{max z}$ [mm]			
	AM50 vs AM95	AMm95 vs AM95	AMm95b vs AM95
bacino	31.5364	31.6836	26.0788
...
media	29.9677	43.2996	40.9207

Tabella 9.12: Scostamento massimo rispetto ad AM95 nella coordinata z

9.2.3 Considerazioni

Sintetizzando generosamente i risultati mostrati, il criterio di mesh morphing introdotto in questo lavoro consente di cogliere efficacemente le variazioni di forma che occorrono al limite della parametrizzazione. Ciò si riflette su uno scostamento medio molto piccolo tra la cinematica a crash frontale di THUMS AM50m95 rispetto a quella di THUMS AM95 originale. In merito allo scostamento massimo tuttavia, valori globali non altrettanto buoni hanno creato incertezza sull'esito assoluto della verifica. A partire dai suddetti presupposti sono state indagate a fondo le cause dell'incongruenza tra ottimi scostamenti medi e "passabili" scostamenti massimi. Si ritiene che esse possano essere identificate nel posizionamento del FE HBM rispetto ai vincoli all'istante iniziale di simulazione. Come si mostra in Figura 9.8, tra THUMS AM95 e la seduta vi è un gap che invece non è presente per THUMS AM50m95. Sempre in merito alla seduta, essa risulta per THUMS AM95 più inclinata di quanto non lo sia per THUMS AM50m95 che oltretutto è seduto più spostato in avanti, disponendo di meno superficie di contatto. L'ultima differenza notevole è rappresentata infine dalla posizione dei piedi dei due modelli rispetto al pavimento dell'autoveicolo: poggiati con la pianta per l'AM50m95 e con le punte per l'AM95. Queste differenze che appaiono piccole e apparentemente di poco conto, e che in

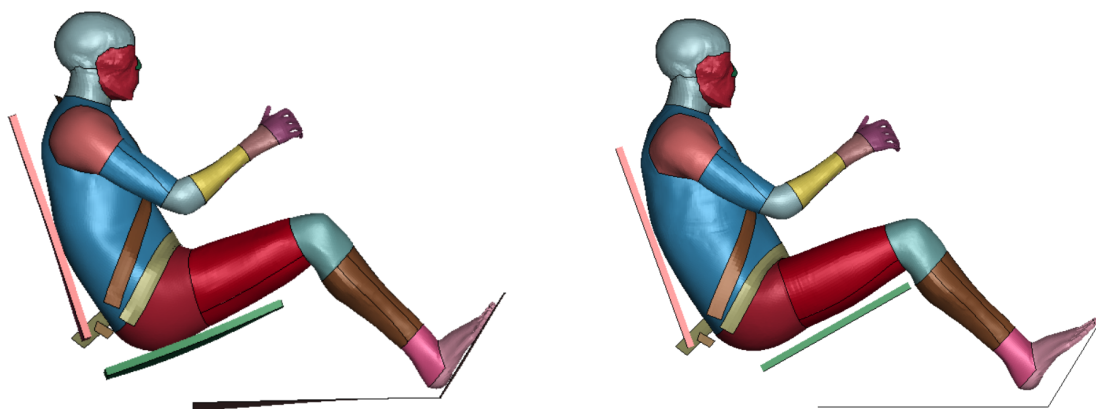


Figura 9.8: Posizionamento iniziale di THUMS AM50m95 e THUMS AM95 rispetto ai vincoli

effetti tali sono state ritenute al momento di generare le diverse configurazioni di simulazione, si crede abbiano in realtà effetti decisivi sull'evoluzione del crash. A titolo di esempio si riporta in Figura 9.9 una fotografia dell'istante di simulazione corrispondente a 75 millesimi di secondo. L'istante in questione è il momento in cui THUMS AM95 recupera definitivamente il gap con la seduta per effetto della forza di gravità. Nello stesso istante THUMS AM50m95, che era già a contatto con la seduta, ha già subito una quota parte della decelerazione trasmessa per attrito e mostra infatti segni di deformazione sulla zona delle cosce e di un incipiente rotazione a ribaltamento del busto. Nella stessa foto si veda ancora come i piedi di THUMS AM50m95 premono decisamente sul pavimento mentre quelli di THUMS AM95 sono appena poggiati. A fine simulazione, Figura 9.10, complici le differenze vincolari appena elencate, THUMS AM50m95 si ritrova molto più avanti sulla seduta, che fornisce dunque una superficie di contatto minore rispetto a quella riscontrata in THUMS AM95, oltretutto più inclinata. L'inerzia di THUMS morphato risulta dunque meno vincolata dal contatto col sedile nella zona del busto e più vincolata nel contatto col pavimento del veicolo nella zona degli arti inferiori. Il risultato è un ripiegamento sulla cintura di sicurezza più accentuato che per il THUMS originale.

Per concludere, si fa presente che quanto scritto in questo paragrafo costituisce un'ipotesi, sebbene ritenuta lecita, per la cui verifica si rimanda ad un'eventuale analisi di sensibilità sul posizionamento di THUMS rispetto ai vincoli di simulazione.

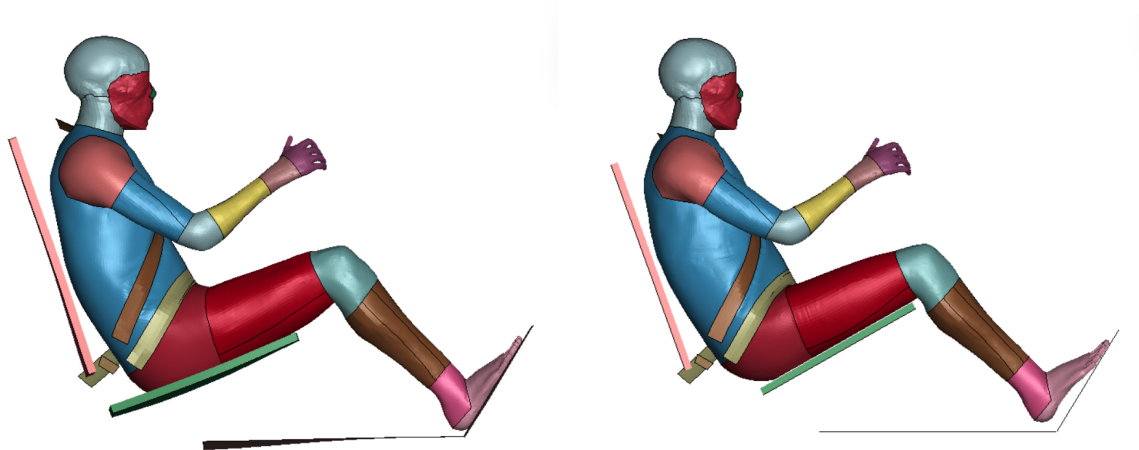


Figura 9.9: Posizionamento di THUMS AM50m95 e THUMS AM95 rispetto ai vincoli a $t = 0.075s$

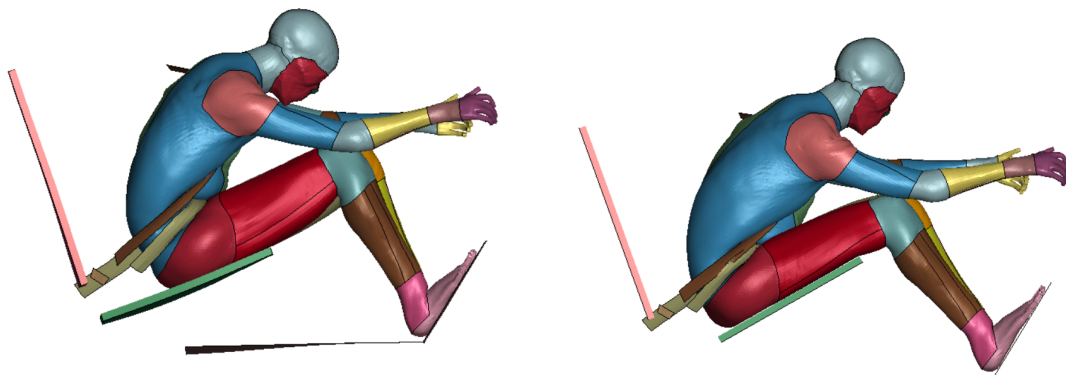


Figura 9.10: Posizionamento di THUMS AM50m95 e THUMS AM95 a fine simulazione

Conclusioni

Il lavoro di tesi ha esplorato la possibilità di realizzare una parametrizzazione del Total Human Model for Safety (THUMS) in base al percentile antropometrico statistico sfruttando il mesh morphing di THUMS AM50 guidato da radial basis function (RBF). Ripercorrendo le fasi di questo studio, si è dapprima stabilito il criterio con il quale implementare il mesh morphing, riconoscendo nell'opportuna modulazione delle differenze di forma tra THUMS AM50 ed AM95 la legge di variabilità antropometrica esistente tra il generico percentile statistico ed il 50°. È stato quindi definito il problema alle RBF selezionando come source points i nodi corrispondenti a 22 edges ben distribuiti sulla forma esterna del FE HBM e formalizzando il calcolo del campo di spostamenti da assegnare loro. A questo punto, ricorrendo ad una versione semplificata di THUMS è stato eseguito un primo tentativo di mesh morphing dal 50° al 95° percentile, mostrando come pur controllando la trasformazione dalla forma esterna del modello, la forma interna si modifica secondo una logica non troppo distante da quella insita nella variabilità antropometrica. Ritornando a lavorare sul THUMS completo, il processo di mesh morphing dal 50° al generico percentile è stato infine completamente automatizzato in tutte le sue fasi, producendo il primo risultato notevole di questo studio: poter realizzare il mesh morphing e la parametrizzazione di THUMS che esso rappresenta in circa 10 secondi.

Spostando il focus del lavoro dalla modellazione alla verifica, si è tentato di valutare l'efficacia della strategia di mesh morphing adottata. Dal punto di vista geometrico sono stati allora definiti i parametri *MDA* e *MDM* con i quali confrontare THUMS AM95 alla sua controparte parametrica AM50m95 e dal punto di vista della simulazione, le configurazioni THUMS AM50m35, THUMS AM50m75, THUMS AM50m95 e THUMS AM50m95b sono state impiegate in quella che è forse la più classica delle applicazioni riguardanti i FE HBM, ovvero il crash frontale, rappresentato dal *Frontal sled test*. La verifica geometrica ha confermato la buona rispondenza tra la forma di THUMS AM95 e quella di AM50m95, che con uno scostamento medio globale calcolato di appena 3.6 mm risulta circa l'80% meno distante da AM95 rispetto a quanto lo sia invece da AM50. Le simulazioni hanno evidenziato invece gli effetti del mesh morphing sulla cinematica a crash dell'occupante. Si è osservato come il comportamento di THUMS in tal senso vari circa linearmente in base al percentile antropometrico e come al limite del dominio di parametrizzazione, ovvero al 95° percentile, esso presenti somiglianze con il comportamento di THUMS AM95 originale. Effettuando il mesh morphing di AM50 in AM50m95, lo scostamento medio globale dal comportamento a crash di AM95 passa infatti da 34.4 mm a 7.9 mm, riducendosi ovvero di circa il 77%. I valori riportati, tuttavia, sono stati messi in dubbio dai valori dello scostamento massimo globale riscontrato, non altrettanto positivi e ritenuti tali a causa di effetti legati a piccole differenze di posizionamento iniziale dei due THUMS rispetto ai vincoli di simulazione.

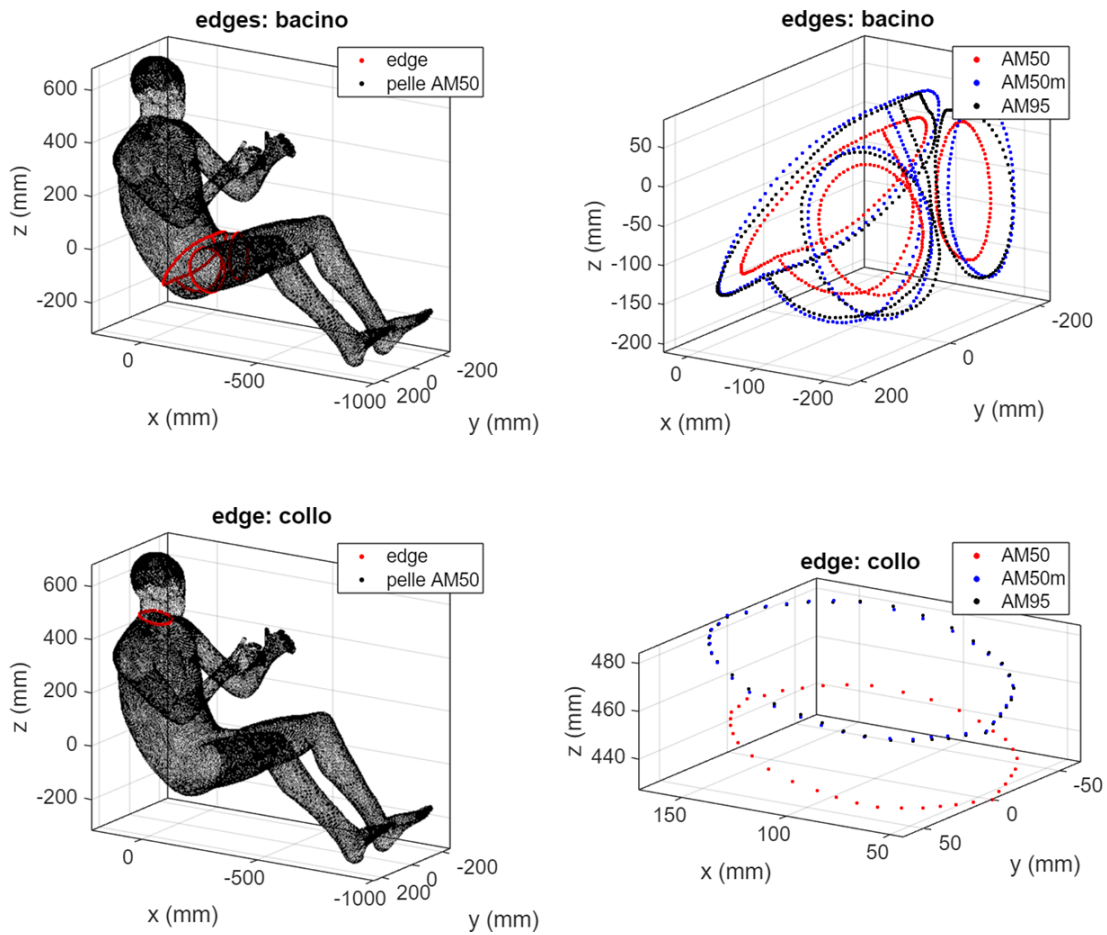
In conclusione, le scelte operate per parametrizzare THUMS in base al percentile antropometrico costituiscono una soluzione certamente efficiente nell'ambito della personalizzazione di FE HBMs, la cui efficacia, tuttavia, nonostante i buoni presupposti rappresentati dai risultati mostrati, rimane ancora da approfondire in eventuali sviluppi futuri.

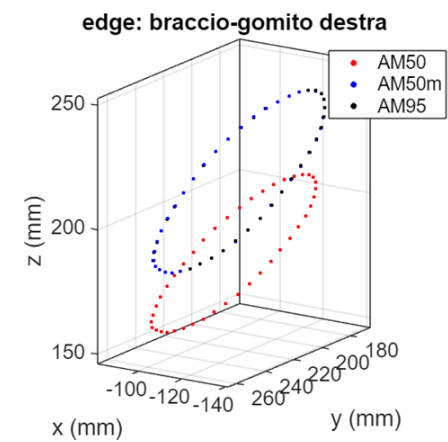
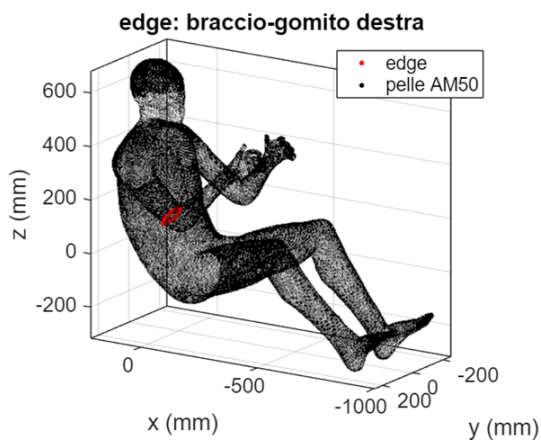
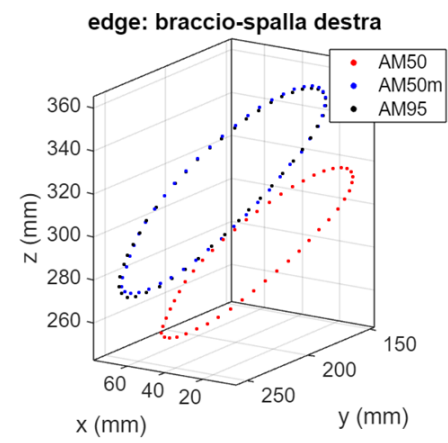
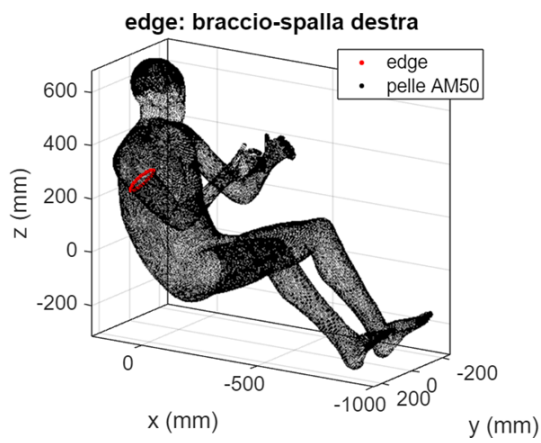
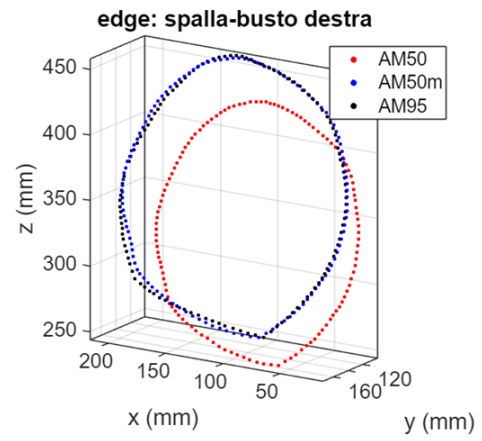
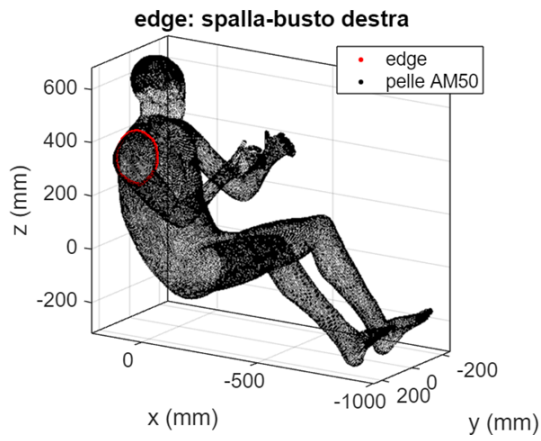
Appendice

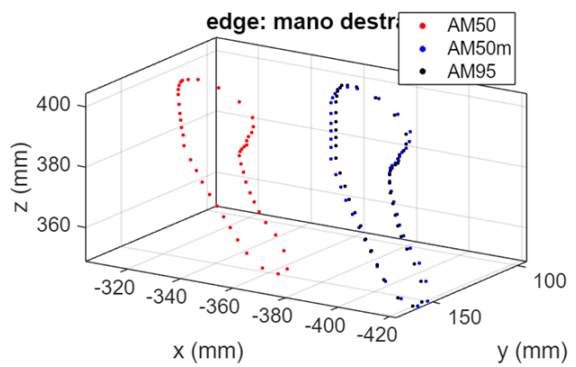
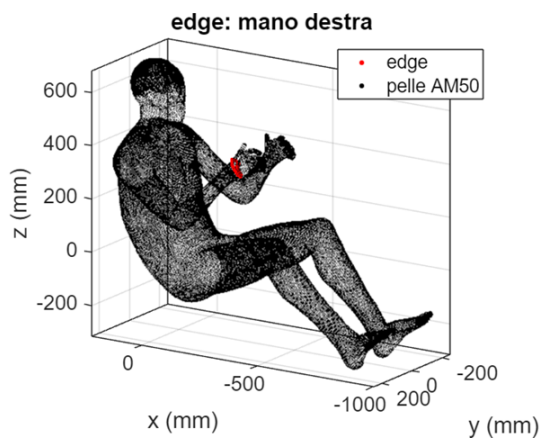
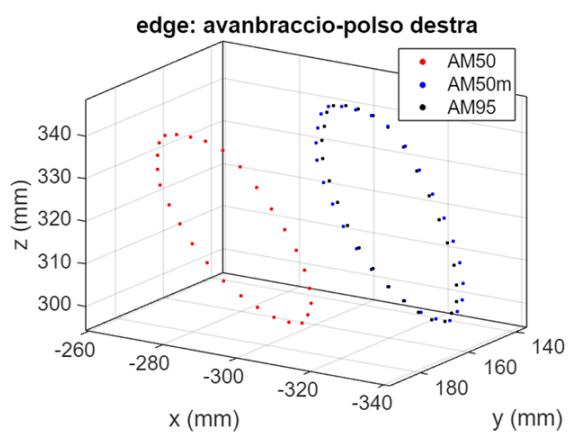
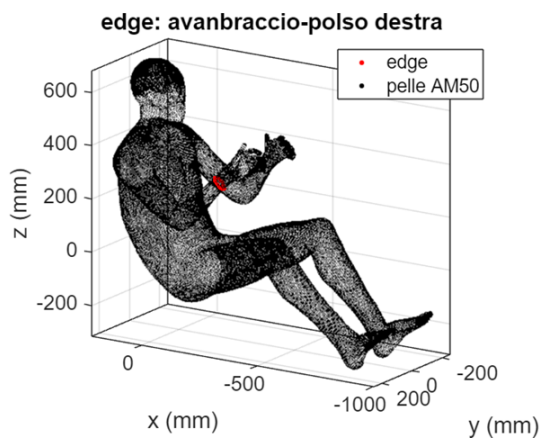
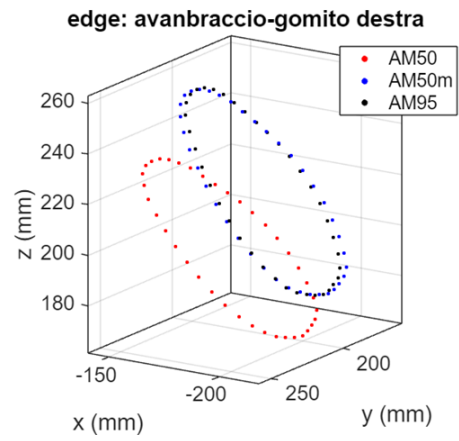
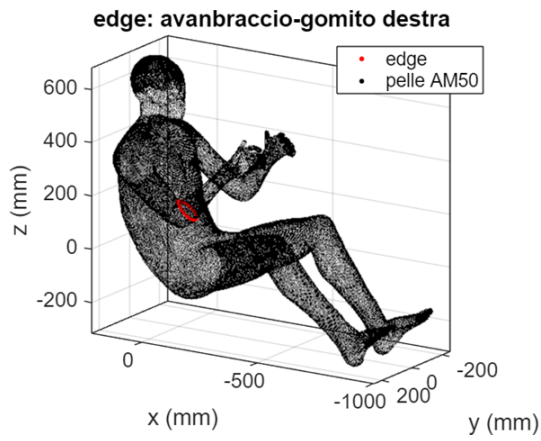
9.3 Grafici

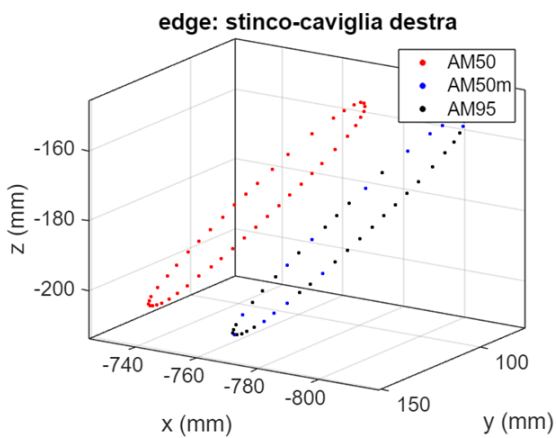
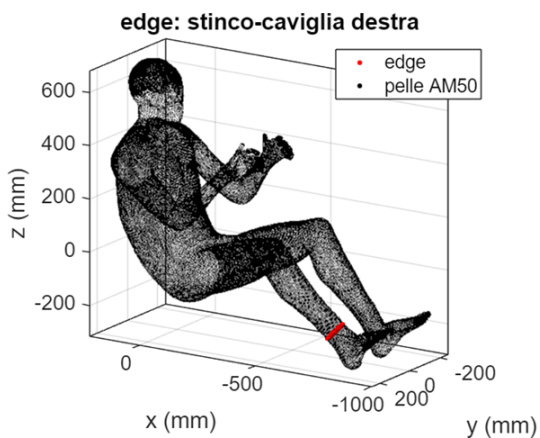
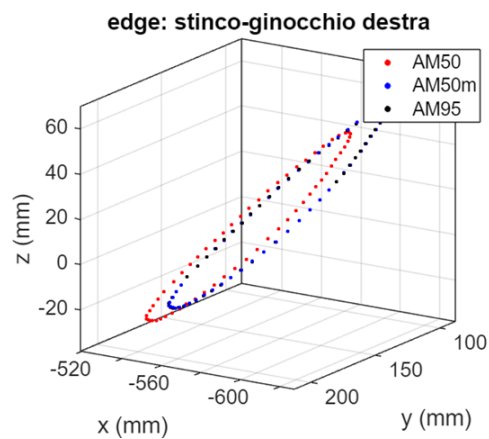
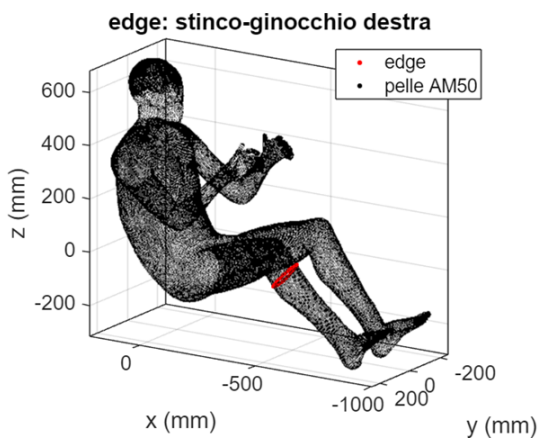
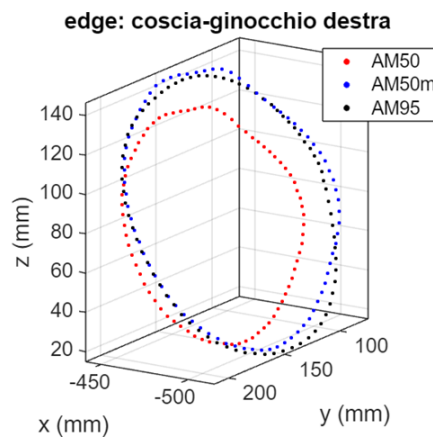
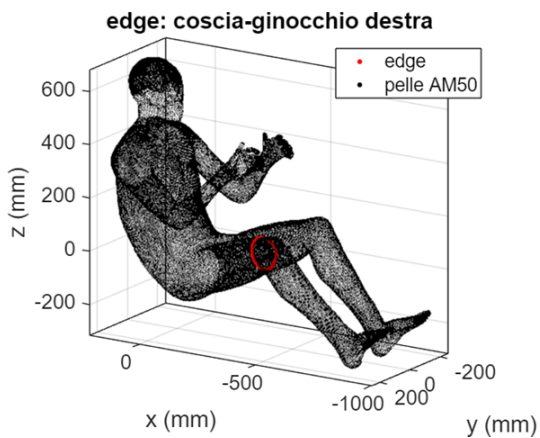
9.3.1 Source points

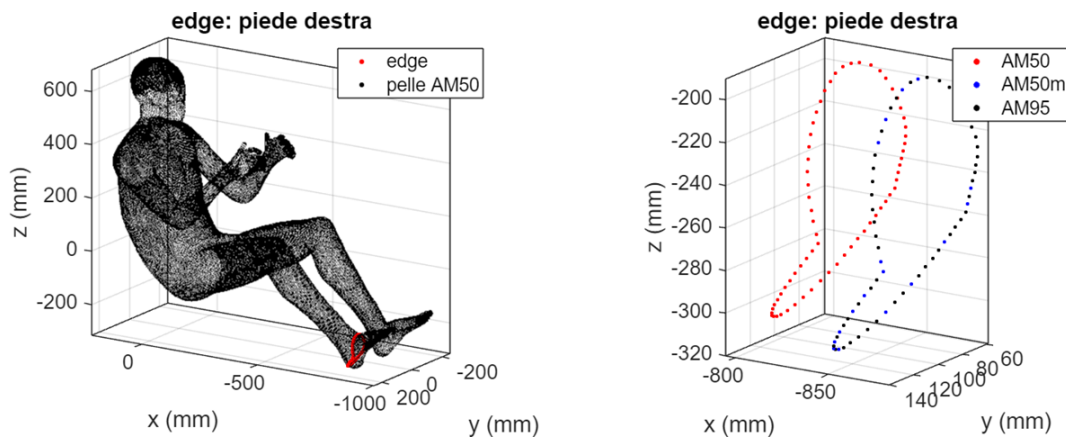
Si riportano a seguire i grafici identificativi dei source points e del campo di spostamenti loro assegnato. Nelle figure di destra si evidenzia la posizione dei source points rispetto alla totalità dei nodi della pelle, nelle figure di destra si confrontano i source points con gli omologhi di AM95 prima (AM50) e dopo il mesh morphing (AM50m).





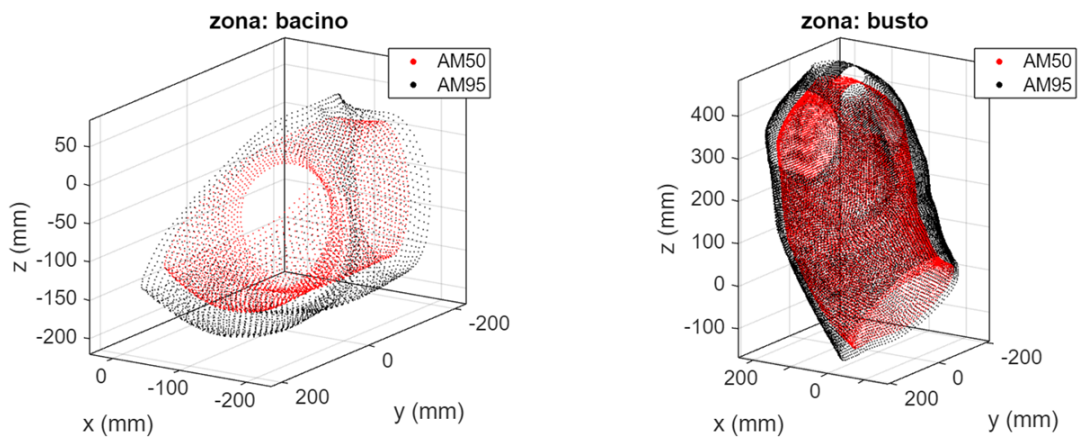


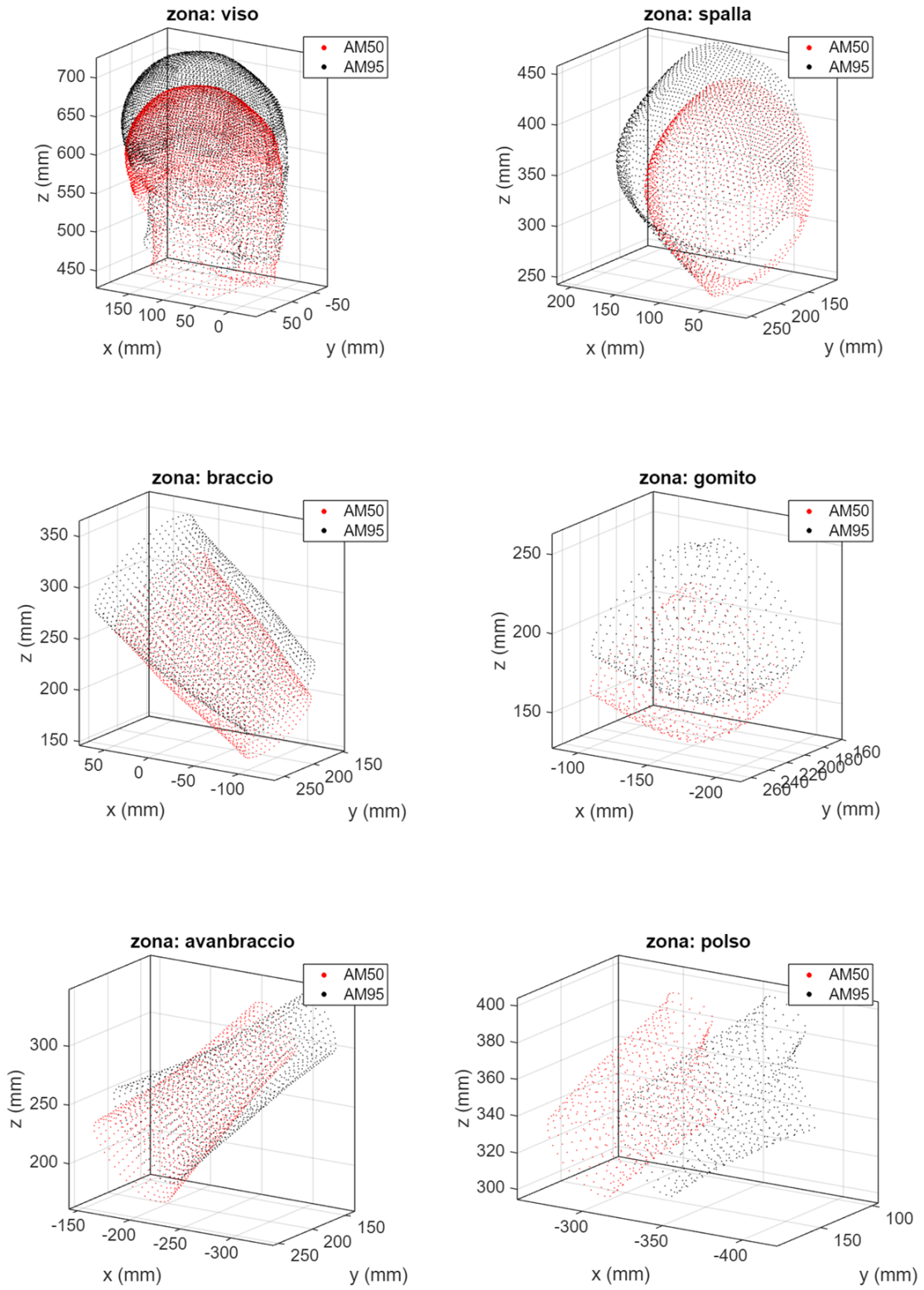


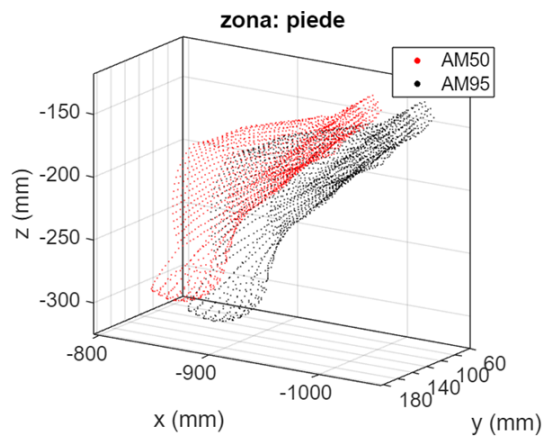
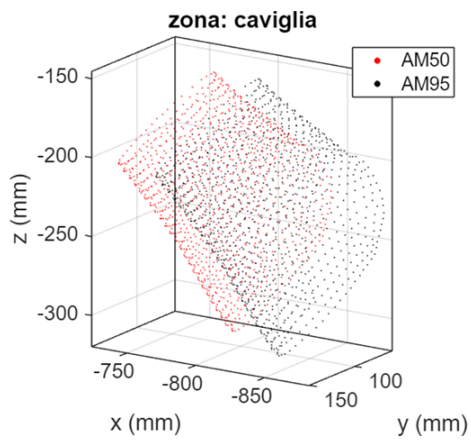
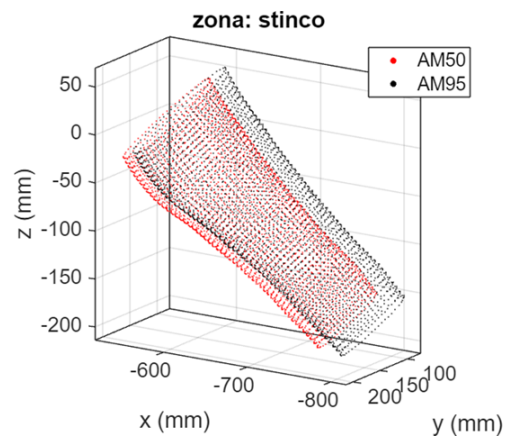
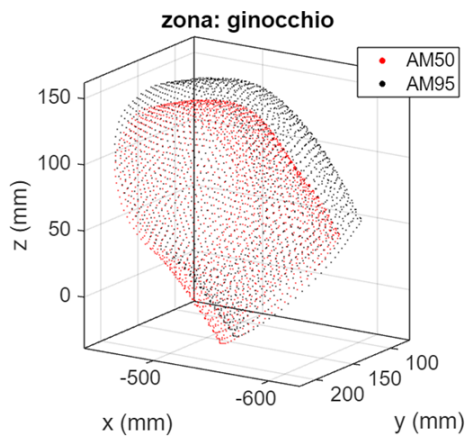
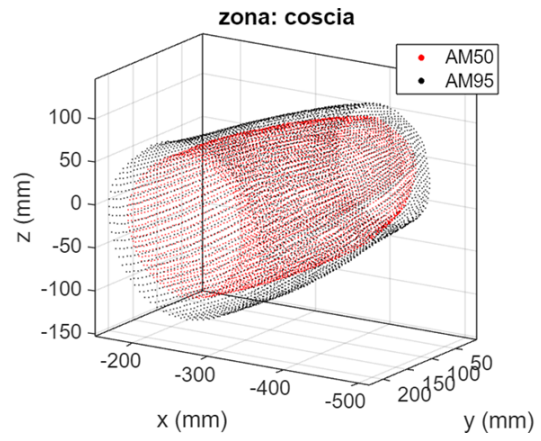
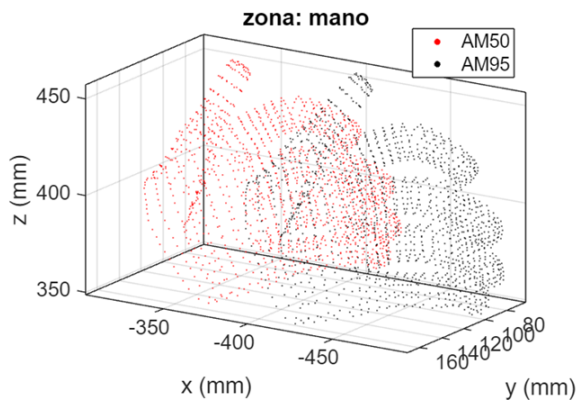


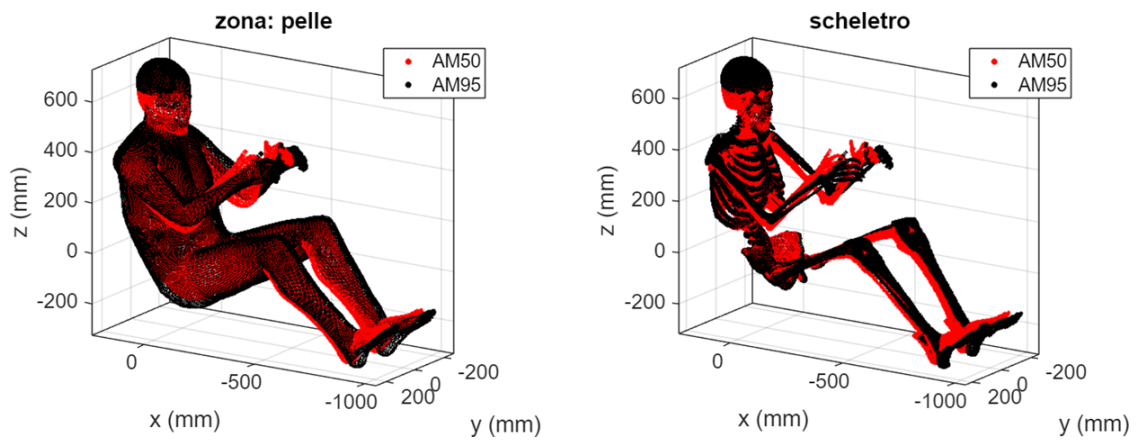
9.3.2 Confronto geometrico: THUMS AM50 vs THUMS AM95

Si riportano a seguire i grafici identificativi del confronto tra le geometrie di THUMS AM50 e THUMS AM95 per ciascuna delle zone ritenute di particolare interesse in questa trattazione.



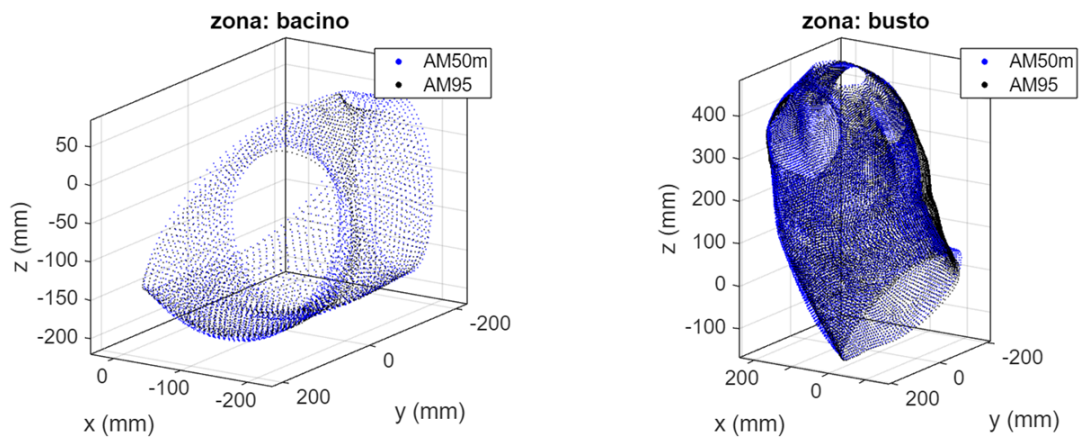


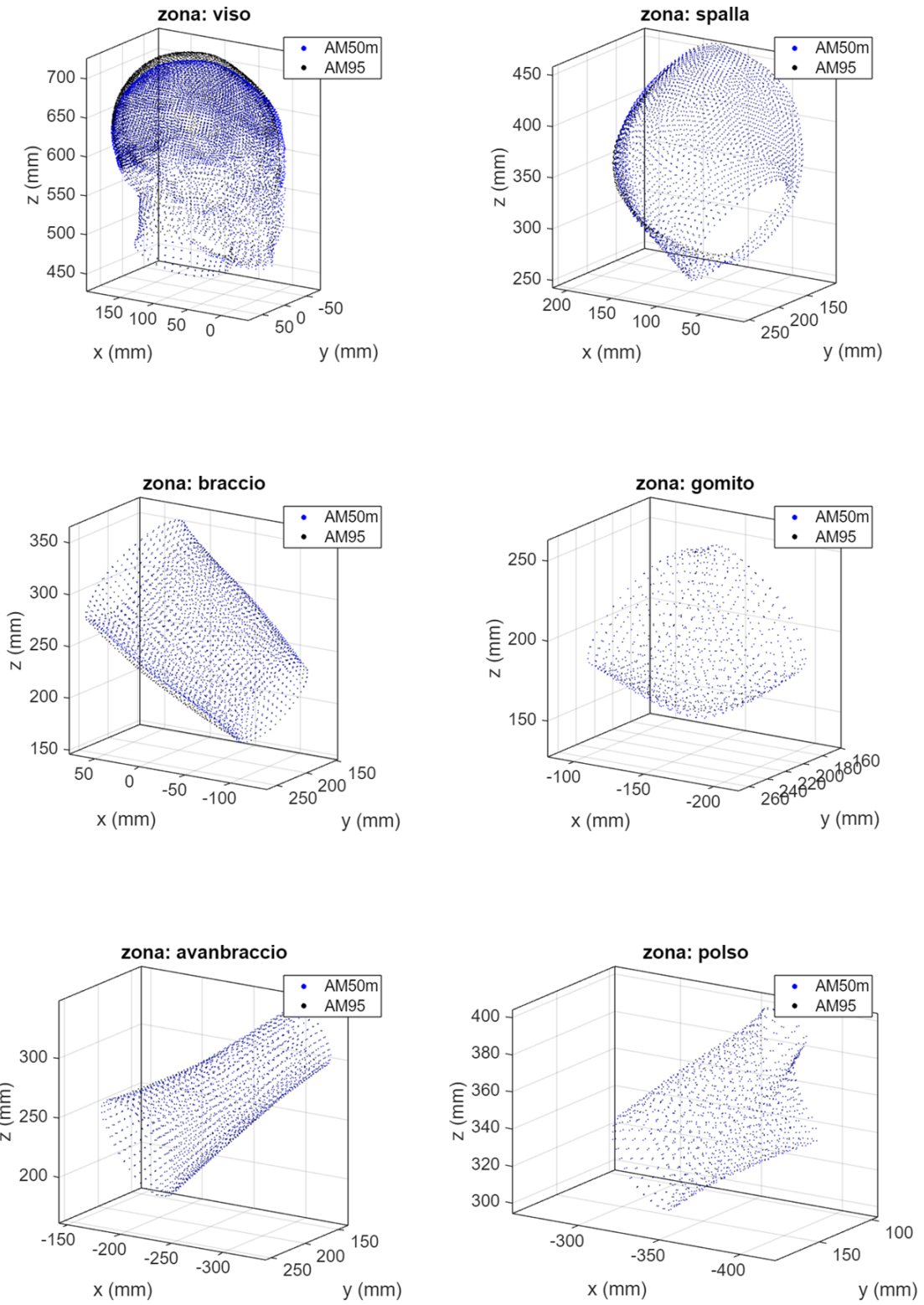


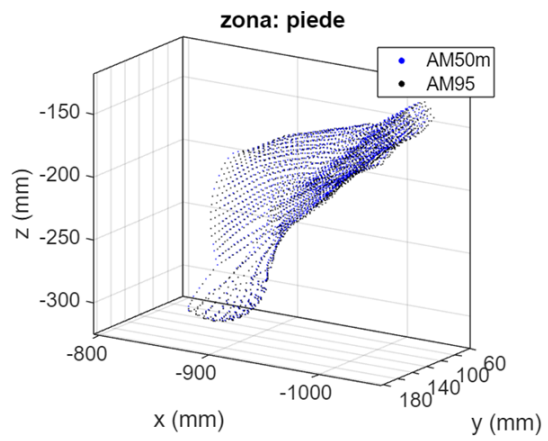
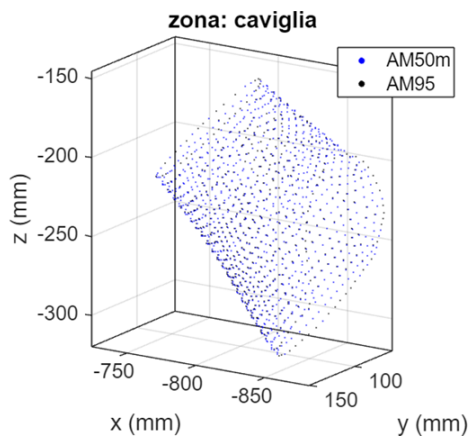
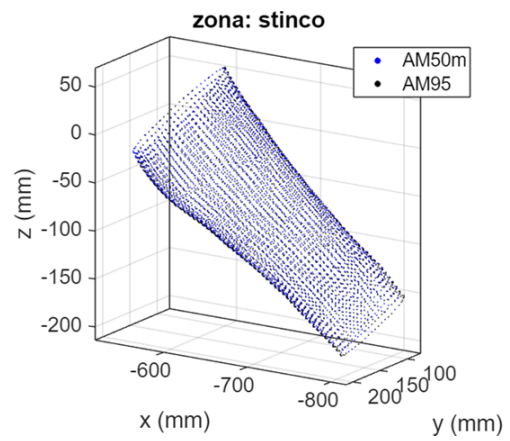
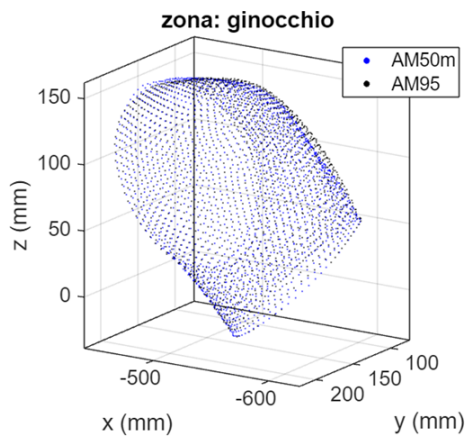
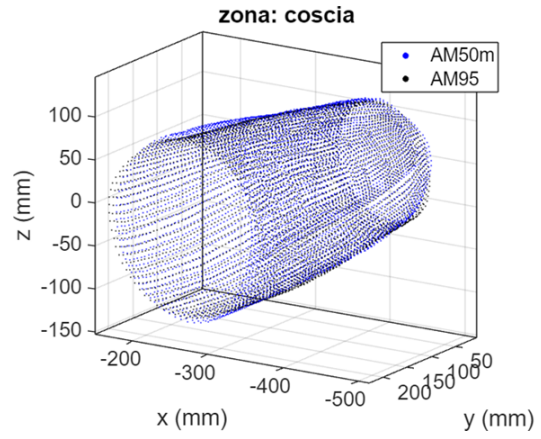
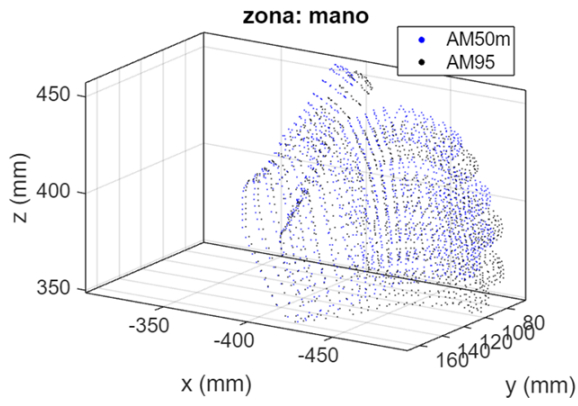


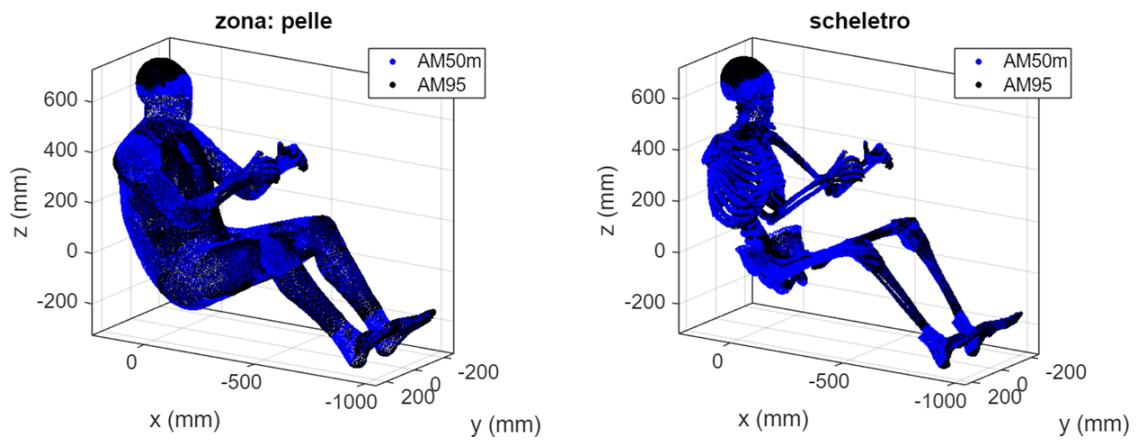
9.3.3 Confronto geometrico: THUMS AM50m vs THUMS AM95

Si riportano a seguire i grafici identificativi del confronto tra le geometrie di THUMS AM50m95 e THUMS AM95 per ciascuna delle zone ritenute di particolare interesse in questa trattazione.



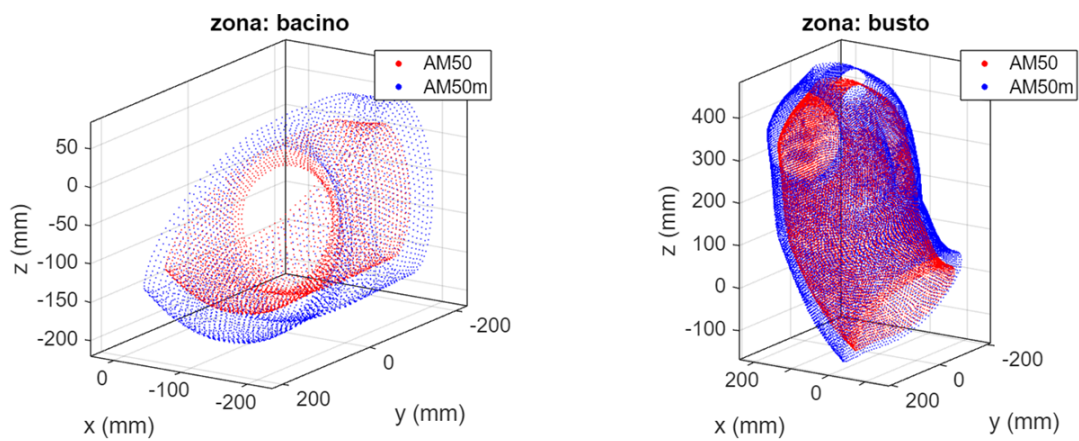


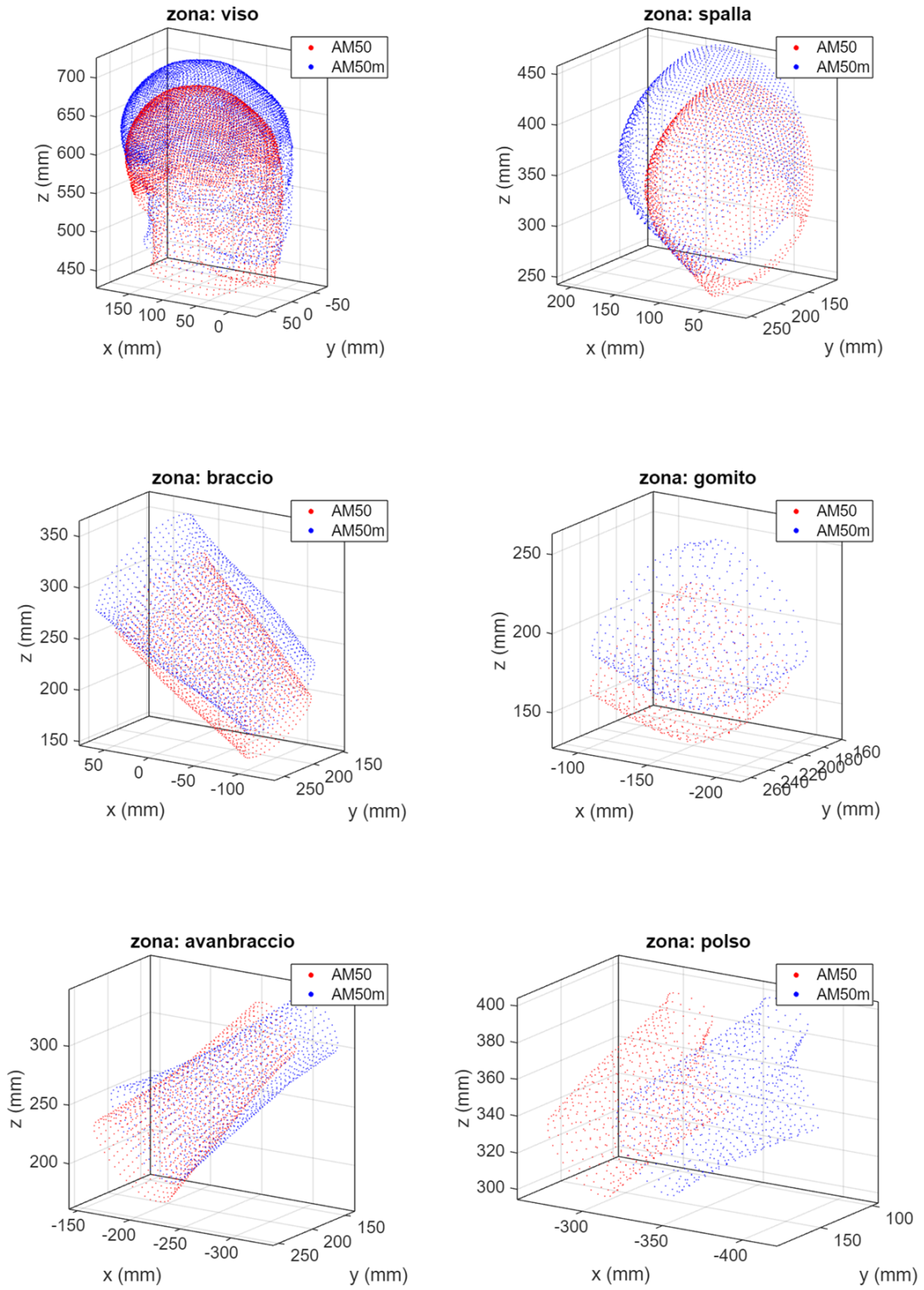


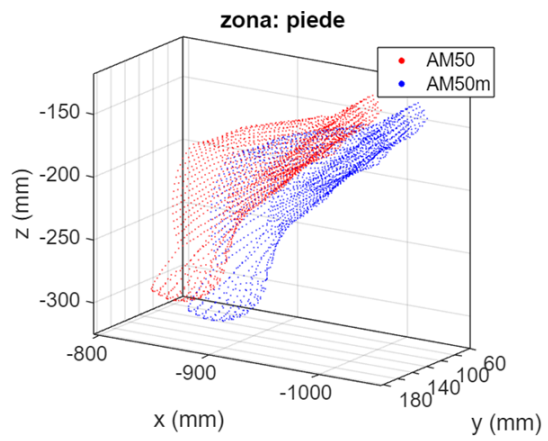
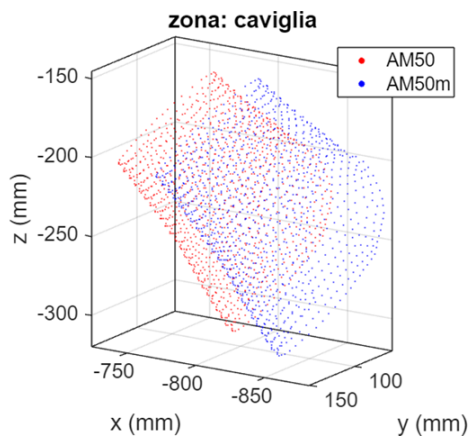
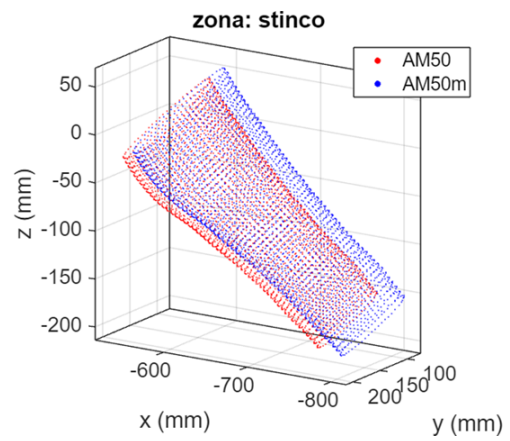
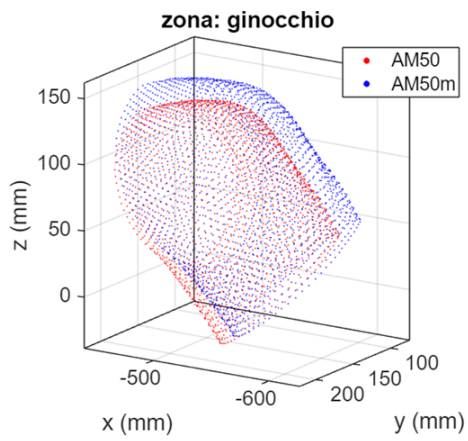
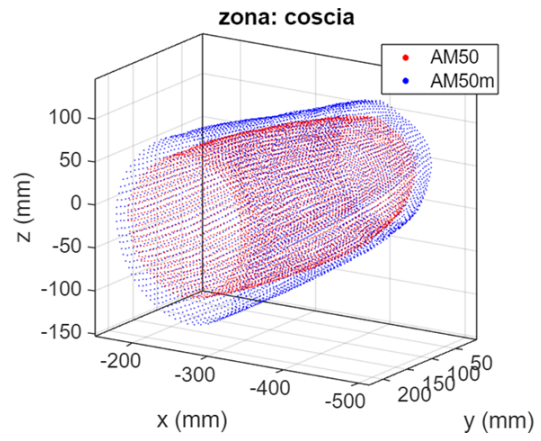
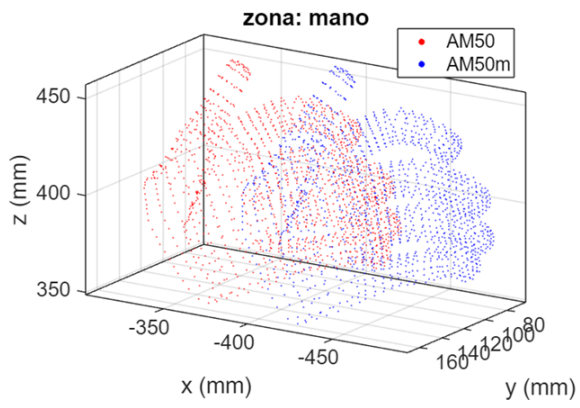


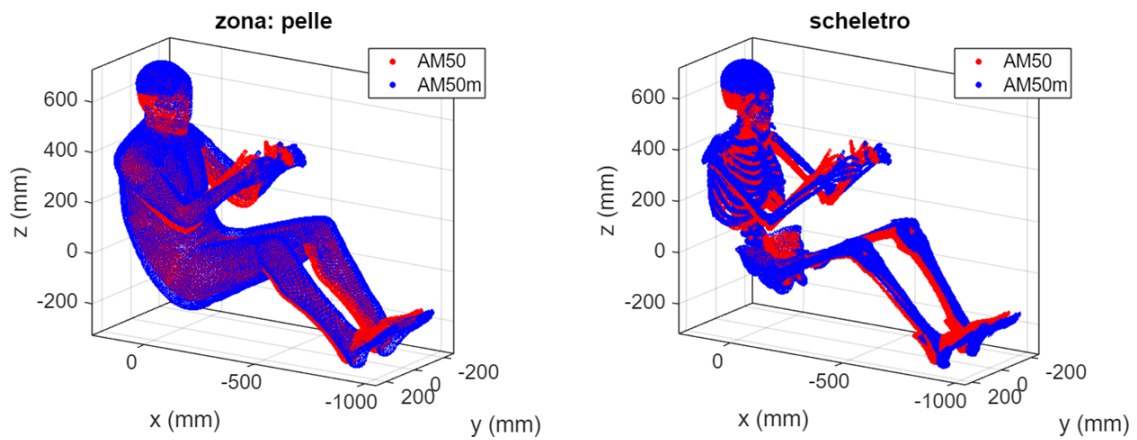
9.3.4 Confronto geometrico: THUMS AM50 vs THUMS AM50m

Si riportano a seguire i grafici identificativi del confronto tra le geometrie di THUMS AM50 e THUMS AM50m95 per ciascuna delle zone ritenute di particolare interesse in questa trattazione.



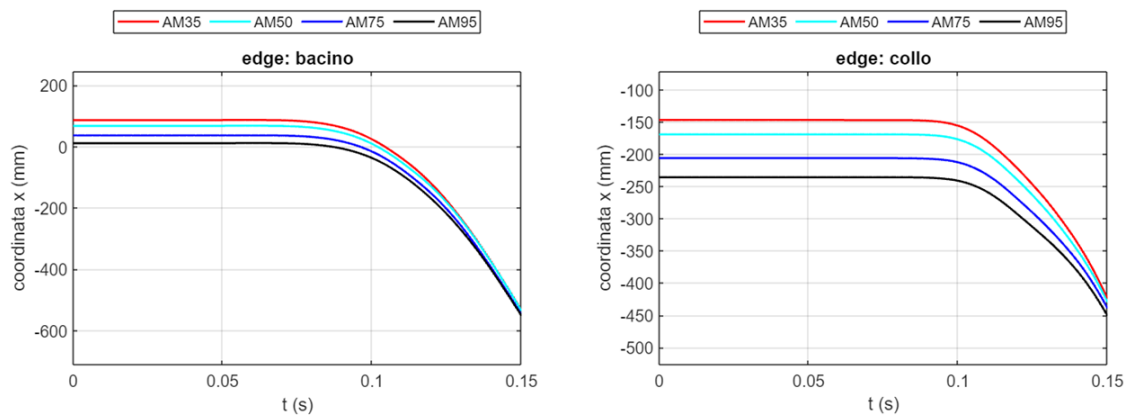


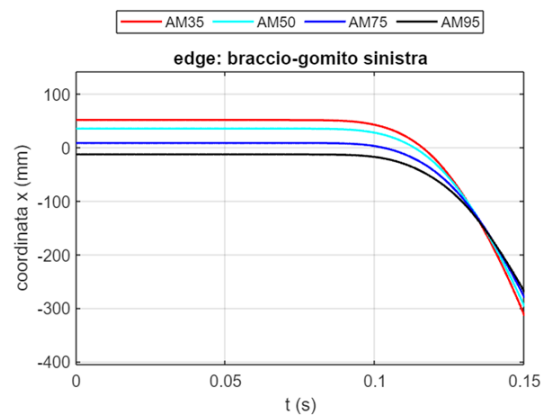
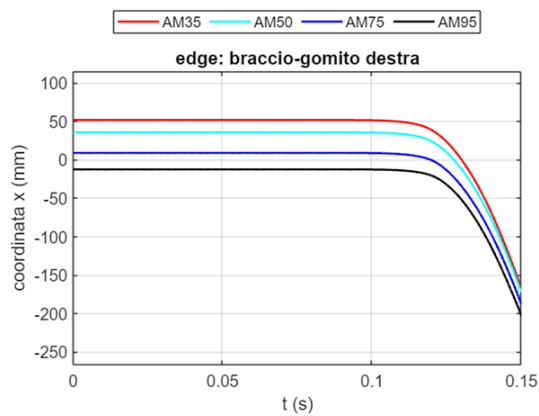
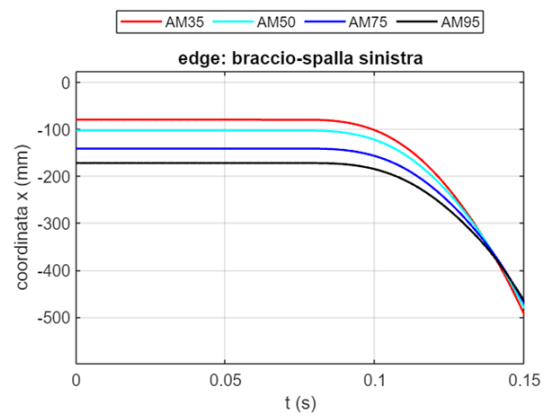
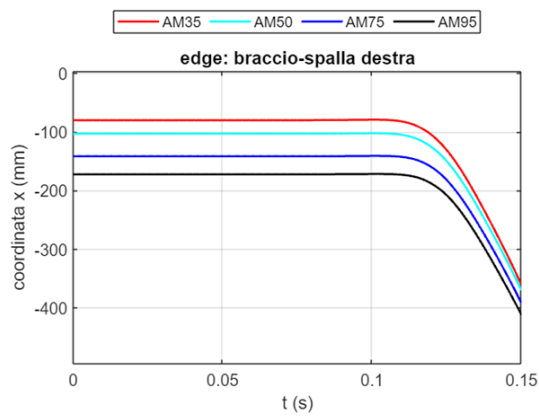
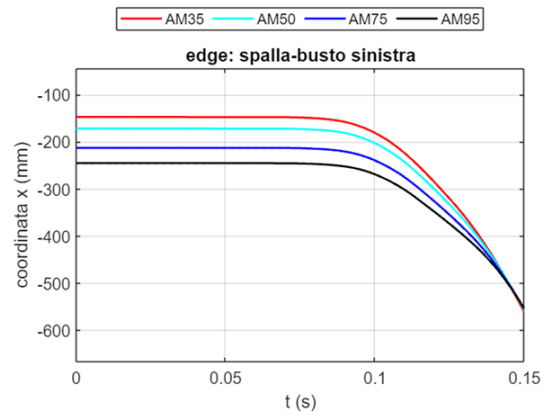
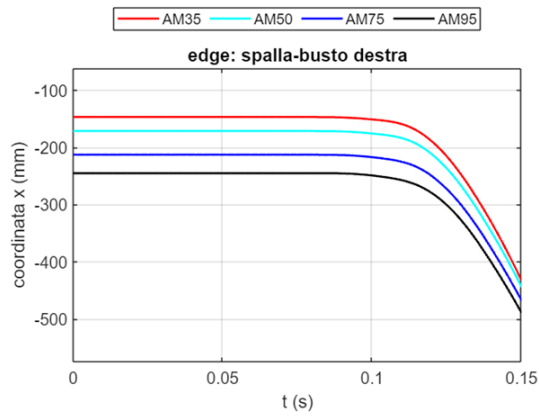


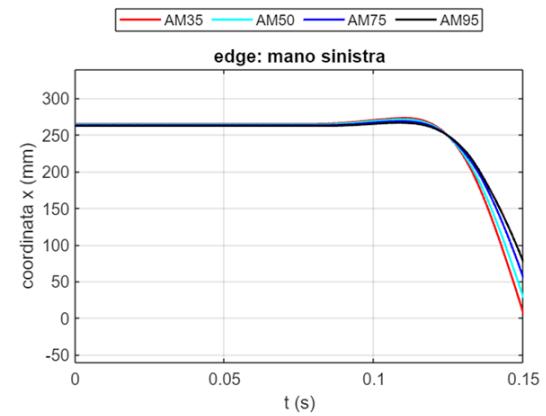
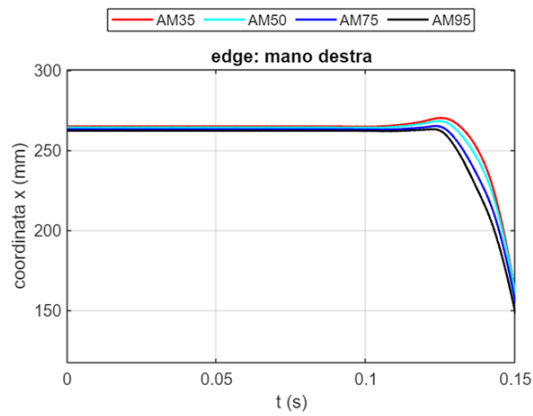
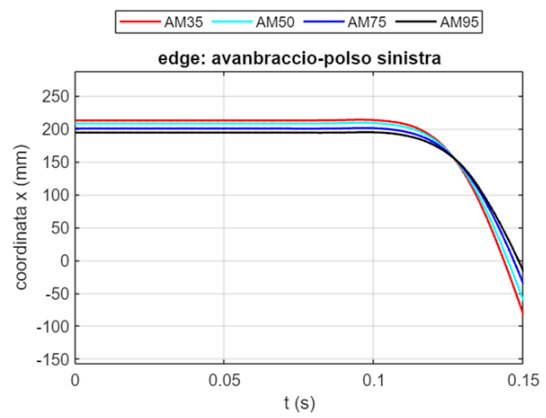
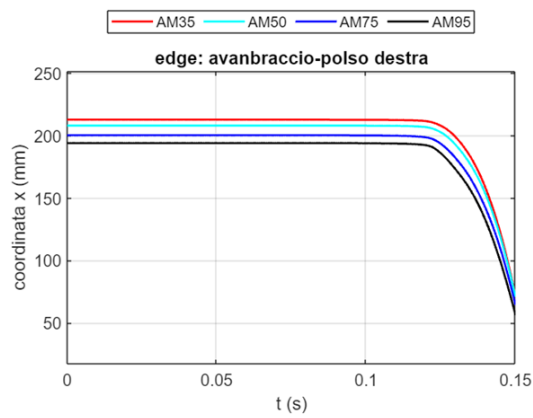
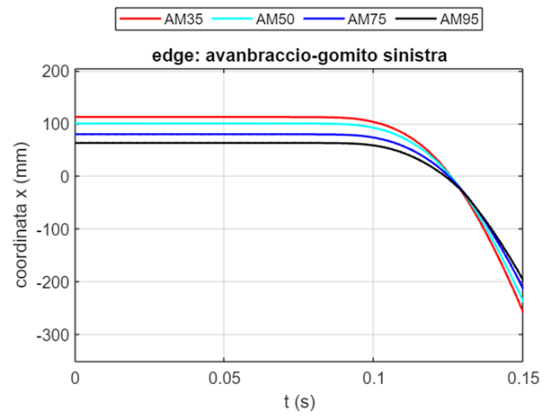
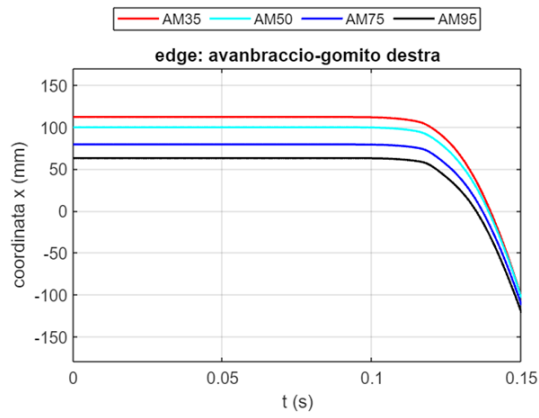


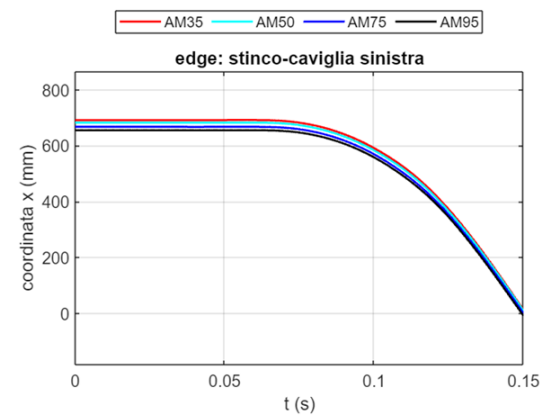
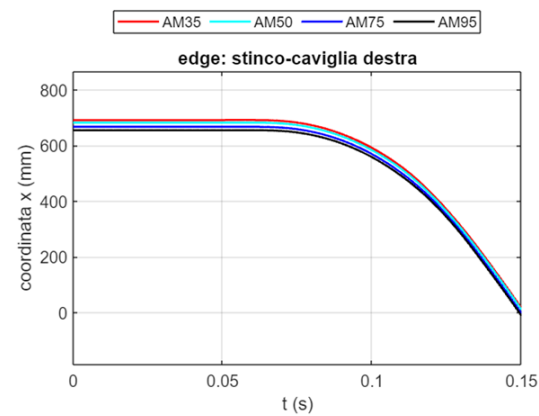
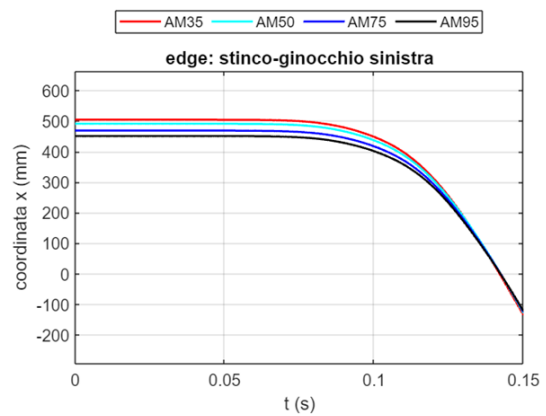
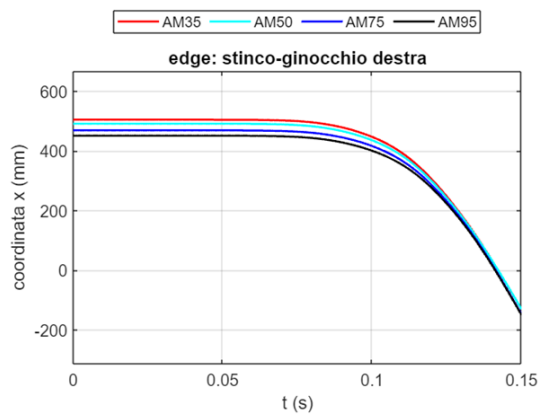
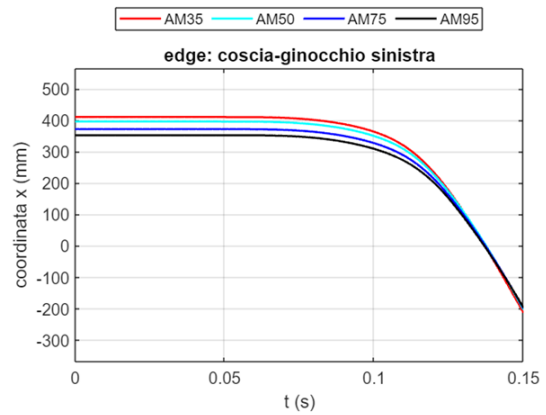
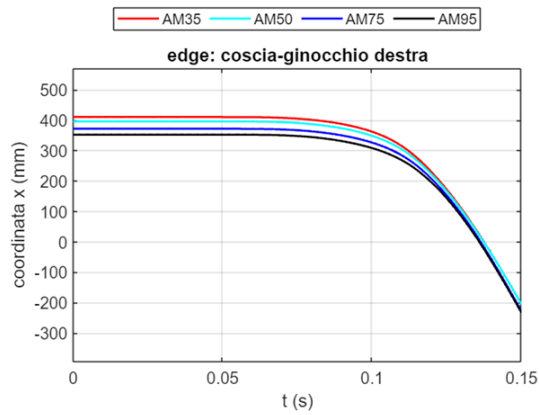
9.3.5 Cinematica dei THUMS morphati

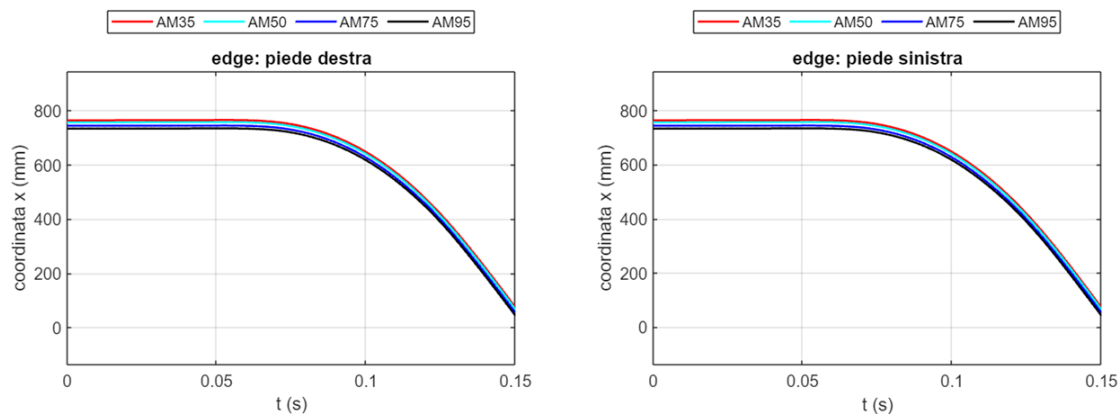
Si riportano a seguire i grafici delle traiettorie nella coordinata x seguite dai punti di controllo durante il *frontal sled test*. Si prendono in esame le configurazioni AM50, AM50m35, AM50m75 e AM50m95.









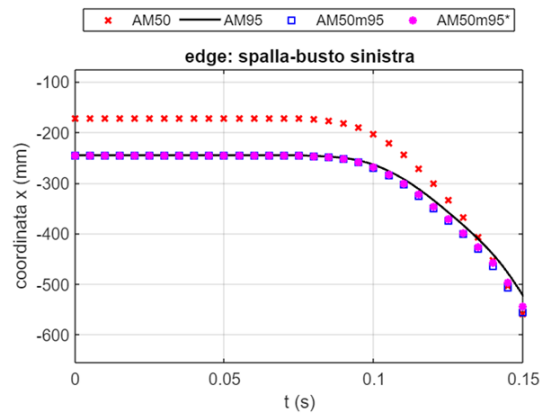
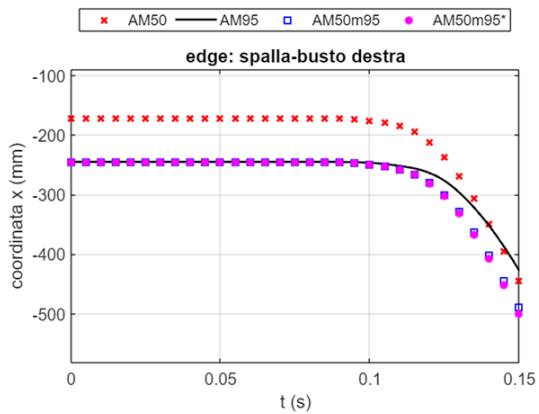
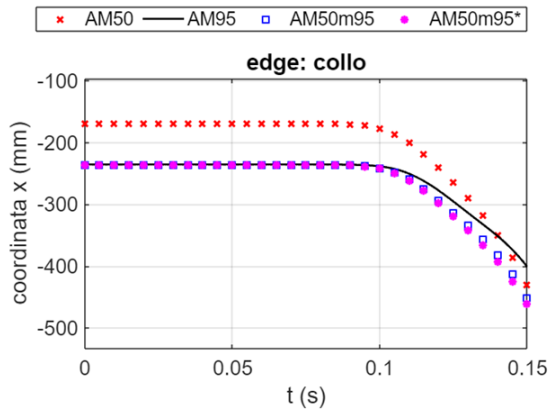
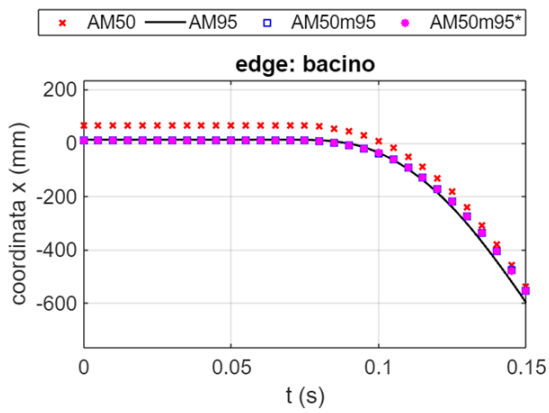


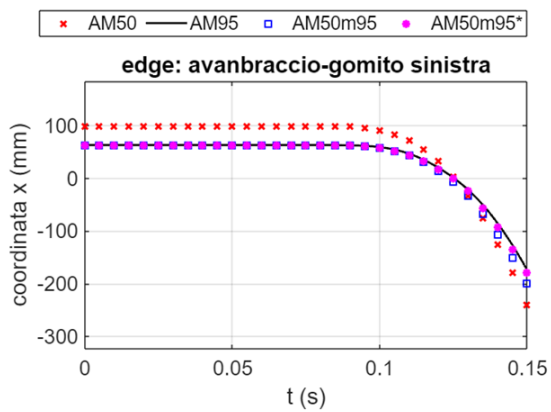
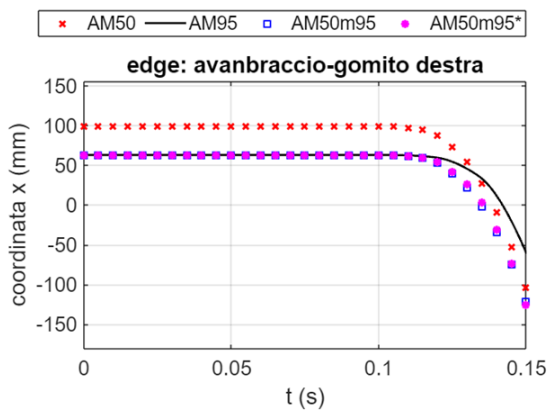
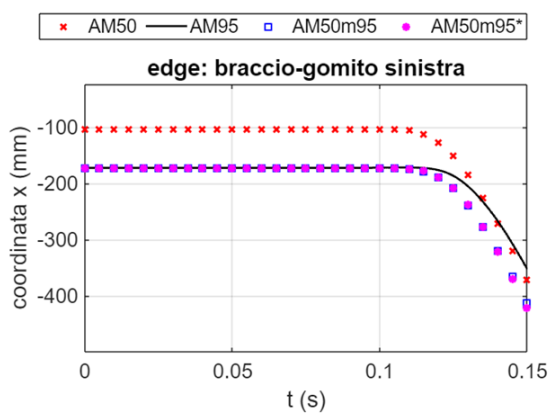
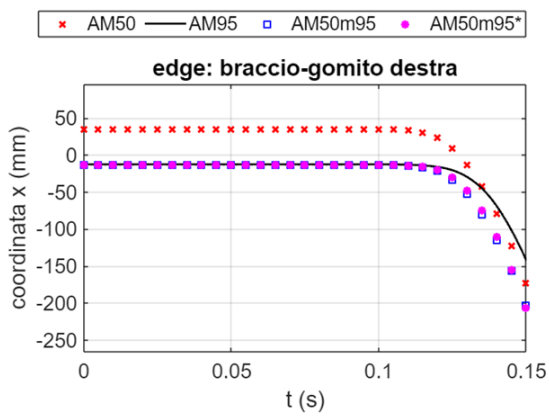
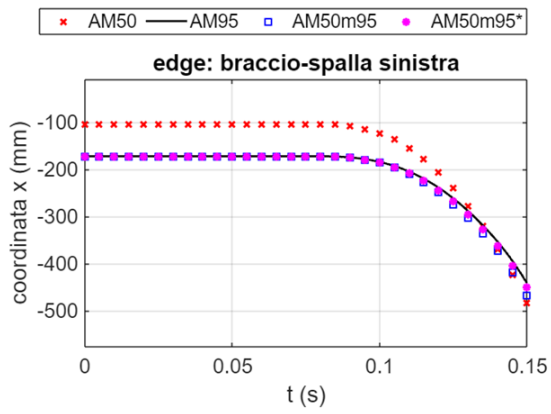
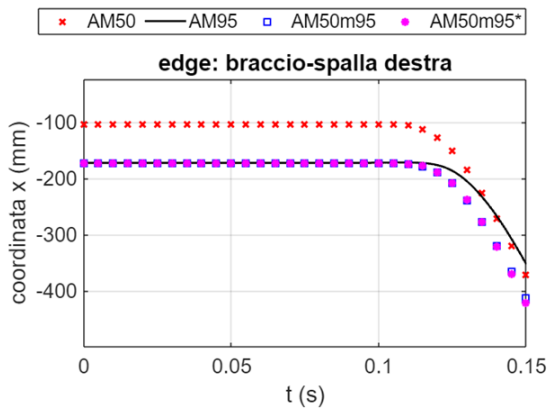
S_{meanx} [mm]			
	AMm35 vs AM50	AMm75 vs AM50	AMm95 vs AM50
bacino	14.5692	27.2591	48.1655
collo	19.9055	32.9198	59.6511
busto-spalla destra	23.0925	39.2095	69.8830
busto-spalla sinistra	19.9685	33.5836	60.7864
braccio-spalla destra	21.3802	36.1754	65.3824
braccio-spalla sinistra	18.7527	31.2618	56.5893
braccio-gomito destra	14.8410	25.4716	46.0103
braccio-gomito sinistra	13.6006	22.2288	40.0445
avanbraccio-gomito destra	11.1429	19.3081	34.9461
avanbraccio-gomito sinistra	10.9458	17.5526	31.2481
avanbraccio-polso destra	4.7046	8.2295	15.0021
avanbraccio-polso	5.3787	8.2168	14.4528
mano destra	1.3058	2.3655	4.4146
mano sinistra	2.4625	3.3628	5.6969
coscia-ginocchio destra	11.4918	21.8664	38.1416
coscia-ginocchio sinistra	12.1709	19.7262	35.7029
stinco-ginocchio destra	10.4424	19.3079	33.8270
stinco-ginocchio sinistra	10.8263	17.9033	32.3466
stinco-caviglia destra	8.3111	14.3203	25.3796
stinco-caviglia sinistra	8.1007	13.9041	25.1061
piede destra	7.6485	12.8227	23.0303
piede sinistra	7.6041	12.7698	23.0461
media	11.7566	19.9894	35.8570

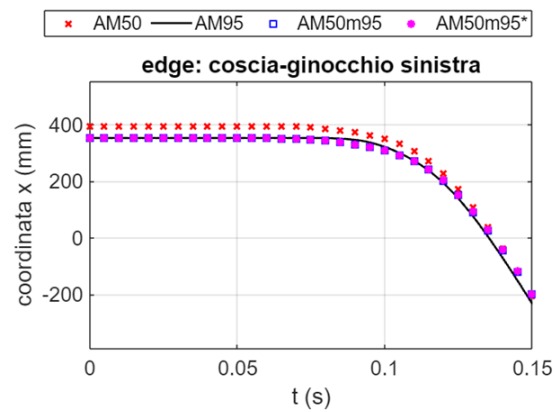
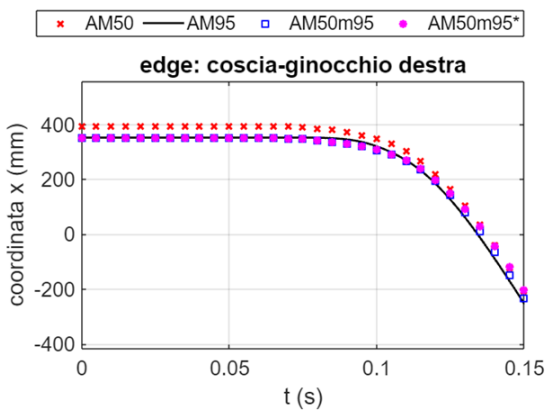
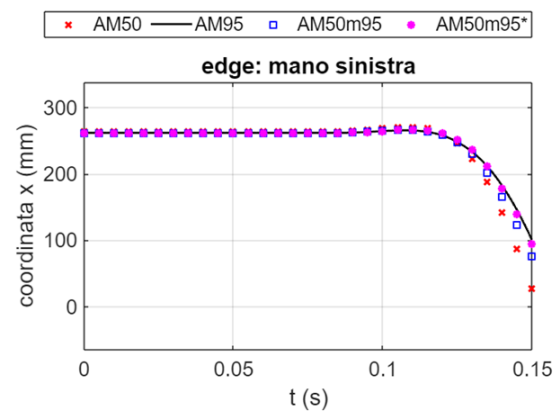
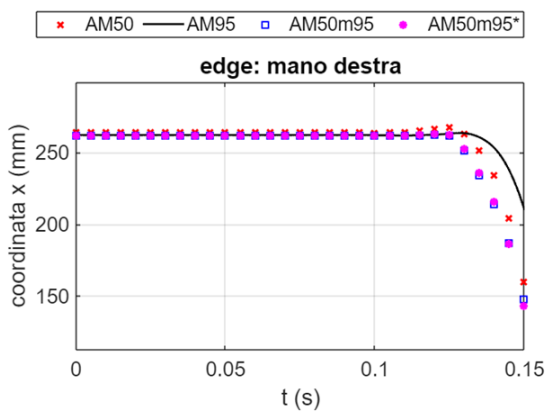
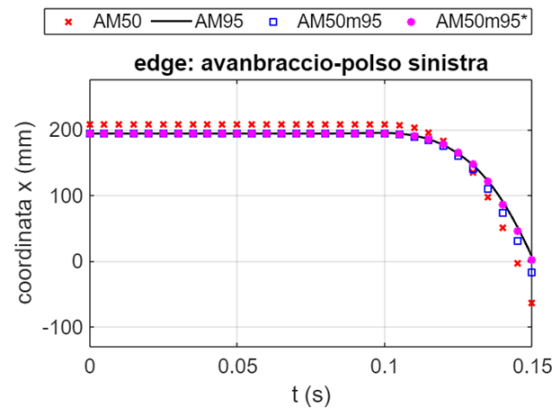
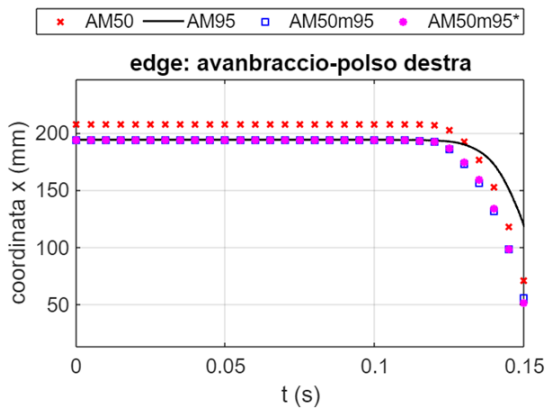
S_{maxx} [mm]	AMm35 vs AM50	AMm75 vs AM50	AMm95 vs AM50
bacino	18.8517	31.4213	56.5584
collo	22.1792	36.9652	66.5374
busto-spalla destra	24.7456	41.6535	73.8678
busto-spalla sinistra	24.4994	40.8319	73.4971
braccio-spalla destra	23.1933	38.6083	69.4777
braccio-spalla sinistra	23.0704	38.4507	69.2112
braccio-gomito destra	16.0872	26.8112	48.2602
braccio-gomito sinistra	17.1024	26.8111	48.2600
avanbraccio-gomito destra	12.2908	20.4840	36.8712
avanbraccio-gomito sinistra	20.3319	23.7588	40.6133
avanbraccio-polso destra	6.3710	11.3194	20.8608
avanbraccio-polso sinistra	21.0442	25.4738	45.4188
mano destra	6.0901	10.6187	19.9462
mano sinistra	21.4565	27.2804	48.8595
coscia-ginocchio destra	14.4851	24.1422	43.4557
coscia-ginocchio sinistra	14.4854	24.1425	43.4563
stinco-ginocchio destra	13.3587	22.2646	40.0762
stinco-ginocchio sinistra	13.3588	22.2646	40.0763
stinco-caviglia destra	9.1343	15.2098	27.3484
stinco-caviglia sinistra	9.1129	15.2134	27.3584
piede destra	7.8998	13.3264	23.7672
piede sinistra	7.8395	12.9646	23.8993
media	15.7722	25.0007	44.8944

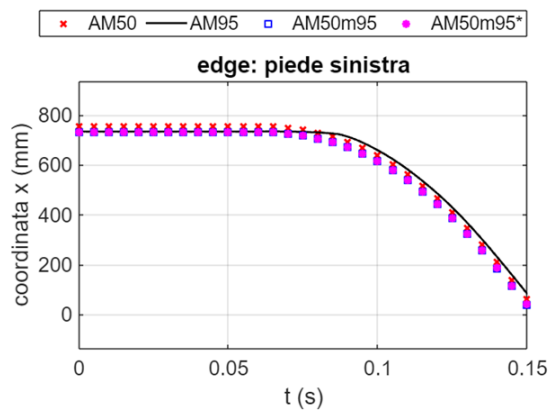
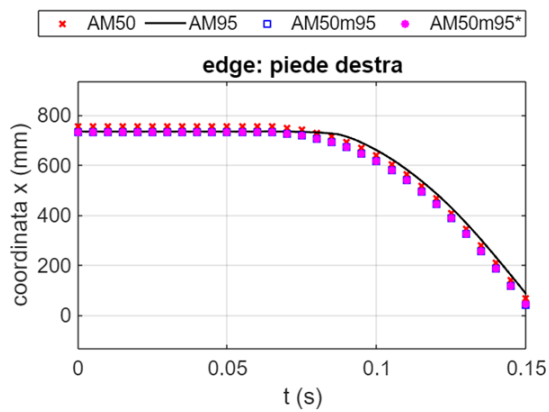
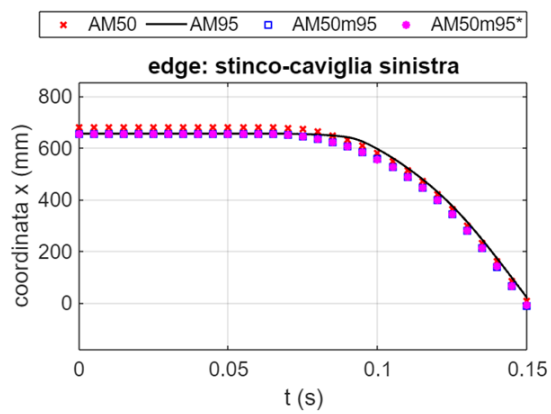
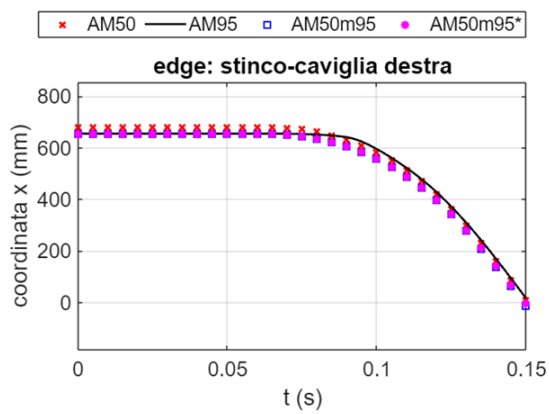
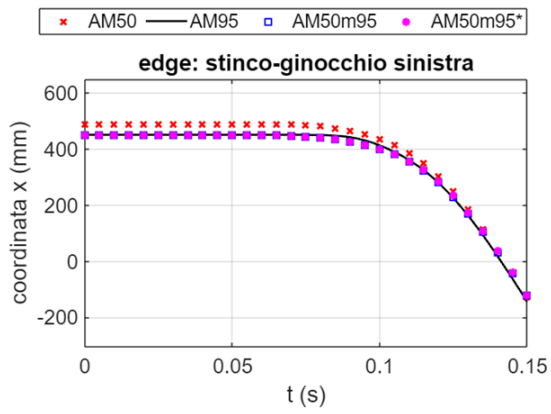
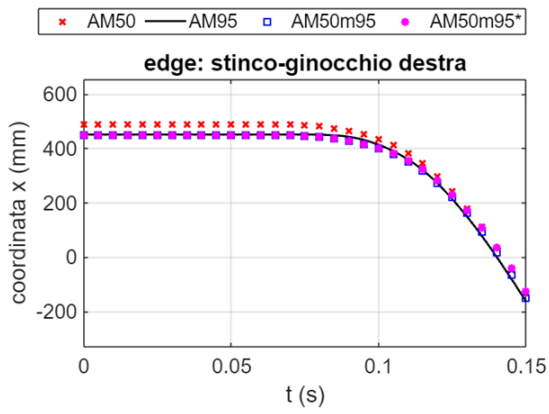
9.3.6 Evoluzione della coordinata x

Si riportano a seguire i grafici delle traiettorie nella coordinata x seguite dai punti di controllo durante il *frontal sled test*. Si prendono in esame le configurazioni AM50, AM50m95, AM50m95b e AM95.







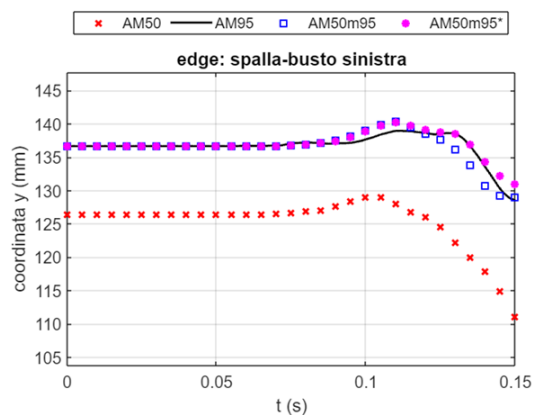
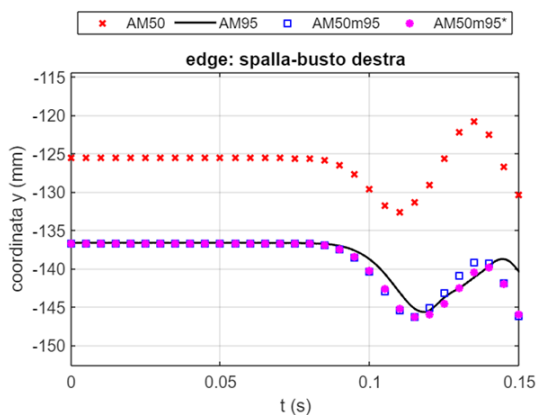
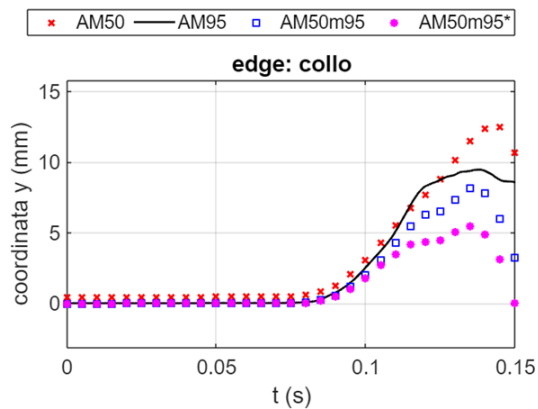
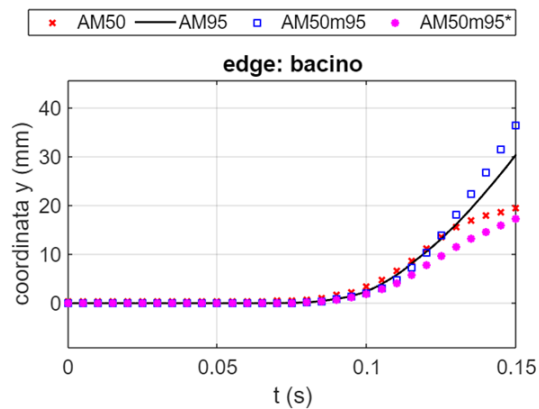


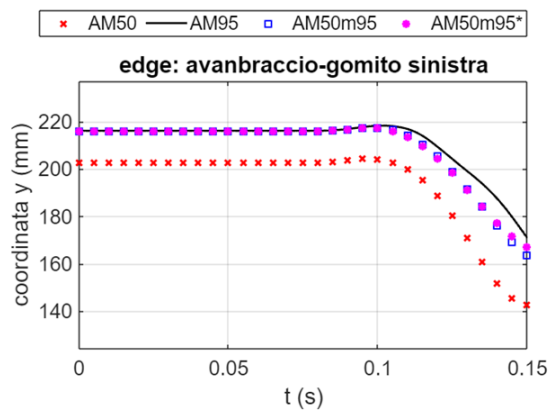
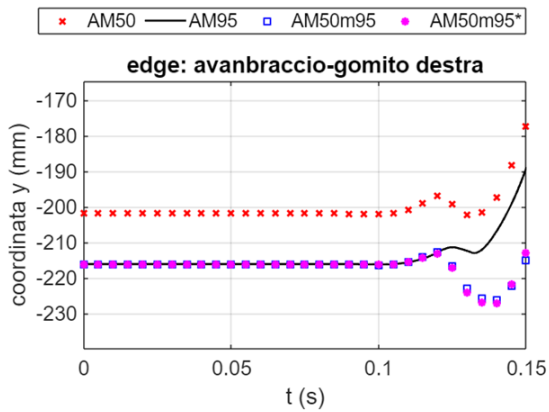
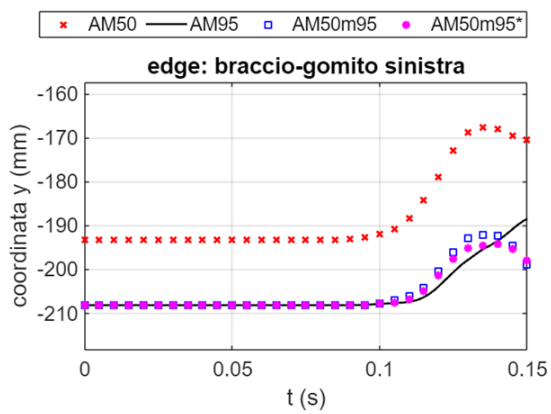
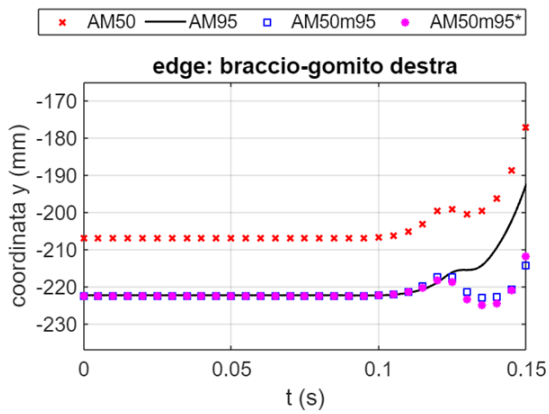
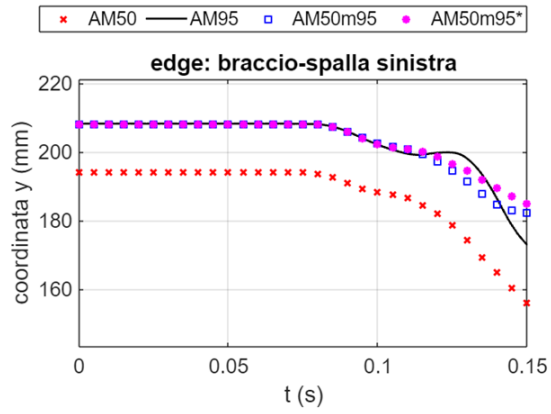
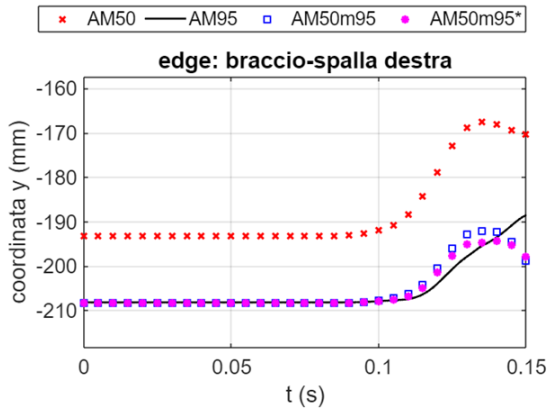
S_{meanx} [mm]			
	AM50 vs AM95	AMm95 vs AM95	AMm95b vs AM95
bacino	55.8978	8.5703	8.6528
collo	54.7108	6.6763	8.7003
busto-spalla destra	61.7294	9.0008	9.8750
busto-spalla sinistra	58.3604	5.6298	4.3405
braccio-spalla destra	58.4107	8.5050	8.7849
braccio-spalla sinistra	56.1065	3.8125	1.6364
braccio-gomito destra	42.0352	6.9175	6.3161
braccio-gomito sinistra	40.9808	3.5321	1.0465
avanbraccio-gomito destra	33.1151	7.0263	6.5018
avanbraccio-gomito sinistra	33.1595	3.3562	0.9831
avanbraccio-polso destra	14.5444	5.8647	5.6235
avanbraccio-polso sinistra	16.4536	2.2589	0.6620
mano destra	4.8533	5.4218	5.3774
mano sinistra	8.1954	2.5092	1.3628
coscia-ginocchio destra	40.5625	7.4457	11.0736
coscia-ginocchio sinistra	38.1619	7.6255	8.2826
stinco-ginocchio destra	34.5765	6.0999	9.2807
stinco-ginocchio sinistra	32.4107	5.4842	6.3898
stinco-caviglia destra	17.3104	14.4397	13.9189
stinco-caviglia sinistra	17.8489	14.7097	14.5714
piede destra	18.9744	19.6219	19.6555
piede sinistra	18.9943	19.6291	19.6121
media	34.4269	7.9153	7.8476

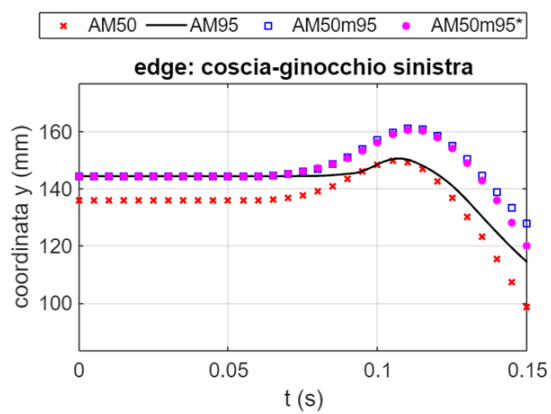
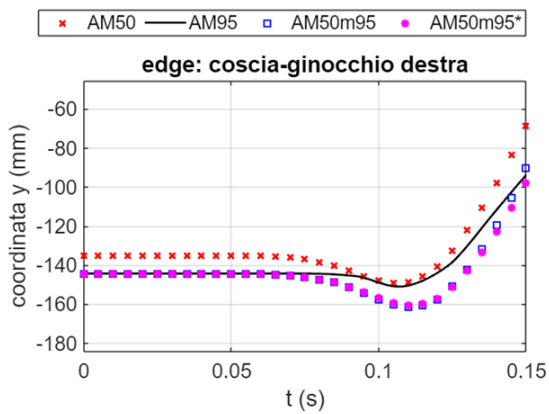
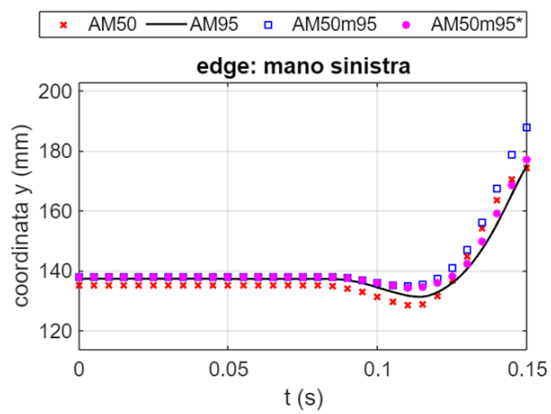
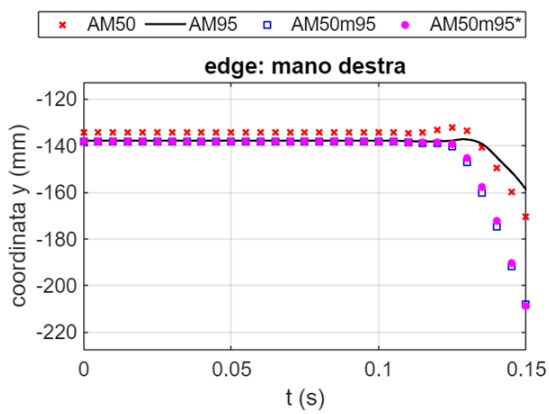
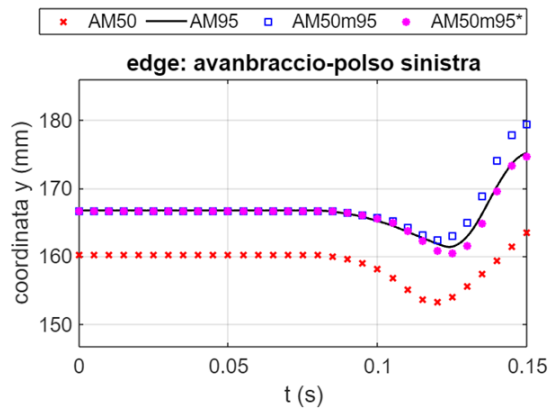
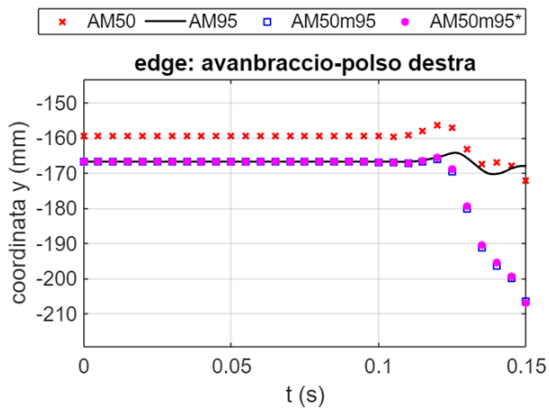
S_{maxx} [mm]	AM50 vs AM95	AMm95 vs AM95	AMm95b vs AM95
bacino	66.0608	48.3838	46.9607
collo	66.5374	48.9306	58.8649
busto-spalla destra	73.4974	60.6840	71.5630
busto-spalla sinistra	73.4961	30.2828	19.4385
braccio-spalla destra	69.6227	59.1353	66.9830
braccio-spalla sinistra	69.2116	23.3649	7.9769
braccio-gomito destra	48.2600	60.3276	62.9140
braccio-gomito sinistra	49.5796	22.4685	4.9736
avanbraccio-gomito destra	42.5880	60.4150	63.6591
avanbraccio-gomito sinistra	63.5258	23.0813	6.2558
avanbraccio-polso destra	46.7413	61.4218	65.6783
avanbraccio-polso sinistra	67.1806	21.7618	4.1335
mano destra	50.1796	62.1707	66.5442
mano sinistra	71.1755	22.3159	5.1407
coscia-ginocchio destra	50.8966	26.4992	53.5750
coscia-ginocchio sinistra	43.4564	37.5773	36.9529
stinco-ginocchio destra	40.0763	20.4830	42.5380
stinco-ginocchio sinistra	40.0763	23.7260	25.4500
stinco-caviglia destra	27.3434	38.9268	38.9879
stinco-caviglia sinistra	27.3531	38.8835	39.1511
piede destra sinistra	23.7706	41.9874	41.9361
piede sinistra	23.7843	42.0085	42.0591
media	51.5643	39.7653	39.6244

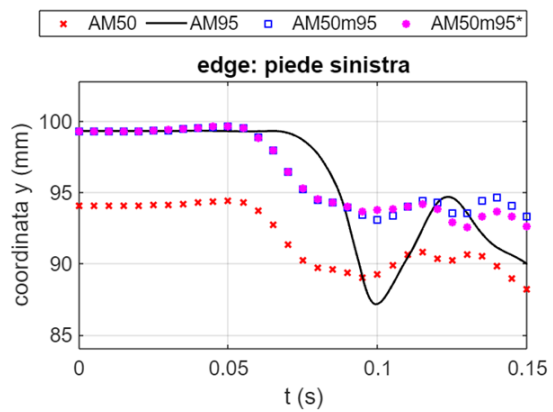
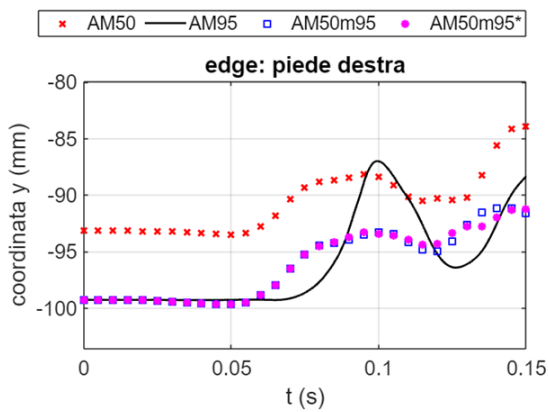
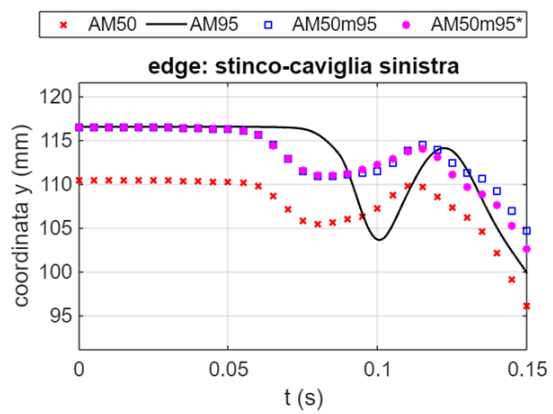
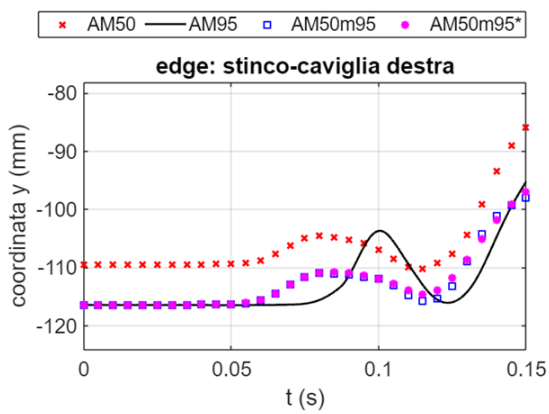
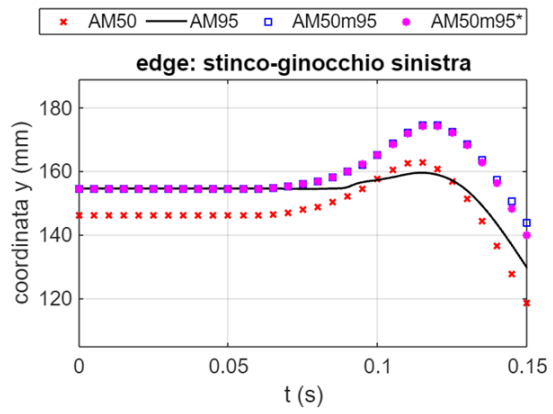
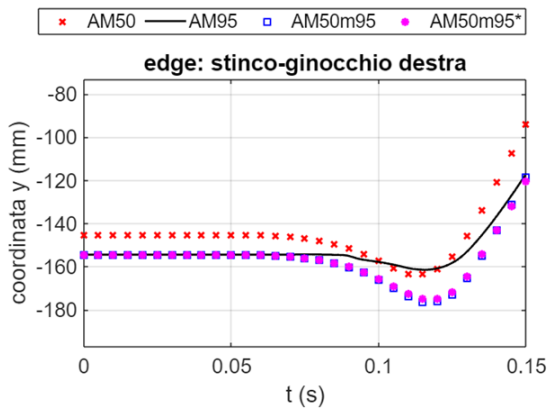
9.3.7 Evoluzione della coordinata y

Si riportano a seguire i grafici delle traiettorie nella coordinata y seguite dai punti di controllo durante il *frontal sled test*. Si prendono in esame le configurazioni AM50, AM50m95, AM50m95b e AM95.







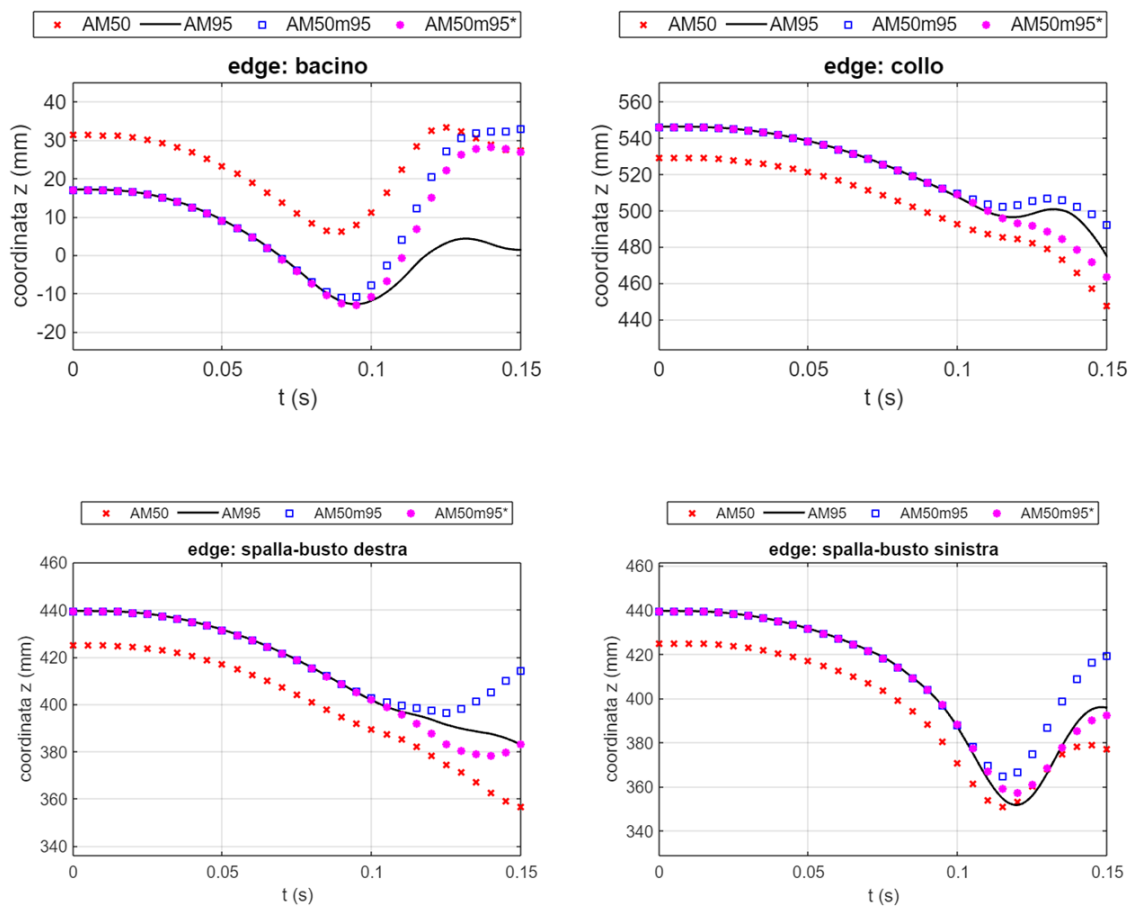


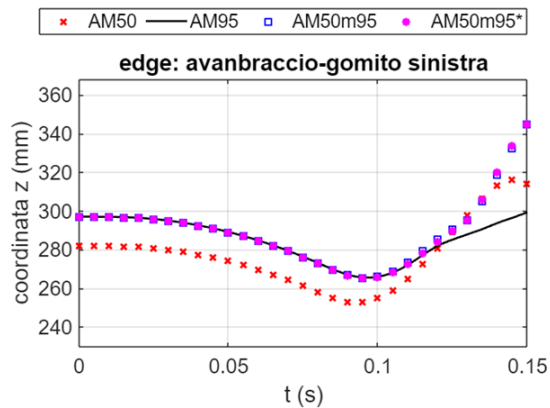
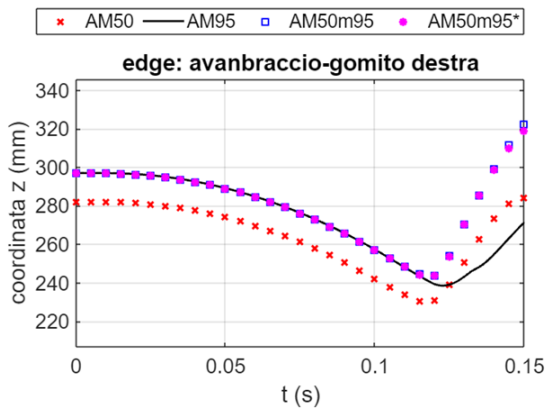
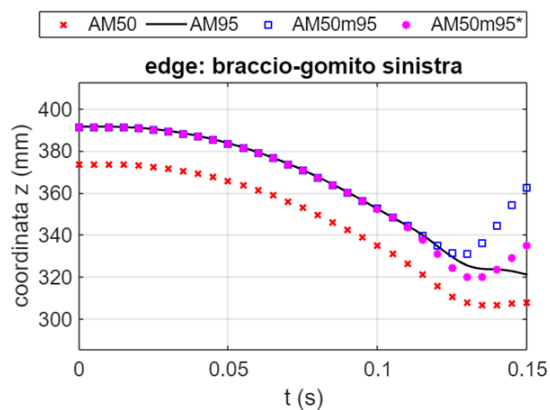
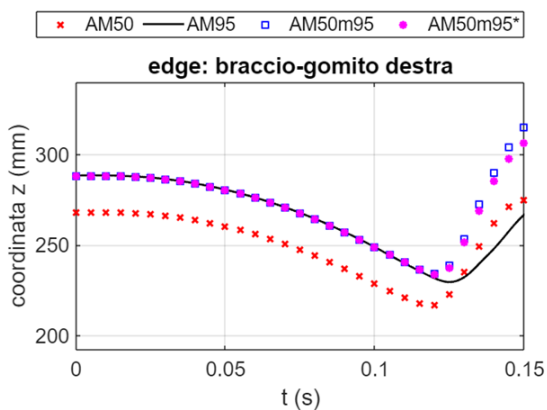
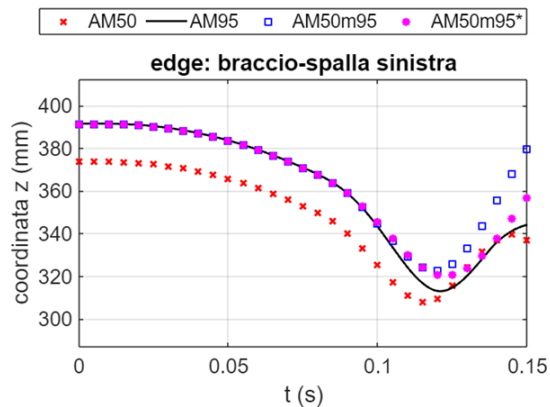
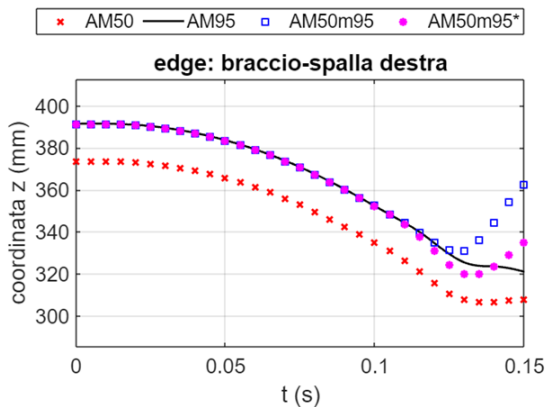
$S_{mean y}$ [mm]	AM50 vs AM95	AMm95 vs AM95	AMm95b vs AM95
bacino	1.2098	0.6931	1.6445
collo	0.7669	0.6119	1.2594
busto-spalla destra	12.2388	0.6506	0.5395
busto-spalla sinistra	11.2203	0.5639	0.4164
braccio-spalla destra	17.7745	1.0316	0.7181
braccio-spalla sinistra	15.4749	1.1245	1.0289
braccio-gomito destra	15.5859	2.0614	2.2160
braccio-gomito sinistra	18.7616	2.4690	2.1487
avanbraccio-gomito destra	13.7529	2.8839	2.9591
avanbraccio-gomito sinistra	17.1351	1.9802	1.9505
avanbraccio-polso destra	6.7444	4.0204	3.8731
avanbraccio-polso sinistra	7.3524	0.6558	0.2719
mano destra	4.0389	4.3301	4.0034
mano sinistra	2.5896	2.7217	1.3685
coscia-ginocchio destra	7.8999	4.1530	4.4664
coscia-ginocchio sinistra	6.2281	5.3191	4.6702
stinco-ginocchio destra	7.7849	4.3204	4.1332
stinco-ginocchio sinistra	6.0050	5.4776	5.2156
stinco-caviglia destra	7.2701	2.2989	2.2474
stinco-caviglia sinistra	5.5833	2.1469	2.0663
piede destra	5.5382	1.6702	1.6266
piede sinistra	4.2320	1.6238	1.5901
media	8.8722	2.4004	2.2915

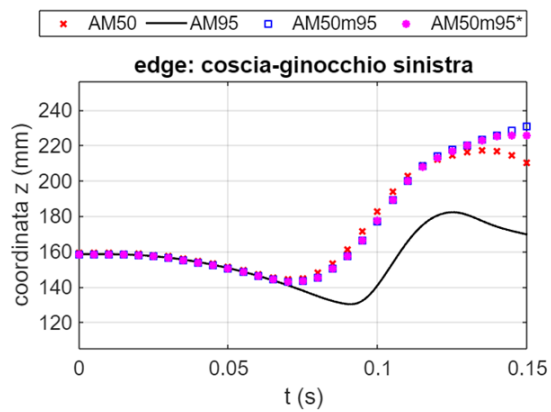
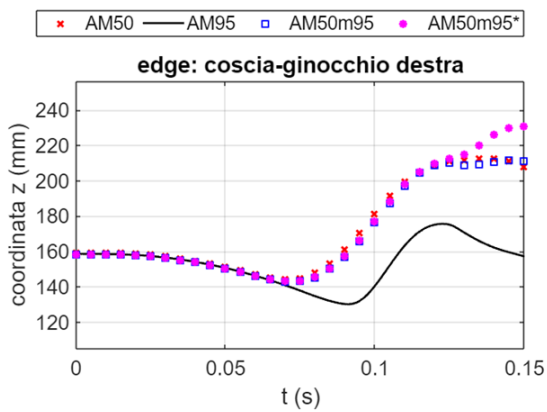
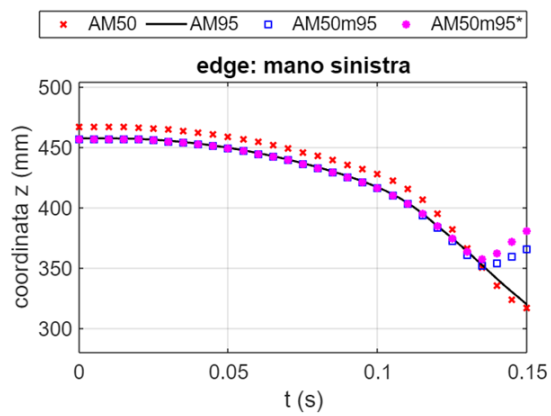
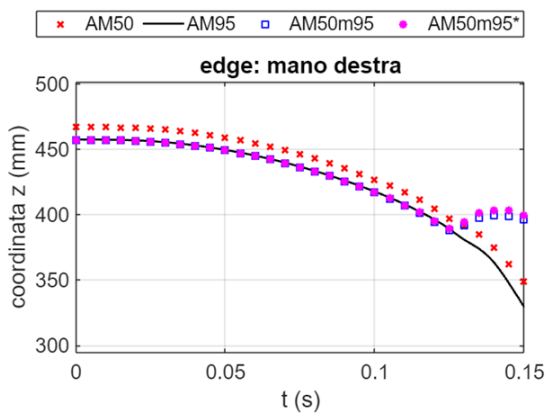
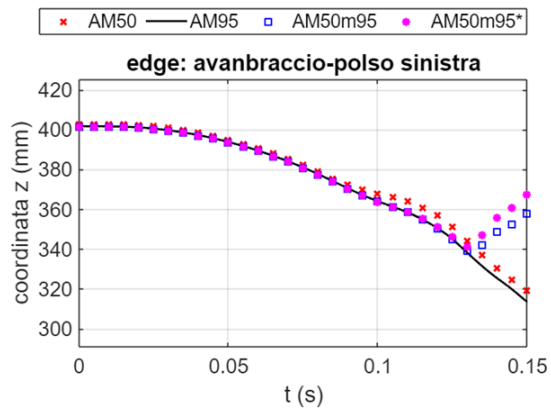
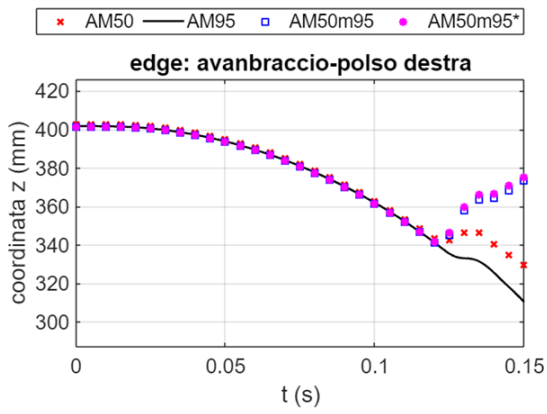
S_{maxy} [mm]	AM50 vs AM95	AMm95 vs AM95	AMm95b vs AM95
bacino	10.9768	6.0695	13.1258
collo	3.7863	5.2759	8.5371
busto-spalla destra	20.6874	5.7021	5.4870
busto-spalla sinistra	17.4846	2.8272	2.6764
braccio-spalla destra	29.1268	10.3298	9.3882
braccio-spalla sinistra	24.0394	9.4966	12.2346
braccio-gomito destra	19.1942	21.8972	19.4021
braccio-gomito sinistra	37.7930	12.9081	10.2113
avanbraccio-gomito destra	16.0728	26.1050	23.8320
avanbraccio-gomito sinistra	35.5869	11.0785	10.1801
avanbraccio-polso destra	9.2029	38.2735	38.6463
avanbraccio-polso sinistra	12.2488	4.2775	1.5989
mano destra	11.3406	48.8235	49.3180
mano sinistra	7.7504	12.7865	3.3571
coscia-ginocchio destra	25.1313	13.7375	13.4556
coscia-ginocchio sinistra	15.3663	14.7881	13.6415
stinco-ginocchio destra	22.9428	15.2884	14.0511
stinco-ginocchio sinistra	10.6259	15.8831	15.5236
stinco-caviglia destra	11.4344	8.2913	8.2496
stinco-caviglia sinistra	10.5471	7.9790	8.6739
piede destra	9.3454	6.2336	6.3978
piede sinistra	8.3929	6.0136	6.6395
media	16.7762	13.8212	13.3922

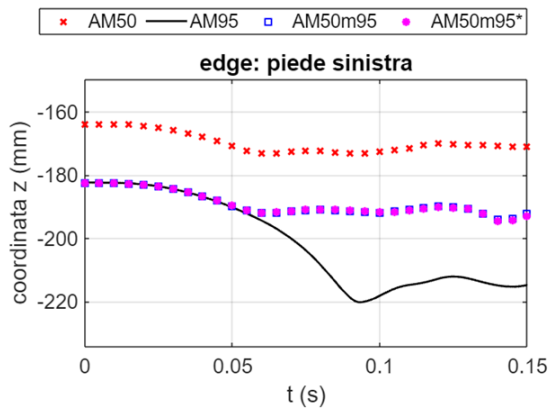
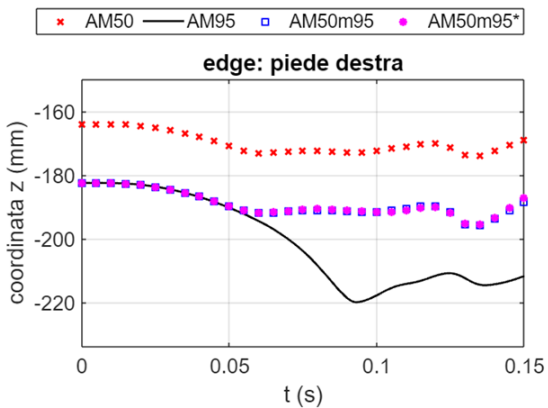
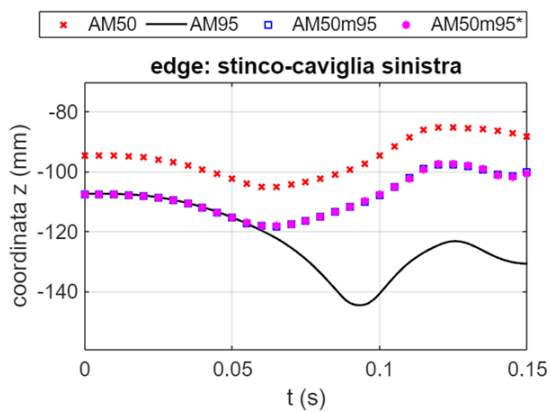
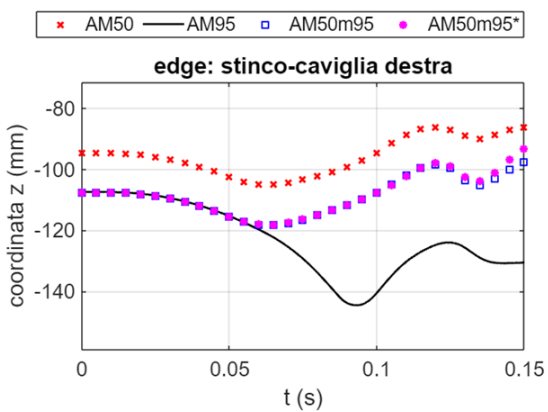
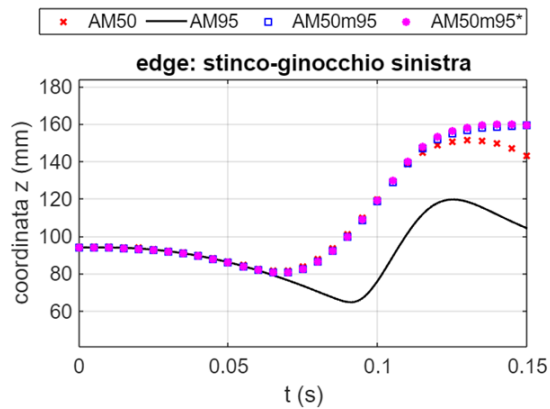
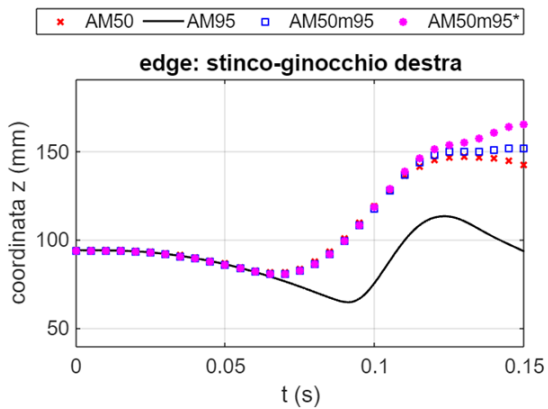
9.3.8 Evoluzione della coordinata z

Si riportano a seguire i grafici delle traiettorie nella coordinata z seguite dai punti di controllo durante il *frontal sled test*. Si prendono in esame le configurazioni AM50, AM50m95, AM50m95b e AM95. .









S_{meanz} [mm]			
	AM50 vs AM95	AMm95 vs AM95	AMm95b vs AM95
bacino	19.4174	7.1945	5.5078
collo	17.7261	2.3648	2.7265
busto-spalla destra	15.4782	3.3205	1.7848
busto-spalla sinistra	12.5332	5.0209	1.3379
braccio-spalla destra	17.5746	3.2200	1.1758
braccio-spalla sinistra	13.9836	4.4813	1.8753
braccio-gomito destra	17.7509	6.1145	5.3059
braccio-gomito sinistra	17.2460	4.2513	3.6387
avanbraccio-gomito destra	14.0568	6.8822	6.7083
avanbraccio-gomito sinistra	13.2474	4.0157	4.0317
avanbraccio-polso destra	3.7250	6.3994	6.7800
avanbraccio-polso sinistra	2.9484	3.1191	4.1252
mano destra	10.2502	5.4892	5.9918
mano sinistra	9.0011	2.6814	3.6805
coscia-ginocchio destra	18.9193	17.3445	19.5996
coscia-ginocchio sinistra	17.7043	17.9286	17.6463
stinco-ginocchio destra	18.8663	19.3729	21.2389
stinco-ginocchio sinistra	17.8323	18.9446	19.3316
stinco-caviglia destra	27.4557	14.3934	14.7717
stinco-caviglia sinistra	27.6126	14.6029	14.6486
piede destra	31.2546	11.9871	12.0785
piede sinistra	31.7230	12.3547	12.3525
media	17.1049	8.7038	8.4699

S_{maxz} [mm]	AM50 vs AM95	AMm95 vs AM95	AMm95b vs AM95
bacino	31.5364	31.6836	26.0788
collo	29.9995	17.8729	16.9849
busto-spalla destra	26.7233	31.4222	9.3722
busto-spalla sinistra	18.6804	23.7861	5.5619
braccio-spalla destra	18.9447	41.5680	14.1817
braccio-spalla sinistra	18.5257	35.5826	12.8650
braccio-gomito destra	20.1360	48.3678	39.7416
braccio-gomito sinistra	20.7053	46.8905	41.5016
avanbraccio-gomito destra	18.8335	51.2577	47.6865
avanbraccio-gomito sinistra	20.1252	45.7934	45.8997
avanbraccio-polso destra	19.4561	63.4872	65.1538
avanbraccio-polso sinistra	6.6073	44.5609	54.3846
mano destra	20.0122	67.4118	70.9769
mano sinistra	12.0902	46.6374	61.8061
coscia-ginocchio destra	51.8716	54.2827	74.0567
coscia-ginocchio sinistra	43.3018	61.6508	56.1976
stinco-ginocchio destra	49.1850	58.8265	72.4012
stinco-ginocchio sinistra	43.7524	55.2518	55.2266
stinco-caviglia destra	47.3058	34.4736	37.4649
stinco-caviglia sinistra	47.2892	34.5173	34.7947
piede destra	47.1099	28.6145	28.8843
piede sinistra	47.0972	28.6519	29.0345
media	29.9677	43.2996	40.9207


```
    path=coordinate[np.isin(coordinate[:,0],vettore)]
    return path

##### PERCORSI FILE #####
#inserire il proprio percorso alla cartella THUMS completo
c0="C:/Users/emanu/Desktop"
c1="THUMS_completo"
c2="thums_model"
nome_pts='RBF_displacements.PTS'
skin='skin.txt'
file50=os.path.join(c0,c1,c2,'THUMS_AM50_V7_Occupant_202301_m95.k')
file95=os.path.join(c0,c1,c2,'THUMS_AM95_V7_Occupant_202301.k')

##### SELEZIONE SOURCE PATH DEL MORPHING #####
# inserire le path relative ai source points del mesh morphing
source_path_list=[]
# specificare le path dei source points da mantenere fissi nel mesh morphing
fixed_path_list=[]

# seleziona i source points tra quelli relativi a:
# 1-bacino
# 2-collo
# 3-spalla_destra
# 4-spalla_sinistra
# 5-braccio_spalla_destra
# 6-braccio_spalla_sinistra
# 7-braccio_gomito_destra
# 8-braccio_gomito_sinistra
# 9-avanbraccio-gomito_destra
# 10-avanbraccio-gomito_sinistra
# 11-avanbraccio_polso_destra
# 12-avanbraccio_polso_sinistra
# 13-mano_destra
# 14-mano_sinistra
# 15-coscia_ginocchio_destra
# 16-coscia_ginocchio_sinistra
# 17-stinco_ginocchio_destra
# 18-stinco_ginocchio_sinistra
# 19-stinco_caviglia_destra
# 20-stinco_caviglia_sinistra
# 21-piede_destro
# 22-piede_sinistra
# 23-osso_sacro
# 24-cassa_toracica
# 25-pelle

for i in source_path_list:
    selezione=i
    if selezione==1:
        kword=["bacino_50",
              "bacino_95"]
        nome="bacino"
    elif selezione==2:
        kword=["collo_50",
              "collo_95"]
        nome="collo"
    elif selezione==3:
        kword=["spalla_destra_50",
              "spalla_destra_95"]
        nome="spalla destra"
    elif selezione==4:
        kword=["spalla_sinistra_50",
              "spalla_sinistra_95"]
        nome="spalla sinistra"
    elif selezione==5:
```



```
        kword=["braccio_spalla_destra_50",
              "braccio_spalla_destra_95"]
        nome="braccio spalla destra"
elif selezione==6:
        kword=["braccio_spalla_sinistra_50",
              "braccio_spalla_sinistra_95"]
        nome="braccio spalla sinistra"
elif selezione==7:
        kword=["braccio_gomito_destra_50",
              "braccio_gomito_destra_95"]
        nome="braccio gomito destra"
elif selezione==8:
        kword=["braccio_gomito_sinistra_50",
              "braccio_gomito_sinistra_95"]
        nome="braccio gomito sinistra"
elif selezione==9:
        kword=["avanbraccio_gomito_destra_50",
              "avanbraccio_gomito_destra_95"]
        nome="avanbraccio gomito destra"
elif selezione==10:
        kword=["avanbraccio_gomito_sinistra_50",
              "avanbraccio_gomito_sinistra_95"]
        nome="avanbraccio gomito sinistra"
elif selezione==11:
        kword=["avanbraccio_polso_destra_50",
              "avanbraccio_polso_destra_95"]
        nome="avanbraccio polso destra"
elif selezione==12:
        kword=["avanbraccio_polso_sinistra_50",
              "avanbraccio_polso_sinistra_95"]
        nome="avanbraccio polso sinistra"
elif selezione==13:
        kword=["mano_destra_50",
              "mano_destra_95"]
        nome="mano destra"
elif selezione==14:
        kword=["mano_sinistra_50",
              "mano_sinistra_95"]
        nome="mano sinistra"
elif selezione==15:
        kword=["coscia_ginocchio_destra_50",
              "coscia_ginocchio_destra_95"]
        nome="coscia ginocchio destra"
elif selezione==16:
        kword=["coscia_ginocchio_sinistra_50",
              "coscia_ginocchio_sinistra_95"]
        nome="coscia ginocchio sinistra"
elif selezione==17:
        kword=["stinco_ginocchio_destra_50",
              "stinco_ginocchio_destra_95"]
        nome="stinco ginocchio destra"
elif selezione==18:
        kword=["stinco_ginocchio_sinistra_50",
              "stinco_ginocchio_sinistra_95"]
        nome="stinco ginocchio sinistra"
elif selezione==19:
        kword=["stinco_caviglia_destra_50",
              "stinco_caviglia_destra_95"]
        nome="stinco caviglia destra"
elif selezione==20:
        kword=["stinco_caviglia_sinistra_50",
              "stinco_caviglia_sinistra_95"]
        nome="stinco caviglia sinistra"
elif selezione==21:
        kword=["piede_destra_50",
```

```

        "piede_destra_95"]
    nome="piede destra"
elif selezione==22:
    kword=["piede_sinistra_50",
           "piede_sinistra_95"]
    nome="piede sinistra"
elif selezione==25:
    kword=["pelle_50",
           "pelle_95"]
    nome="pelle"
elif selezione==23:
    kword=["osso_sacro_50",
           "osso_sacro_95"]
    nome="osso_sacro"
elif selezione==24:
    kword=["cassa_toracica_50",
           "cassa_toracica_95"]
    nome="cassa_toracica"

#inizializzazione vettori contenenti le informazioni da tradurre nel file pts
x_t=[]
y_t=[]
z_t=[]
dx_t=[]
dy_t=[]
dz_t=[]
num_t=[]

##### IDENTIFICAZIONE DEI SOURCE POINTS #####
# si estrae la numerazione relativa alla kword selezionata per AM50
vettore_numerazione50=numerazione_cercata(file50,kword[0])
# si estrapola la path desiderata nell'AM50
path50=individua_path(file50,vettore_numerazione50)
# si estrae la numerazione relativa alla kword selezionata per AM95
vettore_numerazione95=numerazione_cercata(file95,kword[1])
# si estrapola la path desiderata nell'AM95
path95=individua_path(file95,vettore_numerazione95)

##### CALCOLO DEGLI RBF DISPLACEMENT #####
x50 = path50[:,1]
y50 = path50[:,2]
z50 = path50[:,3]
x95 = path95[:,1]
y95 = path95[:,2]
z95 = path95[:,3]

# calcolo dei baricentri degli edge di AM50 e AM95
c50=np.array([np.mean(x50),np.mean(y50),np.mean(z50)])
c95=np.array([np.mean(x95),np.mean(y95),np.mean(z95)])
# traslazione degli edge in posizione di baricentro zero
x50l=x50-c50[0]
y50l=y50-c50[1]
z50l=z50-c50[2]
x95l=x95-c95[0]
y95l=y95-c95[1]
z95l=z95-c95[2]
# calcolo dei fattori di scala
sfx=(max(x95l)-min(x95l))/(max(x50l)-min(x50l))
sfy=(max(y95l)-min(y95l))/(max(y50l)-min(y50l))
sfz=(max(z95l)-min(z95l))/(max(z50l)-min(z50l))
# calcolo degli RBF displacements dovuti allo scaling
rbf_sx=(sfx-1)*x50l
rbf_sy=(sfy-1)*y50l
rbf_sz=(sfz-1)*z50l
# calcolo degli RBF displacements dovuti alla traslazione

```

```

rbf_tx=c95[0]-c50[0]
rbf_ty=c95[1]-c50[1]
rbf_tz=c95[2]-c50[2]
#calcolo degli RBF displacements compressivi
rbf_x=rbf_sx+rbf_tx
rbf_y=rbf_sy+rbf_ty
rbf_z=rbf_sz+rbf_tz

##### CONFRONTO PRIMA DEL MORPHING #####
x=np.concatenate((x50,x95))
y=np.concatenate((y50,y95))
z=np.concatenate((z50,z95))

fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
scatter_set50=ax.scatter(x50,y50,z50,s=10,label='AM50 edge',color='red')
scatter_set95=ax.scatter(x95,y95,z95,s=8,label='AM95 edge',color='black')
ax.view_init(elev=235,azim=-120)
ax.set_xlim(max(x),min(x))
ax.set_ylim(max(y),min(y))
ax.set_zlim(max(z),min(z))
ax.legend()
plt.title(nome)
plt.show()

##### CONFRONTO DOPO IL MORPHING #####
x50m=x50+rbf_x
y50m=y50+rbf_y
z50m=z50+rbf_z

x=np.concatenate((x50m,x95,x50))
y=np.concatenate((y50m,y95,y50))
z=np.concatenate((z50m,z95,z50))

fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')

scatter_set50=ax.scatter(x50,y50,z50,s=5,label='AM50 edge',color='red')
scatter_set50=ax.scatter(x95,y95,z95,s=5,label='AM95 edge',color='black')
scatter_set95=ax.scatter(x50m,y50m,z50m,s=10,label='AM50m edge',color='blue')

ax.view_init(elev=235,azim=-120)
ax.set_xlim(max(x),min(x))
ax.set_ylim(max(y),min(y))
ax.set_zlim(max(z),min(z))
plt.title(nome)
ax.legend()
plt.show()

##### ASSEGNAZIONE VALORI TOTAL #####
num_50=path50[:,0]
for j in range(len(x50)):
    x_t.append(x50[j])
    y_t.append(y50[j])
    z_t.append(z50[j])
    num_t.append(int(num_50[j]))
    if i in fixed_path_list:
        dx_t.append(0.0)
        dy_t.append(0.0)
        dz_t.append(0.0)
    else:
        dx_t.append(rbf_x[j])
        dy_t.append(rbf_y[j])
        dz_t.append(rbf_z[j])

```

```
##### ORDINAMENTO VALORI TOTAL #####
nid_da_ordinare = np.array(num_t)
x_da_ordinare = np.array(x_t)
y_da_ordinare = np.array(y_t)
z_da_ordinare = np.array(z_t)
dx_da_ordinare = np.array(dx_t)
dy_da_ordinare = np.array(dy_t)
dz_da_ordinare = np.array(dz_t)

# Ottenere gli indici di ordinamento del vettore_da_ordinare
indici_ordinamento = sorted(range(len(nid_da_ordinare)),
                             key=lambda k: nid_da_ordinare[k])

# Ordinare il vettore_da_ordinare
num_t = [nid_da_ordinare[i] for i in indici_ordinamento]
# Riordinare anche l'altro_vettore in base agli stessi indici di ordinamento
x_t = [x_da_ordinare[i] for i in indici_ordinamento]
y_t = [y_da_ordinare[i] for i in indici_ordinamento]
z_t = [z_da_ordinare[i] for i in indici_ordinamento]
dx_t = [dx_da_ordinare[i] for i in indici_ordinamento]
dy_t = [dy_da_ordinare[i] for i in indici_ordinamento]
dz_t = [dz_da_ordinare[i] for i in indici_ordinamento]

##### CREAZIONE FILE.PTS #####
#in questa sezione se attivare=1 viene creato il file.PTS contenente degli RBF
#Displacement totali. Si può visualizzare tale file nella cartella PTS FILE

attivare=1
if attivare==1:
    # Combina i vettori in una matrice
    dati = np.column_stack((x_t,y_t,z_t,
                           dx_t,dy_t,dz_t,
                           num_t,['0']*len(x_t),['s']*len(x_t),['p']*len(x_t)))

    # Scrivi i dati nel file .pts
    percorso_file=os.path.join(c0,c1,c2,nome_pts)
    with open(percorso_file,'w') as file:
        file.write(str(len(x_t))+'\n')
        for riga in dati:
            # Converti i valori in stringhe e scrivi la riga nel file
            file.write(' '.join(map(str,riga))+'\n')
    print(f'Il file {nome_pts} è stato generato correttamente')

##### TEMPO DI ESECUZIONE #####
end_time=time.time()
elapsed_time=end_time-start_time
print(f"Tempo di esecuzione: {elapsed_time} secondi")
```

9.4.2 Mesh_Morphing.py

```
import numpy as np
import ctypes
import os
import time
start_time=time.time()

##### PERCORSI FILE #####
#inserire il proprio percorso alla cartella THUMS completo
c0="C:/Users/emanu/Desktop"
c1="THUMS_completo"
c2="thumbs_model"

# inserire il nome del file PTS contenente il mesh morphing
nome_file='RBF_displacements.PTS'
# percorso file al suddetto file
file_pts=os.path.join(c0,c1,c2,nome_file)
```

```

# inserire il nome del k_file_originario
nome_file='THUMS_AM50_V7_Occupant_202301_m95.k'
# percorso file al suddetto file
k_file_originale=os.path.join(c0,c1,c2,nome_file)

# inserire il nome del k_file dopo il morphing
nome_file='THUMS_AM50_V7_Occupant_202301_m95ossa.k'
# percorso file al suddetto file
k_file_morphing=os.path.join(c0,c1,c2,nome_file)

##### RBF MORPH LICENZA E SET UP #####

licenza_percorso=os.path.join(c0,c1,"RBF_FGP.dll")
librbf = ctypes.cdll.LoadLibrary(licenza_percorso)
print('RBF LIB loaded: version ' + str(librbf.RBFMORPH_Version()) + '\n')

debug=False
q=32
tolerance=ctypes.c_double(0.000001)
max_iter=2500
precomp=False
omp_cores=0
use_gpu=False
single_precision=False
rbf_exp=1
CPar=ctypes.c_double(0.0)
apar=ctypes.c_double()
err=ctypes.c_double()
iter=ctypes.c_int()
lcm_distance=ctypes.c_float(0)
lcm_radius =ctypes.c_float(0)
n_direct=1
log="log.txt".encode('utf-8')

##### LETTURA FILE PTS #####
class myvec(ctypes.Structure):
    _fields_ = [('val',ctypes.c_double)]
# inizializzazione degli array che andranno a contenere le coordinate nodali
# dei source points e gli rbf displacements prescritti nel file PTS
X=[]
Y=[]
Z=[]
DX=[]
DY=[]
DZ=[]

# inizializzazione del numero di source points
nsource = 0
scale = 1

# visualizzazione del numero di source points
print(file_pts)
with open (file_pts,'r') as file:
    lines = file.readlines()
print ('lines= ' + str(lines[0]))
nsource = nsource + int(lines[0])
npoints = int(lines[0])

# selezionare il percentile antropomorfo da raggiungere con il morphing
percentile=95
# sulla base della scelta effettuata gli RBF displacements vengono moltiplicati
# per un numero "perc" compreso tra 0 (AM50) e 1 (AM95)
max_percentile=95
min_percentile=50

```

```

perc=(percentile-min_percentile)/(max_percentile-min_percentile)

# assegnazione delle coordinate nodali dei source points
# e degli RBF displacements
for i in range (1, npoints+1):
    a, b, c, d, e, f, g, h, k, l = lines[i].split()

    X.append(np.double(a))
    Y.append(np.double(b))
    Z.append(np.double(c))
    DX.append(np.double(d)*perc)
    DY.append(np.double(e)*perc)
    DZ.append(np.double(f)*perc)

# trasformazione da liste a vettori
X=np.array(X)
Y=np.array(Y)
Z=np.array(Z)
DX=np.array(DX)
DY=np.array(DY)
DZ=np.array(DZ)

# assegnazione della lunghezza dei vettori
myvec_arr = myvec * nsource
# inizializzazione dei vettori
rx = myvec_arr()
ry = myvec_arr()
rz = myvec_arr()
fx = myvec_arr()
fy = myvec_arr()
fz = myvec_arr()
lx = myvec_arr()
ly = myvec_arr()
lz = myvec_arr()
# assegnazione dei valori sulla base del file PTS interpretato
for i in range (nsource):
    rx[i].val = X[i]
    ry[i].val = Y[i]
    rz[i].val = Z[i]
    fx[i].val = DX[i]
    fy[i].val = DY[i]
    fz[i].val = DZ[i]
    lx[i].val = 0.0
    ly[i].val = 0.0
    lz[i].val = 0.0

##### VERIFICA LICENZA #####
checkout = librbf.LicenseCheck("rbf_ans_struct".encode("utf-8"),
                               "1.0".encode("utf-8"))
print ("checkout license = " + str(checkout))

##### ASSEGNAZIONE VALORI SET UP #####
res=librbf.Solve_DP(
    nsource,
    q,
    tolerance,
    max_iter,
    precomp,
    omp_cores,
    use_gpu,
    single_precision,
    rbf_exp,
    CPar,
    ctypes.byref(rx), ctypes.byref(ry), ctypes.byref(rz),
    ctypes.byref(fx), ctypes.byref(fy), ctypes.byref(fz),

```

```

    ctypes.byref(lx), ctypes.byref(ly), ctypes.byref(lz),
    ctypes.byref(apar),
    ctypes.byref(err), ctypes.byref(iter),
    log,
    lcm_distance,
    lcm_radius,
    n_direct
)
#librbf.RBFMORPH_End()
debug = False
if debug:
    for i in range (nsource):
        print (i, end = " ")
        print (lx[i].val, end = " ")
        print (ly[i].val, end = " ")
        print (lz[i].val, end = " ")
        print ()
    print (apar)
print ("res_solve = " + str(res))

##### LETTURA DEI TARGET POINTS #####
# inizializzazione delle liste contenenti le coordinate dei target point e la
# relativa numerazione
x = []
y = []
z = []
nid=[]

keyword1='*NODE'
stop_word='*PART_AVERAGED'
sezione_corretta=False

# il blocco seguente estrae dal k-file le informazioni contenute nella sezione
# tra "*NODE" e "*PART", ovvero le coordinate dei target points
with open(k_file_originale,'r') as file:
    for riga in file:
        if riga.startswith(keyword1):
            sezione_corretta=True
        if sezione_corretta:
            try:
                numeri =list(map(float,riga.split()))
                nid.append(numeri[0])
                x.append(numeri[1])
                y.append(numeri[2])
                z.append(numeri[3])
            except ValueError:
                a=riga.startswith(stop_word)
                if a:
                    break

ntarget = len(x)
target_arr = myvec * ntarget
rx_t = target_arr()
ry_t = target_arr()
rz_t = target_arr()
fx_t = target_arr()
fy_t = target_arr()
fz_t = target_arr()

for i in range (ntarget):
    rx_t[i].val = x[i]
    ry_t[i].val = y[i]
    rz_t[i].val = z[i]
    fx_t[i].val = 0.0
    fy_t[i].val = 0.0

```

```

    fz_t[i].val = 0.0

##### ESECUZIONE DEL MESH MORPHING #####
res = librbf.Morph_DP(
    ntarget,
    ctypes.byref(rx_t), ctypes.byref(ry_t), ctypes.byref(rz_t),
    ctypes.byref(fx_t), ctypes.byref(fy_t), ctypes.byref(fz_t)
)

# inizializzazione dei vettori contenenti le coordinate dei target points
# una volta effettuato il mesh morphing
x2 = []
y2 = []
z2 = []
# assegnazione dei valori
for i in range (ntarget):
    x2.append(x[i] + fx_t[i].val)
    y2.append(y[i] + fy_t[i].val)
    z2.append(z[i] + fz_t[i].val)

##### RISCrittura DEL K FILE #####
# il codice seguente effettua la sostituzione dei target points originari
# con i target points morphati, generando un nuovo k-file dalla struttura
# analoga a quella del k-file originario
sezione_corretta=False
with open(k_file_originale,'r') as input_file:
    lines=input_file.readlines()
with open(k_file_morphing,'w') as output_file:
    sezione_corretta=False
    for j in lines:
        if j.startswith(keyword1):
            sezione_corretta=True
            output_file.write(j)
        if not sezione_corretta:
            output_file.write(j)
        if sezione_corretta and not j.startswith(keyword1):
            for i in range(len(nid)):
                output_file.write(f"{nid[i]:8.0f}{x2[i]:16.8f}{y2[i]:16.8f}
                    {z2[i]:16.8f}{0:8.0f}{0:8.0f}\n")
            break
    sezione_corretta=False
    for j in lines:
        if j.startswith(stop_word):
            sezione_corretta=True
        if sezione_corretta:
            output_file.write(j)
print(f"Le righe tra {keyword1} e {stop_word} nel file {k_file_morphing}
    sono state sostituite con successo.")

##### TEMPO DI ESECUZIONE #####
end_time=time.time()
elapsed_time=end_time-start_time
print(f"Tempo di esecuzione: {elapsed_time} secondi")

```

9.4.3 Confronto.py

```

import numpy as np
import os
import matplotlib.pyplot as plt

##### FUNZIONI IMPIEGATE #####
# Questa funzione permette di generare il vettore contenente la numerazione
# dei nodi appartenenti al node_set creato in LS-PrePost.
# nome_file=inserire il percorso file al K file del modello
# keyword=inserire il nome del node_set desiderato

```



```

def numerazione_cercata(nome_file,keyword):
    elenco=[]
    keyword1='*SET_PART_LIST'
    keyword2='*SET_NODE_LIST_TITLE'
    sezione_corretta=False
    with open(nome_file,'r') as file:
        for riga in file:
            if riga.startswith(keyword):
                sezione_corretta=True
            if sezione_corretta:
                try:
                    numeri =list(map(float,riga.split()))
                    elenco.append(numeri)
                except ValueError:
                    a=riga.startswith(keyword1)
                    b=riga.startswith(keyword2)
                    if a or b:
                        break
    matrice_numerazione=np.array(elenco)
    numerazione=matrice_numerazione.flatten()
    return numerazione

# Questa funzione permette di generare la matrice delle coordinate di tutti i
# nodi presenti nel node set desiderato
# nome_file=inserire il percorso file al K file del modello
# keyword=inserire il vettore con la numerazione dei nodi del node Set desiderato
def individua_path(nome_file,vettore):
    elenco=[]
    keyword1='*NODE'
    keyword2='*PART_AVERAGED'
    sezione_corretta=False
    with open(nome_file,'r') as file:
        for riga in file:
            if riga.startswith(keyword1):
                sezione_corretta=True
            if sezione_corretta:
                try:
                    numeri =list(map(float,riga.split()))
                    elenco.append(numeri)
                except ValueError:
                    a=riga.startswith(keyword2)
                    if a:
                        break
    coordinate=np.array(elenco)
    path=coordinate[np.isin(coordinate[:,0],vettore)]
    return path

# Questa funzione, posti a confronto due insiemi di nodi (I1 e I2), permette di
# determinare il raggio medio della più piccola pin ball region che centrata su
# un nodo di I1 intercetta un nodo di I2
def distanza(x1,y1,z1,x2,y2,z2):
    min_rad=[]
    for i in range(len(x1)):
        r=[]
        for j in range(len(x1)):
            dx=x2[j]-x1[i]
            if abs(dx)>100:
                continue
            dy=y2[j]-y1[i]
            dz=z2[j]-z1[i]
            v=np.array([dx,dy,dz])
            r.append(np.linalg.norm(v))
        vec=np.array(r)
        min_rad.append(min(vec))
    a=np.mean(np.array(min_rad))

```

```
b=np.min(np.array(min_rad))
c=np.max(np.array(min_rad))
return a,b,c

#inserire il proprio percorso alla cartella THUMS completo
c0="C:/Users/emanu/Desktop"
c1="THUMS_completo"
c2="thums_model"

# inserire il nome del k_file_originario
nome_file='THUMS_AM50_V7_Occupant_202301_0.k'
# percorso file al suddetto file
k_file_originale50=os.path.join(c0,c1,c2,nome_file)

# inserire il nome del k_file_originario
nome_file='THUMS_AM95_V7_Occupant_202301.k'
# percorso file al suddetto file
k_file_originale95=os.path.join(c0,c1,c2,nome_file)

# inserire il nome del k_file dopo il morphing
nome_file='THUMS_AM50_V7_Occupant_202301_m95ossa.k'
# percorso file al suddetto file
k_file_morphing=os.path.join(c0,c1,c2,nome_file)

# inserire il nome del file.txt in cui salvare il confronto
nome_file='confronto_solid_ossa.txt'
# percorso file al suddetto file
confronto=os.path.join(c0,c1,c2,nome_file)

##### CONFRONTO ZONA SPECIFICATA #####
DmeanR5095=[]
DminR5095=[]
DmaxR5095=[]

DmeanR50m95=[]
DminR50m95=[]
DmaxR50m95=[]

DmeanR50m50=[]
DminR50m50=[]
DmaxR50m50=[]

# scegliere le zone di interesse
part_list=[]

for i in part_list:
    selezione=i
    if selezione==1:
        nome="bacino_superficie"
    elif selezione==2:
        nome="busto_superficie"
    elif selezione==3:
        nome="viso_superficie"
    elif selezione==4:
        nome="spalla_superficie"
    elif selezione==5:
        nome="braccio_superficie"
    elif selezione==6:
        nome="gomito_superficie"
    elif selezione==7:
        nome="avanbraccio_superficie"
    elif selezione==8:
        nome="polso_superficie"
    elif selezione==9:
        nome="mano_superficie"
```

```
elif selezione==10:
    nome="coscia_superficie"
elif selezione==11:
    nome="ginocchio_superficie"
elif selezione==12:
    nome="stinco_superficie"
elif selezione==13:
    nome="caviglia_superficie"
elif selezione==14:
    nome="piede_superficie"
elif selezione==15:
    nome="cassa_toracica"
elif selezione==16:
    nome="osso_sacro"
elif selezione==17:
    nome="scheletro"

if selezione==21:
    nome="bacino"
elif selezione==22:
    nome="collo"
elif selezione==23:
    nome="spalla_destra"
elif selezione==24:
    nome="spalla_sinistra"
elif selezione==25:
    nome="braccio_spalla_destra"
elif selezione==26:
    nome="braccio_spalla_sinistra"
elif selezione==27:
    nome="braccio_gomito_destra"
elif selezione==28:
    nome="braccio_gomito_sinistra"
elif selezione==29:
    nome="avanbraccio_gomito_destra"
elif selezione==30:
    nome="avanbraccio_gomito_sinistra"
elif selezione==31:
    nome="avanbraccio_polso_destra"
elif selezione==32:
    nome="avanbraccio_polso_sinistra"
elif selezione==33:
    nome="mano_destra"
elif selezione==34:
    nome="mano_sinistra"
elif selezione==35:
    nome="coscia_ginocchio_destra"
elif selezione==36:
    nome="coscia_ginocchio_sinistra"
elif selezione==37:
    nome="stinco_ginocchio_destra"
elif selezione==38:
    nome="stinco_ginocchio_sinistra"
elif selezione==39:
    nome="stinco_caviglia_destra"
elif selezione==40:
    nome="stinco_caviglia_sinistra"
elif selezione==41:
    nome="piede_destra"
elif selezione==42:
    nome="piede_sinistra"

# si estrae la numerazione relativa alla pelle di AM50
vettore_numerazione50=numerazione_cercata(k_file_originale50,nome)
# si estrapola la superficie desiderata nell'AM50
```

```
sup50=individua_path(k_file_originale50,vettore_numerazione50)
# si estrae la numerazione relativa alla pelle di AM50m
vettore_numerazione50m=numerazione_cercata(k_file_morphing,nome)
# si estrapola la superficie desiderata nell'AM50m
sup50m=individua_path(k_file_morphing,vettore_numerazione50m)
# si estrae la numerazione relativa alla pelle di AM95
vettore_numerazione95=numerazione_cercata(k_file_originale95,nome)
# si estrapola la superficie desiderata nell'AM95
sup95=individua_path(k_file_originale95,vettore_numerazione95)

x50=sup50[:,1]
y50=sup50[:,2]
z50=sup50[:,3]

x50m=sup50m[:,1]
y50m=sup50m[:,2]
z50m=sup50m[:,3]

x95=sup95[:,1]
y95=sup95[:,2]
z95=sup95[:,3]

##### minima distanza media/massima #####
dist5095=distanza(x50, y50, z50, x95, y95, z95)
dist95=distanza(x50m, y50m, z50m, x95, y95, z95)
distm50=distanza(x50, y50, z50, x50m, y50m, z50m)

MdmeanR5095.append(PR5095[0])
MdmaxR5095.append(PR5095[2])

MdmeanR50m95.append(PR50m95[0])
MdmaxR50m95.append(PR50m95[2])

MdmeanR50m50.append(PR50m50[0])
MdmaxR50m50.append(PR50m50[2])

print(f"{nome}")
print(f"la minima distanza media tra AM50m e AM95 è {PR50m95[0]}.")
print(f"la minima distanza media tra AM50 e AM95 è {PR5095[0]}.")
print(f"la minima distanza media tra AM50 e AM50m è {PR50m50[0]}.")
print('')
print(f"la minima distanza massima tra AM50m e AM95 è {PR50m95[2]}.")
print(f"la minima distanza massima tra AM50 e AM95 è {PR5095[2]}.")
print(f"la minima distanza massima tra AM50 e AM50m è {PR50m50[2]}.")
```

Bibliografia

- [1] Ansys, Ansys Workbench. <https://www.ansys.com/products/ansys-workbench>, 2024.
- [2] Ansys, LS-DYNA. <https://www.ansys.com/it-it/products/structures/ansys-ls-dyna>, 2024.
- [3] Anoop Chawla Devendra Kumar, Kuldeep Singh and Sudipto Mukherjee. Prediction of pedestrian injuries in motorcycle-pedestrian crashes using thums fe-hbm. *International Journal of Crashworthiness*, 0(0):1–13, 2023.
- [4] Raed E El-Jawahri, Tony R Laituri, Jesse S Ruan, Stephen W Rouhana, and Saeed D Barbat. Development and validation of age-dependent fe human models of a mid-sized male thorax. Technical report, SAE Technical Paper, 2010.
- [5] GHBM. Global human body models consortium. <https://www.ghbmc.com/>, 2016.
- [6] Eberhard Haug, Hyung-Yun Choi, Stéphane Robin, and Muriel Beaugonin. Human models for crash and impact simulation. *Handbook of numerical analysis*, 12:231–452, 2004.
- [7] Thomas Holdgrün, Igor Doric, Thomas Brandmeier, Therese Fuchs, Julia Mühlbauer, Philipp Steinert, and Steffen Peldschus. A virtual reality based approach for researching pedestrian to vehicle collisions. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 1318–1325, 2018.
- [8] EUNJOO Hwang, YULONG Wang, MATTHEW P Reed, JD Rupp, and J Hu. Parametric human body modeling: development and application of new methods for rapid development of diverse human models for crash simulation. *UMTRI Report*, 2014.
- [9] infodoc SERVIZI DI INFORMAZIONE TECNICA. SAE. <https://www.infodoc.it/norme-tecniche/sae/>, 2018.
- [10] Osamu Ito, Yasuhiro Dokko, and Kazuki Ohashi. Development of adult and elderly fe thorax skeletal models. Technical report, SAE Technical Paper, 2009.
- [11] Masami Iwamoto, Yoshikatsu Kisanuki, Isao Watanabe, Katsuya Furusu, Kazuo Miki, and Junji Hasegawa. Development of a finite element model of the total human model for safety (thums) and application to injury reconstruction. In *Proceedings of the international IRCOBI Conference*, pages 18–20, 2002.
- [12] MathWorks, MATLAB. <https://it.mathworks.com/products/matlab.html>, 2024.
- [13] Jonas Östh, Karin Brolin, Stina Carlsson, Jac Wismans, and Johan Davidsson. The occupant response to autonomous braking: a modeling approach that accounts for active musculature. *Traffic injury prevention*, 13(3):265–277, 2012.
- [14] Python. <https://www.python.org/>, 2024.
- [15] RBF Morph. <https://www.rbf-morph.com/>, 2024.
- [16] RBF Morph, RBF Morph Structures. <https://www.rbf-morph.com/act-module/>, 2024.
- [17] Stephen A Ridella, Jonathan D Rupp, and Kristin Poland. Age-related differences in ais 3+ crash injury risk, types, causation and mechanisms. In *IRCOBI conference*, volume 2012, 2012.
- [18] Stéphane Robin. Humos: Human model for safety - a joint effort towards the development of refined human-like car occupant models. Technical report, SAE Technical Paper, 2001.
- [19] Jesse Ruan, Raed El-Jawahri, Li Chai, Saeed Barbat, and Priya Prasad. Prediction and analysis of human thoracic impact responses and injuries in cadaver impacts using a full human body finite element model. Technical report, SAE Technical Paper, 2003.

-
- [20] JD Rupp and CAC Flannagan. Effects of occupant age on ais 3+ injury outcome determined from analyses of fused nass/ciren data. In *SAE 2011 Government/Industry Meeting*, 2011.
- [21] Samantha L Schoell, Ashley A Weaver, Jillian E Urban, Derek A Jones, Joel D Stitzel, Eunjoo Hwang, Matthew P Reed, and Jonathan D Rupp. Development and validation of an older occupant finite element model of a mid-sized male for investigation of age-related injury risk. Technical report, SAE Technical Paper, 2015.
- [22] Xiangnan Shi, Libo Cao, Matthew P Reed, Jonathan D Rupp, and Jingwen Hu. Effects of obesity on occupant responses in frontal crashes: a simulation analysis using human body models. *Computer methods in biomechanics and biomedical engineering*, 18(12):1280–1292, 2015.
- [23] TOYOTA. About thums. <https://www.toyota.co.jp/thums/about/>, 2024.
- [24] TOYOTA. Accesso gratuito al modello THUMS. <https://www.toyota.it/mondo-toyota/news-eventi/2021/toyota-accesso-gratuito-modello-thums>, 2021.
- [25] TOYOTA. Download THUMS. <https://www.toyota.co.jp/thums/>, 2024.
- [26] TOYOTA MOTOR CORPORATION, TOYOTA CENTRAL R&D LABS.,INC. AM50 Occupant Model Version7, (Settembre 2023).
- [27] TOYOTA MOTOR CORPORATION, TOYOTA CENTRAL R&D LABS.,INC. AM95 Occupant Model Version7, (Settembre 2023).
- [28] Philippe Vezin, Karine Bruyère, and François Bermond. Comparison of head and thorax cadaver and hybrid iii responses to a frontal sled deceleration for the validation of a car occupant mathematical model. In *17th International Technical Conference on the Enhanced Safety of Vehicles (ESV)*, pages 10–p, 2001.
- [29] David C Viano, Chantal S Parenteau, and Mark L Edwards. Crash injury risks for obese occupants using a matched-pair analysis. *Traffic injury prevention*, 9(1):59–64, 2008.
-