

POLITECNICO DI TORINO

Department of Mechanical and Aerospace Engineering

Master's Thesis
in
Automotive Engineering

Passive Safety in Autonomous Vehicles: Machine Learning and Artificial Intelligence Applied to Human Body Model Positioning



Tutors

Prof. Eng. Alessandro Scattina

Prof. Eng. Giovanni Belingardi

Candidate

Paolo Ascia

March 2020

Abstract

The development of automated driving systems is raising new challenges for passengers safety. One of the reasons is that the driver and the passengers can engage in a wider range of activities. Thus, their posture varies more from what is nowadays considered to be conventional. Being less expensive, finite element methods are used to investigate new crash scenarios, including those in which passengers are sitting in non-conventional positions. However, positioning a human body model in a non-conventional posture is a very demanding process. The combination of machine learning and artificial intelligence allows to reduce the order of a model, thus drastically lowering both time and computing effort. This paper aims at making a reduced-order model of the THUMS in order to reduce the time for positioning it. With the intention of doing so, four reduced-order sub-models are combined into one model. Each sub-model controls a limb and requires a specific database. Testing is required on the sub-models and the merged model. Both the database and the testing samples are computed using full finite element simulations. The reduced-order model should eventually allow human body model positioning in a few minutes, with a precision nearly as close as a full finite element simulation.

Keywords: *Human Body Model Positioning, Reduced Order Model, Machine Learning, Artificial Intelligence, Automated Driving Systems*

Contents

PREFACE	I
SCENARIO	I
PURPOSE	I
ROADMAP	I
CHAPTER 1: LITERATURE REVIEW	1
1.1 NOMINAL POSTURE.....	1
1.2 AUTONOMOUS VEHICLES.....	1
1.2.1 Automation Levels	2
1.2.2 Non Driving Related Activities	3
1.2.3 Type of Crash	5
1.3 TARGET POSTURE	6
1.4 POSITIONING PROCEDURES	8
1.4.1 Classical Procedure	8
1.4.2 PIPER Tools Smoothing.....	9
1.4.3 Reduced Order Model	10
CHAPTER 2: GENERATING A ROM	13
2.1 ROADMAP	13
2.2 TOOLS.....	14
2.3 DATABASE	16
2.4 INTERPOLATION	19
CHAPTER 3: RESULTS	21
3.1 COMPARISON METHODOLOGY	22
3.1.1 Nodal Displacement	22
3.1.2 Nodal Coordinates.....	22
3.1.3 Bones Volume	22
3.2 COMPARISON WITH REFERENCE.....	23
3.2.1 ROM.....	24
3.2.2 Nodal Displacements Interpolation	25
3.2.3 InvD Interpolation	27
CONCLUSION	29
APPENDIX A POSITIONING INPUT FOR TARGET POSITIONS	31
APPENDIX B DATABASE POSITIONS	33
APPENDIX C ROM TEST RESULTS	41
APPENDIX D DISPLACEMENTE INTERPOLATION TEST RESULTS	57

APPENDIX E INVD INTERPOLATION RESULTS	73
REFERENCES	89
ACKNOWLEDGEMENTS	93

Preface

Vinayak V. Dixit et al. [1] described a challenging future for the automotive industry: *“Autonomous vehicles are predicted to be transformative, with a potential to improve productivity, reduce congestion and improve safety. However, there are many safety and risk related unknowns associated with the autonomous vehicles, with regards to factors affecting disengagements and driver behaviour, at moments requiring manual resumption of vehicle control. The ultimate success of automated vehicles will depend on drivers’ trust in them and on how people choose to use and interact with them, and the ensuing safety risk”*.

Scenario

The introduction of autonomous vehicles in the automotive panorama is raising new challenges. Given the fierce competition, manufacturers need to timely develop new products in order to answer these challenges. Therefore, whoever detains the faster procedure has a winning advantage over other companies. When it comes to vehicle safety, new challenges are arising from the widened range of activities the occupant can engage in. Extensive testing is then required. Nowadays Finite Element (FE) simulations are used to predict physical test results and reduce the number of physical mock-ups required to solve a problem. Among the many FE simulations used, full crash test simulations are very important, but also very demanding to perform. The process of setting up a crash simulation requires many steps. Positioning the Human Body Model (HBM) is one of the fundamental passages. Since it is conventionally performed by running a full FE simulation, it is a very demanding step. In this work, the Total Human Model for Safety version 4.1 (THUMS) [2] is used. Positioning this HBM with a full FE simulation requires on average 6 hours and a half, using 64 cores (Intel® Xeon Scalable Processors Gold 6130 2.10GHz).

Purpose

This thesis work aims at searching for alternative ways to position the THUMS in order to reduce the requirements for both computing power and time. The solution proposed is to follow the steps of Noriyo Ichinose et al. [3] and attempt replicating the work presented at the Automotive CAE Grand Challenge 2020 Carhs conference. The procedure generates a Reduced Order Model (ROM) for predicting the results of a positioning simulation in under a minute on the user personal computer. This model is thought such to allow postures not commonly tested, such as those in which the occupant is using a phone.

Roadmap

In Figure 1 the main steps followed to develop the work are schematically represented. The first one is to familiarize with the tools already available, such as PIPER Tools [4]. This latter is then used in step

2 to set a target position used as a reference to better understand which Degrees of Freedom (DoFs) are required for moving the entire THUMS. The following steps, except the last one, are aimed at generating a ROM capable of positioning this HBM. Firstly, it is important to determine what is the range of motion of the DoFs free to move (step 3). Then comes designing and computing the database (step 4). Therefore, as suggested by Noriyo Ichinose [3], the THUMS is divided into 4 sub-models, each one describing a limb. This subdivision allows to reduce the database size and maintain the prediction quality. Each model has a unique database and each sample point in the database corresponds to a full FE positioning simulation. Once the database is computed, the machine is trained, meaning that, thanks to specific algorithms, relations between all samples are computed (step 5). These relations allow the machine to predict new poses. Testing is then performed in two stages: in the first one (step 6) each ROM is individually tested to evaluate the said predictions. After all models are merged (step 7), the second stage of testing is performed (step 8). This last is carried out to verify that no problems arise from the merging step. Once both verification stages are successful, in the last step, an attempt is made to obtain the target position to prove the effectiveness of this new methodology (step 9). However, it is important to notice that the boundaries of the ROM are much stricter than those of a full FE simulation. Consequently, the range of possible positions is narrower.

For technical difficulties, it was not possible to complete the entire procedure: the last three steps are missing.

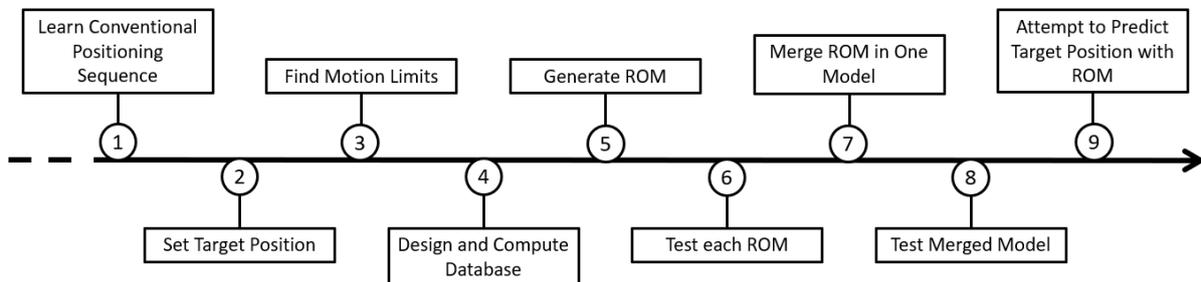


Figure 1. Work roadmap

Chapter 1: Literature Review

An extensive literature review was carried out before starting the thesis work to understand the current state of the art on vehicle safety. This review mainly focused on human body models, crash tests and simulations, autonomous vehicles and their implications on driver and passenger behaviour, and machine learning. The reason why only a few topics are presented in this chapter is that most of them are related to procedures learnt or tools used throughout the work.

1.1 Nominal Posture

In this work, the “nominal posture” is the dummy posture described by Euro NCAP in their MPDB Frontal Impact Testing Protocol [5]. For the purpose of this review, the reader is invited to focus on the following requirements (in brackets the paragraph number of the said protocol):

- “The driver’s upper arms shall be adjacent to the torso as far as is possible” (§ 6.4.1);
- “The driver’s back shall be in contact with the seat back and the centre line of the dummy shall be lined up with the centre line of the seats” (§ 6.4.2);
- “The dummy’s H-point shall be within a square of $\pm 13\text{mm}$ in Z and $\pm 13\text{mm}$ in X of a point 20mm upward and 20mm forwards of the H-point as determined in Section 6.1” (§ 6.4.3);
- “The driver’s hands shall have their palms placed against the steering wheel at a position of a quarter to three” (§ 6.4.7);
- “The upper legs shall be in contact with the seat cushion as far as possible” (§ 6.4.9);
- “The driver dummy’s right foot shall rest on the undepressed accelerator pedal with the heel on the floor” (§ 6.4.10);
- “If a dedicated foot-rest is present, place the left foot fully on this rest providing a normal seating position can still be achieved” (§ 6.4.10).

Any position that does not comply with the EURO NCAP protocol is referred to as Out Of Position (OOP).

1.2 Autonomous Vehicles

In this work, “Autonomous Vehicles” (AVs) are any vehicle fitted with any Driving Automation Systems (DAS) or Technology, defined as “the hardware and software that are collectively capable of performing part or all of the dynamic driving tasks (DDT) on a sustained basis” [6]. Notice that not all AVs can perform the entire DDT, whereas, by the definition given, all vehicles capable of performing at least one DDT autonomously are AVs. As recommended by SAE International, vehicles that can perform the entire DDT on a sustained base are called Automated Driving Systems (ADS). When referring to an ADS it is important to define the DDT Fallback and the Operational Design Domain (ODD). The first is defined as the response of the user or the system in order to perform a DDT or

achieve a minimal risk condition after a failure or exiting ODD, whereas ODD indicates the conditions under which a DAS is specifically designed to function.

1.2.1 Automation Levels

Vehicles are classified depending on their ability to perform DDT without the intervention of a driver. SAE International, in [6], suggests the following six automation levels.

Level 0 – No Driving Automation: the driver is required to perform all DDT, even if the vehicle is enhanced by active safety systems;

Level 1 – Driver Assistance: either the lateral or longitudinal vehicle motion control task of the DDT can be performed by a DAS in a specific ODD;

Level 2 – Partial Driving Automation: both the lateral and longitudinal vehicle motion control task of the DDT can be performed by a DAS in a specific ODD; the driver is expected to complete Object and Event Detection and Response (OEDR) subtasks;

Level 3 – Conditional Driving Automation: an ADS is capable of performing the entire DDT, in a specific ODD; the driver is still expected to be ready to take control as the DDT fallback-ready user;

Level 4 – High Driving Automation: an ADS can perform the entire DDT and DDT fallback, in a specific ODD, without any expectations from the user;

Level 5 – Full Driving Automation: an ADS can perform the entire DDT and DDT fallback, no longer specifically to an ODD.

In Table 1 the reader can find a schematic representation of the six levels proposed. Notice that all but level 0 are AVs, whereas only level 3, 4 and 5 are ADS. As previously stated, this work focuses on a level 3 vehicle, in which a dashboard as the one commonly fitted in cars today is still required.

Table 1. Automation Levels [6]

Level	Name	DDT		DDT fallback	ODD
		Sustained lateral and longitudinal vehicle motion	OEDR		
Driver performs part of all the DDT					
0	No Driving Automation	Driver	Driver	Driver	n/a
1	Driver Assistance	Driver and System	Driver	Driver	Limited
2	Partial Driving Automation	System	Driver	Driver	Limited
ADS performs the entire DDT (while engaged)					
3	Conditional Driving Automation	System	System	Fallback-ready user	Limited
4	High Driving Automation	System	System	System	Limited
5	Full Driving Automation	System	System	System	Unlimited

1.2.2 Non Driving Related Activities

With the introduction of level 3 AVs, drivers can stop paying attention to traffic conditions and engage in other activities. In 2017 Sofia Jorlöv et al [7] carried out a survey to understand public perception of ADS. The paper suggests a link between trust in an ADS, journey length, activities and seats arrangement. This last one is not of interest, since a level 3 AVs is considered to be fitted with a conventional vehicle configuration to allow the driver to take control of the vehicle at all times. On the other hand, it was suggested that, only when the driver fully trusts the vehicle, they engage in other activities.

Assume now the driver trusts ADS: the length of the journey is the second most influencing parameter on activities. The results of the Sofia Jorlöv et al [7] survey are shown in Figure 2. Participants aged 18 or older are given two scenarios: short and long drive. Participants under 18 are given only the long scenario. The short drive is referred to as the interviewee last drive to work, whereas the long drive as the interviewee family trip to their summer house. As shown in Figure 2, longer activities (such as watching a movie or playing a game) are preferred on a longer journey. The population of this survey is small, thus limiting the possibility of generalizing the results.

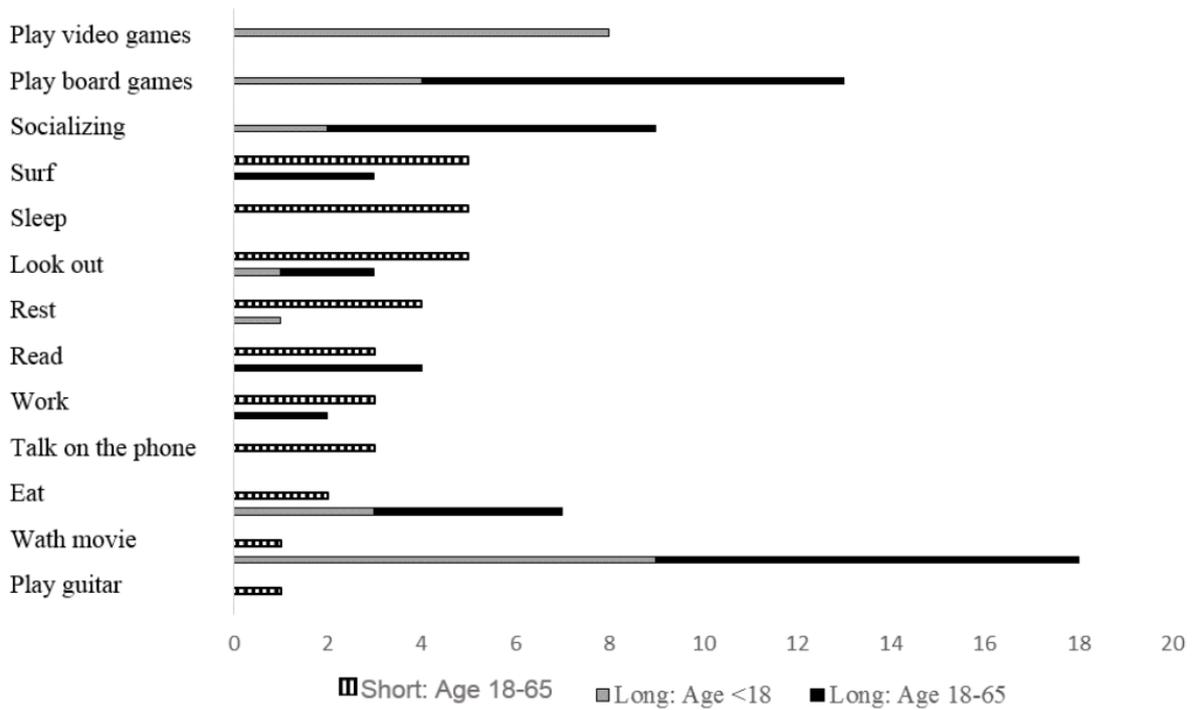


Figure 2. Activities in a level 5 AVs [7]

On the steps of the Swedish research, Matthew P. Reed et al. [8] [9] designed a wider survey to attempt a better estimation of activities in ADS. Both papers are focused on the behaviour of front-seat passengers in conventional vehicles, as it is considered to be the best predictor of AVs future activities. The methodology was the same for both surveys: cameras recording the front passenger of several vehicles for a prolonged period. Later, specific frames are analysed from each journey to quantify the front passenger posture and activity.

In Table 2 the results of [8] are shown. The paper reports that passengers are mostly looking out the windshield or the passenger window. On the other hand, the most frequent interaction is talking to the driver. Other activities relevant to this review are: using the phone (9.9%), eating or drinking (4.8%) and resting (1.1%).

Table 2. Interaction Frequencies [8]

Percent [%]	Behaviour
56.9	Talking
22.9	Nothing
13.4	Other
9.9	Phone
3.1	Food
1.7	Drink
1.1	Resting

Results presented in [9], reported in Table 3, are taken from a video survey more recent than the one in [8]. The use of the phone has more than doubled: 26.4% against 9.9% of Table 2. Nevertheless, the most common interaction is still talking to the driver (46.0%), whereas activities like eating, drinking or resting are much less common. Matthew P. Reed et al. also highlight a correlation between trip length and type of activity: the longer the trip, the more frequent using the phone or resting. However, longer trips are less frequent. However, when trying to adapt the results to an ADS scenario, the paper suggests: *“In this scenario, the already common interaction with handheld devices might replace some or all of the conversation time”* [9].

Table 3. Passenger interaction frequencies [9]

Percent [%]	Behaviour
46.0	Talking to other occupants
26.4	Phone
25.9	Nothing
5.7	Other
3.2	Food
2.2	Resting
1.6	Drink

1.2.3 Type of Crash

In an ADS, if a passenger does not realize they are about to crash, at the time of the impact they are still engaged in their activity. Thus, a key factor is understanding what caused the accident and what happened inside the car just before the collision.

In literature, multiple statistics report the type of crash ADS were involved in [1] [10] [11]. Most incidents are of a small entity and mainly consist of ADS being rear-ended by a conventional vehicle, as shown in Figure 3. The main cause is that ADS braking manoeuvre is often too abrupt for a human driver. Therefore, the conventional vehicle driver fails to brake in time. Another risk, as pointed out by Vinayak V. Dixit et al. [1], is automatic disengagement of ADS, without enough time for the driver to properly take control of the vehicle. Both these types of problems have low frequency and are likely to be solved in the future.

On the other hand, even if not frequent, accidents caused by faulty object detection or faulty vehicle state estimation are more likely to keep happening. To try predicting the passenger behaviour crashes involving level 2 AVs are taken into account because their use resembles the one of a level 3 AV. Some drivers, due to an excessive trust in the DAS fitted to their level 2 AVs, engage in other activities as if they were in an ADS. In these conditions, some accidents have occurred. Police reports state that in all cases the driver activated all DAS available and engaged in other activities [12] [13] [14] [15]. At the

time of the crash, all drivers were still engaged in their activities and did not realize the imminent threat. Police confirmed that the root cause is excessive trust in AVs and failure in object detection. The postures proposed in the following paragraph are based on the assumption that in a crash involving level 3 AV the driver is still engaged in their activity.

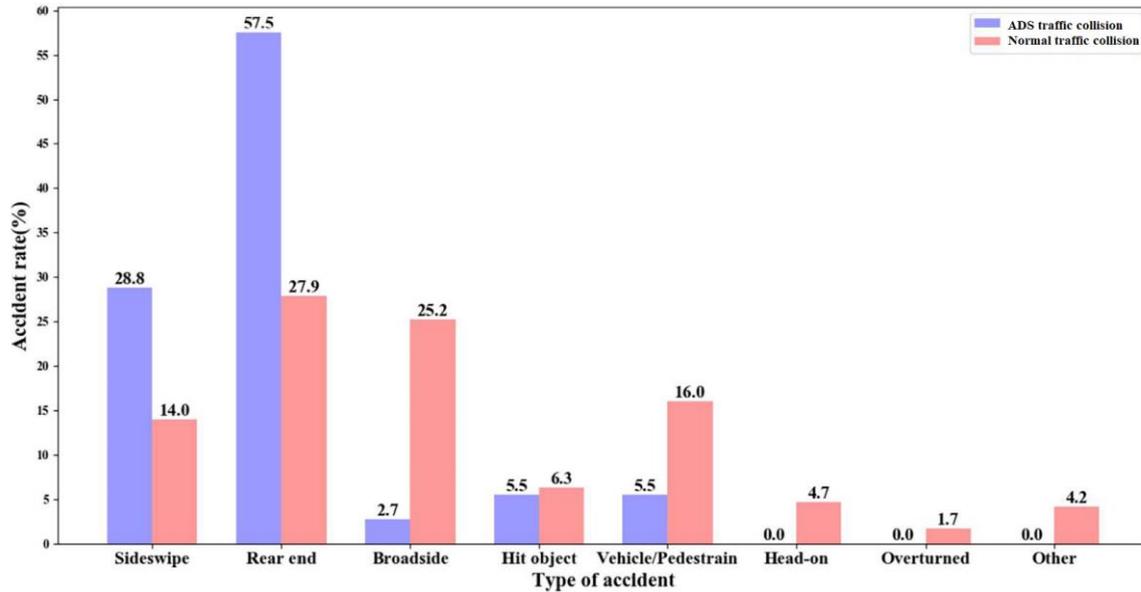


Figure 3. Comparison between crashes involving ADS and normal crashes [11]

1.3 Target Posture

Since in an ADS the driver becomes a passenger, as suggested by Matthew P. Reed et al. [9], the front-seat passenger behaviour is the best predictor for future activities in high automation level vehicle. To find front-passenger postures and activities, Lijiang (Lee) Zhang et al. [16] carried out an interview in 2004. However, the results are in contrast to what has been revealed by video surveys. Another approach is to use traffic cameras data [17]. However, only the upper body could be analysed, thus the results are limited and not relevant to this work.

Matthew P. Reed et al. video surveys [8] [9] are the most detailed and comprehensive research work found. Considering all of the assumptions previously made and analysing the data of both research papers, two positions are analysed in detail: the first one (position 1) being one of the most frequent postures, the second one (position 2) being an interesting case that could increase in popularity.

Both positions share some features. The first one is a slightly rotated pelvis, typical of a more relaxed position. Then, the lower limbs are spread apart: the femur-tibia plane is tilted outwards for the left leg and inwards for the right leg. Figure 4-a is used as a reference for the lower limbs. The driver's feet are resting on the floor. The right upper limb is in contact with the door armrest. However, in position 1 only the right elbow rests on the armrest, whereas in position 2 the entire forearm is resting on it. The left upper limb is interacting with the phone: in the first case the elbow is resting on the left armrest, the

phone is on the left-hand side of the passenger and the head is tilted downwards and to the left as if he or she were looking at the phone screen. As a reference consider the arms positions in Figure 4-a. Meanwhile, in the second case, the phone is near the ear as if the driver were on a call, similarly to Figure 4-b. The end results are shown in Figure 5: position 1 on the left and position 2 on the right. In Appendix A the reader can find the inputs given in PIPER Tools to position both cases.

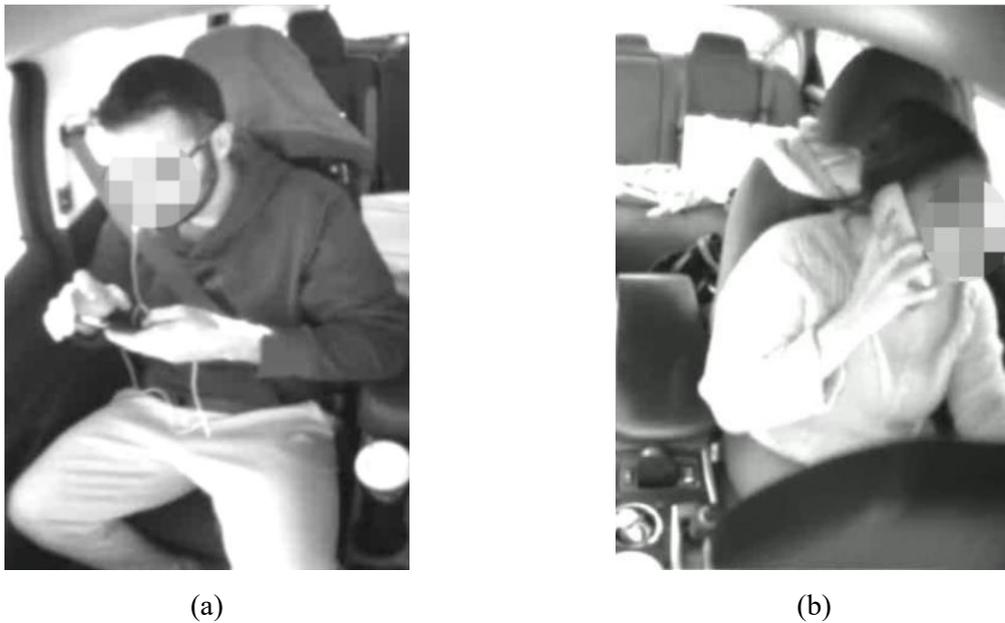


Figure 4. Reference positions: a) reference for a relaxed position, lower limbs position and position 1 phone use; b) reference for position 2 phone use [9]

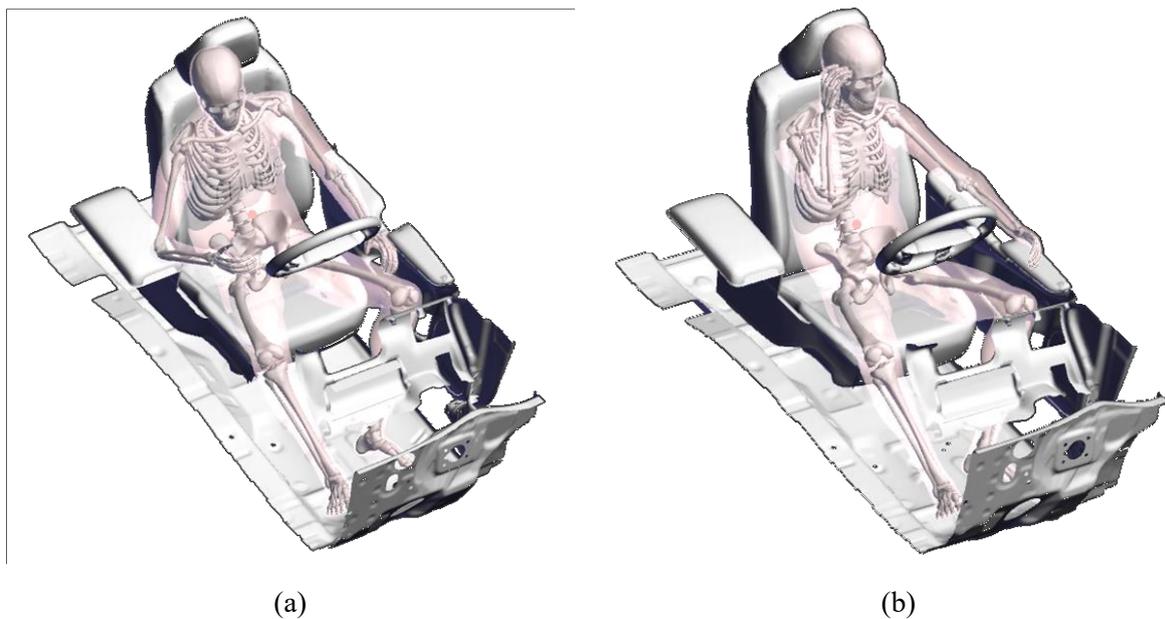


Figure 5. Target positions: a) position 1; b) position 2

1.4 Positioning Procedures

The original position of the THUMS, shown in Figure 6, generally needs to be repositioned to comply with either crash test protocols (e.g.: nominal posture described on page 1) or to reach the desired OOP. To do so, the literature suggests three different methods: one based on FE simulation (hereinafter referred to as “Classical Procedure”), one based on interpolation and one based on machine learning (ML).

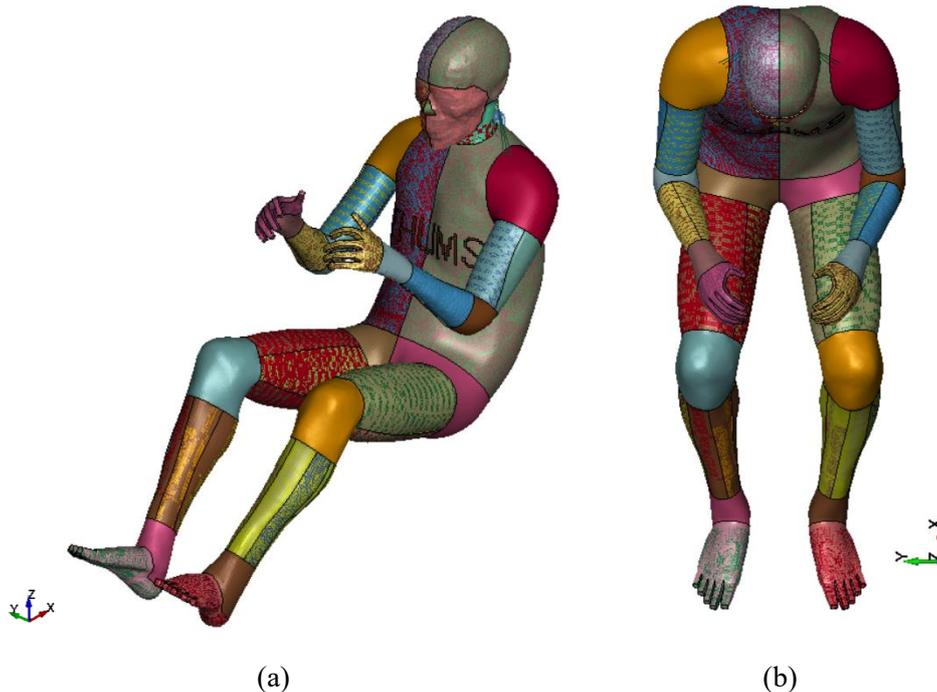


Figure 6. THUMS original position: a) isometric view, b) top view

1.4.1 Classical Procedure

The classical procedure runs a full FE simulation to obtain the desired end position. Loads or displacements are applied to some HBM parts to move it [18]. For example, in Figure 7 it is shown how loads are applied to extend an arm or to lower one foot. This method is the most used one because it is easy to set up and it provides good results. The reason why is its ability to account for all contacts between internal organs without the need for user actions. On top of that, the final result is not affected by the presence of inverted elements. Nevertheless, this method can be time-consuming for large models (as THUMS) and it can lead to some artefacts. These last are because typical biomechanical initial strains and stresses in tissues are not accounted for. Consider Figure 8, the deformation on the HBM belly is an artefact due to the lack of information on how the belly soft tissues should deform. Moreover, defining the target position is a challenge without the correct programs. PIPER Software or OASYS Primer [19] have specific tools to easily position an HBM and visualize a preview without having to run

any simulation. These tools then automatically apply loads and prescribed motions needed in a positioning simulation to the HBM. Since PIPER Software is not a Pre-Processor, the files generated for the motion have to be edited to include the HBM itself. On the other hand, OASYS Primer is a Pre-Processor, thus no additional passages are needed. More details on the positioning tool of PIPER Software in [20]. Instead, more details on OASYS Primer are available on their website [19].

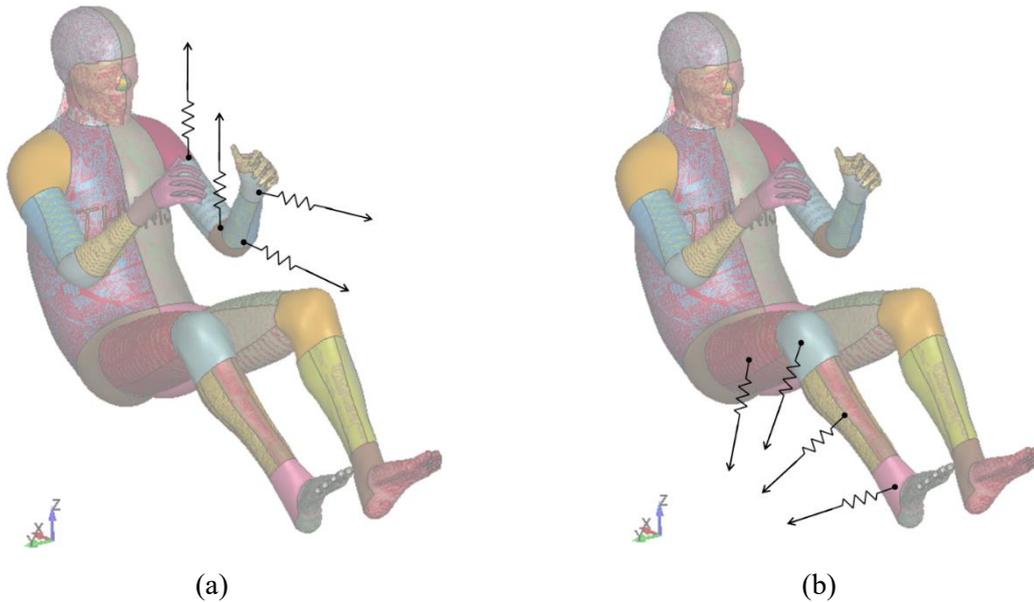


Figure 7. Example of load for HBM repositioning sequence: a) force applied to extend arm; b) force applied to lower one foot

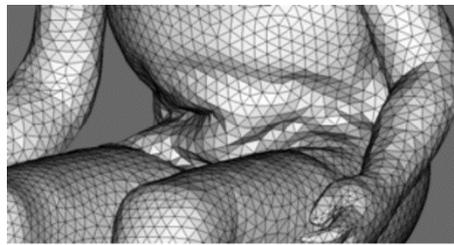


Figure 8. Positioning simulation artefacts on HBM belly [18]

1.4.2 PIPER Tools Smoothing

The second method is based on interpolating algorithms to reconstruct the HBM mesh and counteract excessive deformations and artefacts of a positioning simulation [18]. The only tool found that allows such reconstruction is PIPER Software, thanks to its “Smoothing Module”. The interpolation performed is based on radial basis functions and Dual Kriging Formulation. An example of how powerful this tool can be is shown in Figure 9. More details are available in [21].

Smoothing does not account for contact interaction between internal organs [18]. It repositions only the nodes without considering their relations (i.e.: to which element they belong) because Kriging is a

meshless approach. The positioning process does not require a full FE simulation, but it does not simulate the sliding of internal tissues. On the contrary, it does account for biomechanical motion limits.

On the positive side, the procedure does require less computational power, thus it can be performed on a personal computer. However, as suggested in [22], it still takes a lot of time. Smoothing cannot be done on a global scale, it needs to be broken down into small steps, each one smoothing one part of the HBM. For each step, PIPER Software asks to select the part to be smoothed and the area around it. The undeformed area is used as a reference to set up Kriging and radial basis functions. Consequently, a skilled user is required to select the right portion of the HBM and the right reference.

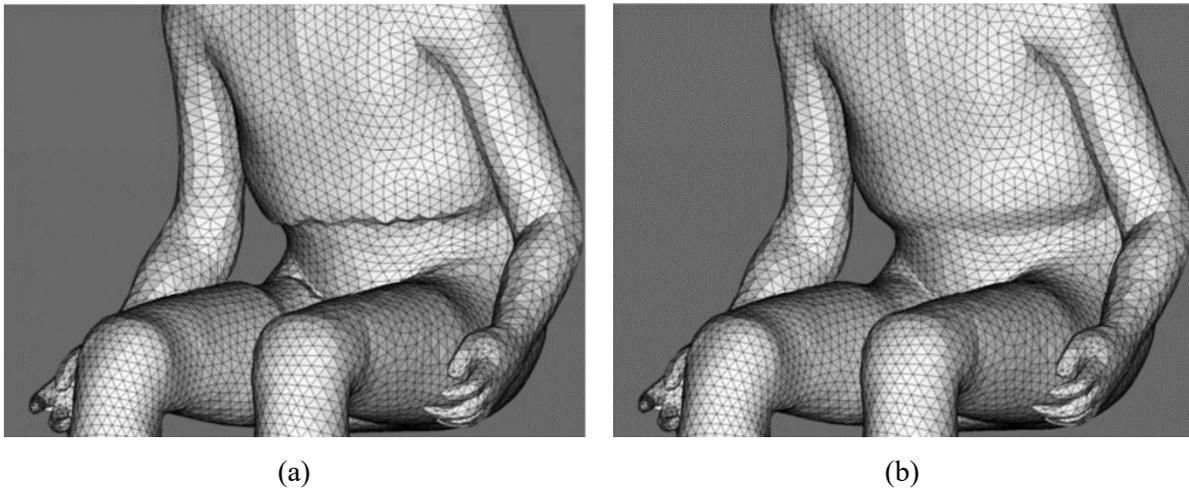


Figure 9. PIPER Software Smooth Module results: a) before smoothing, b) after smoothing [18]

1.4.3 Reduced Order Model

To reduce the time needed for positioning, this method uses the advantages of ML and Artificial Intelligence (AI) to quickly predict the result of a full positioning simulation. To generate a Reduced Order Model (ROM) there are different ways. Goustan Bacquaert et al. [23] describes model reduction as a process *“based on the observation that the solution of many physical problems can be approximated on a low-dimensional manifold which is embedded in the high dimensional space spanned by the model’s degrees of freedom”*. Therefore, a ROM predicts the results of a problem by combining data from a database with a partial solution to the problem. Working with ROM involves two steps [24]: 1) training; 2) testing. Training involves decomposing and compressing (reducing) a database and computing convergence indicators. In this phase, information is analysed and relations are generated to create data patterns. Testing, on the other hand, requires reconstructing the model (predict a solution), comparing the prediction with a reference and checking the database for redundancies and correlations. Testing allows evaluating the quality of the patterns found. At last, some quality indicators are defined to quantify how accurate the prediction is with respect to a full FE simulation.

This positioning method needs a database and a ROM. Generally, to create a database the Classical Procedure is used. As previously stated, there are different ways to generate a ROM. In this review, only

two are considered. The first is Goustan Bacquaert et al. [23] methodology. They worked on the THUMS, creating a subspace from a pool of 21 training positioning simulations (2121 snapshots collected). Despite a rather small pool, they obtained good predictions, thanks to techniques such as gappy data completion and neighbourhood regularization. They claim that the ROM has information on limbs, head and torso motions.

The second methodology is the one developed by Noriyo Ichinose et al. [3]. They used CADLM Lunar software to handle the database and generate a ROM. Their model has already gone through more than one iteration. The first one has a database of 158 simulations. A single ROM is used to predict the entire THUMS. However, despite a high accuracy, in the predicted model there are some penetrations between different parts. This deeply affected the performance in a crash simulation. To solve this problem, in the second iteration, for each limb one ROM was developed. The prediction from each ROM is then merged with the others. Results are more accurate and no penetrations are found. On the downside, the database was resampled and it is now composed of only 40 simulations.

The merged prediction presents a maximum displacement error of 1.5% and a maximum nodal coordinate error around 3 mm. In the testing position, the maximum displacement error is in the right shoulder region and only 1% of the nodes are 3 mm or more away from their reference position. The project managed to allow only limbs movements. Noriyo Ichinose et al. claim that for each position to 17 seconds are needed for a prediction. The research group said that development is still going on and that soon also spine and head movements will be available.

Chapter 2: Generating a ROM

This chapter focuses on the steps needed to train the machine, thus generating a ROM. As Noriyo Ichinose et al. [3] did, four different ROMs are generated, one per limb. Thus, four different databases are computed and four models are created. Consequently, four different testing procedures are needed as well as one for checking the merged solution. The testing phase methodology is described more in depth in Chapter 3.

2.1 Roadmap

In reference to Figure 1 steps from 3 to 9 are broken down into more details in Figure 10, to allow a better understanding of the procedure. Since one ROM is generated per limb, steps from 3 to 7 are repeated four times. The very first action is finding to which DoFs allow freedom and to what extension (step 3). This is done by trial and error, using PIPER Software. Since the dynamics in PIPER Software can diverge from those in a positioning simulation, all limits are verified with a full FE simulation.

Based on the limits found, Design of Experiment (DOE) methodologies are used to generate specific sample points (step 4). As a result of doing this, the subspace of each limb is properly described. A full positioning simulation corresponds to each sample point. To prepare each simulation, PIPER Software and a Pre-Processor are used. The procedure is described in more details on page 8. The results of each simulation are then checked for compliance with the input given in PIPER Software (step 5).

The database is then imported into the CADLM Lunar environment as the training set. For each limb, a ROM is computed (step 6) and tested (step 7). The Classical Procedure is again used to generate the testing sample. If the prediction is not accurate enough, first other interpolation methods are tested and then, if the new results are still not acceptable, the database is enlarged. Once the results are acceptable, the model is merged with the other models (step 8). To do so, the nodal displacements of each prediction are superimposed. The total displacement is then applied to each node in order to compute the new coordinates. The results of this process are tested to verify if problems occurred in the merging step (step 9). This double verification allows to clearly identify whether there is a problem with the merging process or with the error tolerances accepted per limb. If all tests are successful, the prediction is considered to be accurate enough to be used instead of a positioning simulation.

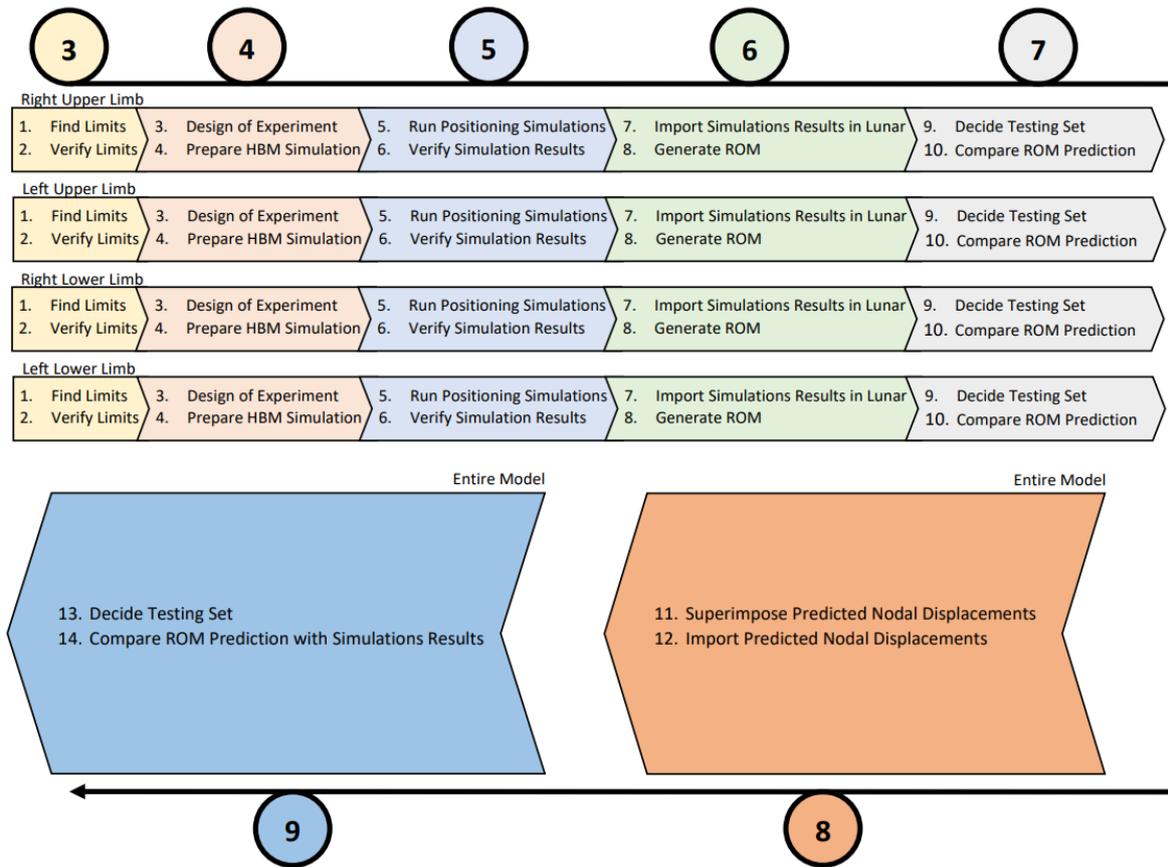


Figure 10. Detailed roadmap to generate the THUMS ROM

2.2 Tools

The following computer programs are used:

- PIPER Software v1.1.0 to position the THUMS and generate the positioning script files;
- HyperMesh 2019 to prepare the THUMS;
- LS PrePost v4.7.20 to prepare the positioning simulation using PIPER Software positioning script files;
- Lunar v4.1.0.0 to design the experiments and elaborate the positioning simulation output for generating a ROM;
- MATLAB 2020b to combine all ROM predictions;
- HyperView 2019 to test each ROM and the final prediction.

The tools used and procedure followed in PIPER Software and LS PrePost are the very same as those used in the Classical Procedure. More details can therefore be found on page 8. HyperMesh on the other hand is used to remove all parts from the THUMS that are not of interest. For example, consider the left upper limb database: all parts not belonging to the left arm are removed, as shown below in Figure 11 – a. The FE model obtained is then used in the positioning simulation instead of the full THUMS. This

substitution allows reducing computational requirements and improving the ROM prediction. Lunar is used both for designing the database sub-space and for reducing the model itself, as explained later on. More information on Lunar can be found in [25]. A MATLAB script is used to combine the predicted displacement and apply it to the nodal coordinates. Lastly, in HyperView bones volume, nodal displacements and nodes coordinates are compared, to evaluate the prediction accuracy in relation to the simulation results.

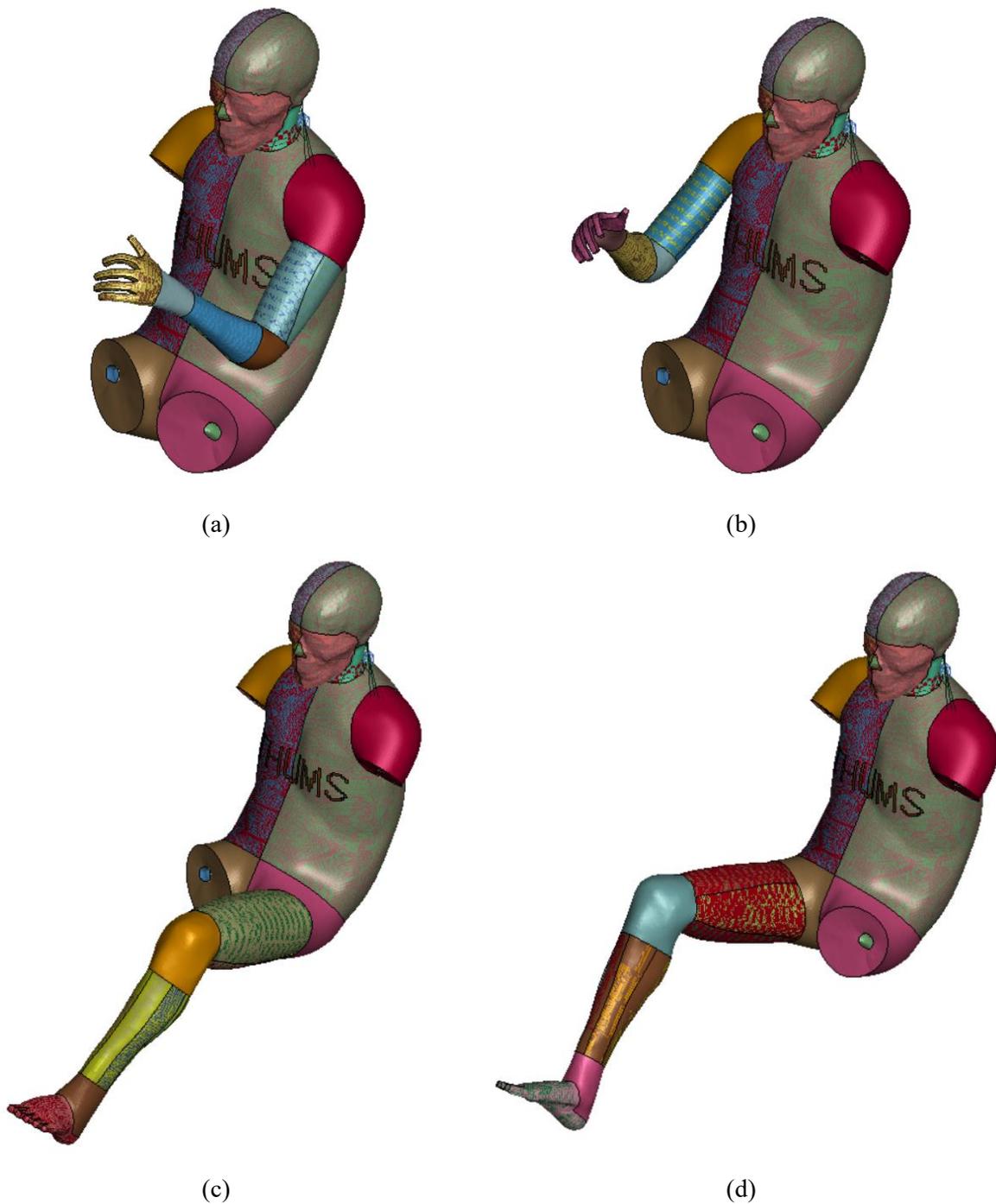


Figure 11. THUMS used for: a) left upper limb database; b) right upper limb database; c) left lower limb database; d) right lower limb database

2.3 Database

As previously stated, four different databases were generated. However, the same procedure was used for all of them.

The first step is to decide which DoFs to allow. The best compromise between computational effort and motion freedom is 4 DoFs per limb: each subspace is then 4-dimensional. As shown in Figure 12, both upper limbs have 3 rotational DoFs on the shoulder and 1 rotational DoF on the elbow, whereas the lower limbs have 3 rotational DoFs on the hip and 1 rotational DoF on the knee.

The next step is identifying the motion limits. Table 4 reports the values of these limits. Where possible, the right and the left side have symmetrical joint angles. Since in a joint there is generally one of the 3 DoFs more important than the others, this joint is allowed a wider motion range. The other limits are then computed according to the elliptical subspace described by the 3 DoFs. For example, consider the left shoulder subspace: the most important motion is the rotation around y-axis; x-axis and z-axis maximum rotations are then set in order to not alter the maximum y rotation. Due to the fact that PIPER Software positioning dynamics may differ from the dynamics in a full FE simulation, all limits are also tested by running simulations: 16 simulations in total are run for this verification. All limits resulted as feasible.

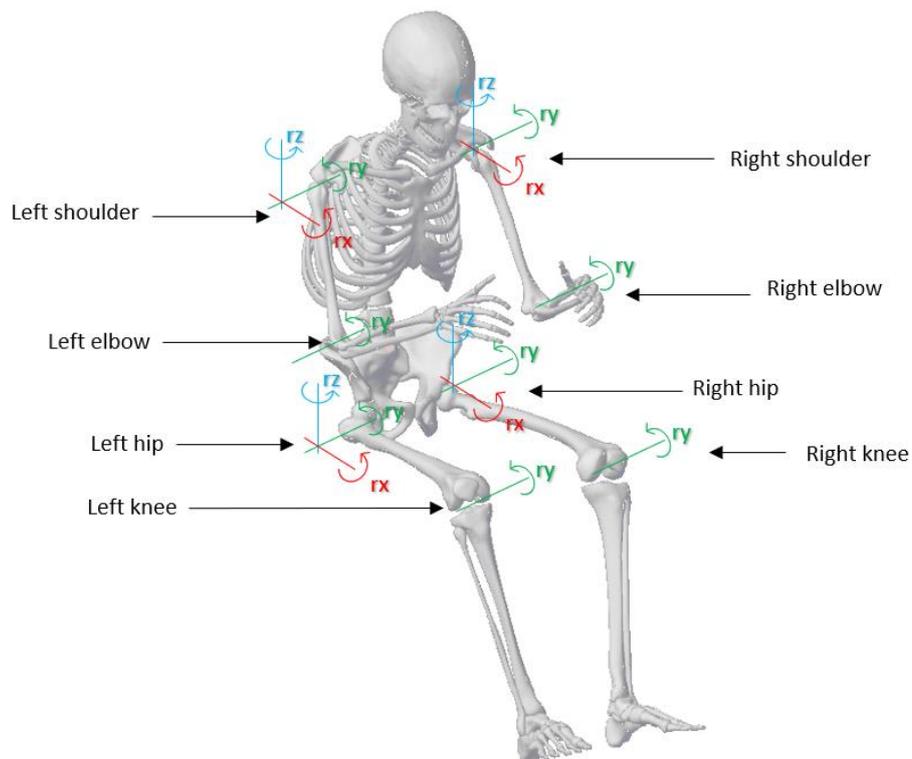


Figure 12. DoFs used for the database: 3 on each shoulder, 1 on each elbow, 3 on each hip, 1 on each knee

Table 4. Lunar parameters with corresponding PIPER joints and limits

Lunar Parameter	Corresponding PIPER Joint	Limits	
		Minimum	Maximum
L_shoulder_x	left_glenohumeral (rx)	-20°	+15°
L_shoulder_y	left_glenohumeral (ry)	-30°	+30°
L_shoulder_z	left_glenohumeral (rz)	-25°	+25°
L_elbow_y	left_humeroradial (ry)	-30°	+30°
L_hip_x	left_hip (rx)	-30°	-10°
L_hip_y	left_hip (ry)	-30°	+30°
L_hip_z	left_hip (rz)	-5°	+15°
L_knee_y	left_knee (ry)	-30°	+30°
R_shoulder_x	right_glenohumeral (rx)	-15°	+20°
R_shoulder_y	right_glenohumeral (ry)	-30°	+30°
R_shoulder_z	right_glenohumeral (rz)	-25°	+25°
R_elbow_y	right_humeroradial (ry)	-30°	+30°
R_hip_x	right_hip (rx)	+11°	+30°
R_hip_y	right_hip (ry)	-30°	+30°
R_hip_z	right_hip (rz)	-14°	+5°
R_knee_y	right_knee (ry)	-30°	+30°

To describe each limb subspace, DoE methodologies are used to describe the sample space. In this phase, CADML Lunar DoE tool is used. Each subspace is described by 8 samples, thus a total of 32 samples. The number of samples is set by balancing computational effort and expected prediction accuracy.

According to the most recent DoE theory [26], there are just a few options available to describe a 4-dimensional space with such a low number of points. Latin HyperCube method is considered to be the best fit. For the purpose of understanding this methodology, consider a 2 DoFs experiment with 4 samples. The subspace is divided into 16 squares. The 4 samples are then placed so that for each column and row there is only one sample. A possible solution is shown in Figure 13. More details on Latin HyperCube sampling methodology are available at [26]. Since this methodology does not consider possible correlations between samples, Lunar offers a modified Latin HyperCube algorithm to reduce the said correlations, called “Improved Latin HyperCube”. This algorithm also positions the samples to describe all DoFs limits [27]. The reason for this is that Lunar uses interpolating ML techniques, thus it requires all limits of the subspace to be fully described.

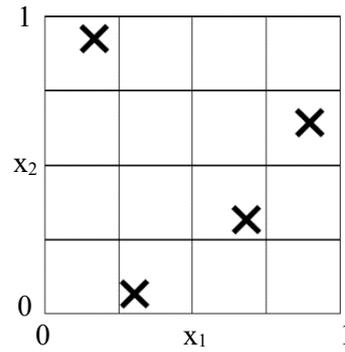


Figure 13. 2 DoFs, 4 points Latin Hypercube sampling distribution

To compute the database, for each sample a full FE simulation is run. The Classical Procedure described on page 8 is used. However, removing all limbs except one reduces the number of constraints on the body, thus the torso nodes move more than what they are supposed to. To counteract this excessive motion, SPC boundary constraints are applied on both shoulders and hips, as shown in Figure 14.

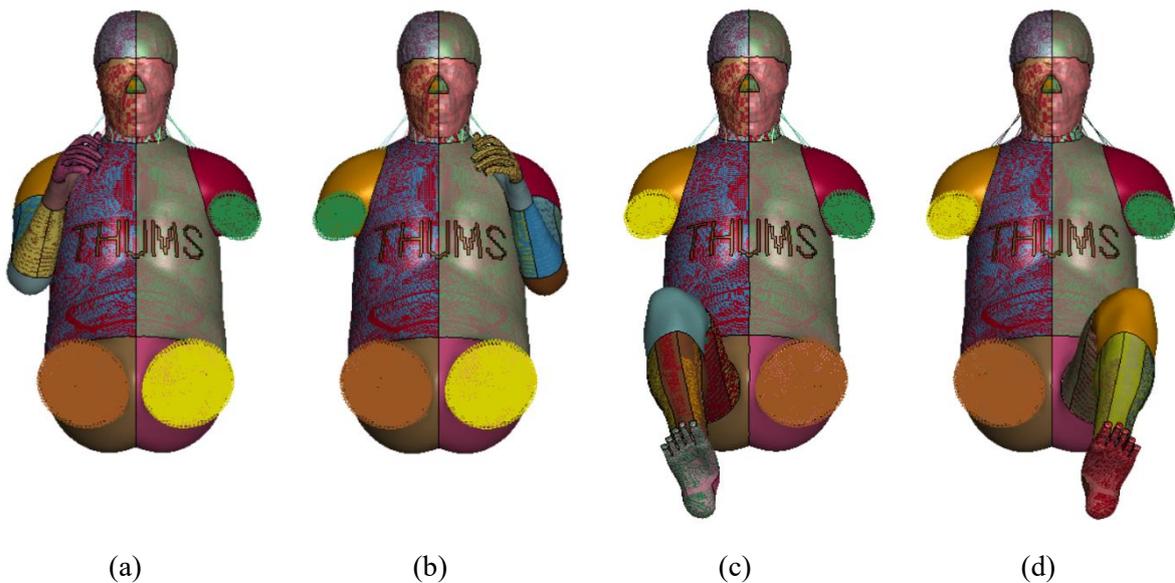


Figure 14. Boundary constraints to prevent unwanted motions: a) right upper limb database SPC configuration; b) left upper limb database SPC configuration; c) right lower limb database SPC configuration; d) left lower limb database SPC configuration

The joints controlled in PIPER Software are listed in Table 4. A CADLM Lunar DoF corresponds to each joint. The DoFs in Lunar follow a naming convention in which the first letter corresponds to the side of the body (R for right and L for left), between underscores the body part is named and the last two letters indicate the rotational axis. For example, “L_shoulder_x” controls the x rotation of the PIPER joint “left_glenohumeral”. The complete list of sample points and inputs used in PIPER Software are available in Appendix B. The reader is invited to notice that to avoid any uncontrolled motion all of the joints in a limb are controlled even if not listed in Table 4. All joints not listed in the said table are set

to 0 so that no motion occurs where none is wanted. For example, in the left lower limb 4 parameters are set as DoFs (i.e.: L_hip_x, L_hip_y, L_hip_z, L_knee_y) whereas, in PIPER Software a total of 9 joints is controlled: left_hip_rx, left_hip_ry, left_hip_rz, left_knee_ry, left_knee_rx, left_knee_rz, left_ankle_joint_rx, left_ankle_joint_ry, left_ankle_joint_rz. Only the first four are varied, whereas all of the others are set to zero.

2.4 Interpolation

Once the database is computed, since results were not accurate enough, three different interpolation methods are used to predict a new posture:

- 1) Clustering, with d3plots as inputs;
- 2) Clustering, with only nodal displacements as inputs;
- 3) InvD, with d3plots as inputs.

The first method is the only one to reduce the model order. This methodology classifies into clusters all information from the database and then evaluates a function to express the relations between clusters [27], [28]. The prediction is then a product of the said functions, therefore all clusters are influencing the interpolated point. In CADLM Lunar, clustering is used in the following formulation:

- Solver: clustering;
- Number of cluster: 0 (default value, corresponds to 2 clusters);
- Interpolation method: RBF (radial basic function).

This method allows to directly generate d3plots, which can then be used to export the new geometry in any post-processor. In this work, LS-PrePost is used.

The second approach attempts to reduce the size of data that need to be processed, thus removing all mesh information and interpolate only nodal displacements. Also in this second method clustering is used. However, using kriging to express the relations between clusters improves the prediction accuracy. In CADLM Lunar, the formulation used is:

- Solver: clustering;
- Number of Clusters: 0 (default value, corresponds to 2);
- Interpolation method: kriging;
- Basis function: constant;
- Stationary: h2.

Each limb requires three separate interpolations, one for the displacement on x-axis, one for the displacement on y-axis and one for the displacement on z-axis. The predictions are then composed to obtain the total displacement. Finally, since no d3plot is generated, an extra step is required to compute new nodal coordinates and import them into the original geometry. MATLAB scripts are used for these steps.

The last interpolation method does not reduce the model order. It is based on inverse distance weighting, a multivariate interpolation method with a set of known points [27], [29]. To evaluate an unknown point, a weighted average is computed considering the number of clusters (also known as neighbours). There is more than one way to compute the weighted average, the most common is Shepard's method. In CADLM Lunar, the formulation used is:

- Solver: InvD;
- Neighbours: 3.

This approach also directly generates d3plots. Therefore, any post-processor is then needed to export the new geometry. LS-PrePost is again used for this purpose.

Chapter 3: Results

Testing not only evaluates the accuracy of the prediction but also defines model usability. As suggested by Noriyo Ichinose et al. [3] work, nodal displacements, nodal coordinates, bones volumes and penetration between parts are the main parameter considered for the evaluation of the prediction. However, the testing phase is not complete, therefore penetrations between parts have not been thoroughly checked. Since the predictions have not been merged, the second stage of testing has not been performed at all. Consequently, the positioned THUMS is not usable yet.

A key feature to understand if the machine is not overtrained on a set is to use a testing set with more than one posture and to guarantee that none of these lasts coincides with a training sample. Consequently, each model is tested with 2 different positions, of which none is part of the training database and at least one per limb has not been used in the training phase to reduce the prediction error. In Table 5 the parameters used for testing each model are listed. Notice, that similarly to the procedure described for the database on page 16, in PIPER all DoFs of each limb are controlled to avoid any uncontrolled motion and reduce variability.

Table 5. Testing set parameters

Test ROM	Parameter	Position 1	Position 2
Right Upper Limb	R_Shoulder_x	12°	20°
	R_Shoulder_y	-25°	-9°
	R_Shoulder_z	-6°	25°
	R_Elbow_y	30°	30°
Left Upper Limb	L_Shoulder_x	-20°	15°
	L_Shoulder_y	19°	30°
	L_Shoulder_z	-25°	25°
	L_Elbow_y	17°	30°
Right Lower Limb	R_Hip_x	11°	30°
	R_Hip_y	-30°	19°
	R_Hip_z	-11°	-14°
	R_Knee_y	-30°	27°
Left Lower Limb	L_Hip_x	-30°	-13°
	L_Hip_y	-11°	-18°
	L_Hip_z	-5°	12°
	L_Knee_y	30°	-26°

3.1 Comparison Methodology

As said before, three parameters are used to evaluate the accuracy of each prediction: nodal displacements, nodal coordinates and bones volume. For each parameter, an error is defined as described below.

3.1.1 Nodal Displacement

For each node, a vector describing the total displacement is defined. To compare the accuracy of the prediction, the reference nodal displacement (the one computed with full FE simulation) is subtracted from the predicted nodal displacement. To compute a relative error, the magnitude of the said difference is divided by the maximum magnitude of the reference nodal displacement, as shown in equation (1), where \vec{d}_{prd} is the predicted nodal displacement and \vec{d}_{ref} is the reference nodal displacement. The results are then plotted on the predicted mesh and statistically analysed to understand the overall performance.

$$ERR_{disp} = \frac{|\vec{d}_{prd} - \vec{d}_{ref}|}{\max|\vec{d}_{ref}|} \quad (1)$$

Notice that some nodes have null displacement, thus dividing by $|\vec{d}_{ref}|$ instead of $\max|\vec{d}_{ref}|$ would yield to either an undetermined or impossible expression.

3.1.2 Nodal Coordinates

Nodal coordinates are also treated as vectors. The error is computed as the difference between the predicted coordinate, \vec{n}_{prd} , and the reference one, \vec{n}_{ref} , as shown in equation (2). The error magnitude is then plotted on the predicted model and statistically analysed to understand the overall performance.

$$ERR_{cord} = |\vec{n}_{prd} - \vec{n}_{ref}| \quad (2)$$

Notice that the error is not expressed in relative quantities. However, when reading these results is important to remember that a THUMS element length is between 3 to 5mm. Moreover, the nodal coordinates error must always be considered with the nodal displacement relative error, to understand if the error is biased by a positioning error or if the mesh has lost too much quality.

3.1.3 Bones Volume

Theoretically, bones volume should not change because of their material properties. However, in the full FE simulations, small variations do occur. These can become bigger in the prediction. Moreover, bones volume also highlights if there are unrealistic deformations in the model. The relative error is computed as shown in equation (3)

$$ERR_{vol} = \frac{|V_{prd} - V_{ref}|}{V_{ref}} \quad (3)$$

where V_{prd} is the bone predicted volume and V_{ref} is the bone original volume.

3.2 Comparison with Reference

Before looking into the results, it is interesting to notice the time required by each model to learn the database and then compute a prediction. The first model, since it has to decompose the model, has the longest training time and consequently the longest time to a prediction, called positioning time in Table 6. On the other hand, the second method, since it just interpolates, it does not require preliminary training. However, due to technical difficulties, precise time measurements are not available. Each positioning required around 2 minutes each. Lastly, the third method is faster than the first, as shown in Table 7. On average the third method has faster training times: on average 6 minutes and 35 seconds against 8 minutes and 10 seconds for a ROM. Positioning times are closer: 1 minute 35 seconds for a ROM meanwhile an InvD interpolation takes 1 minute and 20 seconds.

Table 6. Reduced Order Model training and positioning times

Model	Training time	Positioning time
Left Upper Limb	8 min 10.665 s	1 min 38 s
Right Upper Limb	7 min 31.499 s	1 min 30 s
Left Lower Limb	8 min 18.285 s	1 min 40 s
Right Lower Limb	8 min 32.219 s	1 min 42 s

Table 7. InvD interpolation training and positioning times

Model	Training time	Positioning time
Left Upper Limb	6 min 32.679 s	1 min 27 s
Right Upper Limb	6 min 31.124 s	1 min 29 s
Left Lower Limb	6 min 45.891 s	47 s
Right Lower Limb	6 min 26.966 s	1 min 25 s

3.2.1 ROM

The overall performance of this methodology is poor. As can be seen in Table 8, the predicted nodal displacement has generally a large relative error. The maximum error is 53.87 %, whereas the mean relative error is 18.83%.

In most cases, also parts that are supposed to be constrained seem to have some displacement. This is highlighted by the contour and the bar graphs in Appendix C. The reason for the said displacements is a scaling effect that resizes the model, or parts of it, as shown in Figure 15. The bones volume gives an idea of how much a model is scaled: most bones have a smaller volume than their reference. Using the relations between predicted and reference bones volume one could be tempted to rescale the entire model. However, this solution has two major flaws: the first is that, as shown in Figure 15 – a, if only part of the model is scaled, it is difficult to describe the smooth transition between scaled and not-scaled parts. The second flaw is the need for a reference for each positioning since the scaling factor is not constant.

CADLM has declared that the scaling effect is as of now a problem of the Lunar commercial software, linked to the failure definition of some materials. They also declared that this issue is being fixed, thus, in a future release, the scaling factor should no longer be present.

Table 8. Relative Error statistical analysis

Limb	Position #	Minimum Relative Error [%]	Maximum Relative Error [%]	Average Relative Error [%]	Standard Deviation Relative Error [%]
Right Upper Limb	1	0.05	76.00	19.88	11.42
Right Upper Limb	2	0.09	74.95	39.16	21.18
Left Upper Limb	1	0	7.27	0.30	0.95
Left Upper Limb	2	0	52.51	0.94	4.73
Right Lower Limb	1	0.07	57.80	20.01	11.33
Right Lower Limb	2	0.05	59.04	27.59	13.99
Left Lower Limb	1	0.05	66.95	27.13	14.18
Left Lower Limb	2	0.05	36.40	15.62	8.27
Average		0.05	53.87	18.83	10.76

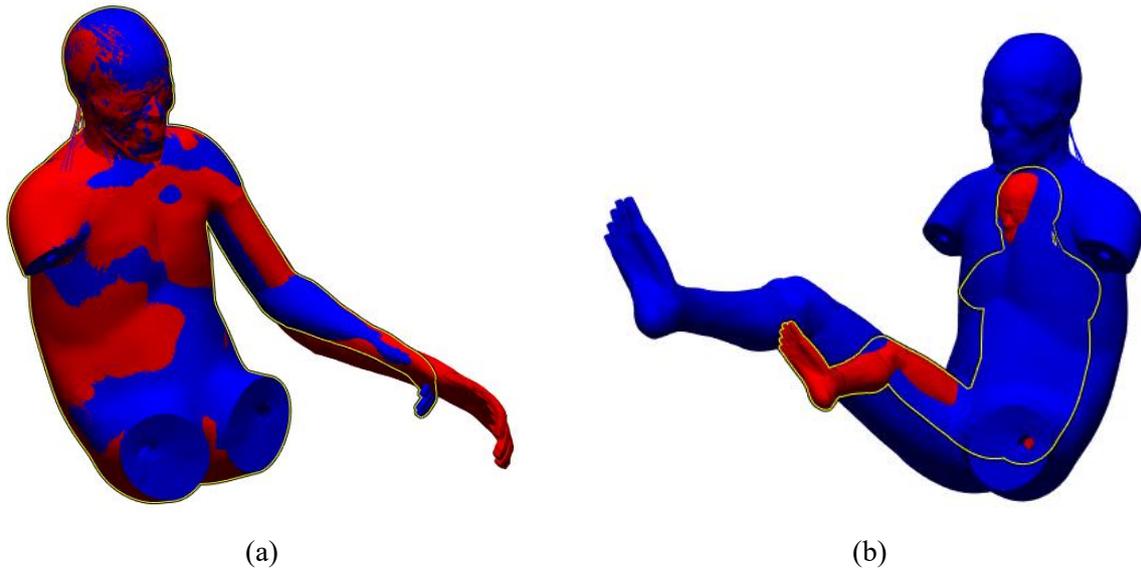


Figure 15. Comparison between reference model (blue) and predicted model (red): a) only the left arm is upscaled; b) the entire model is downscaled

3.2.2 Nodal Displacements Interpolation

In the second methodology, the displacement prediction has greatly improved as shown by the values in Table 9. Although the maximum relative error has increased on average, both the standard deviation and the mean value have decreased respectively to 9.05 % and 3.29 %. This, along with the contour plots and bar graphs in Appendix D, proves that most nodes are much closer to their reference.

Nevertheless, all predictions present unrealistic deformations in the limbs, like the example shown in Figure 16. Moreover, as shown in Figure 17, also well-predicted parts, such as the torso, have very low quality because of penetrations between parts. The root cause of these penetrations is that this interpolation model does not have any information on the existing relations between all nodes: there are no rules for reconstructing the mesh. Consequently, even a small error deeply alters the mesh quality because it can result in penetration between two parts. This methodology, however, helps to prove that without decomposing the model, there is no scaling factor affecting the prediction.

Table 9. Relative Error statistical analysis

Limb	Position #	Minimum Relative Error [%]	Maximum Relative Error [%]	Average Relative Error [%]	Standard Deviation Relative Error [%]
Right Upper Limb	1	0	83.91	2.60	7.03
Right Upper Limb	2	0	95.11	2.18	8.40
Left Upper Limb	1	0	28.32	1.05	3.05
Left Upper Limb	2	0	111.90	2.69	10.90
Right Lower Limb	1	0	76.16	5.76	13.28
Right Lower Limb	2	0	89.49	5.52	14.12
Left Lower Limb	1	0	68.96	3.98	9.39
Left Lower Limb	2	0	35.13	2.53	6.26
Average		0	73.62	3.29	9.05

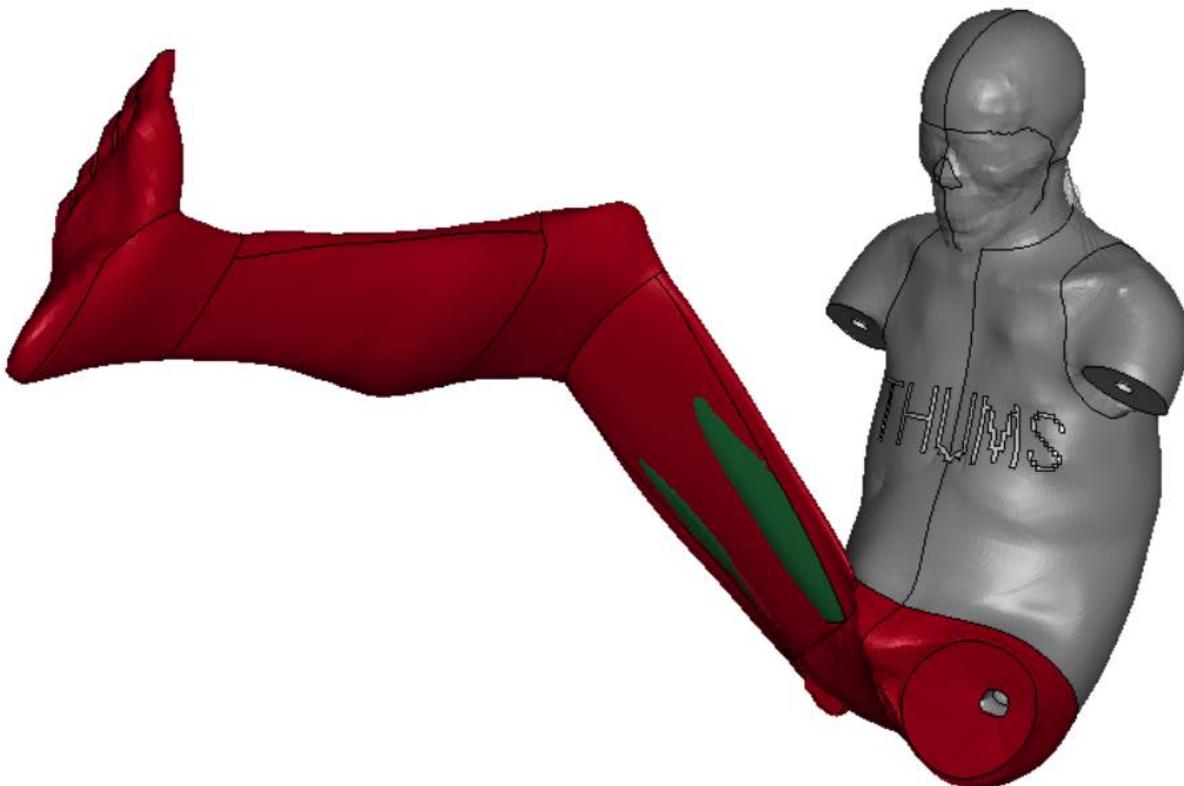


Figure 16. Example of excessive bad prediction: in red unrealistic deformations; in green visible penetrating part

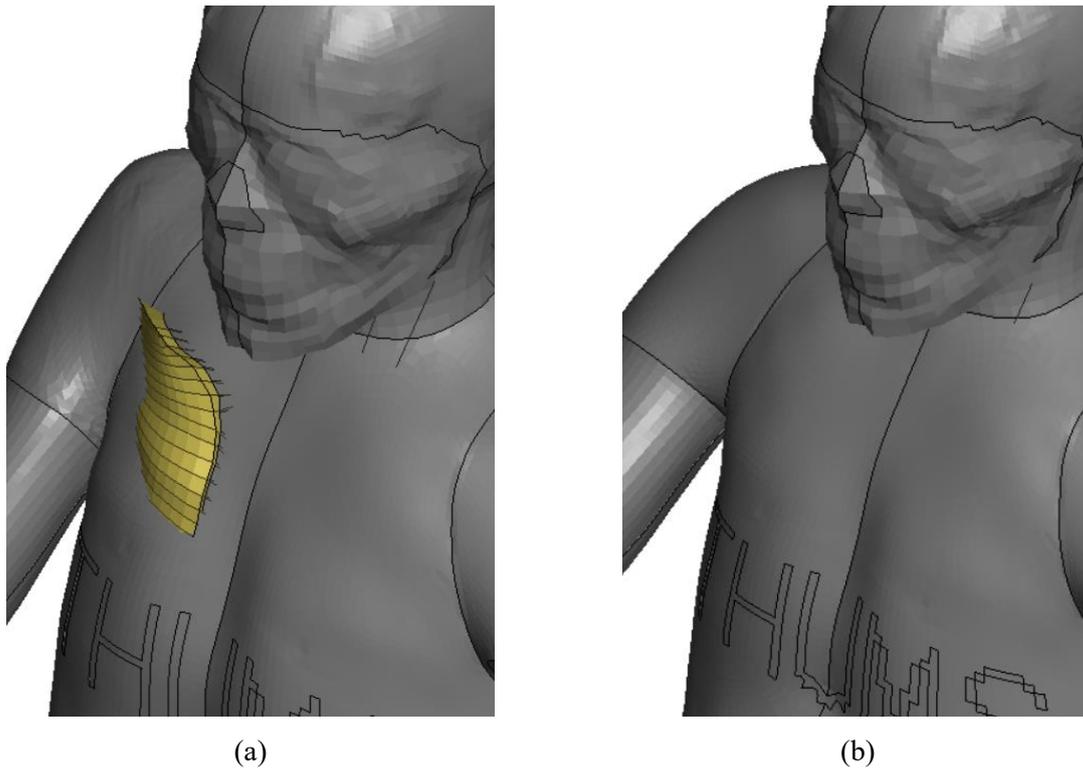


Figure 17. Example of penetration: a) right muscle penetrating the torso skin; b) reference

3.2.3 InvD Interpolation

The last methodology has the best performance, despite still presenting values for the maximum error on average higher than the first methodology, as can be seen in Table 10. However, the average error decreases furthermore to 2.62% as well as the standard deviation (8.31 %). This accuracy increase is the result of having apparently solved not only the scaling problem but also both the unnatural deformations and the penetration. Contour plots and bar graphs in Appendix E confirm the previous statement.

On the downside, as highlighted by the contour plots in Appendix E, the accuracy lowers in the limbs. As shown in Figure 18, the cause for this is the difficulty for the prediction to match the position given as input. Consider wanting an angle of 20° on the elbow, the prediction is not capable of matching the request, thus the smaller angle at the elbow would be smaller. The effect of this angular mistake accentuates at the tips of the limbs.

Since InvD only interpolates, without any order reduction, the model generated extrapolates less information from a single sample. The database used is therefore too little to provide an accurate prediction.

Table 10. Relative Error statistical analysis

Limb	Position #	Minimum Relative Error [%]	Maximum Relative Error [%]	Average Relative Error [%]	Standard Deviation Relative Error [%]
Right Upper Limb	1	0	69.06	1.74	7.37
Right Upper Limb	2	0	60.28	1.55	6.72
Left Upper Limb	1	0	48.57	1.10	5.14
Left Upper Limb	2	0	76.25	1.36	7.13
Right Lower Limb	1	0	72.68	4.24	11.87
Right Lower Limb	2	0	74.11	4.35	11.40
Left Lower Limb	1	0	32.10	2.59	5.99
Left Lower Limb	2	0	64.47	4.05	10.84
Average		0	62.19	2.62	8.31

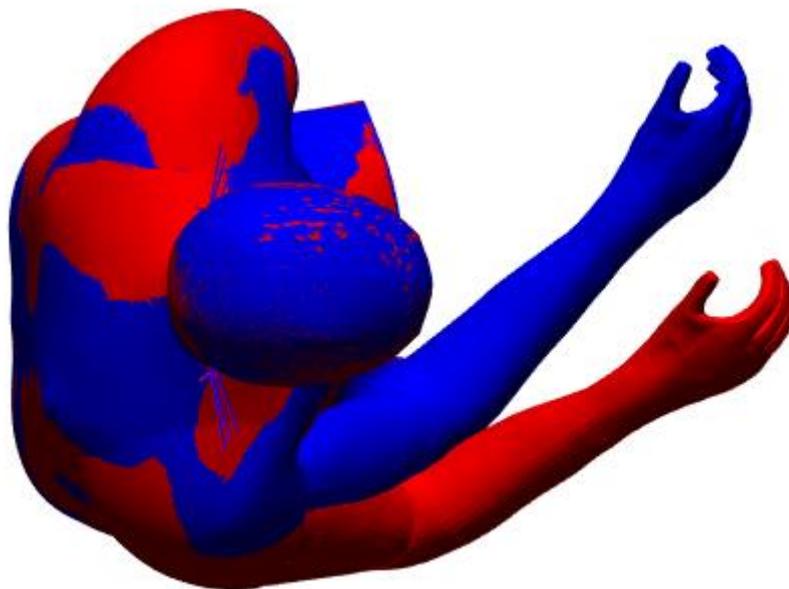


Figure 18. Angle difference on the right arm: predicted model in red; reference model in blue

Conclusion

Positioning an HBM is a demanding process in terms of both time and computing power. Considering the corresponding costs, it is understandable that developing new safety features in cars can be limited also by the number of occupant positions manufacturers test. Therefore, reducing the costs of positioning an HBM can enlarge the possibilities for developers in the vehicle safety field.

As discussed previously, there is more than one approach to position an HBM, however, the ML-based one has proven to be the quickest. This work tries to replicate one of the ML-based methodologies. Despite not being finished for technical difficulties, it is already possible to analyse some benefits and drawbacks of developing such a methodology.

On the bright side, the improvements that each iteration have brought to the work are a sign that a good result can be achieved. If this should actually be the case, the entire positioning sequence would take less than 15 minutes in total on the user's personal computer. The trade-off is reducing the fidelity with the human body. Depending on the application this loss can be acceptable or not. In any case, having set the target maximum relative error to 3 %, most applications can use the predicted model, as long as there are no penetrations between parts. With future improvements of the interpolating algorithms, the prediction usability will become bigger and bigger.

On the less bright side, the creation of this model requires a lot of resources. Up to this stage of the development, to create the database and test the predictions a total of 56 simulations were run: 16 to test the limbs limits, 32 to compute the database sample and 8 to compute the testing set. However, since the database is not big enough, the total number of simulations must be increased. Moreover, the testing set is very limited and it does not manage to characterize the entire prediction subspace, increasing furthermore the number of simulations required. Ultimately, considering the resources required until now, creating such a model could result in a financial loss if a company does not need to position the HBM often. A way of making this ML-approach more affordable and increase its usage is by sharing the database and the interpolating algorithms. One way to do so is by integrating this methodology in a computer program already widely spread among users, like PIPER Software. This not only makes the model available to a wider range of users, but it can also yield to develop a model reduction specific to the HBM, thus further improving the machine performance.

Another point of weakness of this work is the testing set used. It is not big enough to characterize the machine performance over the entire subspace. Therefore, the performance is evaluated only where the worst conditions are thought to be. However, it is not verified if in better conditions the machine effectively performs better. To properly validate the model, testing must be performed more extensively. A good practice would be subdividing the subspace and characterize with at least one test each region and map the subspace with the results obtained.

To sum up, ML is a tool that in the right hands has great potential. Like all tools, it requires dedication and time to develop the right experience to obtain good results. On top of that, ML must be used carefully. With the loss of accuracy, a ROM could end up not satisfying the quality requirements of a job, yielding inaccurate results. A lot of resources have to be spent not only in creating the model but also in training developers on ML. A small company can lack the resources to invest in the development of this model. Sharing not only the HBM but also its ROM could benefit a lot of users, allowing quicker development in a field as important as vehicle safety.

Appendix A Positioning input for target positions

	Position 1			
	Pelvis Rotation			
Reference Frame	rx [°]	ry [°]	rz [°]	k
W_Origin_to_B_pelvic_Skeleton	0.0	10.0	0.0	5e+9
Lower Limbs				
Joints	rx [°]	ry [°]	rz [°]	k
Left_hip	-15.0	5.0	13.0	1e+9
Left_knee	0.0	15.0	0.0	1e+9
Right_hip	10.0	9.0	-20.0	1e+9
Right_knee	0.0	-10.0	0.0	1e+9
Right Upper Limb				
Joints	rx [°]	ry [°]	rz [°]	k
Right_glenohumeral	-29.0	25.0	-1.0	1e+9
Right_humeroradial	10.0	0.0	10.0	1e+9
Right_radiocarpal_joint	58.0	-1.0	0.0	1e+8
Landmarks	x [mm]	y [mm]	z [mm]	k
shoulderlr	*	203.8	*	1e+8
Left Upper Limb				
Joints	rx [°]	ry [°]	rz [°]	k
Left_humeroradial	0.0	20.0	0.0	1e+8
Left_radiocarpal_joint	15.0	-23.0	5.0	1e+8
Landmarks	x [mm]	y [mm]	z [mm]	k
Lateral_epicondyle_of_left_humerus	*	-302.9	81.1	1e+8
wrist_circuml	*	-270.1	72.3	1e+8
Head				
Reference Frame	rx [°]	ry [°]	rz [°]	k
Pelvic_frame_to_B_Skull	-10.0	-20.0	-10.0	1e+8

* parameter not controlled.

Position 2				
Pelvis Rotation				
Reference Frame	rx [°]	ry [°]	rz [°]	k
W_Origin_to_B_pelvic_Skeleton	0.0	10.0	0.0	5e+9
Lower Limbs				
Joints	rx [°]	ry [°]	rz [°]	k
Left_hip	-15.0	5.0	13.0	1e+9
Left_knee	0.0	15.0	0.0	1e+9
Right_hip	10.0	9.0	-20.0	1e+9
Right_knee	0.0	-10.0	0.0	1e+9
Right Upper Limb				
Joints	rx [°]	ry [°]	rz [°]	k
Right_glenohumeral	-9.0	-29.0	15.0	1e+8
Right_humeroradial	0.0	-55.0	0.0	1e+9
Right_radiocarpal_joint	6.0	18.0	-26.0	1e+8
Landmarks	x [mm]	y [mm]	z [mm]	k
upperarmr_proximal	*	*	262.5	1e+8
Lateral_epicondyle_of_right_humerus	*	267.2	289.3	1e+8
Left Upper Limb				
Joints	rx [°]	ry [°]	rz [°]	k
Left_humeroradial	0.0	42.0	-15.0	1e+9
Left_radiocarpal_joint	-18.0	-22.0	-1.0	1e+8
Left_glenohumeral	24.0	3.0	7.0	1e+9
Landmarks	x [mm]	y [mm]	z [mm]	k
wrist_circuml	*	-347.4	126.0	1e+8

* parameter not controlled.

Appendix B Database Positions

Right Upper Limb

Table 11. Right Upper Limb Database: PIPER Software inputs

Joint	Shoulder			Elbow			Wrist		
	Right_glenohumeral			Right_humeroradial			Right_radiocarpal_joint		
DoF	rx	ry	rz	rx	ry	rz	rx	ry	rz
Position 1	10	13	4	0	-30	0	0	0	0
Position 2	20	-13	-11	0	4	0	0	0	0
Position 3	-5	-30	11	0	-13	0	0	0	0
Position 4	0	30	-25	0	-21	0	0	0	0
Position 5	15	21	-4	0	-4	0	0	0	0
Position 6	-10	-4	-11	0	30	0	0	0	0
Position 7	5	4	25	0	13	0	0	0	0
Position 8	-15	-21	18	0	21	0	0	0	0

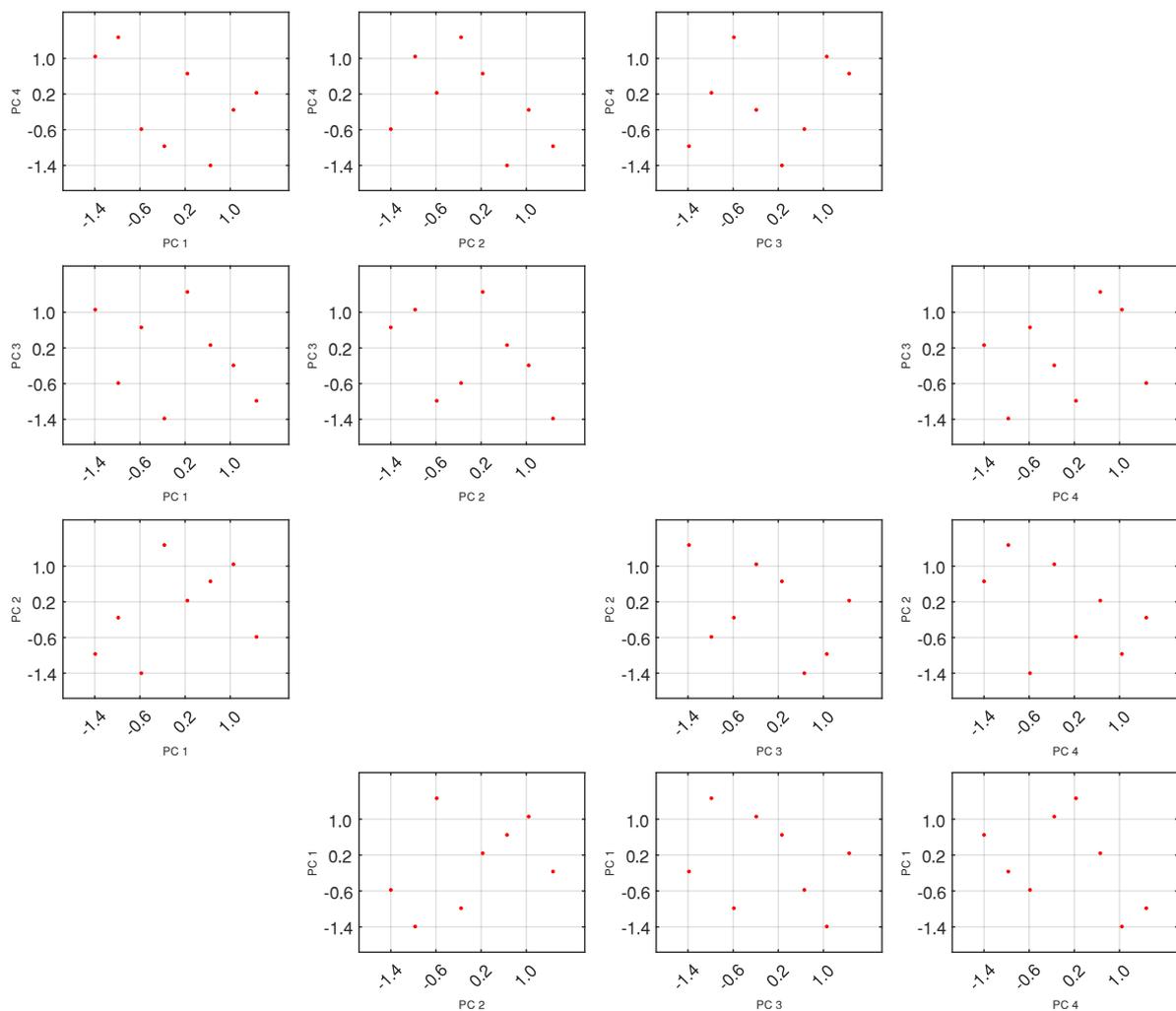


Figure 19. Right Upper Limb Database: Principal Component Analysis Plot

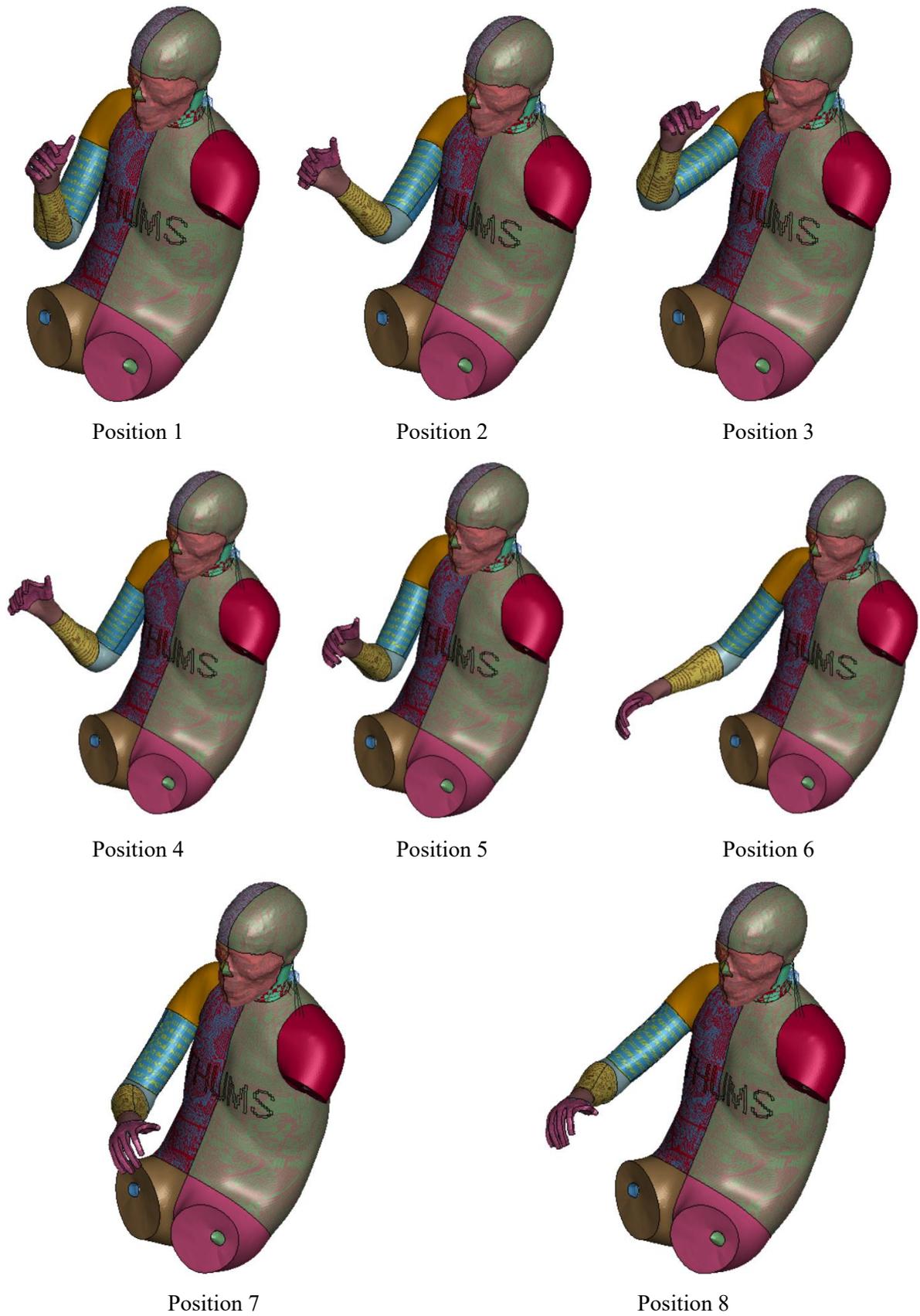


Figure 20. Right Upper Limb Database: Positioning Simulation Results

Left Upper Limb

Table 12. Left Upper Limb Database: PIPER Software inputs

Joint	Shoulder			Elbow			Wrist		
	Left_glenohumeral			Left_humeroradial			Left_radiocarpal_joint		
DoF	rx	ry	rz	rx	ry	rz	rx	ry	rz
Position 1	5	13	4	0	-30	0	0	0	0
Position 2	15	-13	-18	0	4	0	0	0	0
Position 3	-10	-30	11	0	-13	0	0	0	0
Position 4	-5	30	-25	0	-21	0	0	0	0
Position 5	10	21	-4	0	-4	0	0	0	0
Position 6	-15	-4	-11	0	30	0	0	0	0
Position 7	0	4	25	0	13	0	0	0	0
Position 8	-20	-21	18	0	21	0	0	0	0

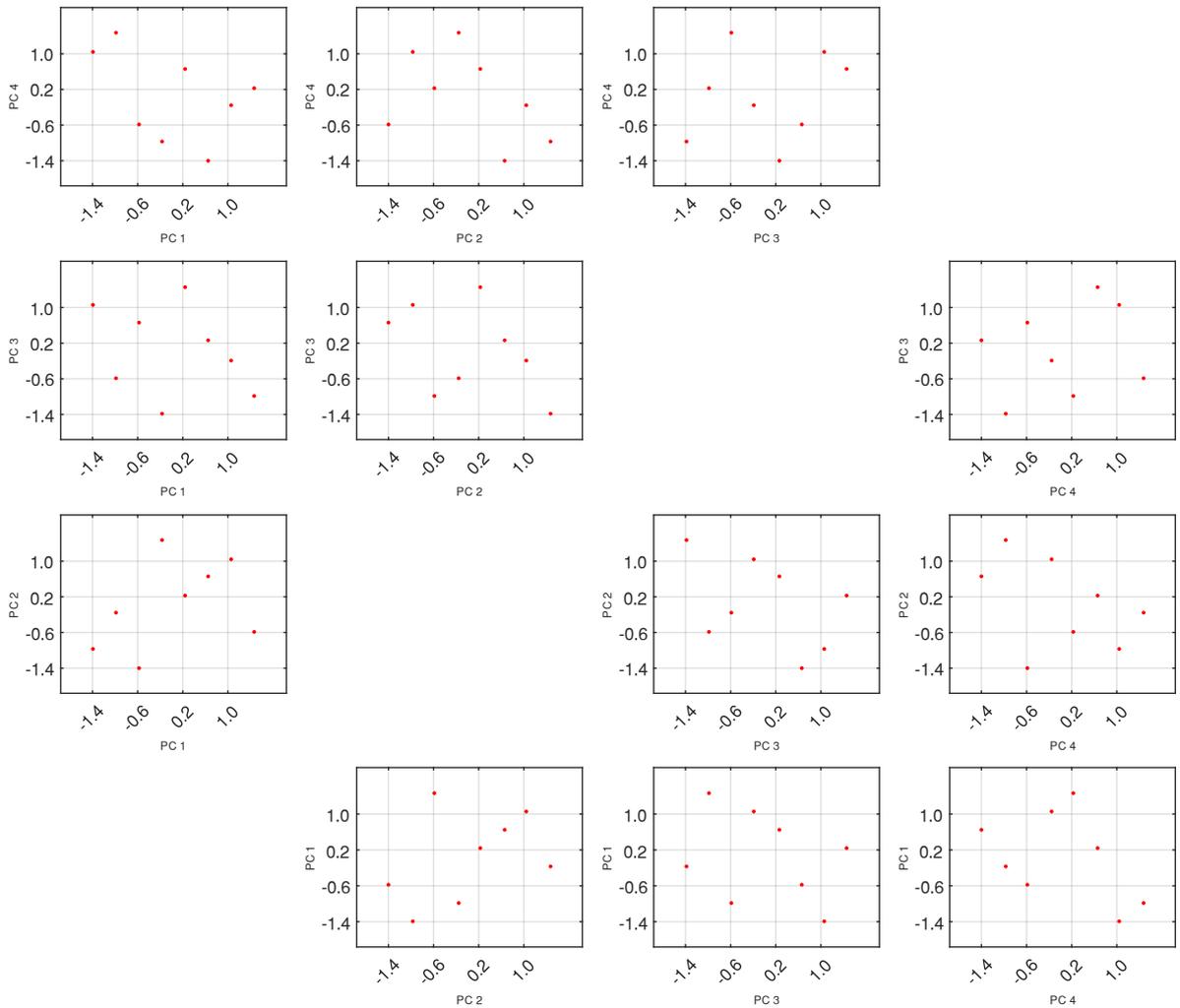


Figure 21. Left Upper Limb Database: Principal Component Analysis Plot

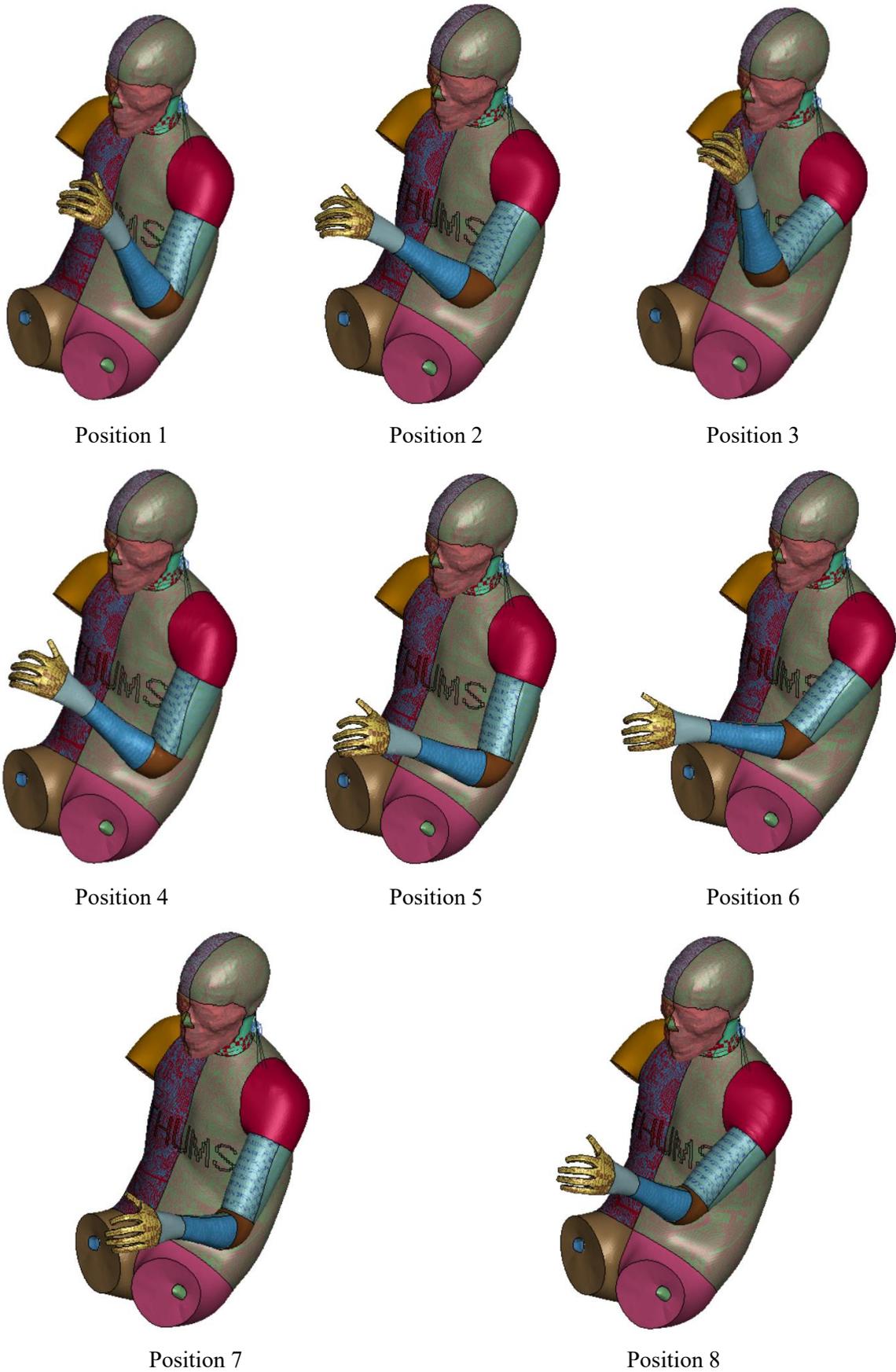


Figure 22. Left Upper Limb Database: Positioning Simulation Results

Right Lower Limb

Table 13. Right Lower Limb Database: PIPER Software inputs

Joint	Hip			Knee			Ankle		
	Righ_hip			Right_knee			Right_ankle_joint		
DoF	rx	ry	rz	rx	ry	rz	rx	ry	rz
Position 1	25	13	-3	0	-30	0	0	0	0
Position 2	30	-13	-11	0	4	0	0	0	0
Position 3	16	-30	0	0	-13	0	0	0	0
Position 4	19	30	-14	0	-21	0	0	0	0
Position 5	27	21	-6	0	-4	0	0	0	0
Position 6	14	-4	-9	0	30	0	0	0	0
Position 7	22	4	5	0	13	0	0	0	0
Position 8	11	-21	2	0	21	0	0	0	0

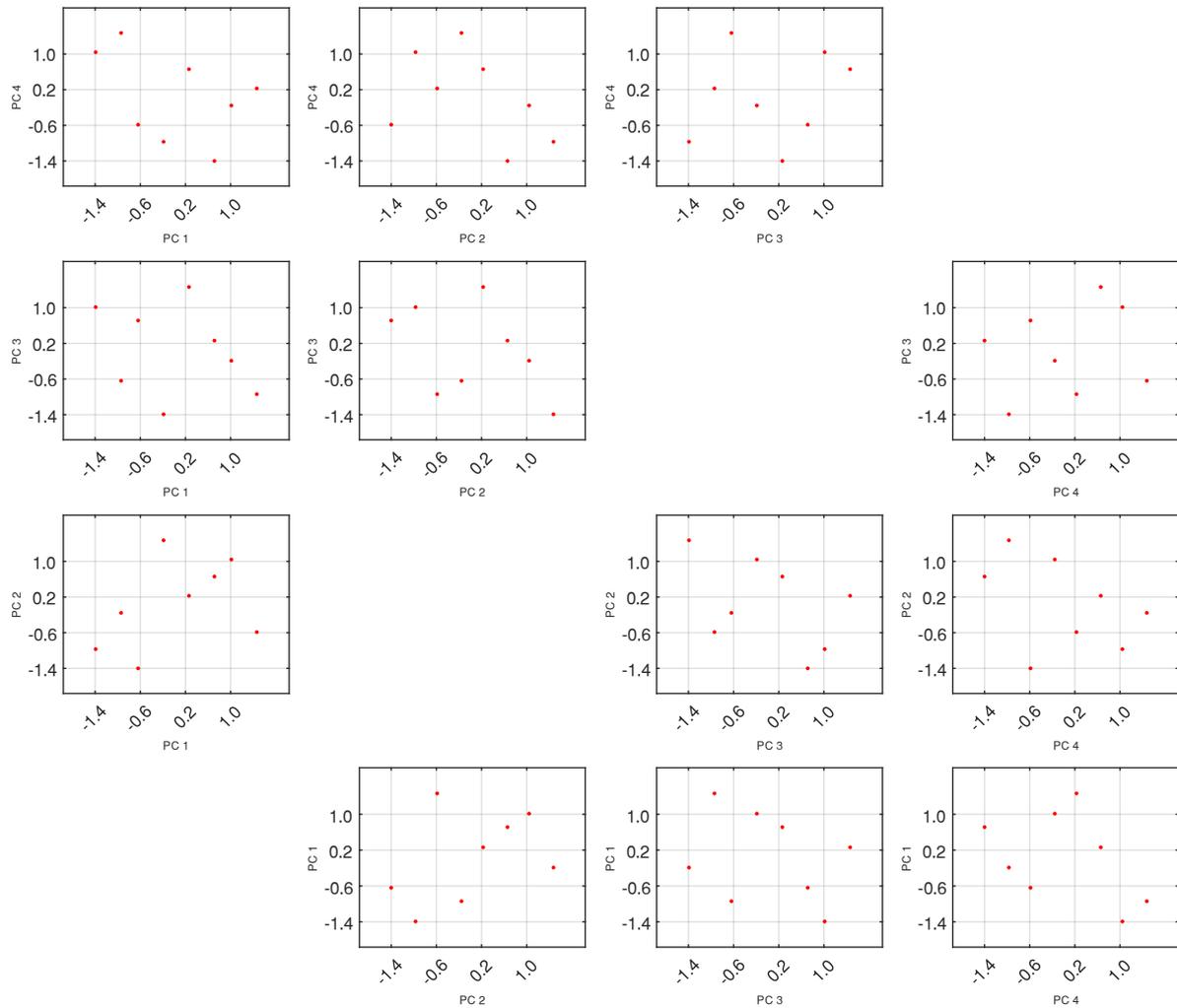


Figure 23. Right Lower Limb Database: Principal Component Analysis Plot

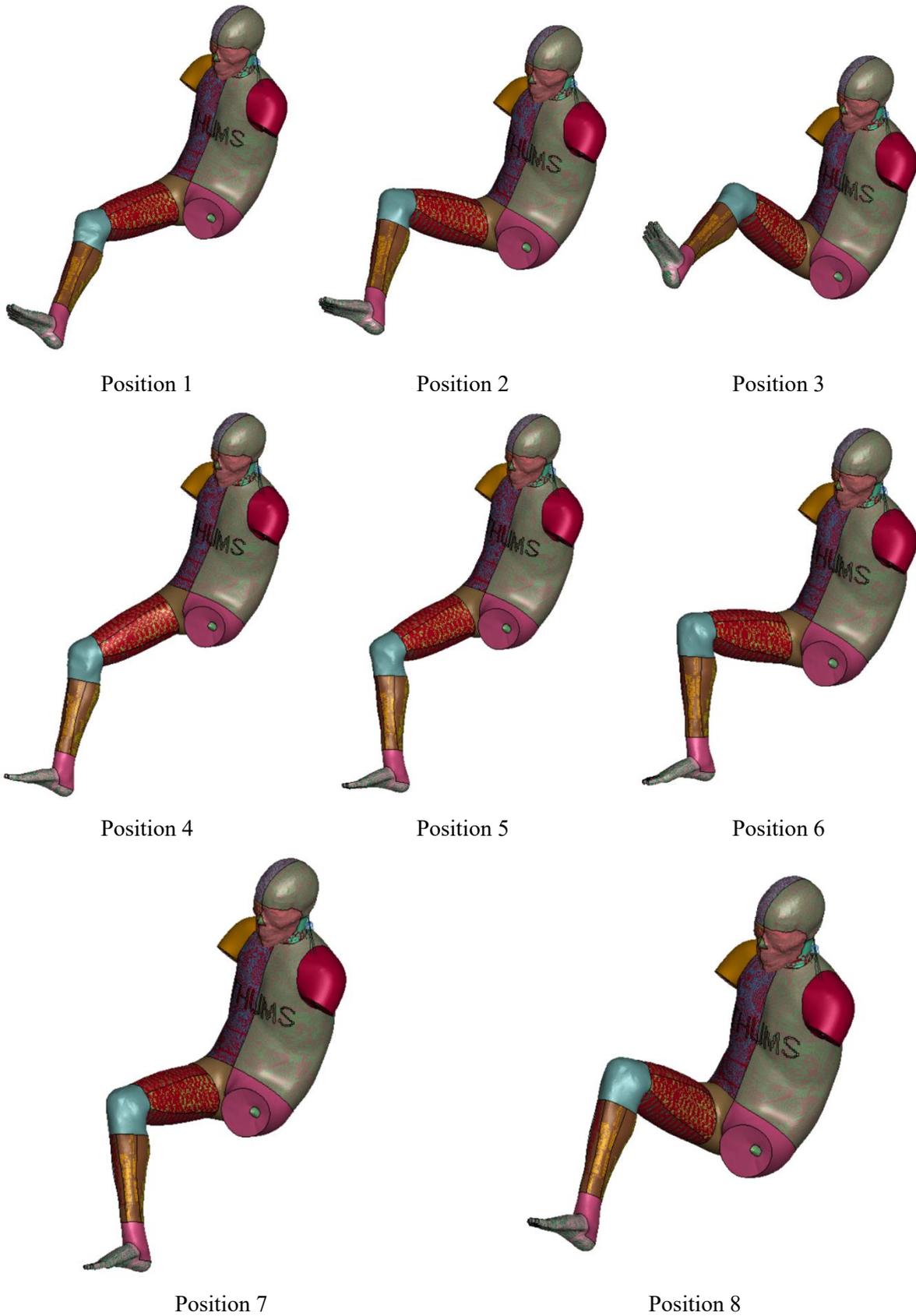


Figure 24. Right Lower Limb Database: Positioning Simulation Results

Left Lower Limb

Table 14. Left Lower Limb Database: PIPER Software inputs

Joint	Hip			Knee			Ankle		
	Left_hip			Left_knee			Left_ankle_joint		
DoF	rx	ry	rz	rx	ry	rz	rx	ry	rz
Position 1	-16	13	6	0	-30	0	0	0	0
Position 2	-10	-13	-2	0	4	0	0	0	0
Position 3	-24	-30	9	0	-13	0	0	0	0
Position 4	-21	30	-5	0	-21	0	0	0	0
Position 5	-13	21	4	0	-4	0	0	0	0
Position 6	-27	-4	1	0	30	0	0	0	0
Position 7	-29	4	15	0	13	0	0	0	0
Position 8	-30	-21	12	0	21	0	0	0	0

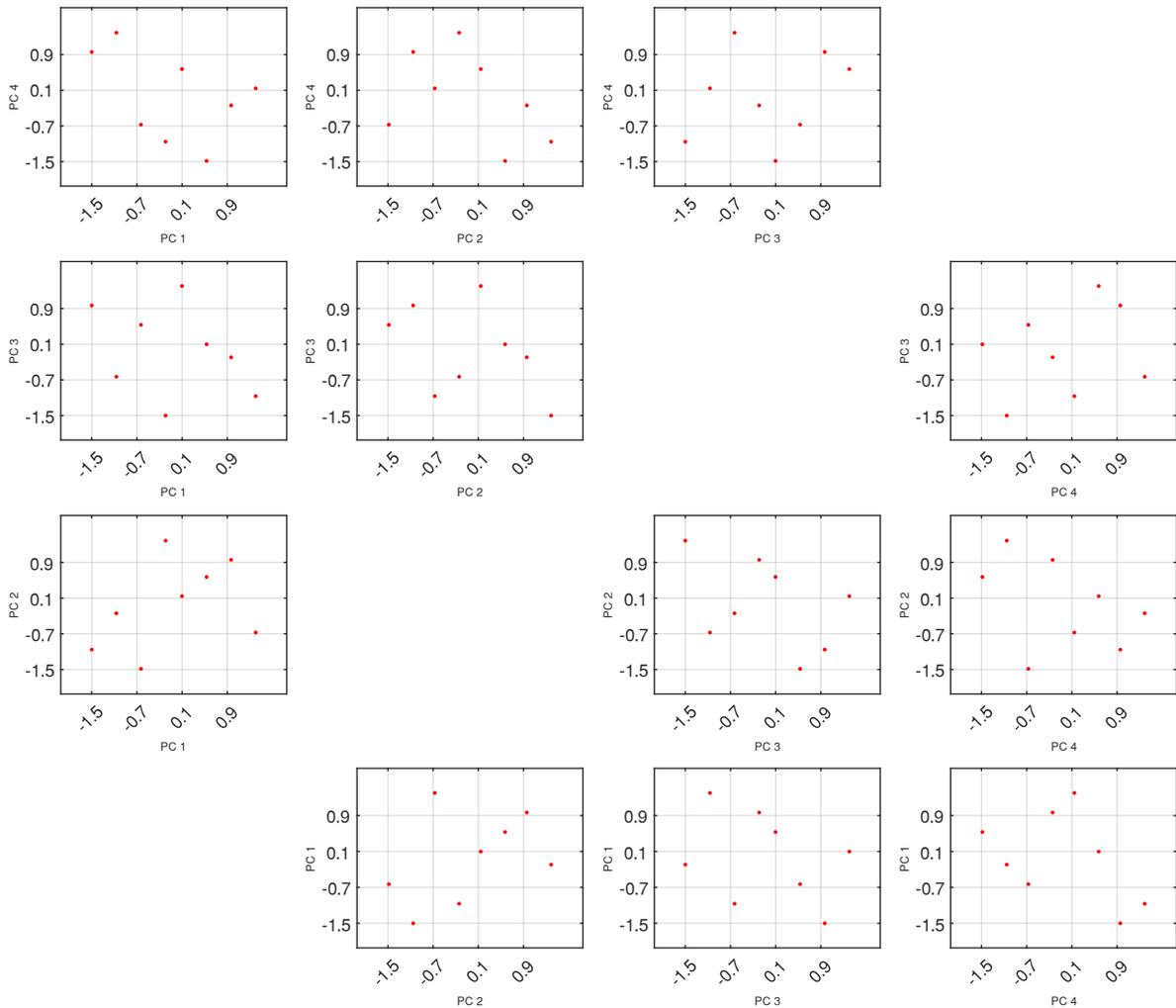


Figure 25. Left Lower Limb Database: Principal Component Analysis Plot

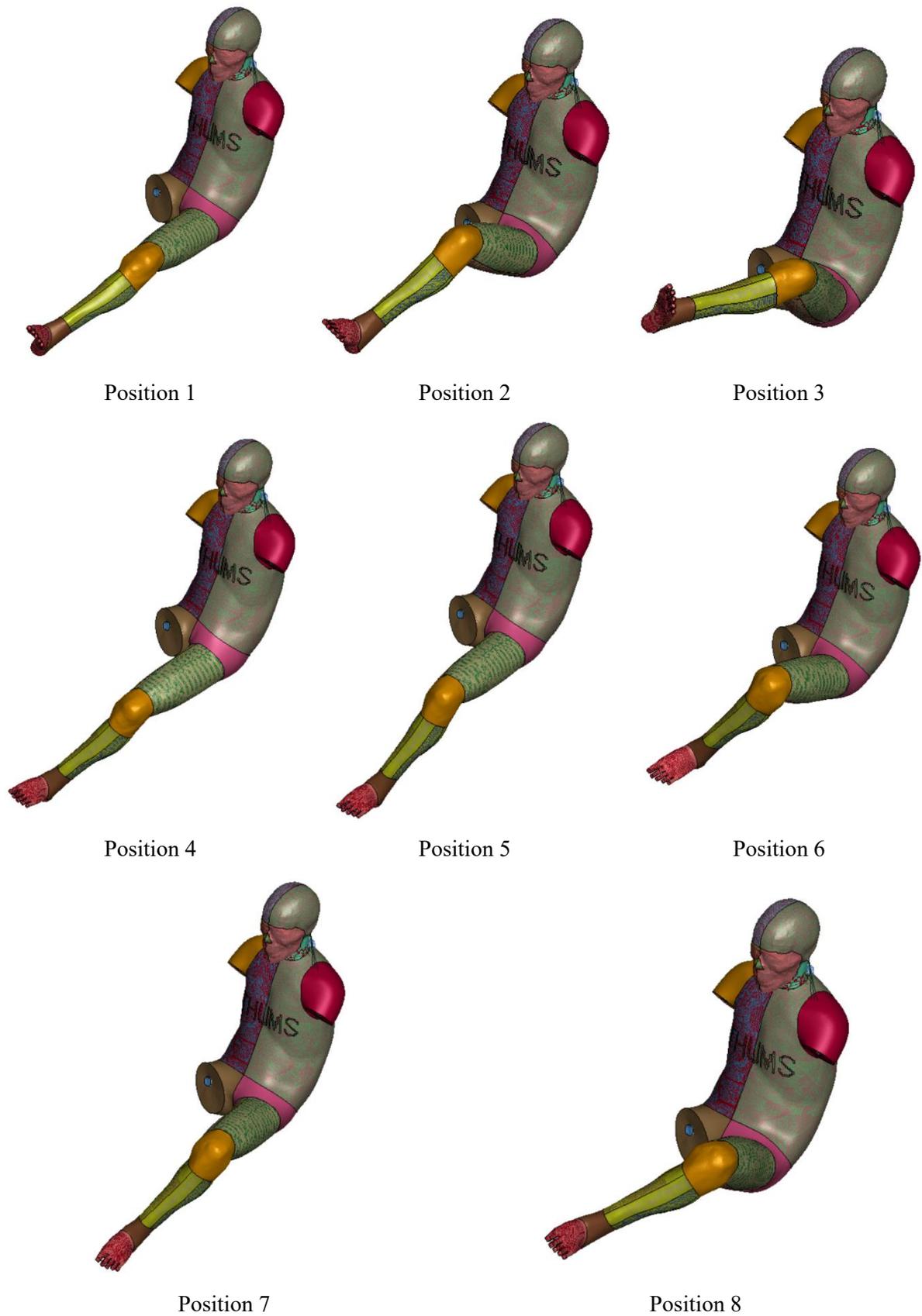


Figure 26. Left Lower Limb Database: Positioning Simulation Results

Appendix C ROM Test Results

Right Upper Limb Position 1

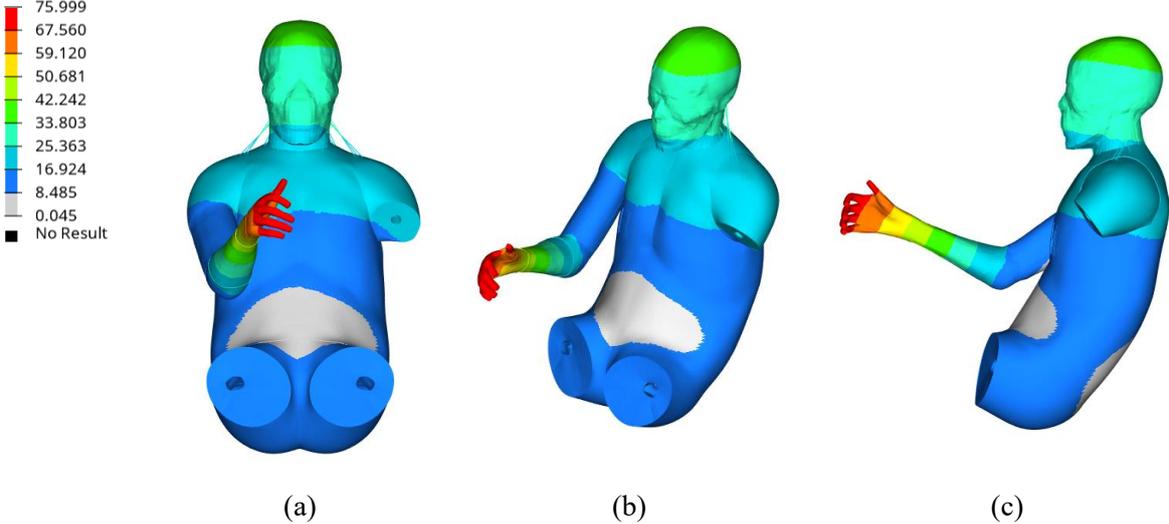


Figure 27. Right Upper Limb Position 1, nodal displacement relative error plots: a) front view; b) iso view; c) top view

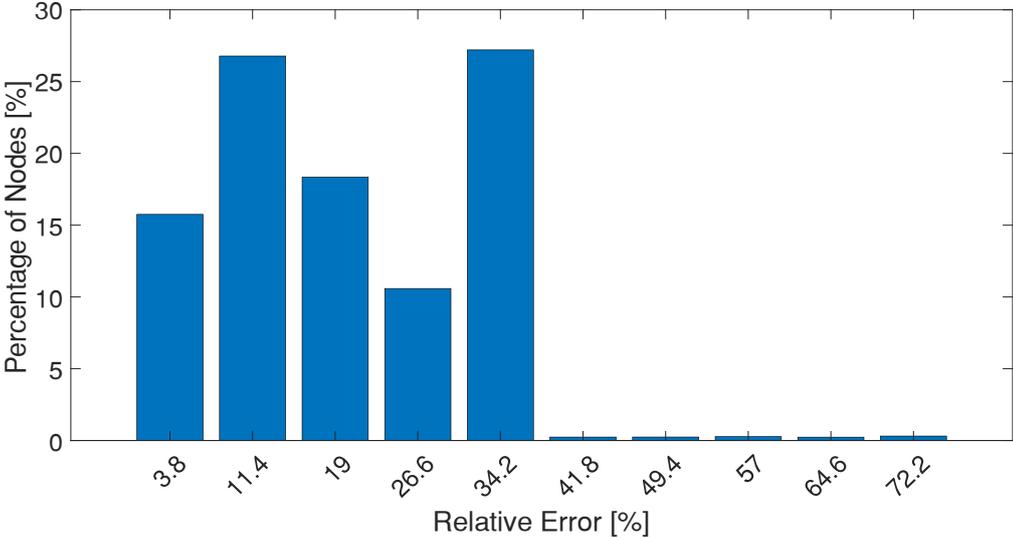


Figure 28. Right Upper Limb Position 1, nodal displacement relative error bar graph

Table 15. Right Upper Limb Position 1, nodal displacement relative error statistical analysis

	Value
Minimum	0.045 %
Maximum	75.999 %
Average	19.8836 %
Standard Deviation	11.4171 %

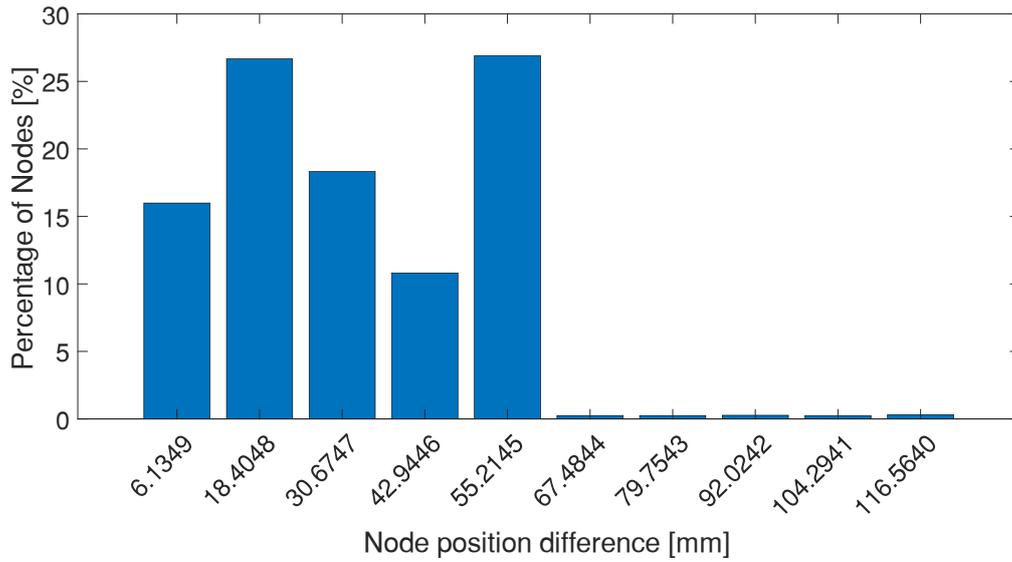


Figure 29. Right Upper Limb Position 1, nodal coordinates error bar graph

Table 16. Right Upper Limb Position 1, nodal coordinates error statistical analysis

	Value
Minimum	0.0731 mm
Maximum	122.6990 mm
Average	32.1066 mm
Standard Deviation	18.4345 mm

Table 17. Right Upper Limb Position 1, bones volume relative error

Bone	Reference Volume [mm³]	Predicted Volume [mm³]	Relative Error [%]
Humerus	8.48 e+04	5.58 e+04	34.19
Radius	1.93 e+04	1.08 e+04	44.15

Right Upper Limb Position 2

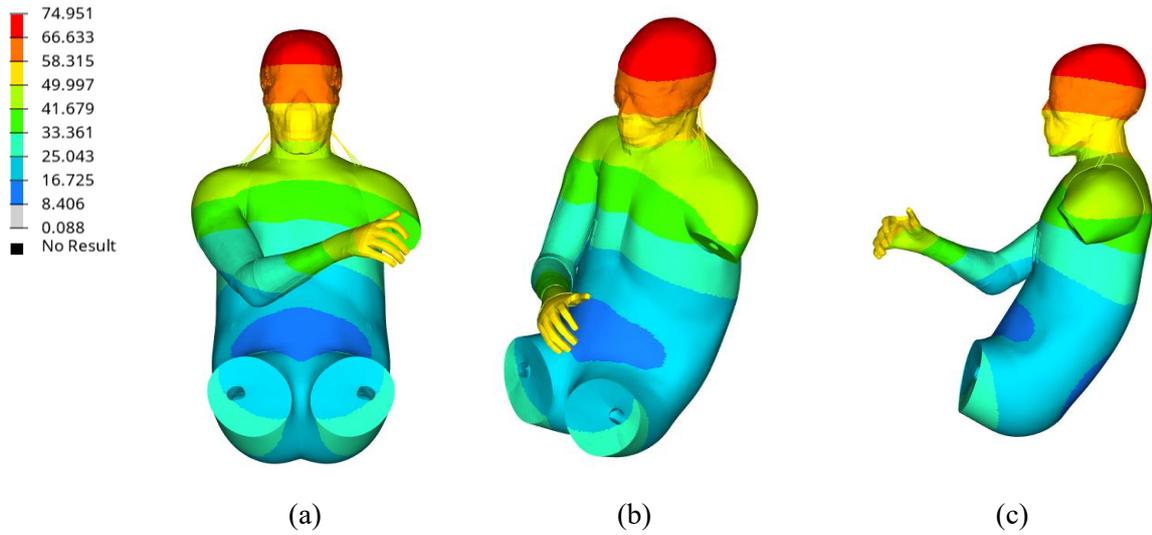


Figure 30. Right Upper Limb Position 2, nodal displacement relative error plots: a) front view; b) iso view; c) top view

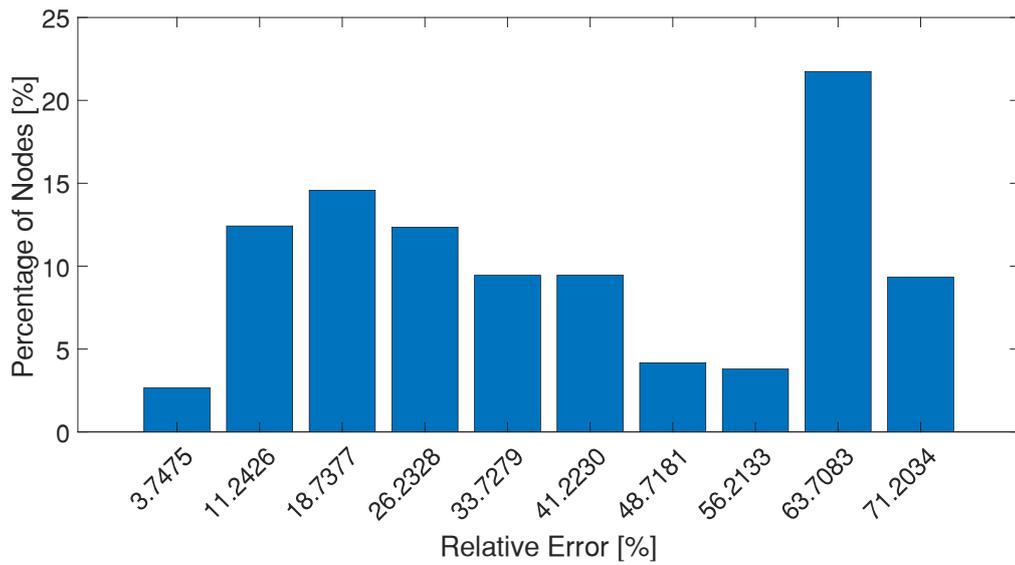


Figure 31. Right Upper Limb Position 2, nodal displacement relative error bar graph

Table 18. Right Upper Limb Position 2, nodal displacement relative error statistical analysis

	Value
Minimum	0.088 %
Maximum	74.951 %
Average	39.1575 %
Standard Deviation	21.1823 %

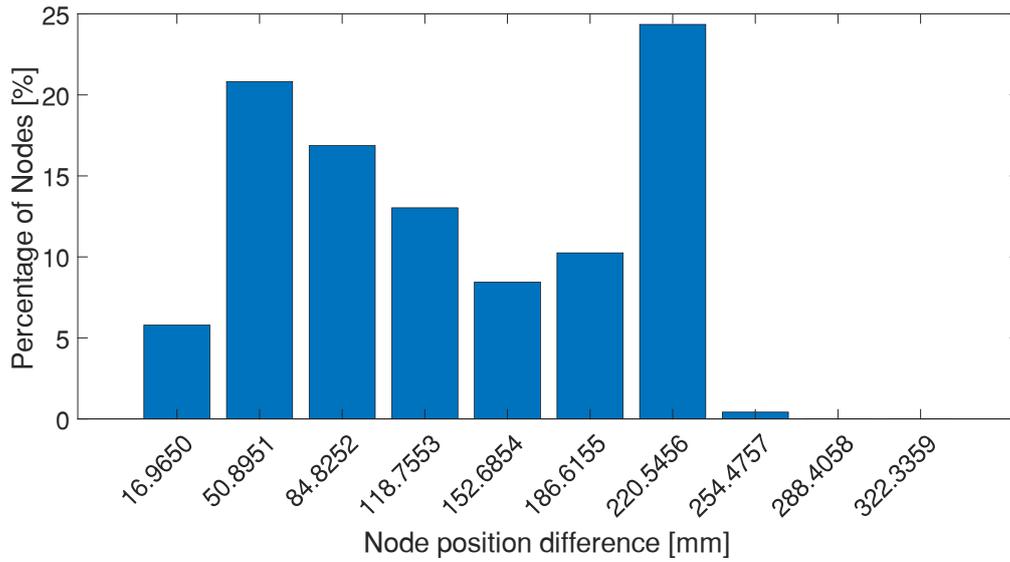


Figure 32. Right Upper Limb Position 2, nodal coordinates error bar graph

Table 19. Right Upper Limb Position 2, nodal coordinates error statistical analysis

	Value
Minimum	0.2893 mm
Maximum	339.3010 mm
Average	128.1903 mm
Standard Deviation	69.3443 mm

Table 20. Right Upper Limb Position 2, bones volume relative error

Bone	Reference Volume [mm³]	Predicted Volume [mm³]	Relative Error [%]
Humerus	8.48 e+04	2.21 e+04	73.93
Radius	1.93 e+04	5.51 e+03	71.44

Left Upper Limb Position 1

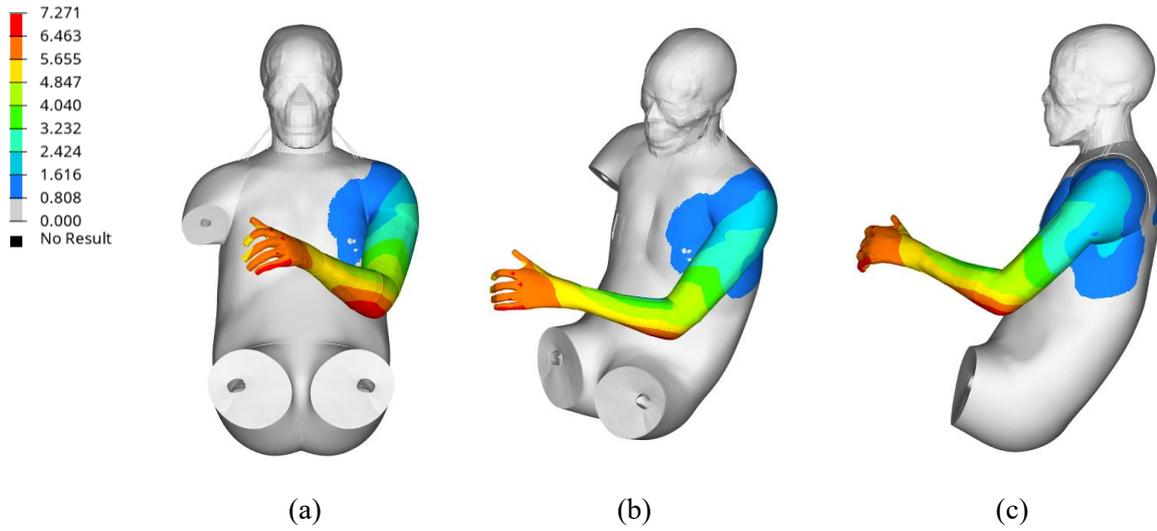


Figure 33. Left Upper Limb Position 1, nodal displacement relative error plots: a) front view; b) iso view; c) top view

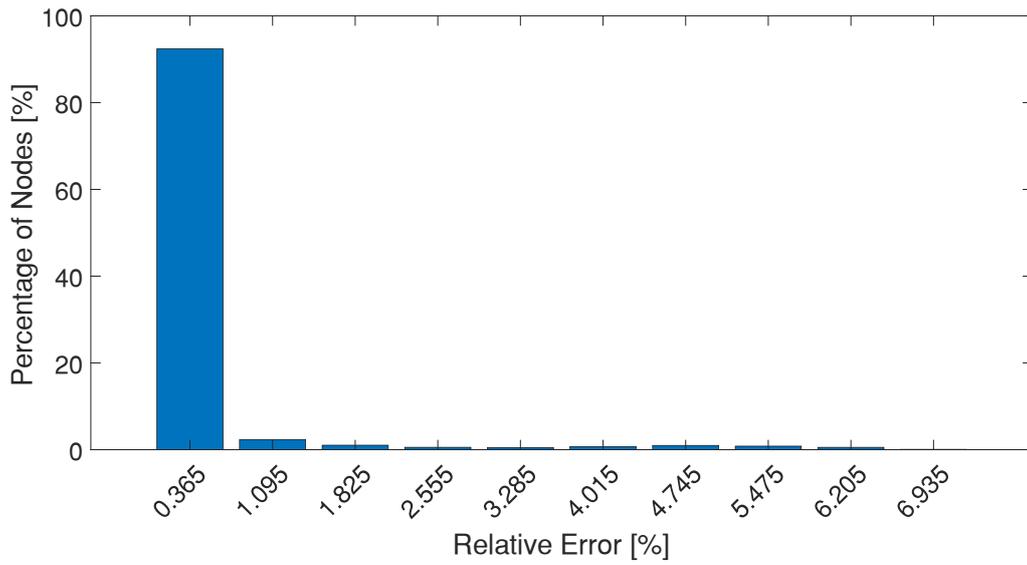


Figure 34. Left Upper Limb Position 1, nodal displacement relative error bar graph

Table 21. Left Upper Limb Position 1, nodal displacement relative error statistical analysis

	Value
Minimum	0
Maximum	7.271 %
Average	0.3033 %
Standard Deviation	0.9525 %

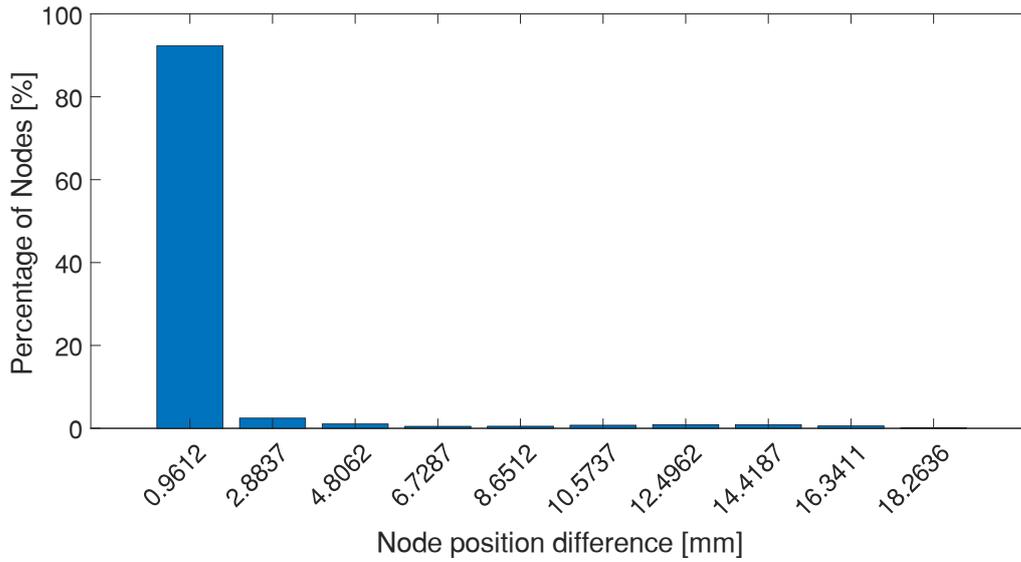


Figure 35. Left Upper Limb Position 1, nodal coordinates error bar graph

Table 22. Left Upper Limb Position 1, nodal coordinates error statistical analysis

	Value
Minimum	0
Maximum	19.2249 mm
Average	0.8086 mm
Standard Deviation	2.5152 mm

Table 23. Left Upper Limb Position 1, bones volume relative error

Bone	Reference Volume [mm³]	Predicted Volume [mm³]	Relative Error [%]
Humerus	8.48 e+04	9.26 e+04	9.23
Radius	1.93 e+04	2.20 e+04	14.17

Left Upper Limb Position 2

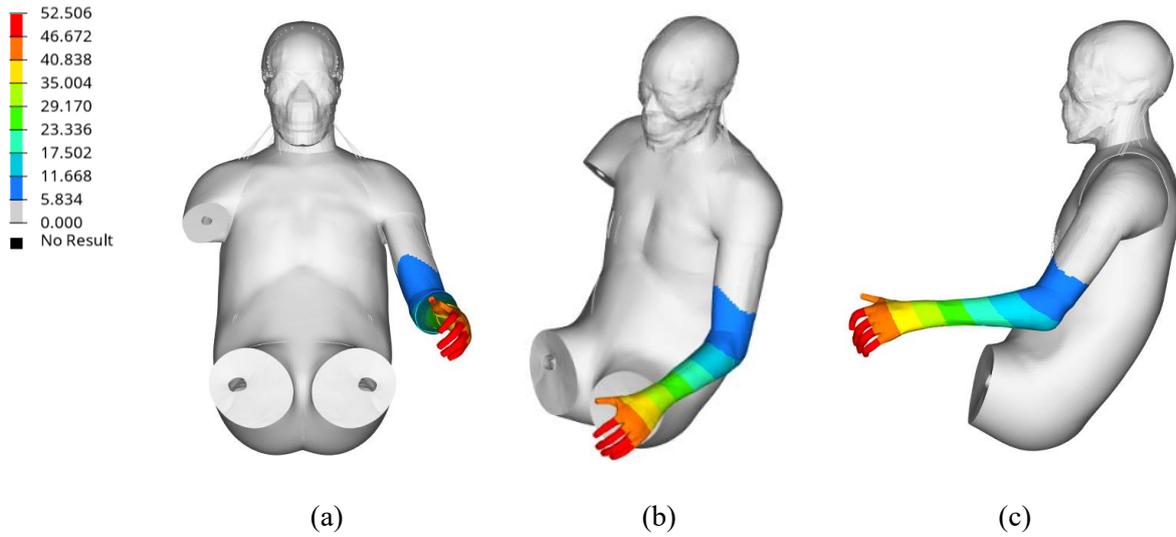


Figure 36. Left Upper Limb Position 2, nodal displacement relative error plots: a) front view; b) iso view; c) top view

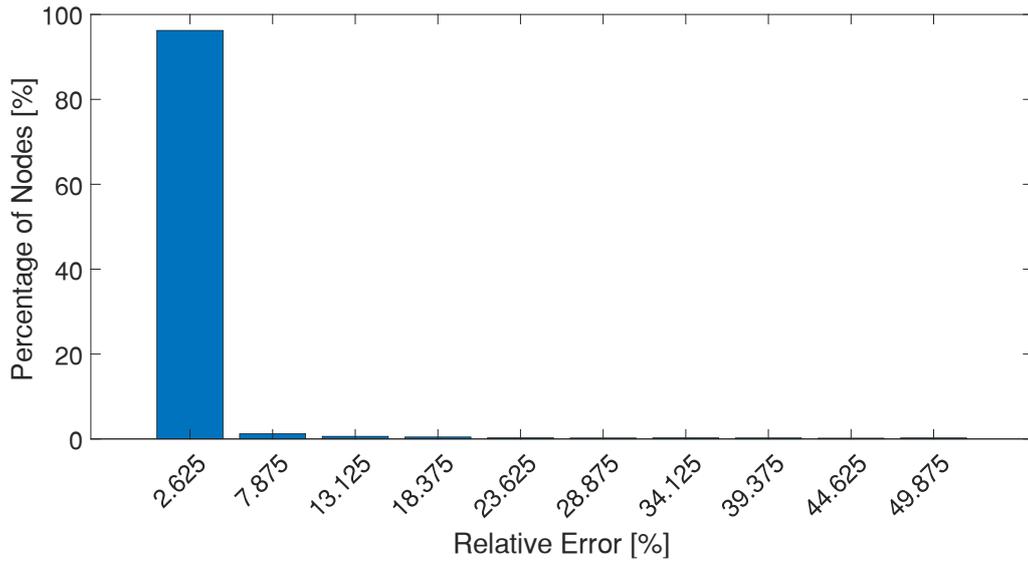


Figure 37. Left Upper Limb Position 2, nodal displacement relative error bar graph

Table 24. Left Upper Limb Position 2, nodal displacement relative error statistical analysis

	Value
Minimum	0.0
Maximum	52.506 %
Average	0.9395 %
Standard Deviation	4.7309 %

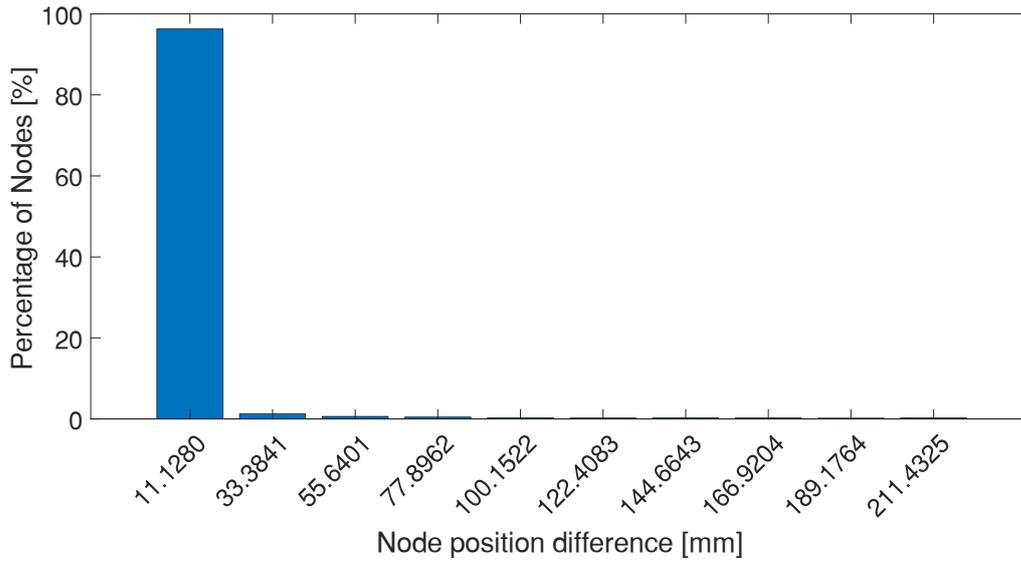


Figure 38. Left Upper Limb Position 2, nodal coordinates error bar graph

Table 25. Left Upper Limb Position 2, nodal coordinates error statistical analysis

	Value
Minimum	0
Maximum	222.5605 mm
Average	3.9924 mm
Standard Deviation	20.0481 mm

Table 26. Left Upper Limb Position 2, bones volume relative error

Bone	Reference Volume [mm³]	Predicted Volume [mm³]	Relative Error [%]
Humerus	8.48 e+04	1.11 e+05	31.28
Radius	1.93 e+04	4.32 e+04	124.16

Right Lower Limb Position 1

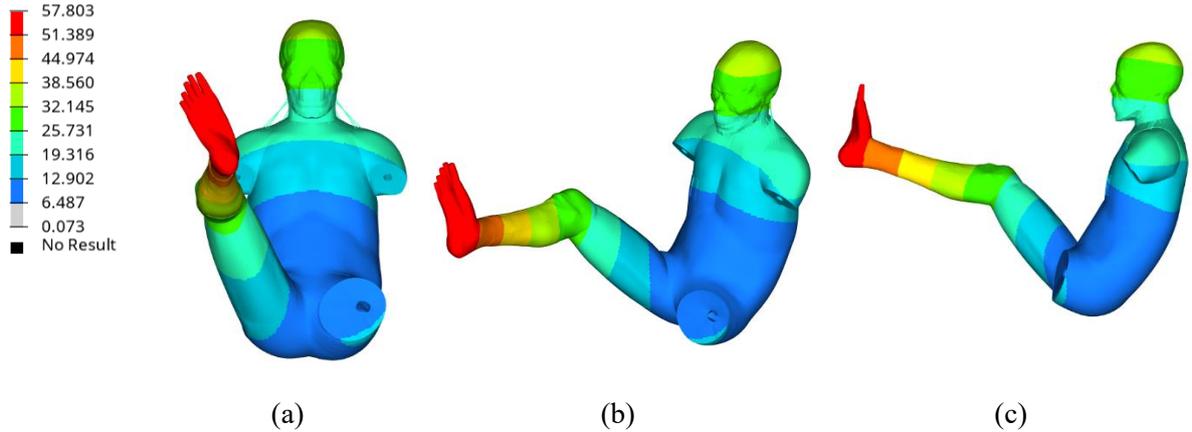


Figure 39. Right Lower Limb Position 1, nodal displacement relative error plots: a) front view; b) iso view; c) top view

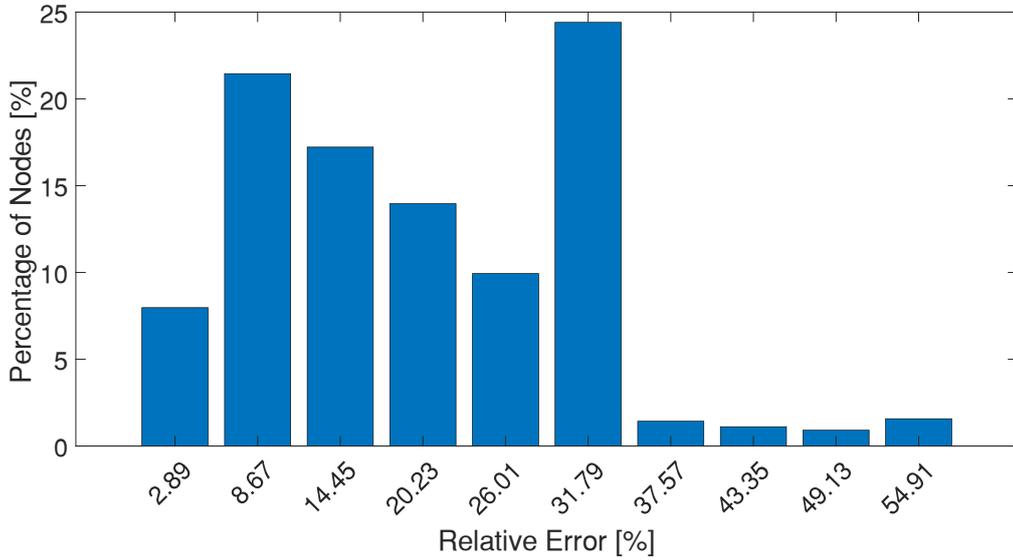


Figure 40. Right Lower Limb Position 1, nodal displacement relative error bar graph

Table 27. Right Lower Limb Position 1, nodal displacement relative error statistical analysis

	Value
Minimum	0.073 %
Maximum	57.803 %
Average	20.0093 %
Standard Deviation	11.3278 %

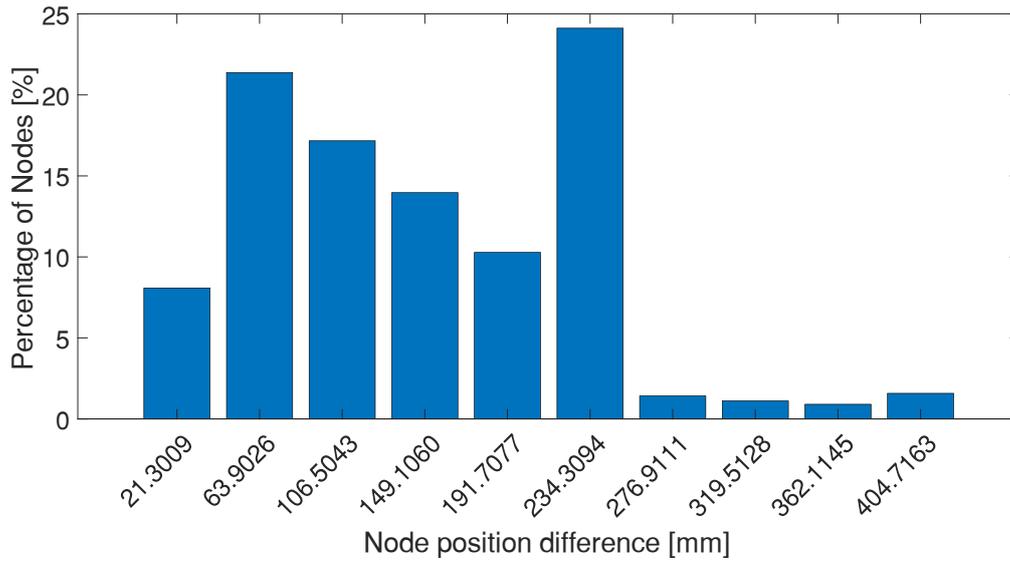


Figure 41. Right Lower Limb Position 1, nodal coordinates error bar graph

Table 28. Right Lower Limb Position 1, nodal coordinates error statistical analysis

	Value
Minimum	0.5361 mm
Maximum	426.0171 mm
Average	147.4807 mm
Standard Deviation	83.4851 mm

Table 29. Right Lower Limb Position 1, bones volume relative error

Bone	Reference Volume [mm³]	Predicted Volume [mm³]	Relative Error [%]
Femur	4.04 e+05	8.78 e+04	78.24
Tibia	2.36 e+05	6.05 e+04	74.32

Right Lower Limb Position 2

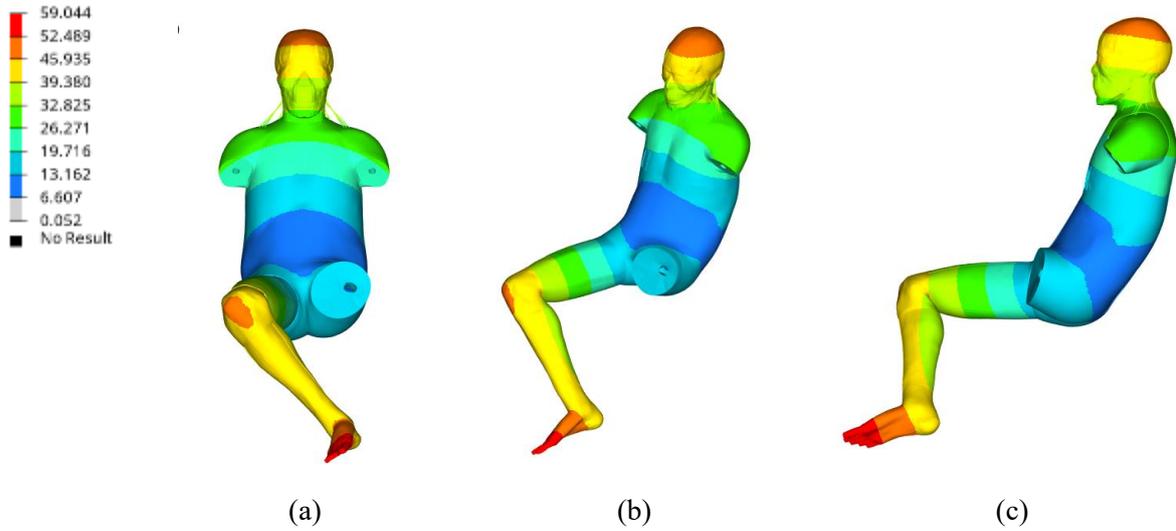


Figure 42. Right Lower Limb Position 2, nodal displacement relative error plots: a) front view; b) iso view; c) top view

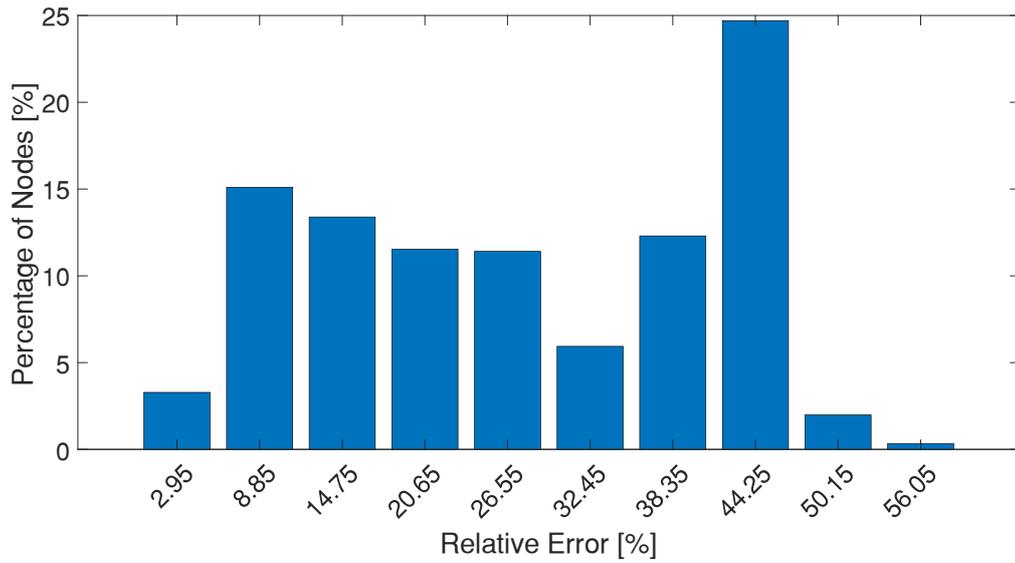


Figure 43. Right Lower Limb Position 2, nodal displacement relative error bar graph

Table 30. Right Lower Limb Position 2, nodal displacement relative error statistical analysis

	Value
Minimum	0.052 %
Maximum	59.044 %
Average	27.5857 %
Standard Deviation	13.9872 %

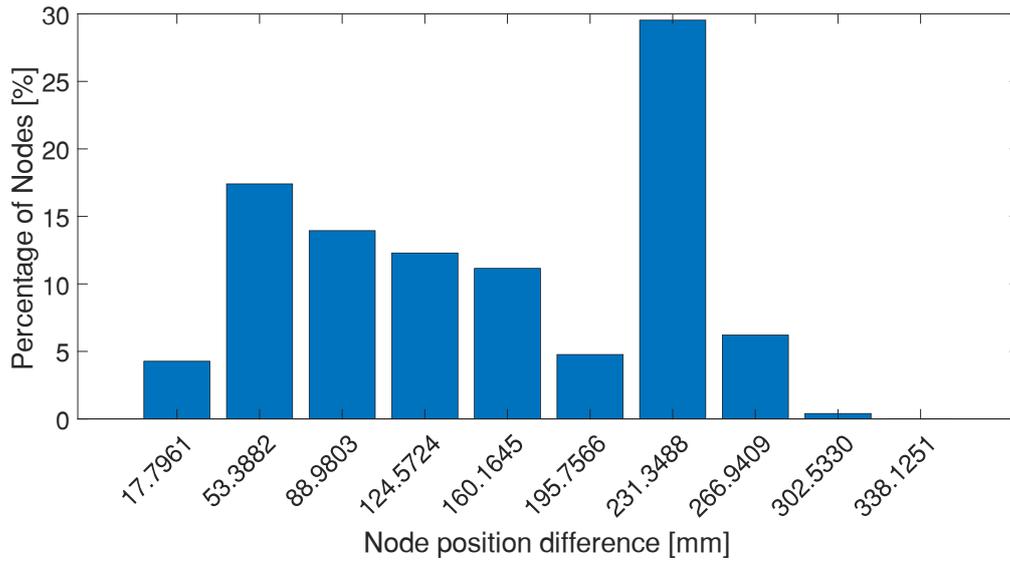


Figure 44. Right Lower Limb Position 2, nodal coordinates error bar graph

Table 31. Right Lower Limb Position 2, nodal coordinates error statistical analysis

	Value
Minimum	0.2868 mm
Maximum	355.9212 mm
Average	151.1325 mm
Standard Deviation	76.6262 mm

Table 32. Right Lower Limb Position 2, bones volume relative error

Bone	Reference Volume [mm³]	Predicted Volume [mm³]	Relative Error [%]
Femur	4.04 e+05	8.43 e+04	79.10
Tibia	2.36 e+05	6.62 e+04	71.89

Left Lower Limb Position 1

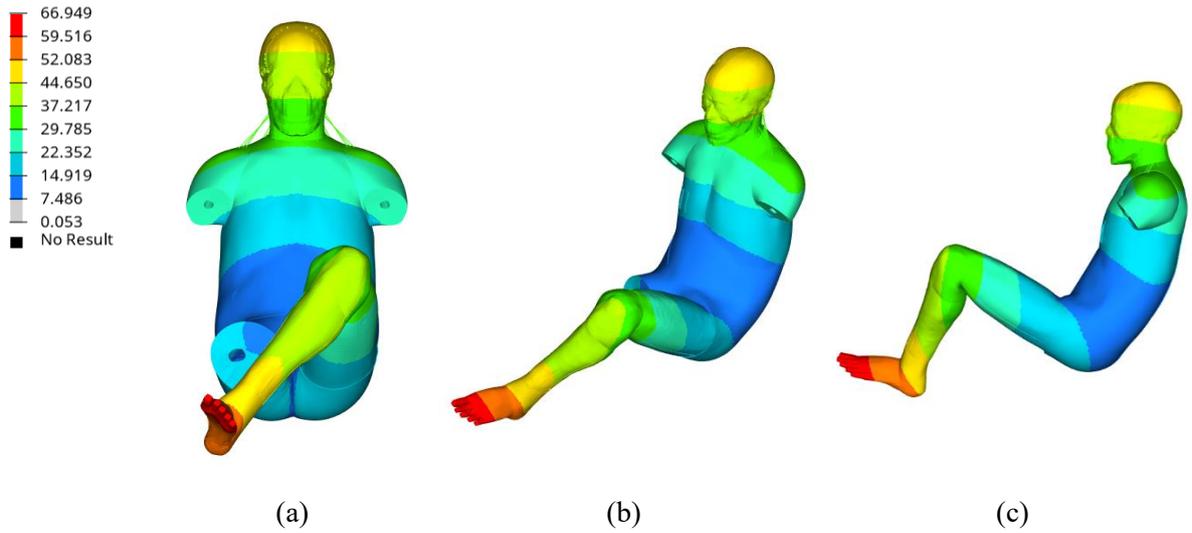


Figure 45. Left Lower Limb Position 1, nodal displacement relative error plots: a) front view; b) iso view; c) top view

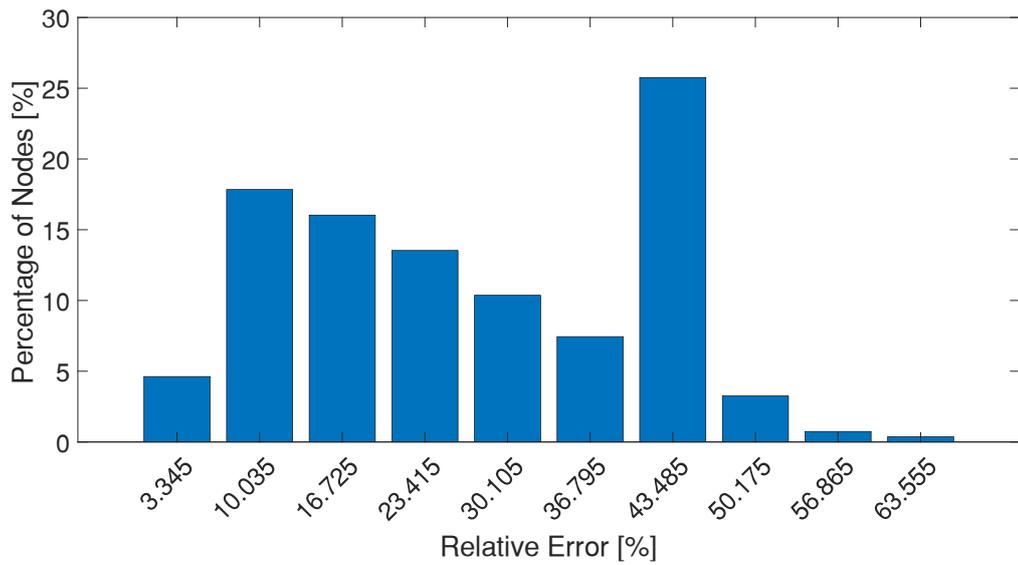


Figure 46. Left Lower Limb Position 1, nodal displacement relative error bar graph

Table 33. Left Lower Limb Position 1, nodal displacement relative error statistical analysis

	Value
Minimum	0.053 %
Maximum	66.949 %
Average	27.1337 %
Standard Deviation	14.1752 %

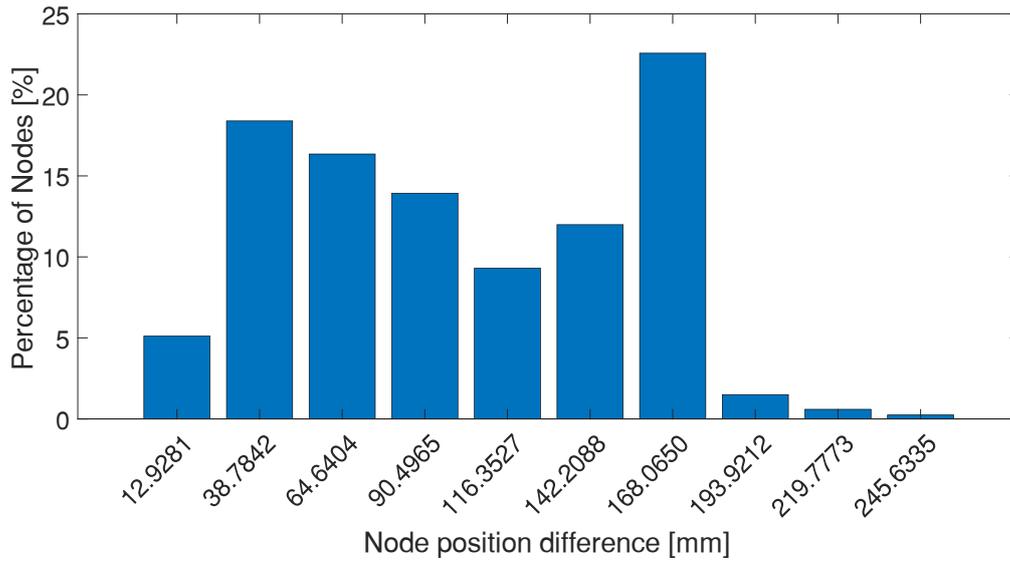


Figure 47. Left Lower Limb Position 1, nodal coordinates error bar graph

Table 34. Left Lower Limb Position 1, nodal coordinates error statistical analysis

	Value
Minimum	0.1983 mm
Maximum	258.5615 mm
Average	101.6327 mm
Standard Deviation	53.0915 mm

Table 35. Left Lower Limb Position 1, bones volume relative error

Bone	Reference Volume [mm ³]	Predicted Volume [mm ³]	Relative Error [%]
Femur	4.04 e+05	1.68 e+05	58.49
Tibia	2.36 e+05	9.03 e+04	61.69

Left Lower Limb Position 2

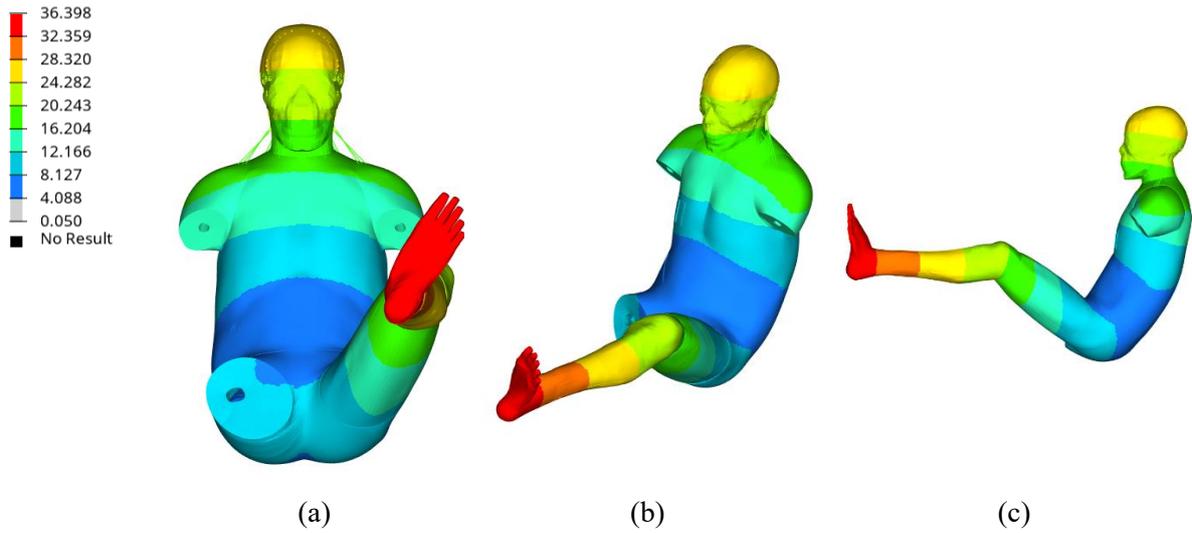


Figure 48. Left Lower Limb Position 2, nodal displacement relative error plots: a) front view; b) iso view; c) top view

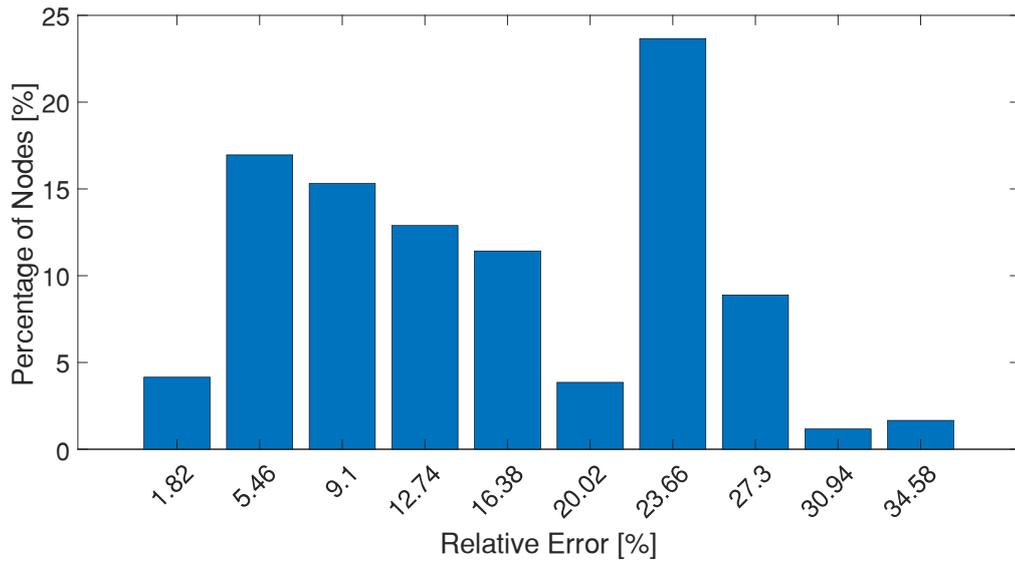


Figure 49. Left Lower Limb Position 2, nodal displacement relative error bar graph

Table 36. Left Lower Limb Position 2, nodal displacement relative error statistical analysis

	Value
Minimum	0.050 %
Maximum	36.398 %
Average	15.6222 %
Standard Deviation	8.2745 %

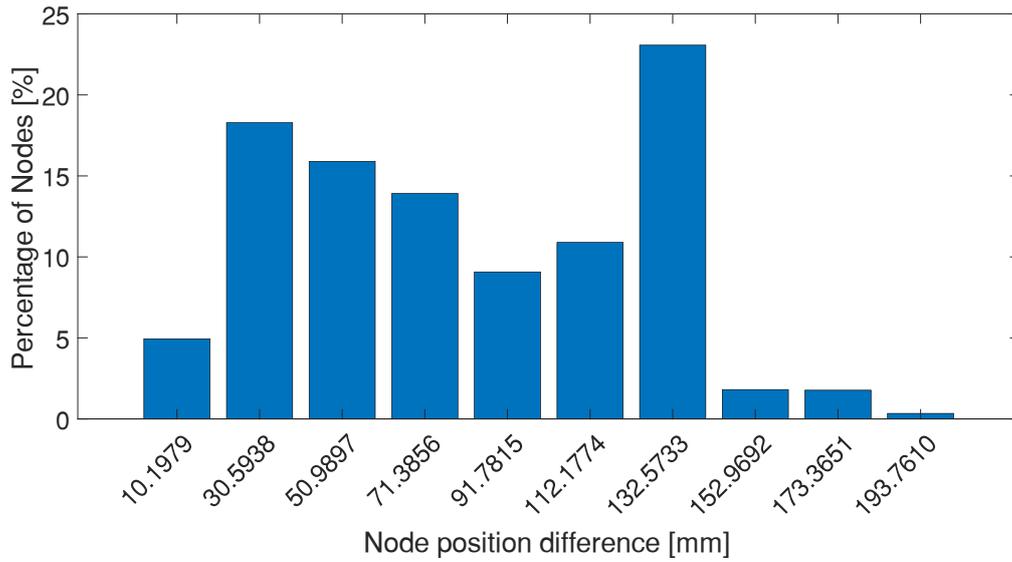


Figure 50. Left Lower Limb Position 2, nodal coordinates error bar graph

Table 37. Left Lower Limb Position 2, nodal coordinates error statistical analysis

	Value
Minimum	0.2594 mm
Maximum	203.9590 mm
Average	81.8356 mm
Standard Deviation	43.3417 mm

Table 38. Left Lower Limb Position 2, bones volume relative error

Bone	Reference Volume [mm ³]	Predicted Volume [mm ³]	Relative Error [%]
Femur	4.04 e+05	2.02 e+05	50.04
Tibia	2.36 e+05	1.36 e+05	42.15

Appendix D Displacements Interpolation Test Results

Right Upper Limb Position 1

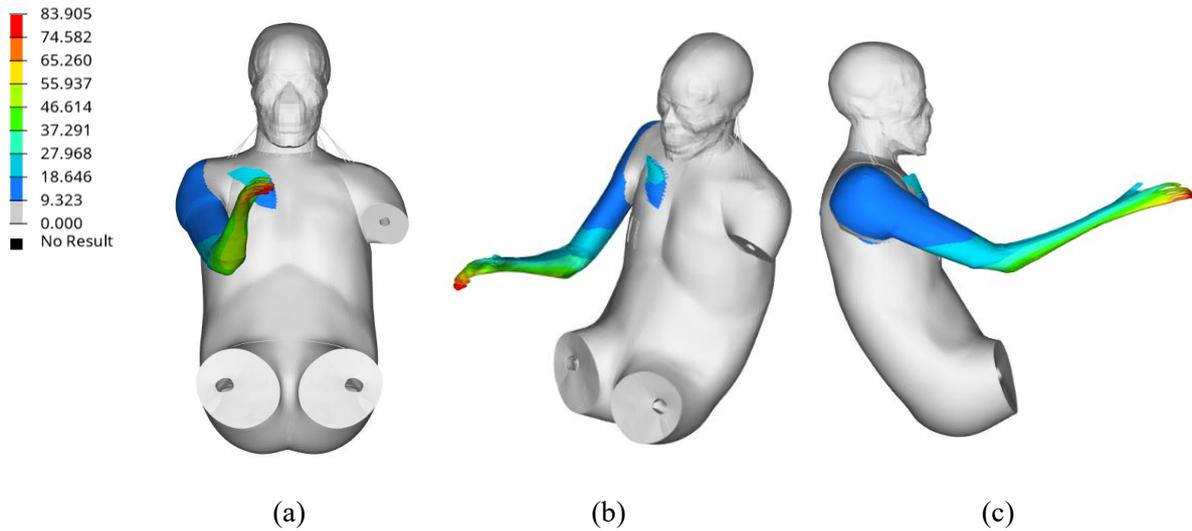


Figure 51. Right Upper Limb Position 1, nodal displacement relative error plots: a) front view; b) iso view; c) top view

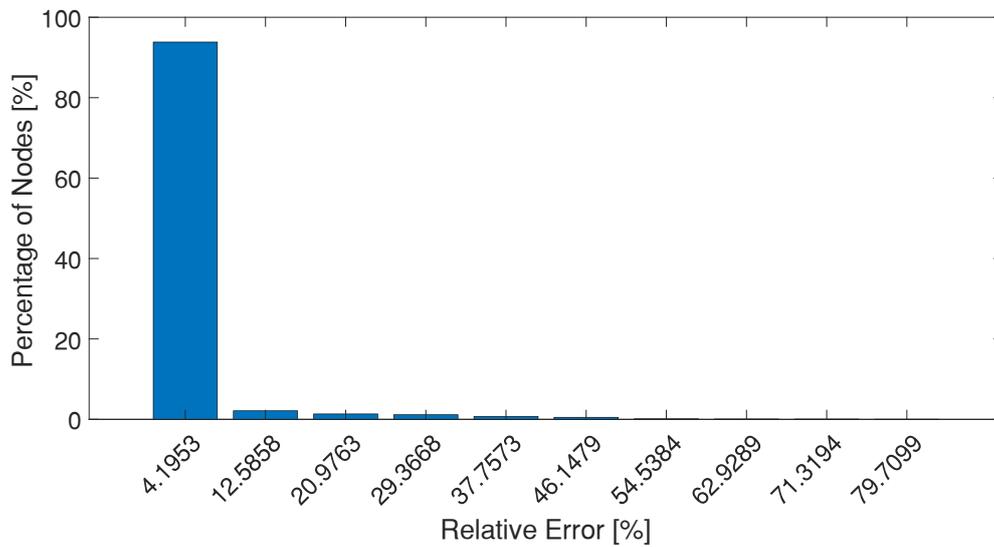


Figure 52. Right Upper Limb Position 1, nodal displacement relative error bar graph

Table 39. Right Upper Limb Position 1, nodal displacement relative error statistical analysis

	Value
Minimum	0.0
Maximum	83.9052 %
Average	2.5963 %
Standard Deviation	7.0321 %

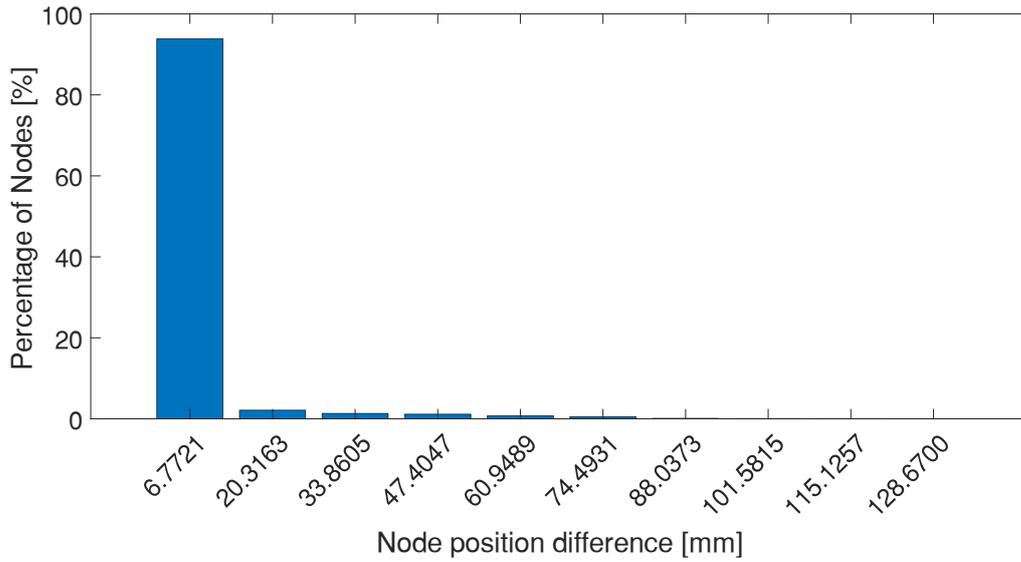


Figure 53. Right Upper Limb Position 1, nodal coordinates error bar graph

Table 40. Right Upper Limb Position 1, nodal coordinates error statistical analysis

	Value
Minimum	0.0
Maximum	135.4421 mm
Average	4.1910 mm
Standard Deviation	11.3514 mm

Table 41. Right Upper Limb Position 1, bones volume relative error

Bone	Reference Volume [mm³]	Predicted Volume [mm³]	Relative Error [%]
Humerus	8.48 e+04	6.02 e+04	29.03
Radius	1.93 e+04	6.95 e+04	260.07

Right Upper Limb Position 2

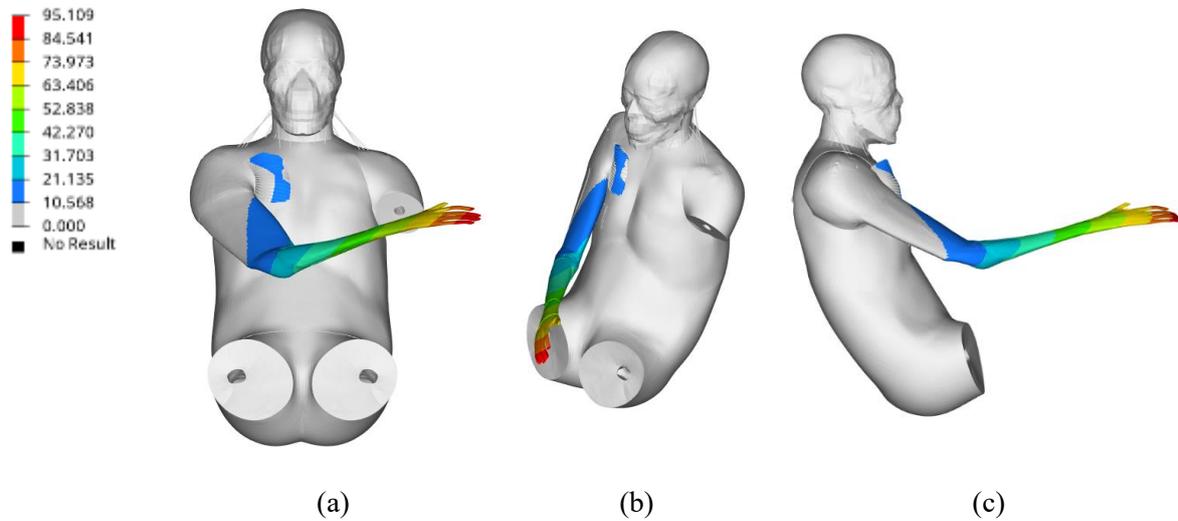


Figure 54. Right Upper Limb Position 2, nodal displacement relative error plots: a) front view; b) iso view; c) top view

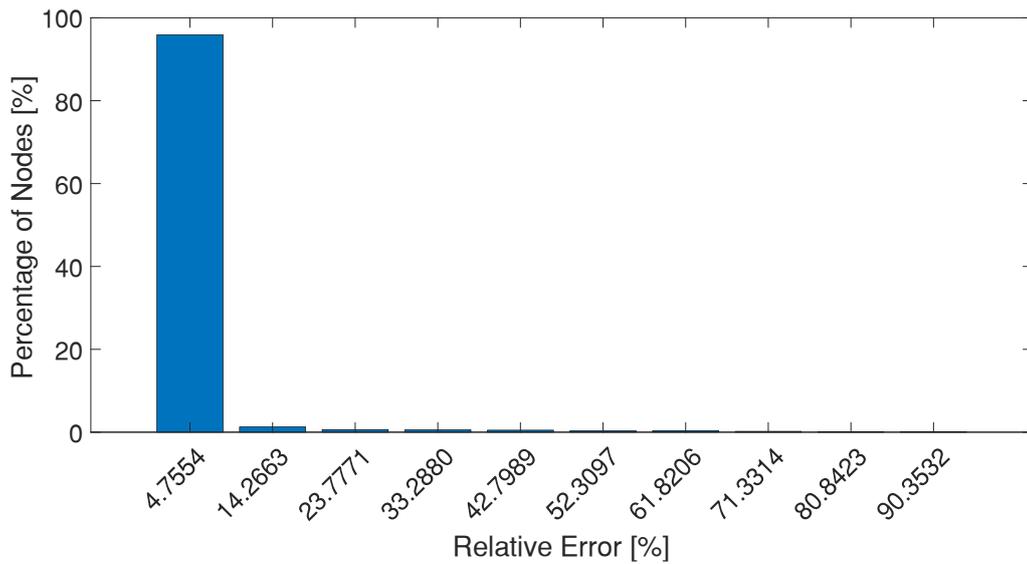


Figure 55. Right Upper Limb Position 2, nodal displacement relative error bar graph

Table 42. Right Upper Limb Position 2, nodal displacement relative error statistical analysis

	Value
Minimum	0.0
Maximum	95.1086 %
Average	2.1821 %
Standard Deviation	8.3988 %

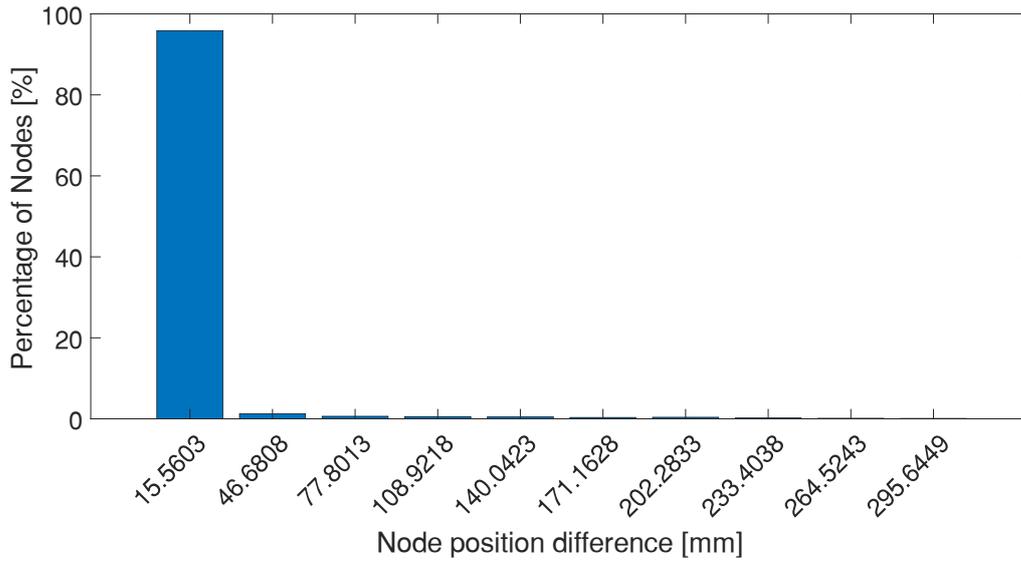


Figure 56. Right Upper Limb Position 2, nodal coordinates error bar graph

Table 43. Right Upper Limb Position 2, nodal coordinates error statistical analysis

	Value
Minimum	0.0 mm
Maximum	311.2051 mm
Average	7.1400 mm
Standard Deviation	27.4816 mm

Table 44. Right Upper Limb Position 2, bones volume relative error

Bone	Reference Volume [mm³]	Predicted Volume [mm³]	Relative Error [%]
Humerus	8.48 e+04	5.36 e+04	36.75
Radius	1.93 e+04	6.19 e+04	220.62

Left Upper Limb Position 1

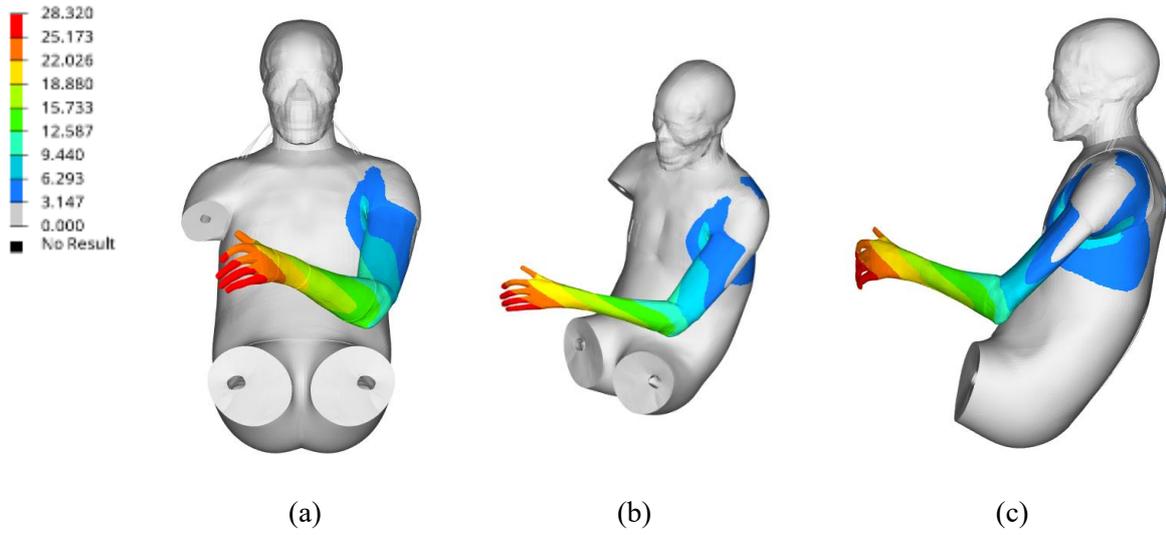


Figure 57. Left Upper Limb Position 1, nodal displacement relative error plots: a) front view; b) iso view; c) top view

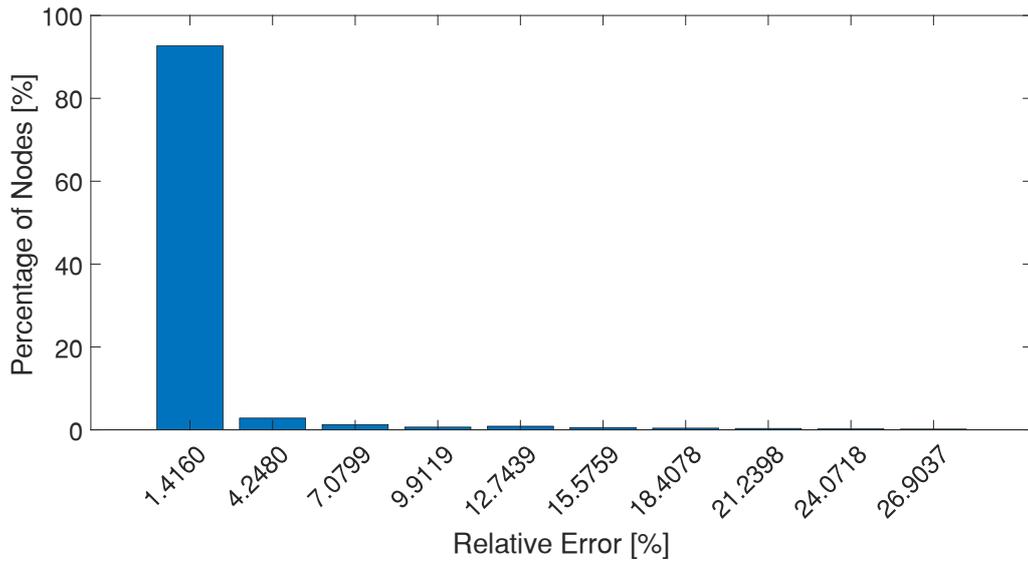


Figure 58. Left Upper Limb Position 1, nodal displacement relative error bar graph

Table 45. Left Upper Limb Position 1, nodal displacement relative error statistical analysis

	Value
Minimum	0.0
Maximum	28.3197 %
Average	1.0491 %
Standard Deviation	3.0472 %

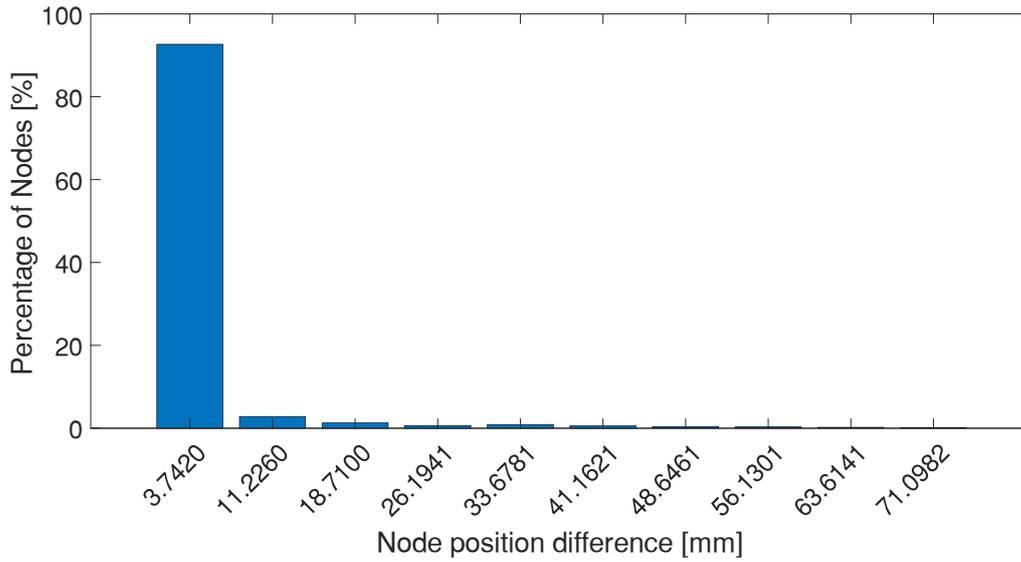


Figure 59. Left Upper Limb Position 1, nodal coordinates error bar graph

Table 46. Left Upper Limb Position 1, nodal coordinates error statistical analysis

	Value
Minimum	0.0
Maximum	75.8402 mm
Average	2.7723 mm
Standard Deviation	8.0528 mm

Table 47. Left Upper Limb Position 1, bones volume relative error

Bone	Reference Volume [mm³]	Predicted Volume [mm³]	Relative Error [%]
Humerus	8.48 e+04	5.91 e+04	30.32
Radius	1.93 e+04	7.33 e+04	279.78

Left Upper Limb Position 2

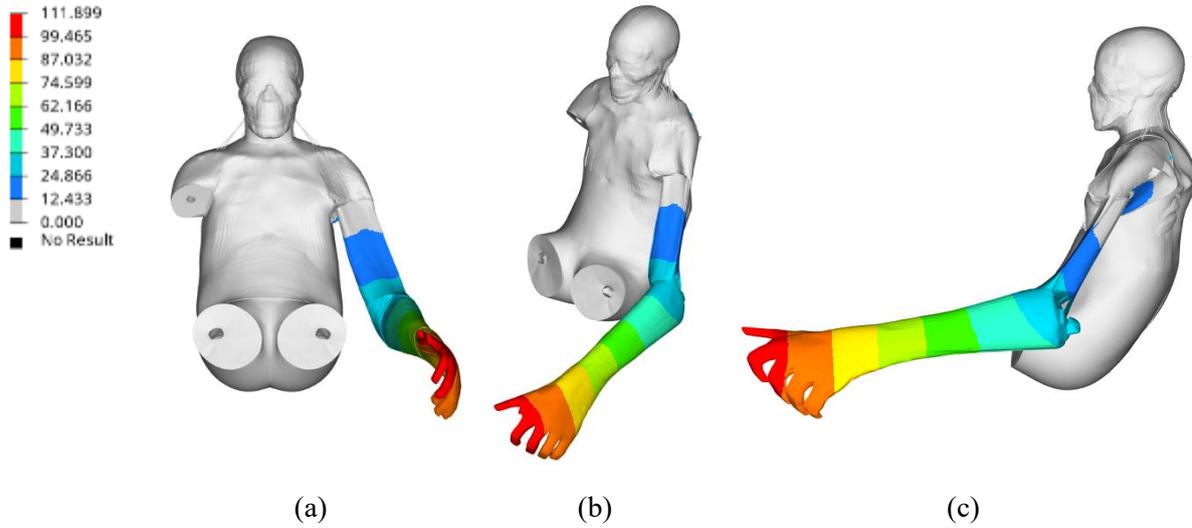


Figure 60. Left Upper Limb Position 2, nodal displacement relative error plots: a) front view; b) iso view; c) top view

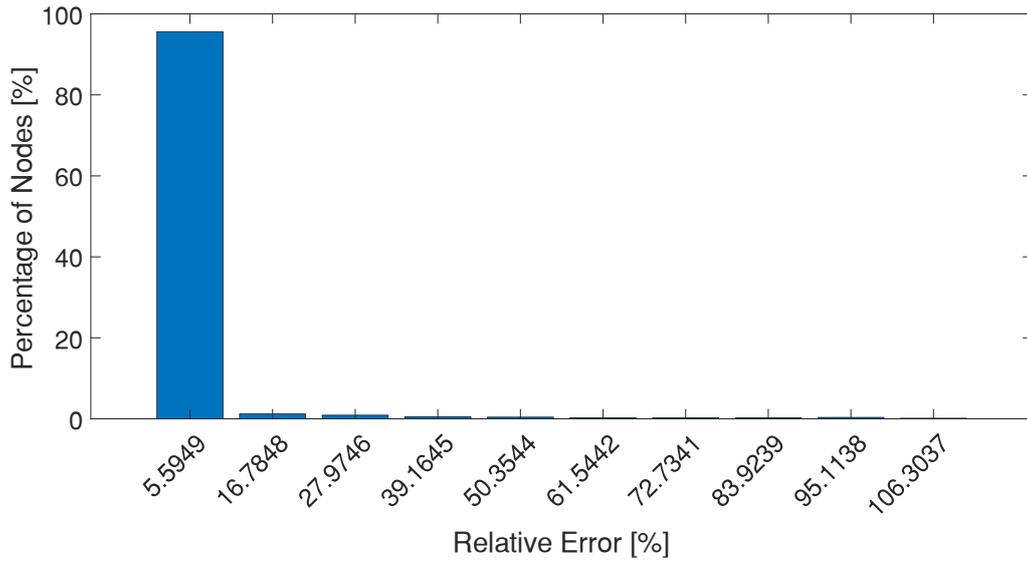


Figure 61. Left Upper Limb Position 2, nodal displacement relative error bar graph

Table 48. Left Upper Limb Position 2, nodal displacement relative error statistical analysis

	Value
Minimum	0.0
Maximum	111.8986 %
Average	2.6927 %
Standard Deviation	10.8984 %

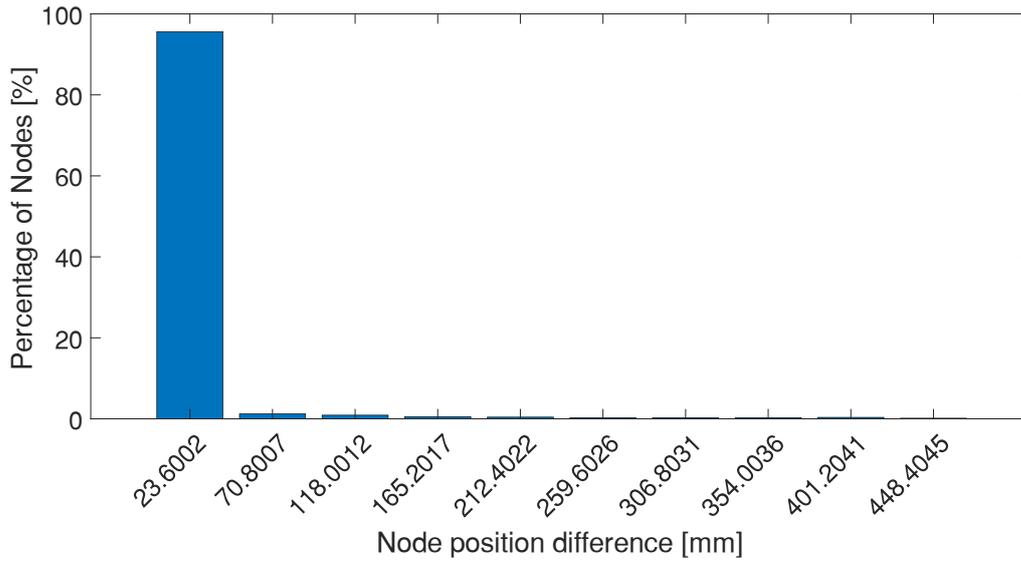


Figure 62. Left Upper Limb Position 2, nodal coordinates error bar graph

Table 49. Left Upper Limb Position 2, nodal coordinates error statistical analysis

	Value
Minimum	0.0
Maximum	472.0048 mm
Average	11.3583 mm
Standard Deviation	45.9712 mm

Table 50. Left Upper Limb Position 2, bones volume relative error

Bone	Reference Volume [mm³]	Predicted Volume [mm³]	Relative Error [%]
Humerus	8.48 e+04	7.94 e+03	90.64
Radius	1.93 e+04	8.99 e+04	365.93

Right Lower Limb Position 1

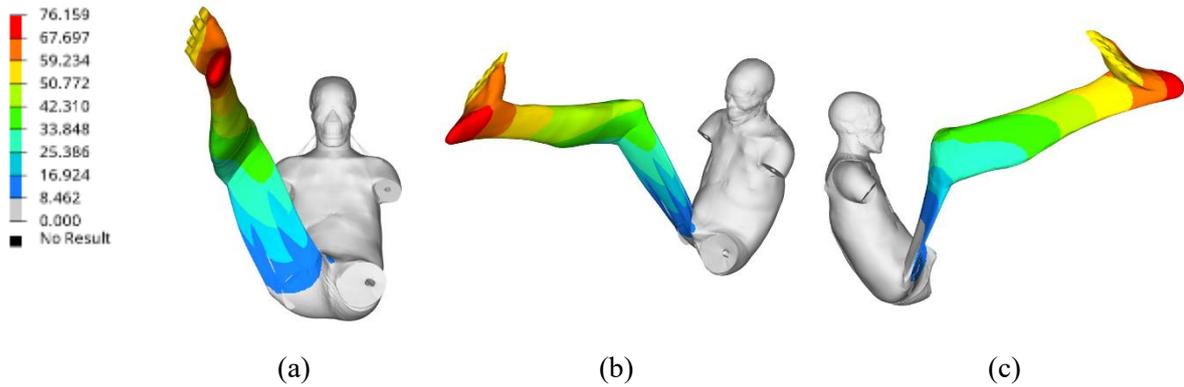


Figure 63. Right Lower Limb Position 1, nodal displacement relative error plots: a) front view; b) iso view; c) top view

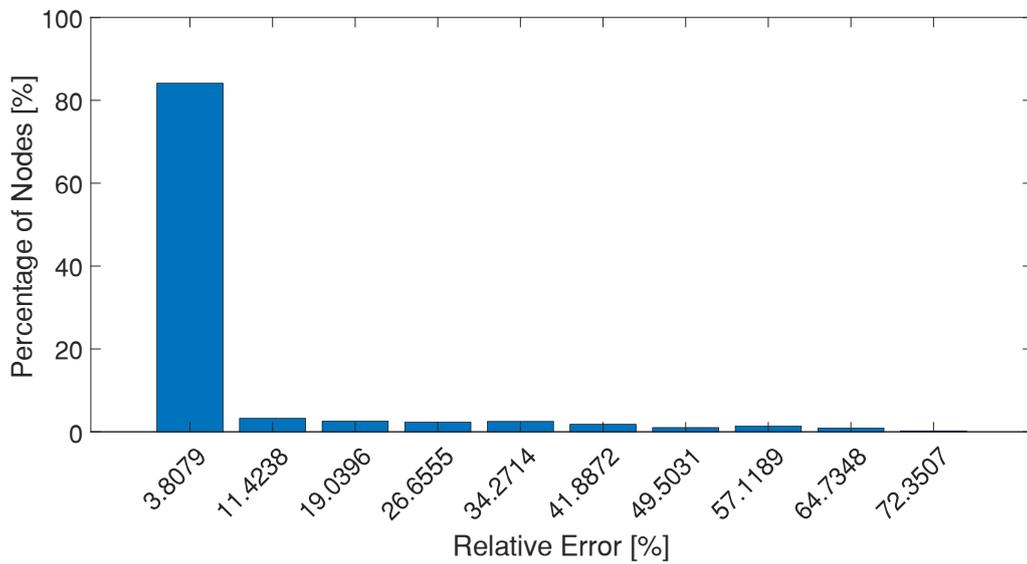


Figure 64. Right Lower Limb Position 1, nodal displacement relative error bar graph

Table 51. Right Lower Limb Position 1, nodal displacement relative error statistical analysis

	Value
Minimum	0.0
Maximum	76.1586 %
Average	5.7635 %
Standard Deviation	13.2814 %

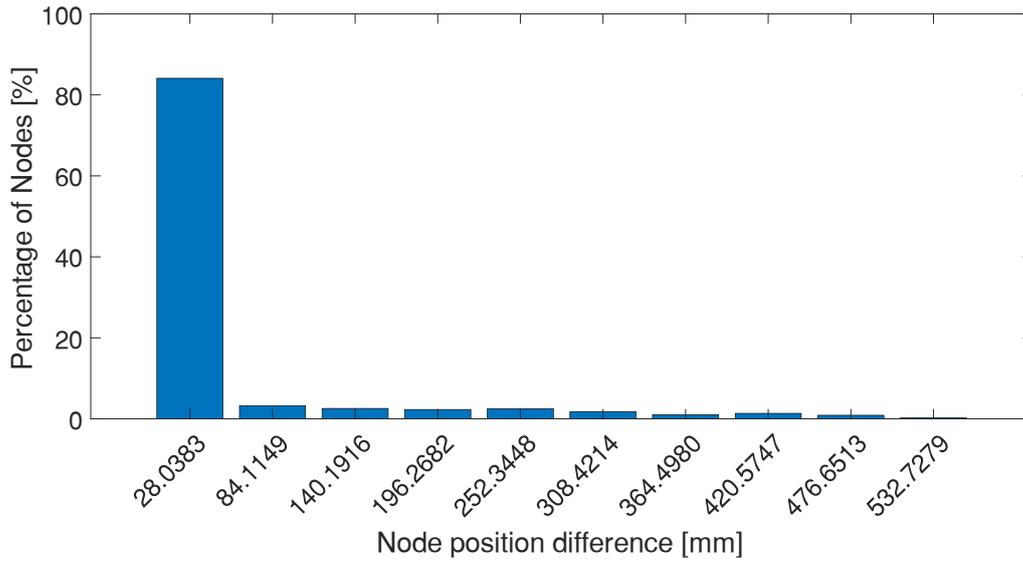


Figure 65. Right Lower Limb Position 1, nodal coordinates error bar graph

Table 52. Right Lower Limb Position 1, nodal coordinates error statistical analysis

	Value
Minimum	0.0 mm
Maximum	560.7662 mm
Average	42.4373 mm
Standard Deviation	97.7930 mm

Table 53. Right Lower Limb Position 1, bones volume relative error

Bone	Reference Volume [mm³]	Predicted Volume [mm³]	Relative Error [%]
Femur	4.04 e+05	4.60 e+04	88.59
Tibia	2.36 e+05	8.44 e+05	257.97

Right Lower Limb Position 2

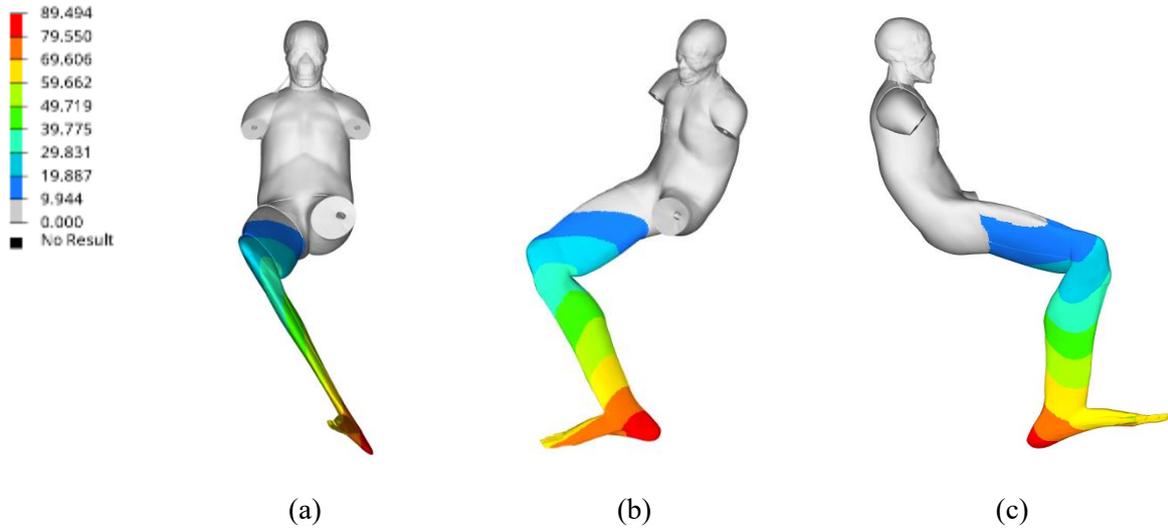


Figure 66. Right Lower Limb Position 2, nodal displacement relative error plots: a) front view; b) iso view; c) top view

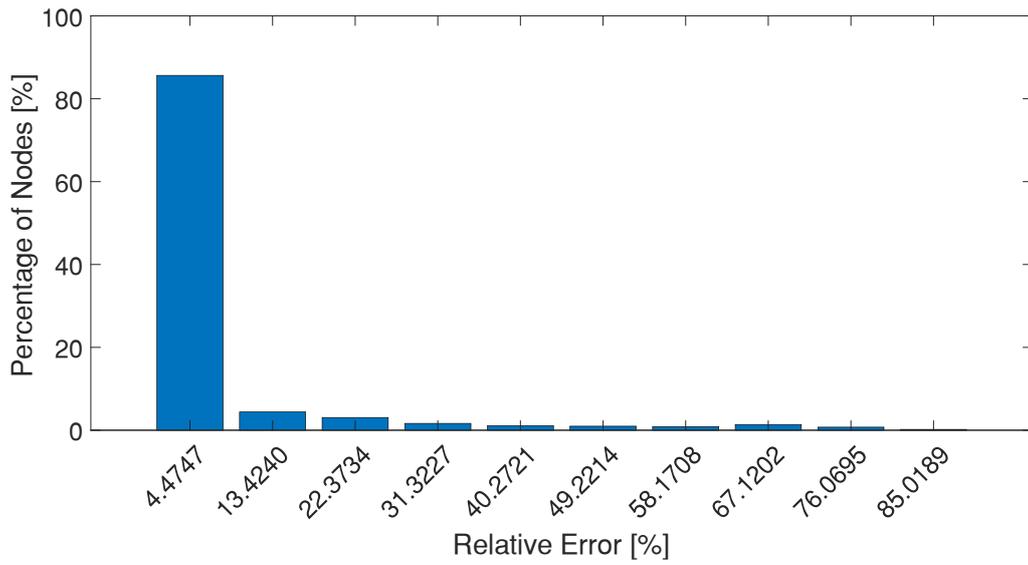


Figure 67. Right Lower Limb Position 2, nodal displacement relative error bar graph

Table 54. Right Lower Limb Position 2, nodal displacement relative error statistical analysis

	Value
Minimum	0.0
Maximum	89.4935 %
Average	5.5214 %
Standard Deviation	14.1171 %

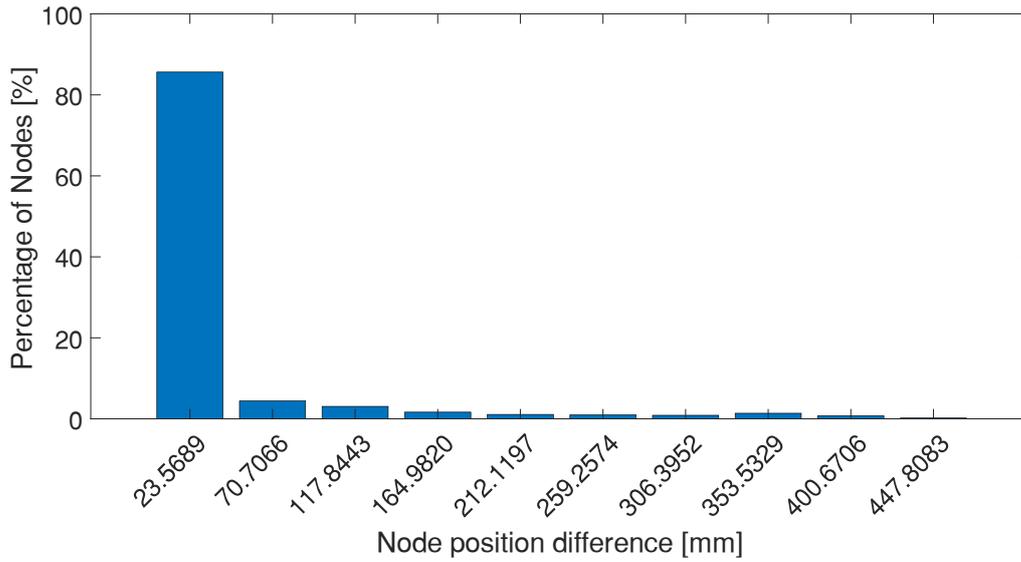


Figure 68. Right Lower Limb Position 2, nodal coordinates error bar graph

Table 55. Right Lower Limb Position 2, nodal coordinates error statistical analysis

	Value
Minimum	0.0 mm
Maximum	471.3772 mm
Average	29.0820 mm
Standard Deviation	74.3570 mm

Table 56. Right Lower Limb Position 2, bones volume relative error

Bone	Reference Volume [mm³]	Predicted Volume [mm³]	Relative Error [%]
Femur	4.04 e+05	3.78 e+05	6.26
Tibia	2.36 e+05	7.72 e+05	227.83

Left Lower Limb Position 1

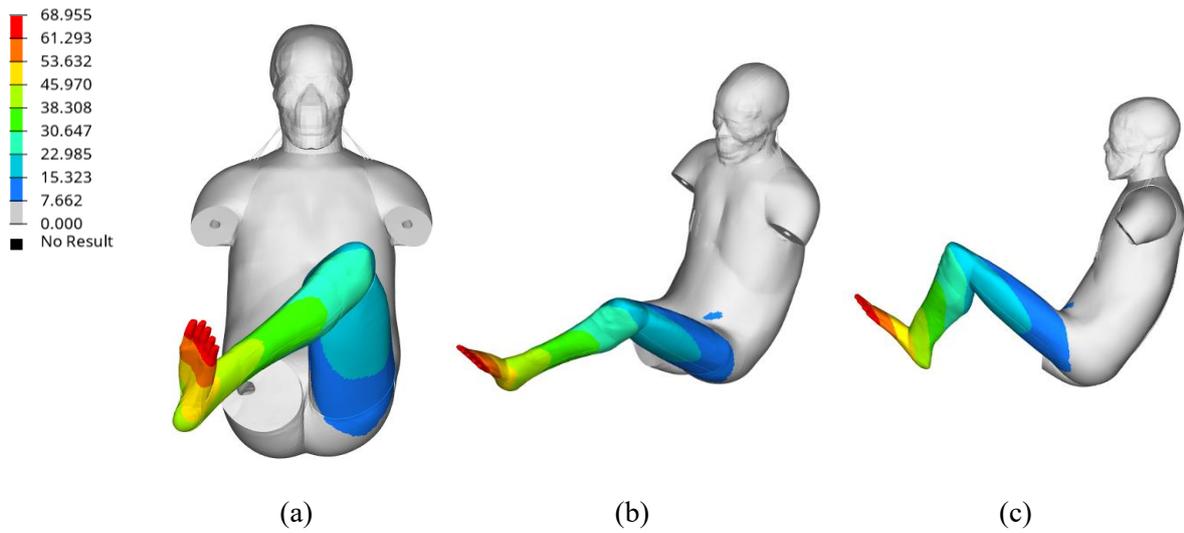


Figure 69. Left Lower Limb Position 1, nodal displacement relative error plots: a) front view; b) iso view; c) top view

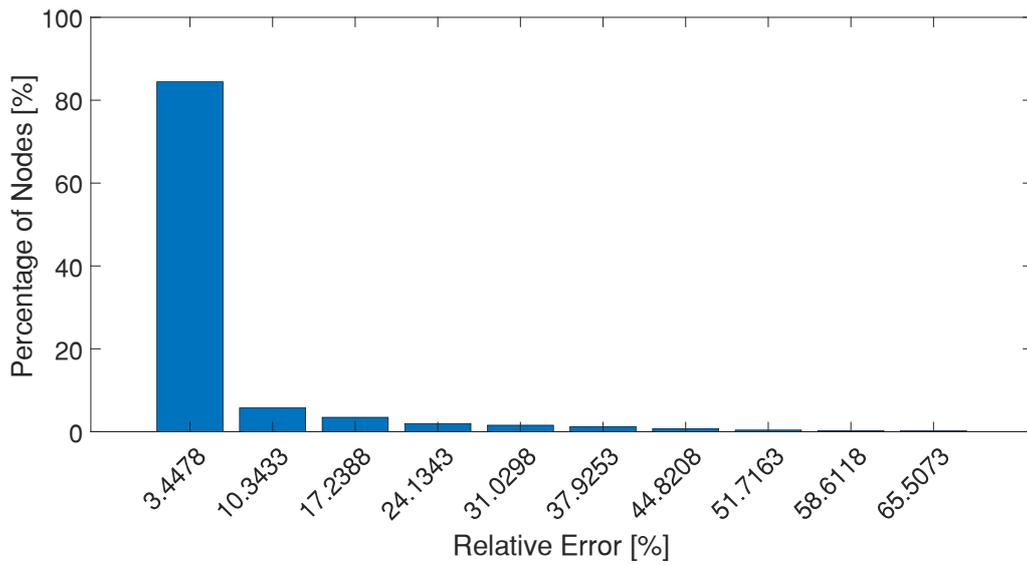


Figure 70. Left Lower Limb Position 1, nodal displacement relative error bar graph

Table 57. Left Lower Limb Position 1, nodal displacement relative error statistical analysis

	Value
Minimum	0.0
Maximum	68.9551 %
Average	3.9786 %
Standard Deviation	9.3942 %

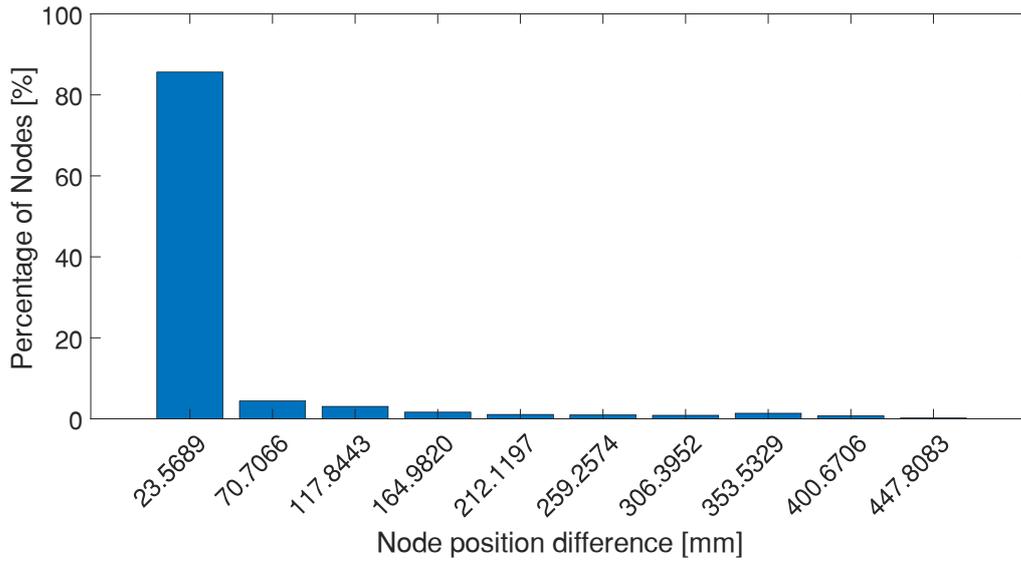


Figure 71. Left Lower Limb Position 1, nodal coordinates error bar graph

Table 58. Left Lower Limb Position 1, nodal coordinates error statistical analysis

	Value
Minimum	0.0
Maximum	251.5472 mm
Average	14.5138 mm
Standard Deviation	34.2699 mm

Table 59. Left Lower Limb Position 1, bones volume relative error

Bone	Reference Volume [mm³]	Predicted Volume [mm³]	Relative Error [%]
Femur	4.04 e+05	3.20 e+05	20.65
Tibia	2.36 e+05	4.95 e+05	109.90

Left Lower Limb Position 2

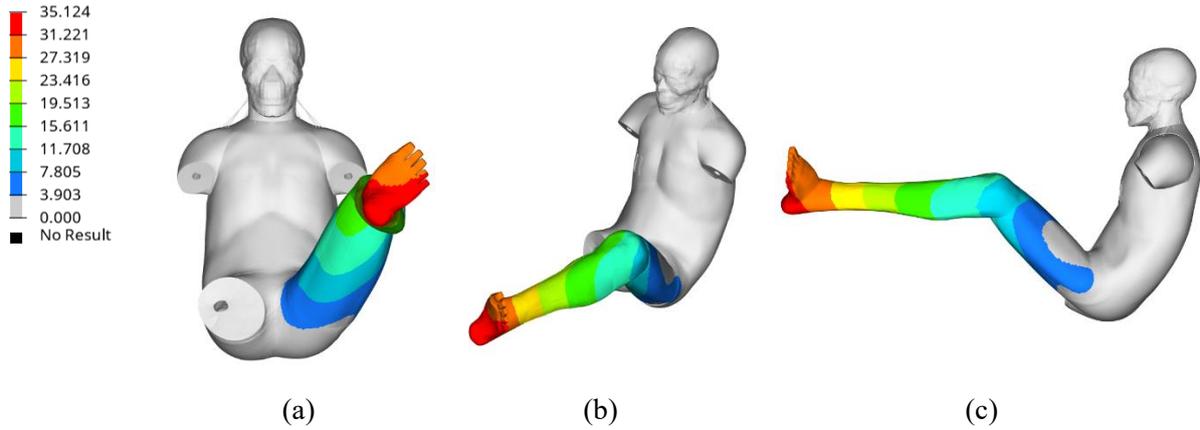


Figure 72. Left Lower Limb Position 2, nodal displacement relative error plots: a) front view; b) iso view; c) top view

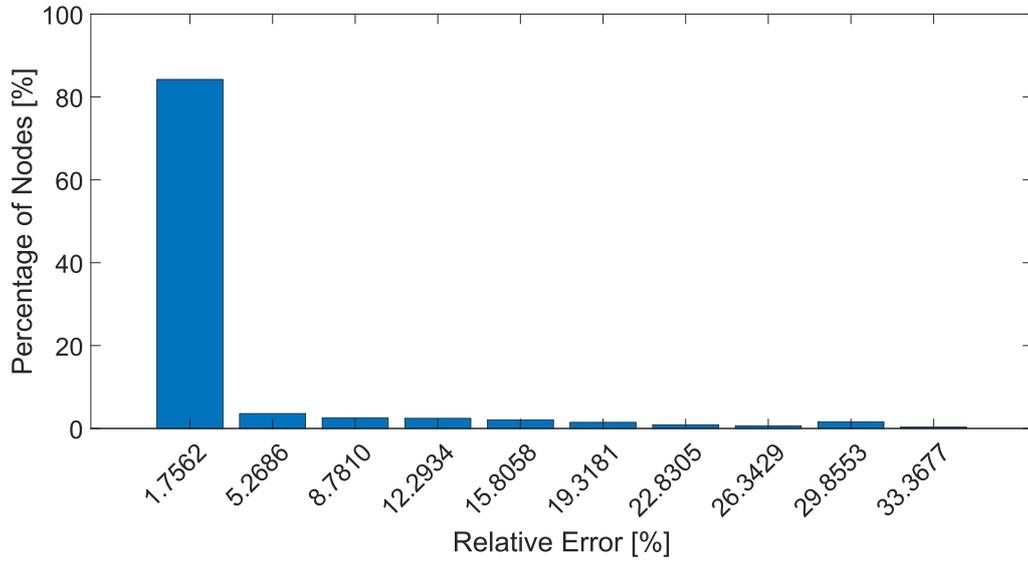


Figure 73. Left Lower Limb Position 2, nodal displacement relative error bar graph

Table 60. Left Lower Limb Position 2, nodal displacement relative error statistical analysis

	Value
Minimum	0.0
Maximum	35.1239 %
Average	2.5302 %
Standard Deviation	6.2588 %

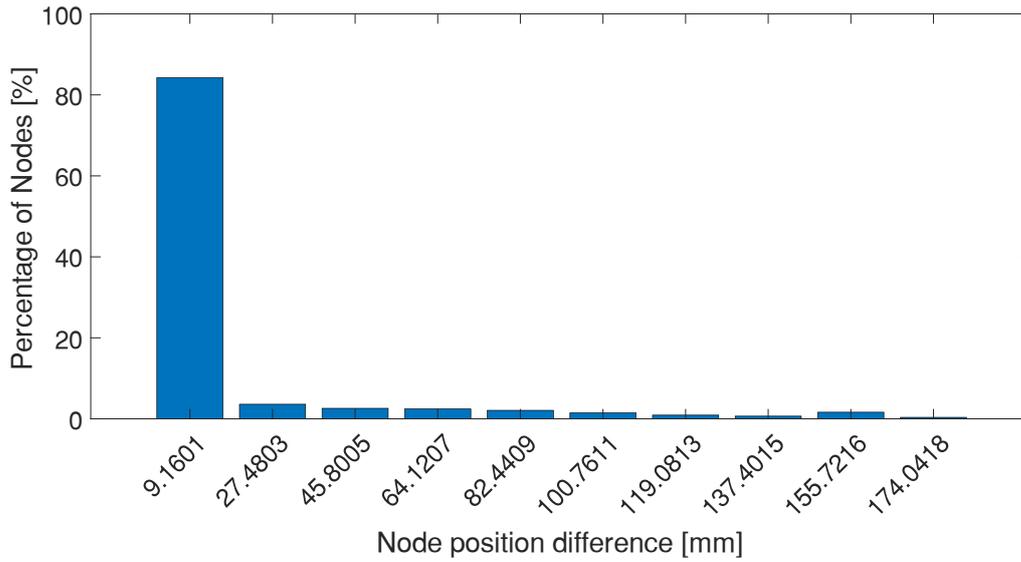


Figure 74. Left Lower Limb Position 2, nodal coordinates error bar graph

Table 61. Left Lower Limb Position 2, nodal coordinates error statistical analysis

	Value
Minimum	0.0
Maximum	183.2019 mm
Average	13.1974 mm
Standard Deviation	32.6450 mm

Table 62. Left Lower Limb Position 2, bones volume relative error

Bone	Reference Volume [mm³]	Predicted Volume [mm³]	Relative Error [%]
Femur	4.04 e+05	4.43 e+05	9.84
Tibia	2.36 e+05	7.71 e+05	227.26

Appendix E InvD Interpolation Results

Right Upper Limb Position 1

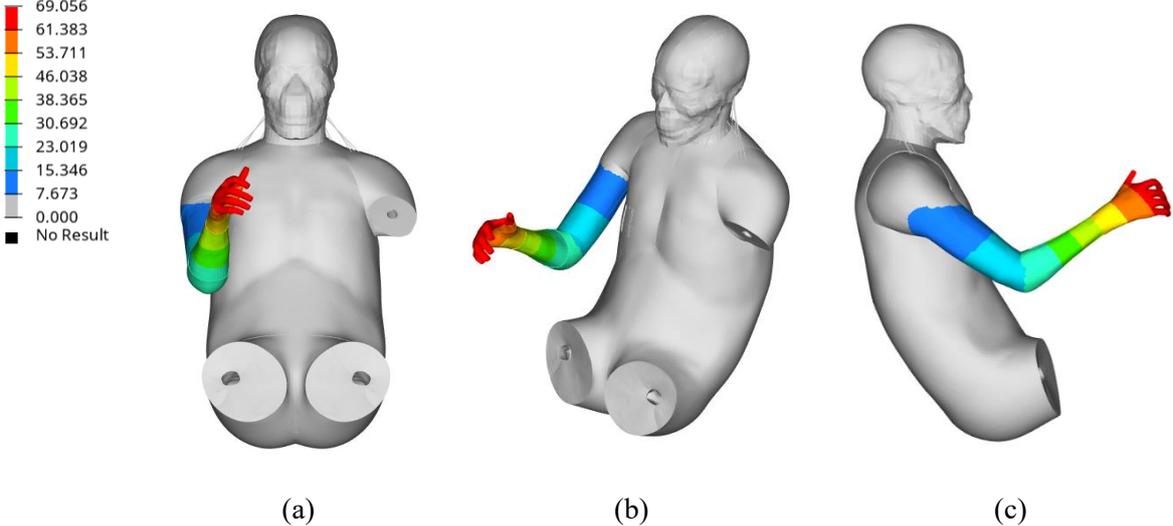


Figure 75. Right Upper Limb Position 1, nodal displacement relative error plots: a) front view; b) iso view; c) top view

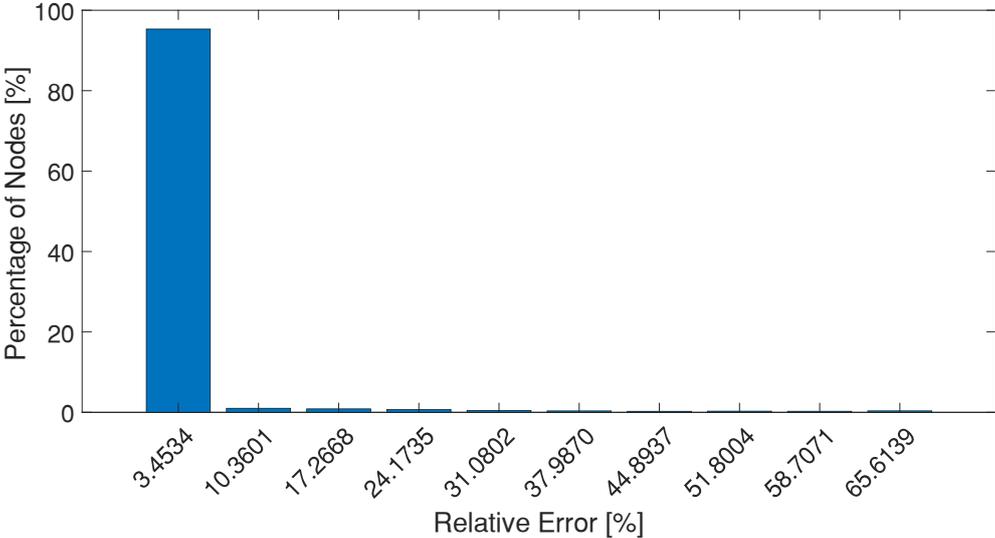


Figure 76. Right Upper Limb Position 1, nodal displacement relative error bar graph

Table 63. Right Upper Limb Position 1, nodal displacement relative error statistical analysis

	Value
Minimum	0.0
Maximum	69.056 %
Average	1.7389 %
Standard Deviation	7.3693 %

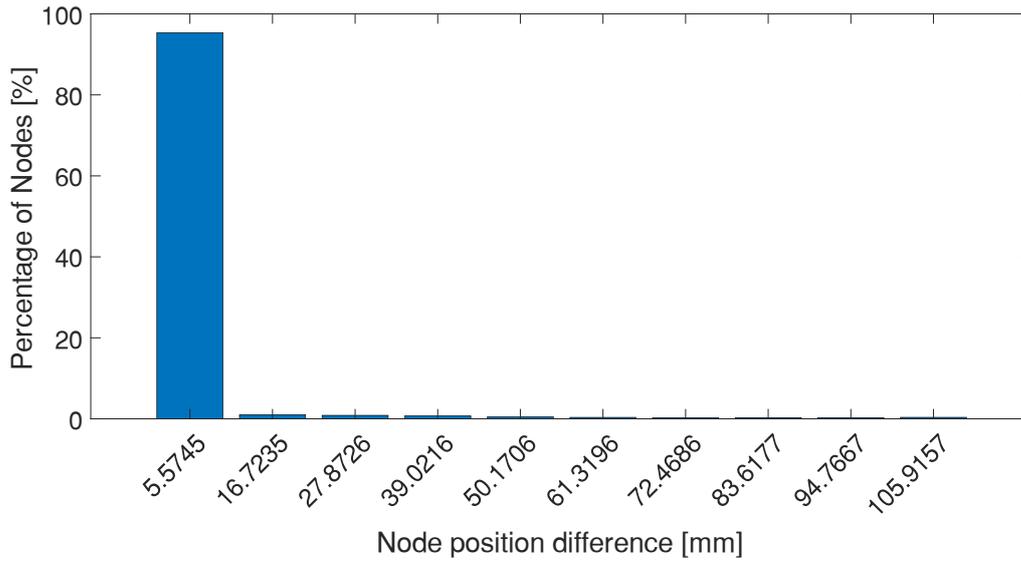


Figure 77. Right Upper Limb Position 1, nodal coordinates error bar graph

Table 64. Right Upper Limb Position 1, nodal coordinates error statistical analysis

	Value
Minimum	0.0
Maximum	111.4902 mm
Average	2.8071 mm
Standard Deviation	11.8957 mm

Table 65. Right Upper Limb Position 1, bones volume relative error

Bone	Reference Volume [mm³]	Predicted Volume [mm³]	Relative Error [%]
Humerus	8.48 e+04	7.41 e+04	12.60
Radius	1.93 e+04	1.48 e+04	23.38

Right Upper Limb Position 2

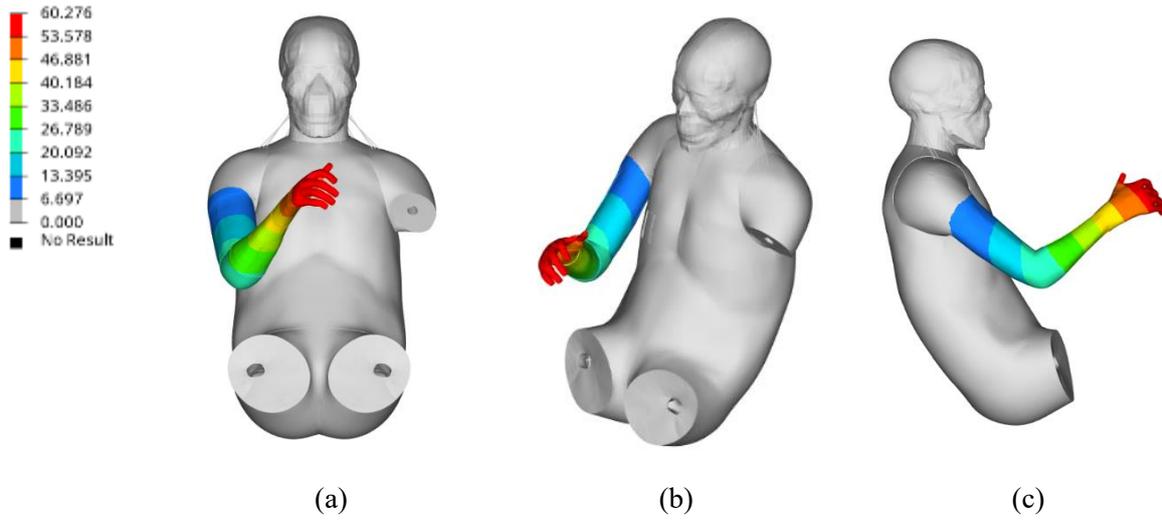


Figure 78. Right Upper Limb Position 2, nodal displacement relative error plots: a) front view; b) iso view; c) top view

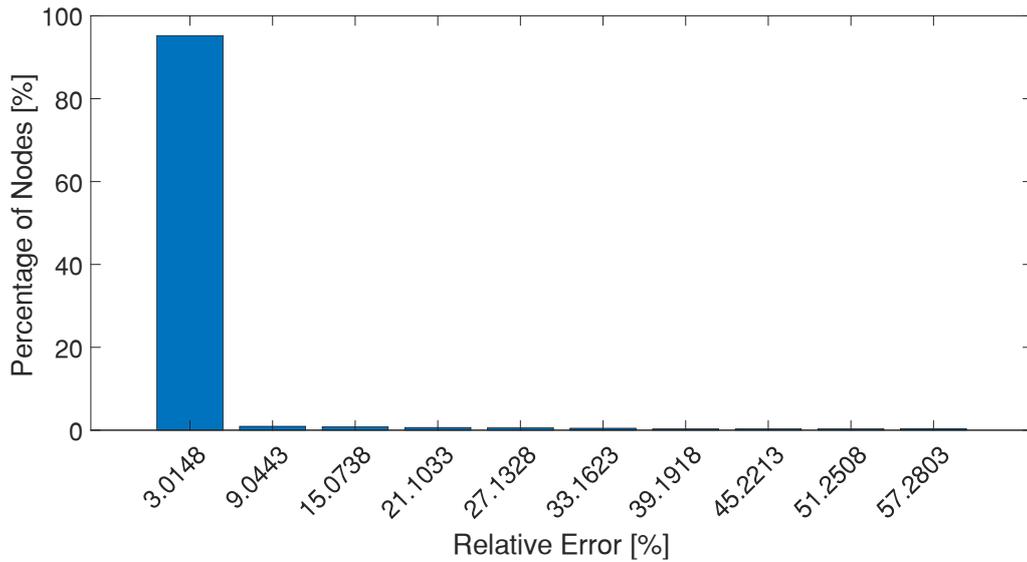


Figure 79. Right Upper Limb Position 2, nodal displacement relative error bar graph

Table 66. Right Upper Limb Position 2, nodal displacement relative error statistical analysis

	Value
Minimum	0.0
Maximum	60.276 %
Average	1.5503 %
Standard Deviation	6.7174 %

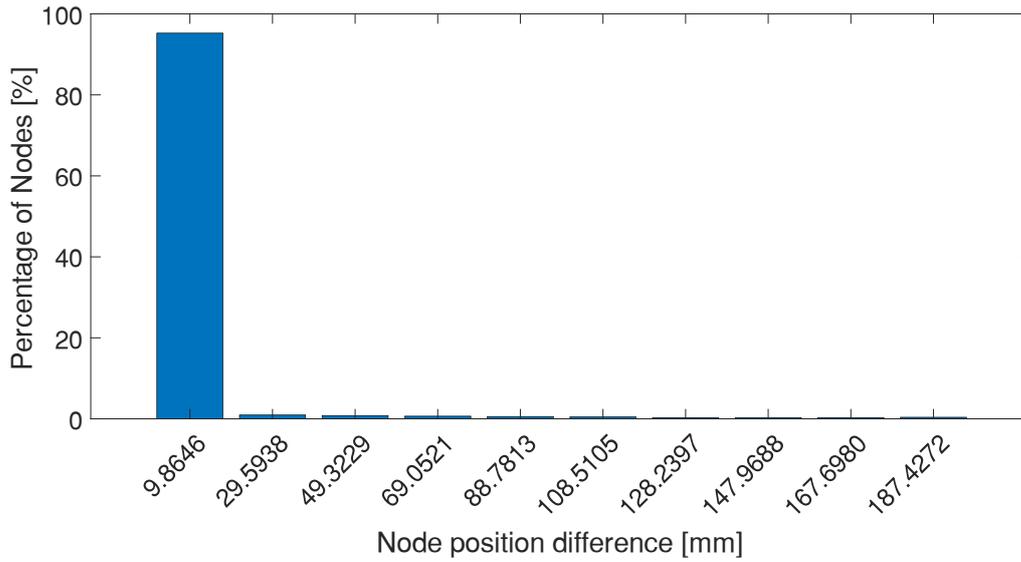


Figure 80. Right Upper Limb Position 2, nodal coordinates error bar graph

Table 67. Right Upper Limb Position 2, nodal coordinates error statistical analysis

	Value
Minimum	0.0 mm
Maximum	197.2918 mm
Average	5.0727 mm
Standard Deviation	21.9802 mm

Table 68. Right Upper Limb Position 2, bones volume relative error

Bone	Reference Volume [mm³]	Predicted Volume [mm³]	Relative Error [%]
Humerus	8.48 e+04	7.65 e+04	9.76
Radius	1.93 e+04	1.68 e+04	13.14

Left Upper Limb Position 1

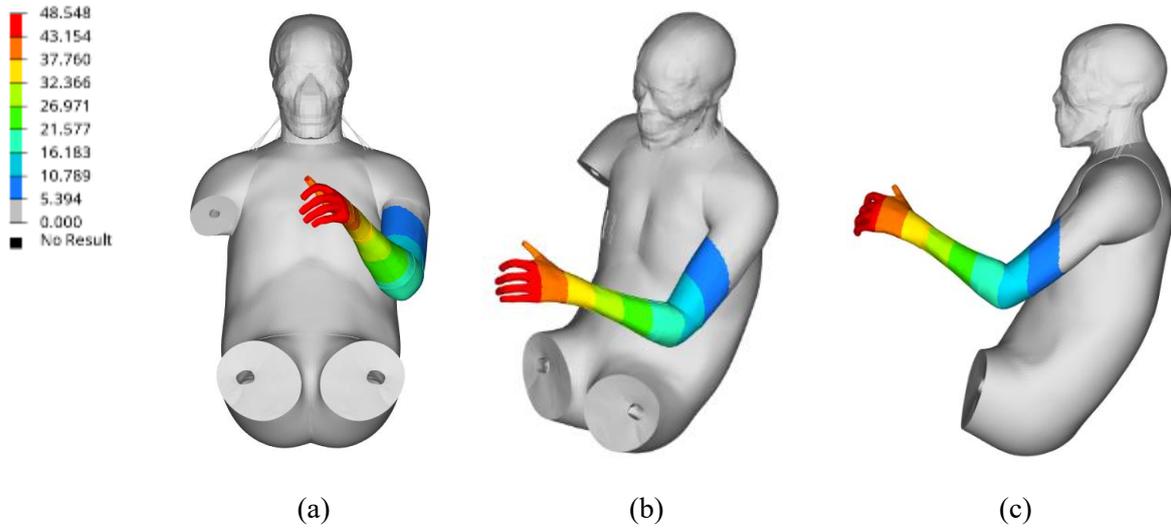


Figure 81. Left Upper Limb Position 1, nodal displacement relative error plots: a) front view; b) iso view; c) top view

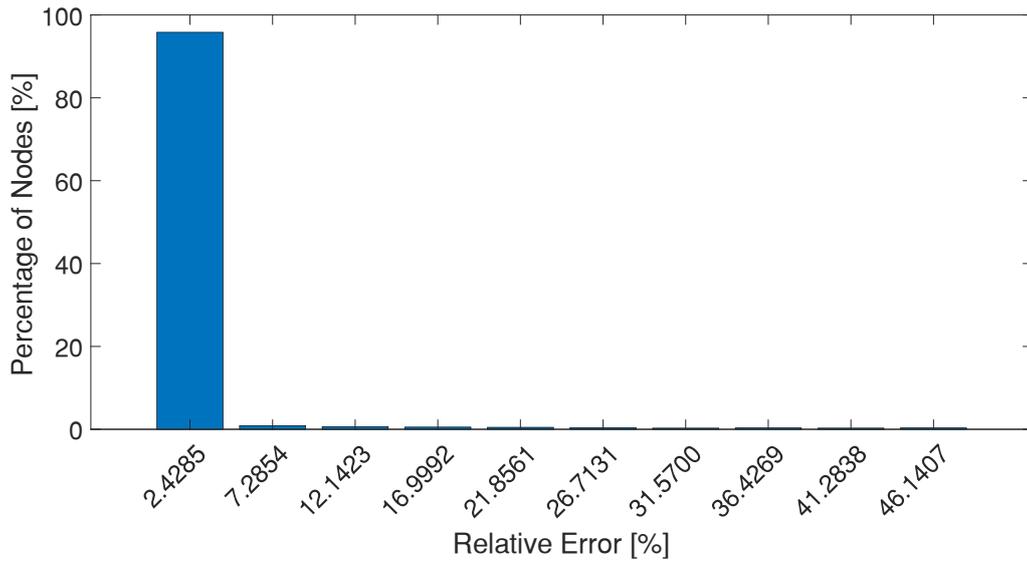


Figure 82. Left Upper Limb Position 1, nodal displacement relative error bar graph

Table 69. Left Upper Limb Position 1, nodal displacement relative error statistical analysis

	Value
Minimum	0.0
Maximum	48.5692 %
Average	1.0958 %
Standard Deviation	5.1390 %

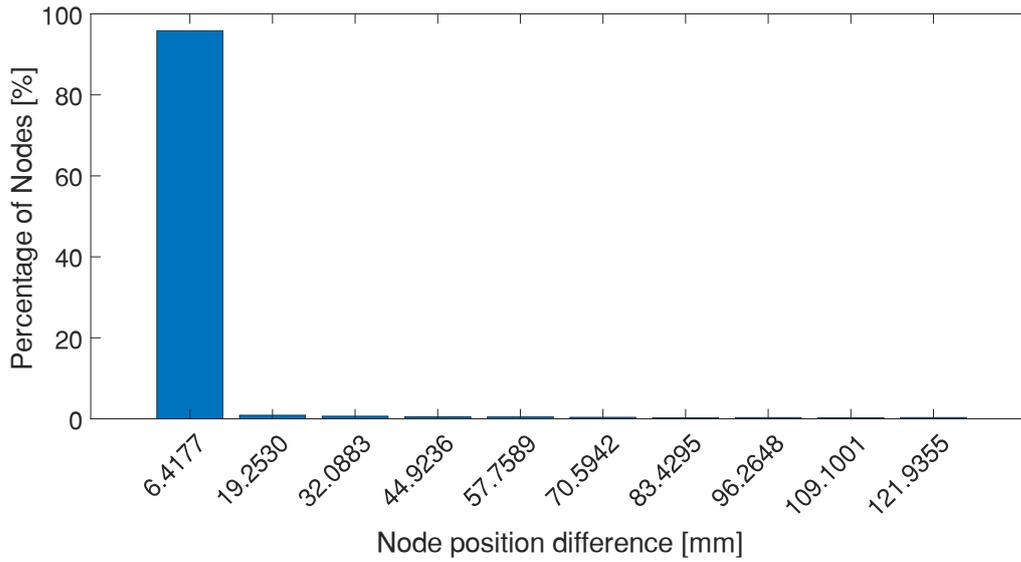


Figure 83. Left Upper Limb Position 1, nodal coordinates error bar graph

Table 70. Left Upper Limb Position 1, nodal coordinates error statistical analysis

	Value
Minimum	0.0
Maximum	128.3531 mm
Average	2.8959 mm
Standard Deviation	13.5807 mm

Table 71. Left Upper Limb Position 1, bones volume relative error

Bone	Reference Volume [mm³]	Predicted Volume [mm³]	Relative Error [%]
Humerus	8.48 e+04	8.12 e+04	4.16
Radius	1.93 e+04	1.81 e+04	6.32

Left Upper Limb Position 2

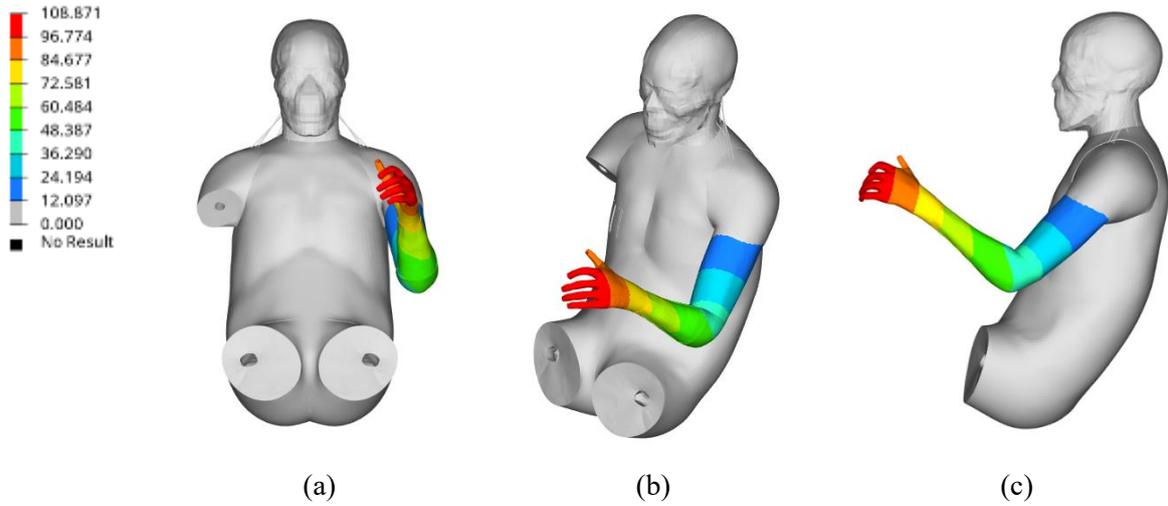


Figure 84. Left Upper Limb Position 2, nodal displacement relative error plots: a) front view; b) iso view; c) top view

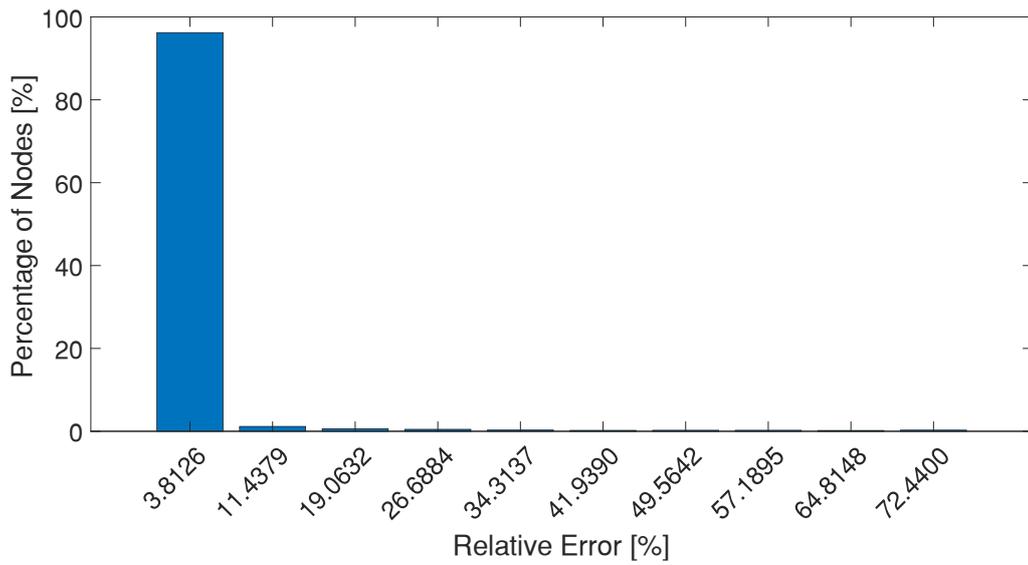


Figure 85. Left Upper Limb Position 2, nodal displacement relative error bar graph

Table 72. Left Upper Limb Position 2, nodal displacement relative error statistical analysis

	Value
Minimum	0.0
Maximum	76.2527 %
Average	1.3563 %
Standard Deviation	7.1298 %

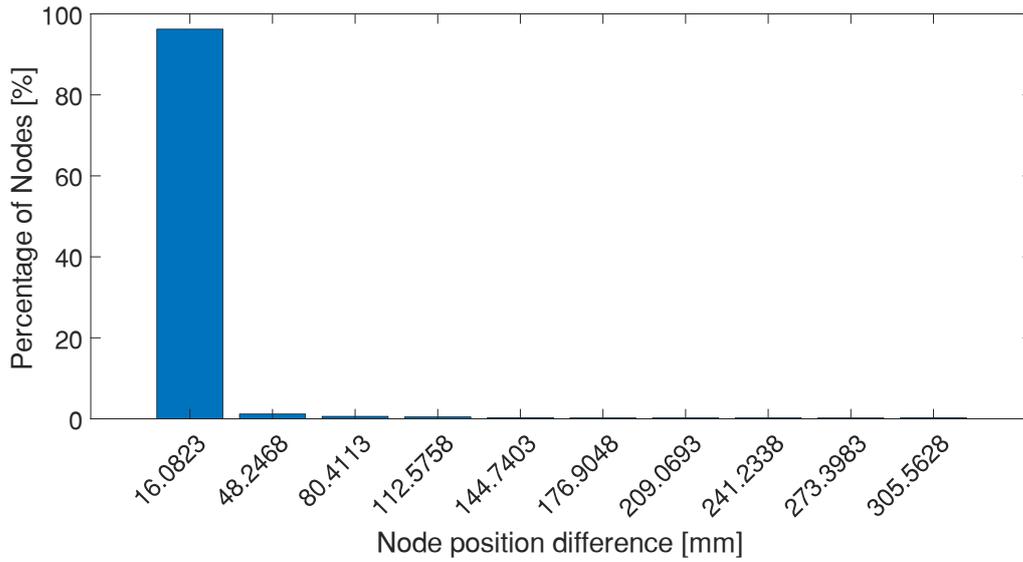


Figure 86. Left Upper Limb Position 2, nodal coordinates error bar graph

Table 73. Left Upper Limb Position 2, nodal coordinates error statistical analysis

	Value
Minimum	0.0
Maximum	321.6450 mm
Average	5.7212 mm
Standard Deviation	30.0745 mm

Table 74. Left Upper Limb Position 2, bones volume relative error

Bone	Reference Volume [mm³]	Predicted Volume [mm³]	Relative Error [%]
Humerus	8.48 e+04	8.28 e+04	2.34
Radius	1.93 e+04	1.76 e+04	8.57

Right Lower Limb Position 1

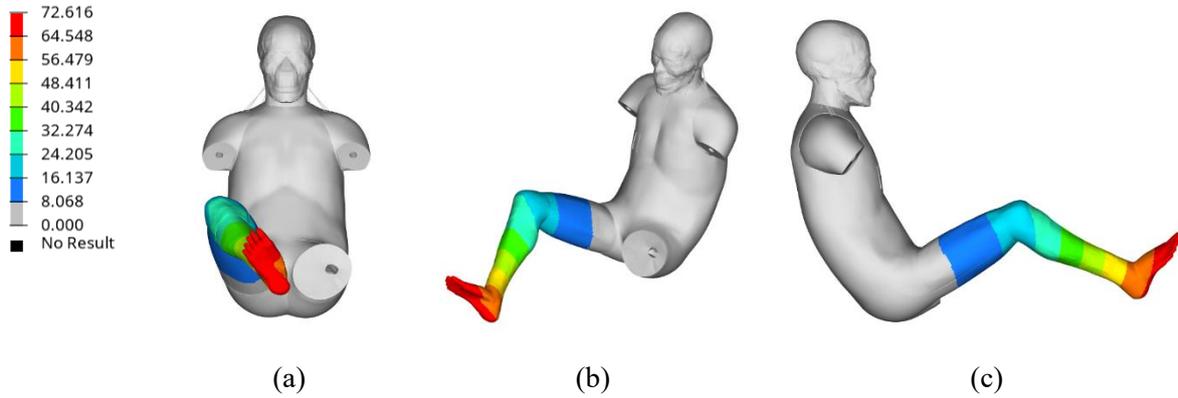


Figure 87. Right Lower Limb Position 1, nodal displacement relative error plots: a) front view; b) iso view; c) top view

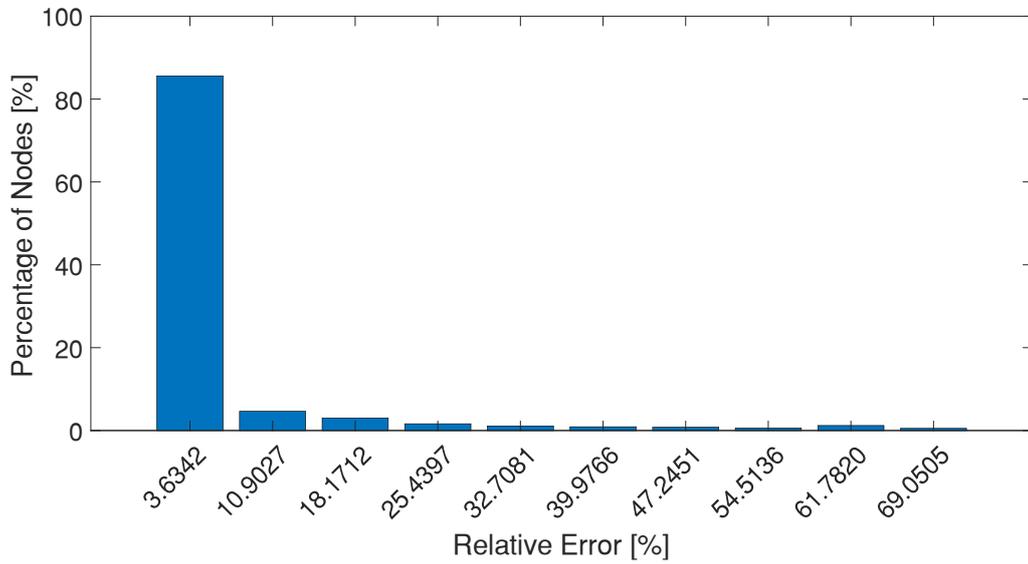


Figure 88. Right Lower Limb Position 1, nodal displacement relative error bar graph

Table 75. Right Lower Limb Position 1, nodal displacement relative error statistical analysis

	Value
Minimum	0.0
Maximum	72.6848 %
Average	4.2403 %
Standard Deviation	11.8698 %

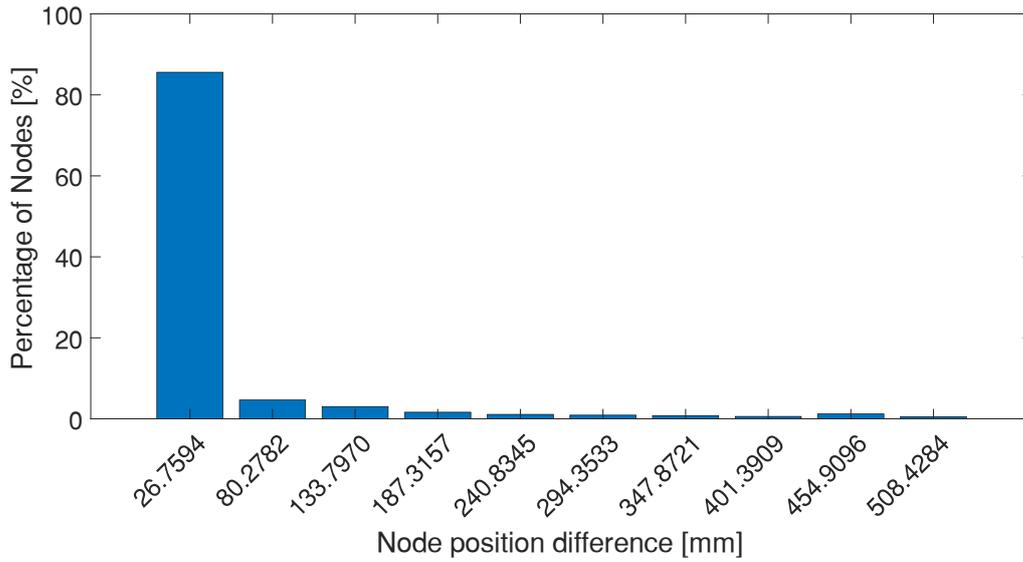


Figure 89. Right Lower Limb Position 1, nodal coordinates error bar graph

Table 76. Right Lower Limb Position 1, nodal coordinates error statistical analysis

	Value
Minimum	0.0 mm
Maximum	535.1878 mm
Average	31.2223 mm
Standard Deviation	87.3990 mm

Table 77. Right Lower Limb Position 1, bones volume relative error

Bone	Reference Volume [mm³]	Predicted Volume [mm³]	Relative Error [%]
Femur	4.04 e+05	3.34 e+05	17.38
Tibia	2.36 e+05	1.74 e+05	26.11

Right Lower Limb Position 2

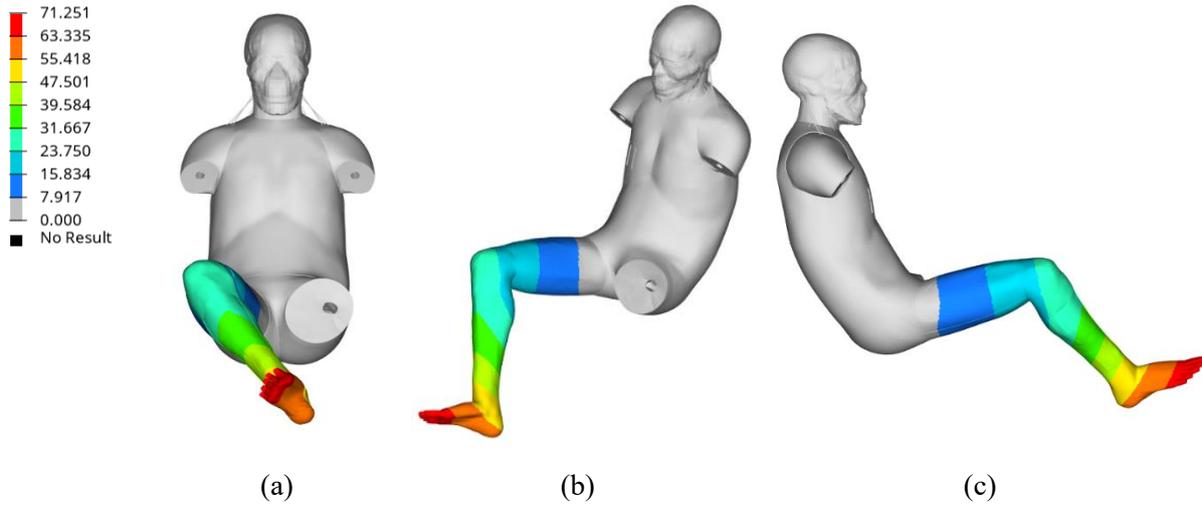


Figure 90. Right Lower Limb Position 2, nodal displacement relative error plots: a) front view; b) iso view; c) top view

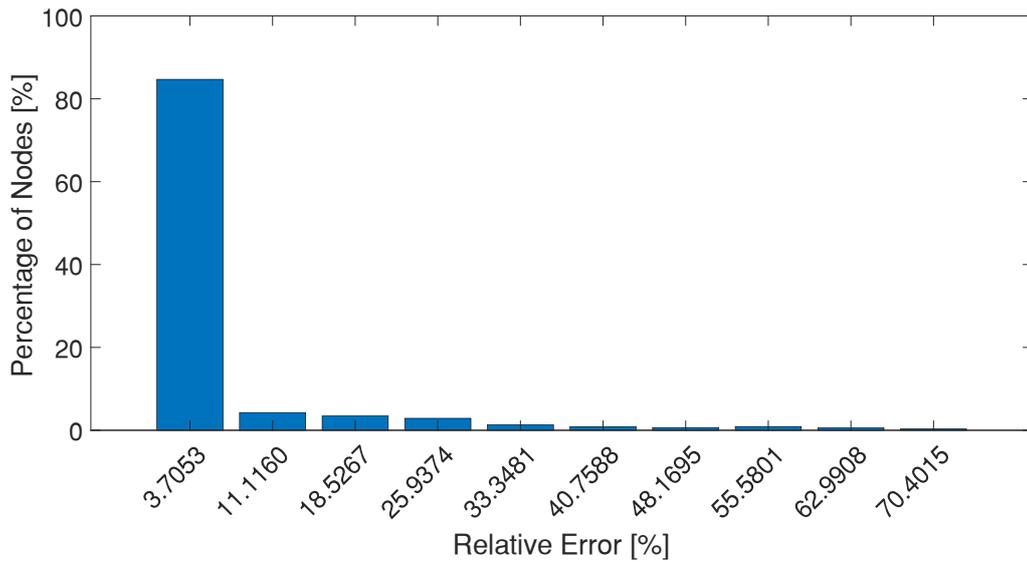


Figure 91. Right Lower Limb Position 2, nodal displacement relative error bar graph

Table 78. Right Lower Limb Position 2, nodal displacement relative error statistical analysis

	Value
Minimum	0.0
Maximum	74.1069 %
Average	4.3497 %
Standard Deviation	11.3953 %

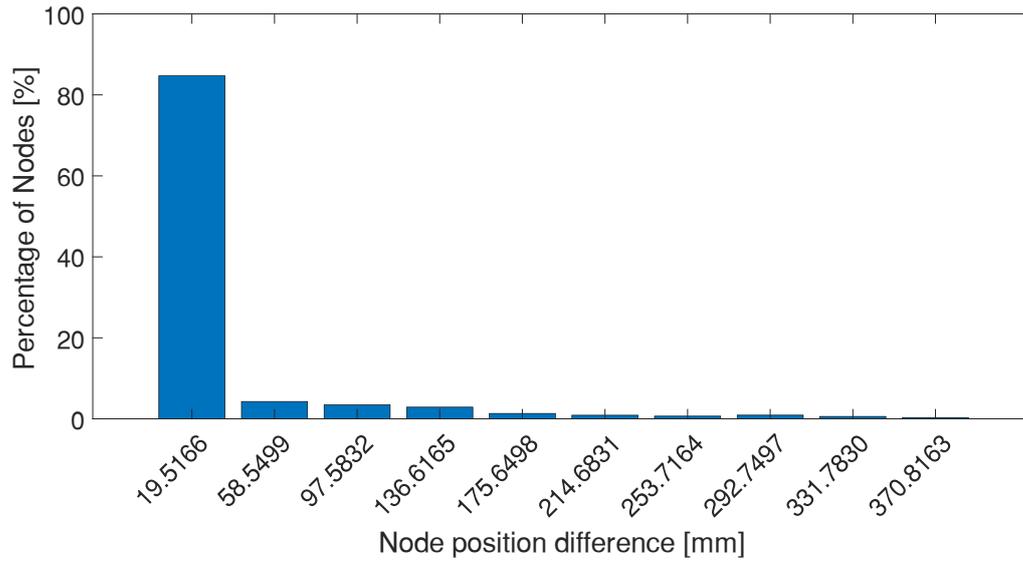


Figure 92. Right Lower Limb Position 2, nodal coordinates error bar graph

Table 79. Right Lower Limb Position 2, nodal coordinates error statistical analysis

	Value
Minimum	0.0 mm
Maximum	390.3330 mm
Average	22.9053 mm
Standard Deviation	60.0207 mm

Table 80. Right Lower Limb Position 2, bones volume relative error

Bone	Reference Volume [mm³]	Predicted Volume [mm³]	Relative Error [%]
Femur	4.04 e+05	3.72 e+05	7.86
Tibia	2.36 e+05	2.19 e+05	6.93

Left Lower Limb Position 1

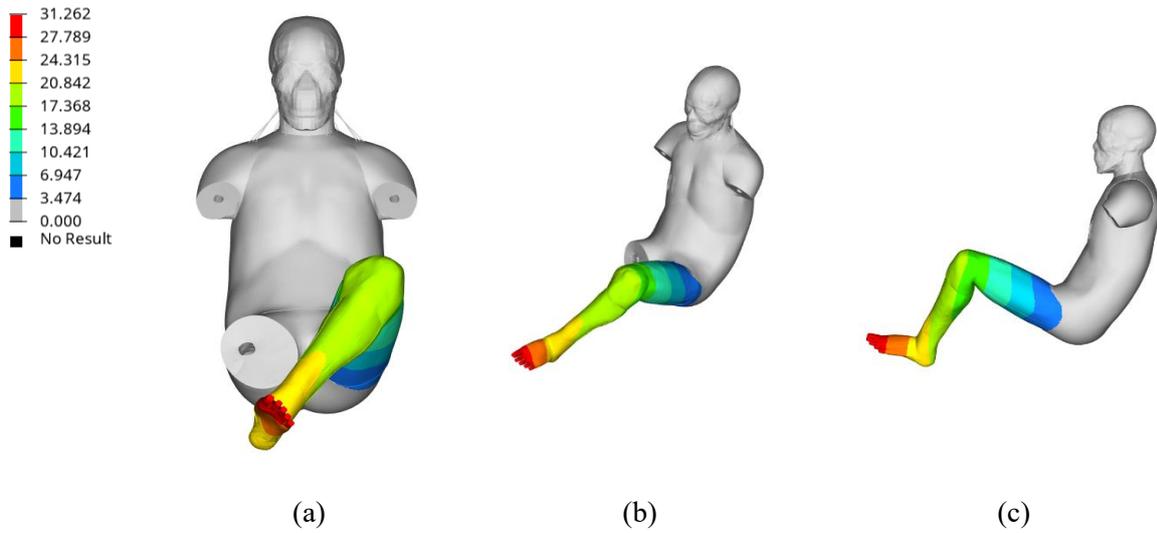


Figure 93. Left Lower Limb Position 1, nodal displacement relative error plots: a) front view; b) iso view; c) top view

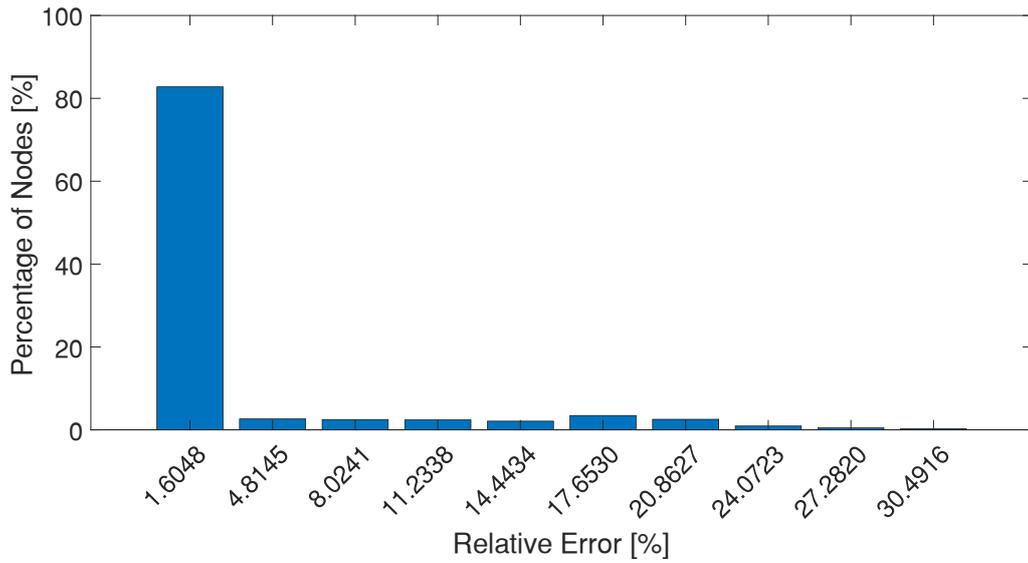


Figure 94. Left Lower Limb Position 1, nodal displacement relative error bar graph

Table 81. Left Lower Limb Position 1, nodal displacement relative error statistical analysis

	Value
Minimum	0.0
Maximum	32.0964 %
Average	2.5905 %
Standard Deviation	5.9857 %

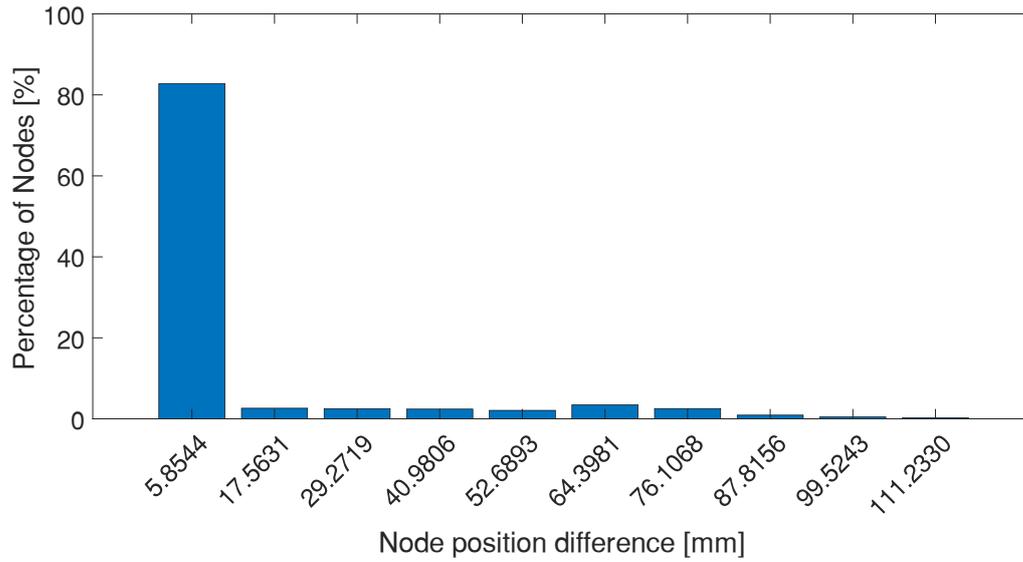


Figure 95. Left Lower Limb Position 1, nodal coordinates error bar graph

Table 82. Left Lower Limb Position 1, nodal coordinates error statistical analysis

	Value
Minimum	0.0
Maximum	117.0874 mm
Average	9.4503 mm
Standard Deviation	21.8357 mm

Table 83. Left Lower Limb Position 1, bones volume relative error

Bone	Reference Volume [mm³]	Predicted Volume [mm³]	Relative Error [%]
Femur	4.04 e+05	3.93 e+05	2.67
Tibia	2.36 e+05	2.09 e+05	11.48

Left Lower Limb Position 2

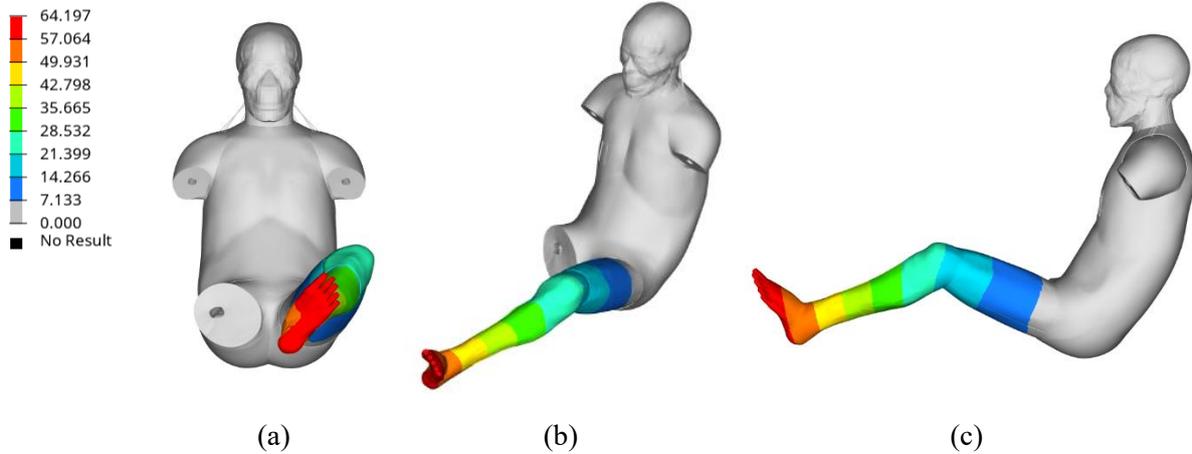


Figure 96. Left Lower Limb Position 2, nodal displacement relative error plots: a) front view; b) iso view; c) top view

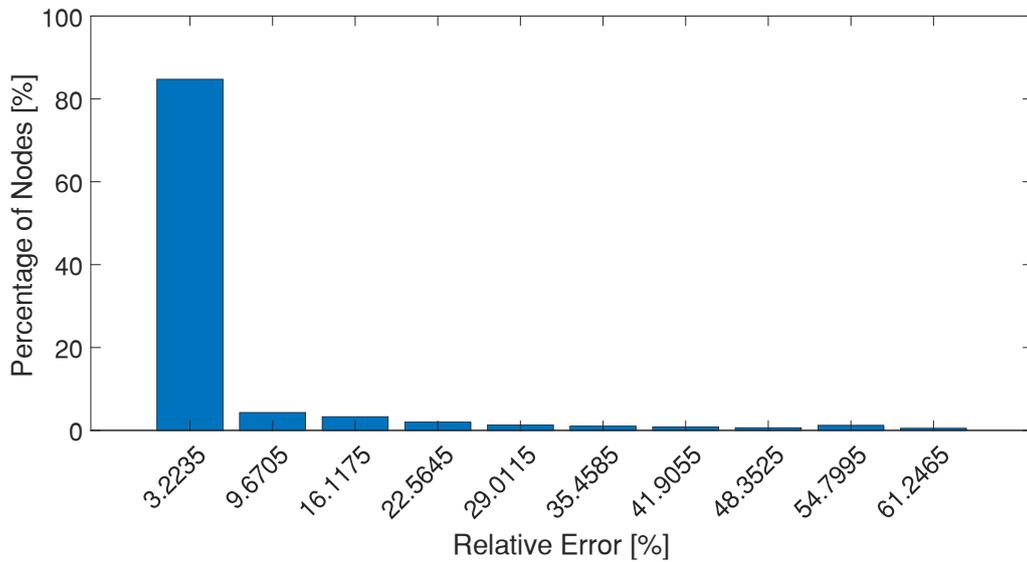


Figure 97. Left Lower Limb Position 2, nodal displacement relative error bar graph

Table 84. Left Lower Limb Position 2, nodal displacement relative error statistical analysis

	Value
Minimum	0.0
Maximum	64.4700 %
Average	4.0509 %
Standard Deviation	10.8357 %

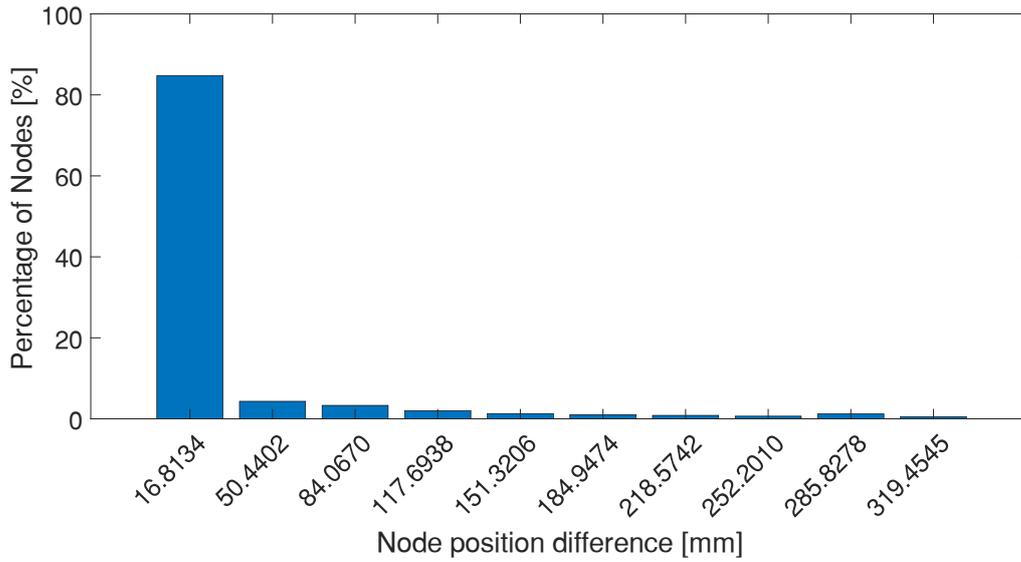


Figure 98. Left Lower Limb Position 2, nodal coordinates error bar graph

Table 85. Left Lower Limb Position 2, nodal coordinates error statistical analysis

	Value
Minimum	0.0
Maximum	336.2679 mm
Average	21.1291 mm
Standard Deviation	56.5176 mm

Table 86. Left Lower Limb Position 2, bones volume relative error

Bone	Reference Volume [mm³]	Predicted Volume [mm³]	Relative Error [%]
Femur	4.04 e+05	3.57 e+05	11.47
Tibia	2.36 e+05	2.01 e+05	14.73

References

- [1] V. V. Dixit, S. Chand and D. J. Nair, "Autonomous Vehicles: Disengagements, Accidents and Reaction Times," *Plose One*, December 2016.
- [2] Toyota Motor Company, "Documentation: Total Human Model for Safety (THUMS), AM50 Occupant Model, Version 4.1," April, 2020.
- [3] N. Ichinose, M. Takeda, N. Taki and T. Yoshimach, "Efficient Pre-Simulation Method for Human Body Model with Odyssey," in *Automotive CAE Grand Challenge*, Hanau, Germany, 2020.
- [4] PIPER Project, "PIPER Software v1.1.0," [Online]. Available: http://www.piper-project.eu/#piper_110.
- [5] Euro NCAP, "MPDB FRONTAL IMPACT TESTING PROTOCOL," October 2019. [Online]. Available: <https://cdn.euroncap.com/media/55858/euro-ncap-mpdb-testing-protocol-v111.pdf>. [Accessed 25 January 2021].
- [6] SAE International, "Information report J3016: Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles," SAE International, 2018.
- [7] S. Jorlöv, K. Bohman and A. Larsson, "Seating positions and activities in highly automated cars - A qualitative study of future automated driving scenarios," in *International Research Council on the Biomechanics of Injury, IRCOBI*, Antwerp, 2017.
- [8] M. P. Reed, S. M. Ebert and M. L. Jones, "Road Vehicle Passenger Behaviors: A Video Study," UMTRI - Transportation Research Institute, University of Michigan, 2019.
- [9] M. P. Reed, S. M. Ebert and M. L. Jones, "Naturalistic Passenger Behaviour: Posture and Activities," UMTRI - Transportation Research Institute, University of Michigan, 2020.
- [10] S. Wang and Z. Li, "Exploring the mechanism of crashes with automated vehicles using statistical modeling approaches," *Plos ONE*, 28 March 2019.
- [11] C. Xu, Z. Ding, C. Wang and Z. Li, "Statistical analysis of the Patterns and Characteristics of Connected and Autonomous Vehicle Involved Crashes," *Journal of Safety Research*, vol. 71, pp. 41 - 47, 2019.

- [12] National Transportation Safety Board, “Collision Between a Car Operating With Automated Vehicle Control Systems and a Tractor-Semitrailer Truck Near Williston, Florida, May 7, 2016,” National Transportation Safety Board, Washington D.C., 2016.
- [13] National Transportation Safety Board, “Rear-End Collision Between a Car Operating with Advanced Driver Assistance Systems and a Stationary Fire Truck, Culver City, California, January 22, 2018,” National Transportation Safety Board, Washington D.C, 2018.
- [14] United States District Court for the Northern District of California, San Jose Division, “Complaint for Damages,” United States District Court for the Northern District of California, San Jose Division, Irvine, California, 2020.
- [15] Police Sergeant Samuel Winkler, “Final Report - Crash involving Tesla Model S – 10400 South Bangerter Highway,” South Jordan Police Department, South Jordan, Utah, 2018.
- [16] L. L. Zhang, L. Chen, A. Vertiz and R. Balci, “Survey of Front Passenger Posture Usage in Passenger Vehicles,” in *SAE World Congress*, Detroit, 2004.
- [17] L. Bingley, R. Morris and G. Cross, “Determination of Real World Occupant Postures by Photo Studies to Aid Smart Restraint Development,” in *19th International Technical Conference on the Enhanced Safety Vehicles*, Washington DC, 2005.
- [18] T. Janak, Y. Lafon, P. Y. Petit and P. D. Beillas, “Transformation Smoothing to use after Positioning of Finite Element Human Body Models,” in *IRCOBI*, Athens, 2018.
- [19] OASYS, “Human Body Model Dummy Trees,” [Online]. Available: <https://www.oasys-software.com/dyna/models/hbm-dummy-trees/>. [Accessed 02 02 2021].
- [20] M. Torunier, M. Nesme, B. Gilles and F. Faure, “Stable Constrained Dynamics. ACM Transactions on Graphics, Association for Computing Machinery,” in *SIGGRAPH*, Los Angeles, 2015.
- [21] PIPER Project, “PIPER Software Framework and Application: User Guide,” 2017.
- [22] F. Germanetti, G. Belingardi, A. Scattina and F. Cappellino, “Finite Element Simulation of Impact of Autonomous Vehicle with Human Body Model in Out-of-Position Configuration,” Politecnico di Torino, Turin, 2019.

- [23] G. Bacquaert, C. Bach, D. Draper, S. Peldschus and F. Duddeck, "Positioning Human Body Models for Crashworthiness Using Model Order Reduction," *Computer Methods in Biomechanics and Biomedical Engineering*, vol. 23, no. 11, pp. 734-743, 2020.
- [24] K. Bouchiba, K. Kayvantash and K. Hanna, "The Emergence of Artificial Intelligence and Machine Learning in CAE Simulation," *Manufacturing Intelligence - hexagonmi.com - mscsoftware.com*, 18 November 2020.
- [25] "ODYSSEE (CADLM)," MSC Software, [Online]. Available: <https://www.mscsoftware.com/product/odyssee-cadlm>. [Accessed 05 02 2021].
- [26] A. A. Giunta, S. F. Wojtkiewicz Jr. and M. S. Eldred, "Overview of Modern Design of Experiments Methods for Computational Simulations," in *American Institute of Aeronautics and Astronautics (AIAA)*, Reno, Nevada, 2003.
- [27] CADLM, "Lunar User Guide," 2020.
- [28] D. Forsyth, *Applied Machine Learning*, Springer, 2019.
- [29] "Inverse distance weighting," Wikipedia, the free encyclopedia, 13 01 2021. [Online]. Available: https://en.wikipedia.org/wiki/Inverse_distance_weighting. [Accessed 15 03 2021].

Computational resources provided by HPC@POLITO (<http://www.hpc.polito.it>)

Acknowledgements

I would like to thank CADLM for the support provided. A special mention goes to Mr Dorian Salin for the time spent explaining how to use their software and helping me overcoming difficulties. I would like then to thank also HPC staff for the technical support provided with their clusters. Last but not least I would like to thank both my tutors: Prof. Alessandro Scattina for his constant support, for the opportunities offered and for the trust given me; Prof. Giovanni Belingardi for his helpful insight.

This work marks the end of a long journey that started 6 years ago when I decided to study engineering in Turin. I shared the ups and downs of this journey with my brother, my father, my mother, all my relatives and my friends. I would like to share with them also this last work as a sign of gratitude for all they have given me, hoping we will walk together new paths.

Thank Dad, thank Mum, Thank Elia

Thanks to everyone who shared this journey with me

Con questo lavoro si conclude un percorso iniziato sei anni fa quando decisi di studiare ingegneria a Torino. Ho condiviso gli alti e bassi di questo percorso con mio fratello, mio padre, mia madre, i miei parenti e tutti i miei amici. Voglio condividere con loro anche quest'ultimo lavoro in segno di gratitudine per tutto quello che mi hanno dato, sperando di poter percorrere nuovamente insieme i futuri cammini.

Grazie papà, grazie mamma, grazie Elia

Grazie a tutti coloro con cui ho condiviso questo cammino

