A Method to Use Kriging With Large Sets of Control Points to Morph Finite Element Models of the Human Body

Tomás Janak, Yoann Lafon, Philippe Petit, Philippe Beillas

A method to use kriging with large sets of control points to morph finite element models of the human body

Tomáš Janák[1], Yoann Lafon[1], Philippe Petit[2], Philippe Beillas[1]
1. Univ Lyon, Université Claude Bernard Lyon 1, Univ Gustave Eiffel, IFSTTAR, LBMC UMR_T 9406, F69622 Lyon, France. 2. LAB PSA – Renault, Nanterre, France

ABSTRACT
As developing finite element human body models for automotive impact is a time consuming process, morphing using interpolation methods such as kriging has often been used to rapidly generate models of different shapes and sizes. Kriging can be computationally expensive when many control points are used, i.e. for very detailed target geometry (e.g. shape of bones and skin). It can also lead to element quality issues (up to inverted elements) preventing the use of the morphed models for finite element simulation. This paper presents a workflow combining iterative subsampling and spatial subdivision methodology that effectively reduces the computational costs and allows for the generation of usable models through kriging with hundreds of thousands of control points. As subdivision introduces discontinuities in the interpolation function that can cause distortion of elements on the boundaries of individual subdivision areas, algorithms for smoothing the interpolation over those boundaries are proposed and compared. Those techniques and their combinations were tested and evaluated in a scenario of mass change on the detailed 50th percentile male model of the Global Human Body Models Consortium: the model, which has body mass index 25.34, was morphed towards a statistical surface model of a person with body mass index 20, 22.7 and 35. 234 777 control points were used to successfully morph the model in less than 15 minutes on an office PC. Open source implementation is provided.

1. Introduction

Finite Element (FE) human body models (HBM) have become valuable tools to study the human response to mechanical loading, in particular in the context of motor vehicle accidents. HBMs are extensively used in research and are being introduced in car evaluation protocols (e.g. Euro NCAP pedestrian protocol [1]). There are now two main HBM families, the Global Human Body Models Consortium (GHBMC)[2] and the Total Human Model For Safety (THUMS) [3]. Models from both families describe all main anatomical structures and are validated against many experimental datasets. Each family includes a few models corresponding to the sizes of crash test dummies, i.e. 50th percentile male, 5th percentile female and 95th percentile male. This means that HBM or crash test dummies do not explicitly represent large parts of the population. This can be an issue: for example the obese population, which is not well represented by current surrogates, is likely to interact with restraint systems differently than the non-obese population [4,5].

As developing completely new HBMs is a lengthy process, recent efforts have been focused on morphing existing HBMs to represent different population characteristics. For example, Vavalle et al. [6] morphed the GHBMC 50th percentile male model towards the geometry of a 95th percentile male derived from multimodality imaging. Hu et al. [7] and Zhang et al.[8] morphed the same model towards Statistical Shape Models (SSMs) of the ribcage, pelvis, femur, tibia and external shape; Hwang et al. [9] did the same with the THUMS model. Schoell et al. [10] morphed the GHBMC 50th percentile male towards a 65 year old, including also changes to material properties related to age. Beillas and Berthet [11] morphed the GHBMC 50th percentile male and 5th percentile female models towards the anthropometry of 52 subjects. Morphing was also proposed to be used in the context of positional change in combination with other methods [12]. All these studies used Radial Basis Functions (RBF) or kriging interpolation approaches, which, for the parameters used in these studies, lead to an identical interpolation function [13] (see Appendix A for more details about the method).

The interpolation field is defined by a set of Control Points (CPs) in a known initial (source) and a final (target) position. In the extreme case, one CP per FE node would be needed for a complete control of the shape of the mesh. This would mean for example 217 495 CPs to completely control the bone surfaces of the GHBMC M50 detailed occupant model version 4.1 used in this study (GHBMC M50-O in further text). But the aforementioned studies used much lower amount of CPs. For example, Zhang et al. [8] remarked that using more than 30 000 CPs at once resulted in *"significant increase in computational time"*, which led them to partition the model into several parts. The root cause of the long computation times is that the numerical solution requires a Lower-Upper (LU) factorization of a dense n by n covariance matrix (where n is the number of CPs) [14], an operation with $O(n^3)$ time complexity that demands $O(n^2)$ memory [15].

Therefore, directly using a high number of control points is not practical with personal computers.

General methods to reduce the cost were proposed in the literature for other applications. Some approaches use a simplified covariance model based on some *a priori* assumptions about the character of the data. For example, solutions found in [16,17] use covariance functions which are non-zero only in a limited area around each CP, yielding sparse linear systems in place of the original dense ones. Alternatively, the "low-rank kriging" methods use a small amount of basis functions tailored for the particular data set, yielding a very small linear system [18–20]. However, such approaches tend to over-smooth the data in cases where the observations are measured very densely [21]. This is problematic for the problem at hand: preserving a high level of geometrical details is a primary aim of using such large amounts of CPs to morph HBM.

Haas [22] used a "moving window" approach for kriging, where the estimation at each point is computed based on only a local neighborhood. However, this moving window method is computationally expensive, as it requires building the covariance matrix and solving the interpolation problem for each point. This method is impracticable when those points are the millions of nodes of the most detailed HBM.

Stein [21] instead proposed an approach that splits the CPs into blocks, leading to a block sparse covariance matrix and therefore low computational demands. Nevertheless, the authors noted that the assumption of block independence is idealistic and therefore flawed when applied to real problems. Sang, Jun, and Huang [23] proposed to use a combination of kriging methods, with a low-rank interpolation over the entire data set (i.e. reduced level of detail), followed by an interpolation under the assumption of independent blocks to finally capture the full details. However, it still leads to discontinuities at the region boundaries [24], which is an issue for morphing finite element models when nodes belonging to a same element are on the block boundaries, they would be transformed by different blocks, depreciating dramatically the quality of these elements.

Other past studies restricting the kriging computation to anatomical "domains", either by anatomical regions or by structures (e.g. torso, extremities, pelvis etc. [8], or *"a single bone, ligament, muscle, or internal organ*" [25]), can be understood as a special case of this approach where the definition of the domains was tailored to the specific model and transformation. An implementation facilitating the domain definition is also available within the open source PIPER software [26]. While successful, the domain definition requires user expertise that may prevent automating such transformations, especially for large numbers of CPs for which many domains would be needed.

Overall, the methods reported in the literature to reduce the computation cost of kriging do not seem directly applicable to morph a HBM by a large amount of CPs while maintaining element quality necessary for finite element simulation. The aim of this study is to develop and implement a methodology to solve this problem. As good initial results were observed by morphing HBM per parts after manual splitting, an approach similar to [23] was selected as the basis for the current study, with first a low-detail transformation capturing the global features and then block transformation to deal with local features. This was implemented by employing both subsampling and spatial subdivision techniques. Additional measures to deal with the discontinuities at boundaries were investigated as part of this study. The methods were tested on the GHBMC M50-O in a mass change scenario, morphing it towards three statistical surface models with Body Mass Index (BMI) 20, 22.7 and 35 (the original model has BMI 25.34).

2. Method

2.1. Test case

The test case simulates a change of skin surface while keeping all bones unchanged. This could be understood as a simplified representation of a rapid weight gain or loss for an individual. It was designed to evaluate the morphing method but does not attempt to account for all factors that could be associated with weight changes in reality.

The GHBMC M50-O model [2] version 4.1 was used in all tests. The model is composed of 1 255 225 nodes and 2 313 366 elements organized in parts describing various anatomical structures. The model stature and masses are 174.9 cm and 77.53 Kg, respectively, for a corresponding BMI of 25.34 kg/m².

Four skin surfaces were generated using a publicly available statistical shape model (SSM) ([27], available at *http://humanshape.org* [28]*)*. One skin surface was generated for anthropometric parameters similar to those of the GHBMC M50-O (BMI 25.34 kg/m²) and it was defined as the baseline mesh (source). Additional target surfaces were generated with the same anthropometric parameters except for the BMI. Initially, two targets were created, one with an increased BMI (35) and one with decreased (20). As it will be shown in Section 4, as the thinning proved to be challenging, an additional target half way between the BMI 20 and the baseline model was used as well, i.e. BMI

22.7. The surface meshes had several quality issues that required changes before using them as targets: areas around armpits and crotch were smoothed, and the hand surface meshes were manually replaced by those of the GHBMC M50-O.

Since the GHBMC M50-O and humanshape.org postures are different (arms are in a different position, see Fig. 1), the GHBMC M50-O was first morphed to the baseline skin surface as follows: the baseline skin surface mesh was resampled to obtain a more homogenous distribution of 17 282 vertices. Then, it was aligned with the GHBMC M50-0 using Iterative Closest Point in Meshlab (ISTI – CNR, Italy) to minimize rigid transformations needed to register the surfaces. After the alignment, the skin of GHBMC M50-O was registered to it using the *mHBM* registration software [29]. The GHBMC M50-O was morphed by standard kriging using the registered vertices as source CPs and the baseline vertices as target CPs. This morphed model, referred to as $GHBMC_{25.3}$ in the following text, was subsequently used as the source model for the actual morphing tests.

The mHBM software was also used to register the (resampled) target surfaces to the $GHBMC_{25.3}$ mesh, although it required less landmarks thanks to the now more similar posture (typically only shoulders, knees and in the BMI 35 case also sternum, elbows and thighs). After registration, small adjustment of the target skin surfaces were required as the BMI 20 and BMI 22.7 skin surfaces intersected the bones of the $GHBMC_{25.3}$ around the ribcage and pelvis. To correct this issue, the surfaces were locally adjusted by manual cage morphing followed by smoothing in Ls-Prepost (LSTC, CA), enforcing about a 10 mm gap between skin and bones in the thoracic region and pelvis. It was also assumed that the BMI 35 surface should be outside the $GHBMC_{25.3}$ skin, which was not the case everywhere. This was also solved by manual morphing in LS-Prepost, mainly on arms and legs. These manual adjustments required approximately half a day of work per target. This is indicative as this number, which is very specific to the test case, may also vary significantly from operator to operator. Appendix C shows the meshes before and after the manual adjustments.

Finally, the resampled skin was registered onto these adjusted skin surfaces using the mHBM software. All 17 282 vertices of the registered baseline skin surface were used as control points to morph the GHBMC25.3 onto the various BMI SSM skin surfaces. In addition, to avoid any deformation of the bones, all 217 495 nodes on the surface of bones of the GHBMC25.3 were used as fixed CPs (i.e. CPs with identical source and target positions). This led to a total of 234 777 CPs. Illustrations of three of the resulting surfaces created to serve as targets for full model morphing are provided in Fig. 1 along with the original GHBMC25.3 skin surface.

2.2. Subsampling

Decimating the CPs reduces the size of the correlation matrix by directly removing some of the CPs, which reduces the computational cost. The subsampling algorithm implemented here subdivides the CP space in a regular cubical grid and then assigns CPs to its cells. One CP per cell is kept at most (the one closest to the center of the cell) and the rest is discarded. A large numbers of CPs and a fine grid will lead to a homogenous distribution of the remaining CPs across the HBM, distributing the loss of details across the model evenly.

The size of the grid is controlled by a single parameter specified by the user: the number of grid cells along the X axis of the bounding box of the model. The size of the cubic cell is then computed as the bounding box size in the X axis divided by the input parameter. For example, if the user specifies the number of cells as 60 and the bounding box of the model is 900 mm wide, the space will be divided in to a grid of cubes with edge length 15 mm (900 / 60). These cubic cells will fill the whole bounding box of the model. For example, if the bounding box is 1500 mm large in the Y direction and 1200 mm in the Z direction, there will be a total of 60 x 100 x 80 cells (900 / 15, 1500 / 15, 1200 / 15, respectively). There can be at most the same number of CPs, since at most one CP is allowed in a cell. In practice, many cells will not contain even a single CP, therefore the total number of CPs will be lower. This method is very fast due to its simplicity, but does not allow for exactly specifying the desired number of CPs.

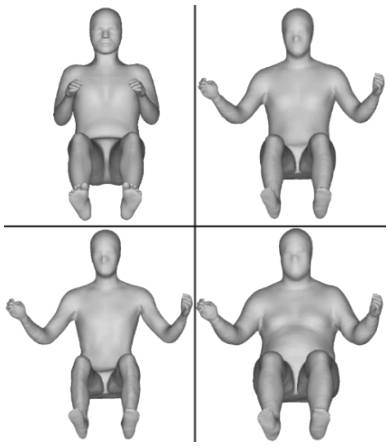

Fig. 1. Overview of the targets shapes after smoothing and local adjustments used for the current study: top right is the “baseline” surface (same BMI as the GHBMC M50-O); bottom left BMI 20 and bottom right the BMI 35 surface. The original GHBM25.3 skin surface is shown in top left for comparison.

2.3. Spatial subdivision

The basic principle of spatial subdivision is to divide the space into "cells", assign CPs and nodes of the HBM to those cells and perform kriging in each cell separately. The core idea of this approach is that the global transformation can be approximated within a cell precisely enough by using only the CPs present in that cell. This is based on the property of kriging that the influence of CPs on the transformation of a node diminishes with the CP-to-node distance [13]. The following subdivision scheme was implemented to generate the cells: the bounding box of the model is recursively divided into equal, axis aligned halves until there is no cell containing more than a specified amount of CPs. Cells are always split by the midpoint of their longest axis. This means that no knowledge about the topology of the model is used. Fig. 2 illustrates the subdivision principle in two dimensions. As the CPs near a cell boundary will still have similar effects on the nodes on both sides, spatial subdivision may create discontinuities in the transformation near the boundaries when such CPs are ignored on one side of the boundary, as illustrated in Fig. 3b. The discontinuities near the boundaries could degrade the element quality of the transformed mesh, which is a critical issue for finite element simulation. A way to mitigate these discontinuities is to include CPs from neighboring cells when defining the cell transformation. Two strategies used to mitigate discontinuities between cells created this way were implemented: "overlapping" (Fig. 3c) and "neighborhood extension" (Fig. 3d). In the overlapping strategy, each cell size is increased by a constant percentage after all cells are generated. This makes neighboring cells overlap each other. The transformation of nodes within a region overlapped by multiple cells are computed by averaging the transformation predicted by each overlapping cell.

As an alternative, a heuristic approach to obtain a finite set of CPs closed to each face of the cell was implemented: the neighborhood extension (Fig. 3d). First, for each face of the cell, starting from a corner, the N-nearest CPs are found (N being a user parameter). Then, a square with edge length equal to the distance to the farthest of those N CPs is considered as "covered". The remainder of the face is subdivided into three quadrilaterals as shown in Fig. 4.

These are recursively processed the same way until the entire area of the original face is covered. All the CPs collected are used with the ones inside the cell to compute the transformation field inside the cell. The final positions for transformed nodes are then always computed within a single cell, but as Fig. 3d shows, the transformation is expected to behave more similarly to the global one near the boundaries of the cells, possibly leading to smaller discontinuities.

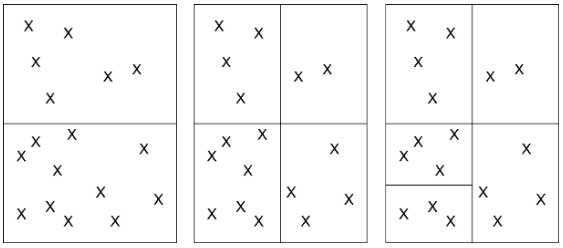

Fig. 2. Subdivision approach: illustration of the principles in two dimensions. The algorithm recursively divides the bounding box of the points in half along the longest edge until the maximum number of points per cell (four in this example) is achieved. In this example, three levels of subdivision are needed: first, the area is split in half (left), both of the children rectangles are divided next (middle) and finally only one of the four children needs to be divided.

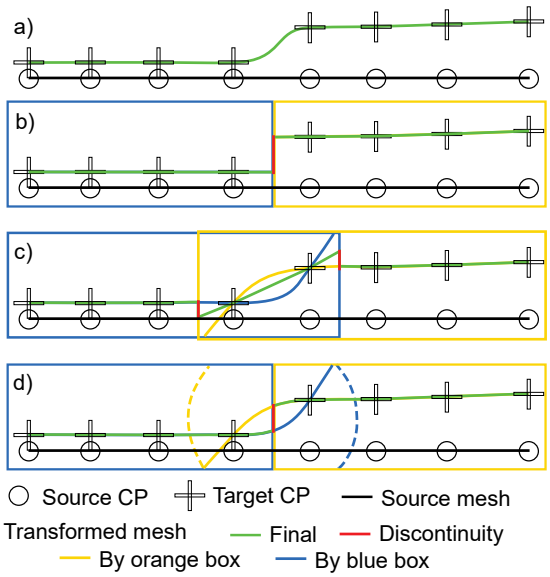

Fig. 3. Illustration of the effect of spatial subdivision. The top image (a) shows the transformation when no spatial subdivision is done. In the middle image (b), spatial subdivision by two cells (blue and orange) is visualized. The blue and orange lines show the results after applying only transformation defined by CPs in the blue or orange cells, respectively. These lines represent the final concatenated result (orange transformation applied only within the orange cell, blue only in blue cell), including the red steep discontinuity. The bottom images (c,d) visualize the two strategies to mitigate discontinuities: overlapping (c) and neighborhood extension (d), which results in smoother transformation, although some discontinuities (red) might still appear if the overlap is not large enough (here only two CPs in both cases (c,d)). The dashed lines in (d) signify the neighborhood of the blue and orange cell.

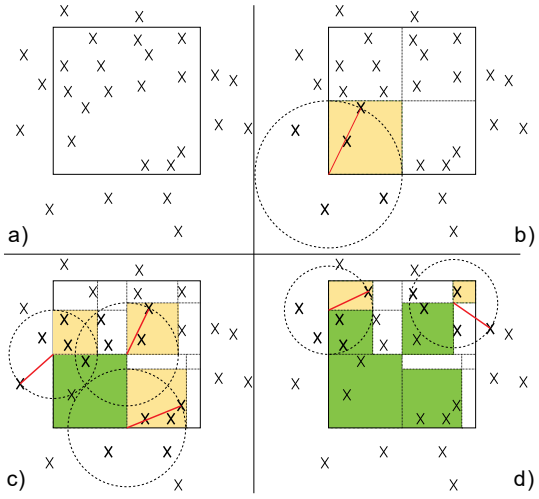

Fig. 4. Three iterations of the algorithm for finding closest CPs to a face of a cell. First (a), only the face (black rectangle) and the CPs in its vicinity (crosses) are shown. Then (b) the algorithm starts from bottom left corner, finding 5 nearest CPs. The red line shows the distance to the farthest one, which becomes the diameter of the circle (dashed) that lays out the "covered" part of the face (shaded in yellow). Then (c) three remaining parts of the face (beyond the dotted line) are processed recursively. The part covered in the previous iteration is shaded in green. The last image (d) shows only a part of a next iteration – 2 out of the 9 rectangles that need to be processed – for better clarity. It shows that for those two parts, the algorithm will reach the end of the face and will terminate (for that direction).

2.4. Workflow combining subdivision and subsampling

As Fig. 3 illustrated, discontinuities may still occur at cell boundaries even when overlapping or neighborhood extension are used. To reduce them further, the following iterative Subsampling + Spatial Subdivision workflow (abbreviated as "3S") was formulated to limit the amplitude of the transformation at the cell boundaries:

1. Subsample the CPs down to such an amount that allows kriging without the need for subdivision.
2. Transform the model using the subsampled CPs to generate an "intermediate" morphed mesh. The source position of all CPs (not just the subsampled) are transformed as well.
3. Re-iterate steps 1 and 2, progressively decreasing the number of CPs discarded in step 1. The intermediate mesh and CPs obtained by transformations in previous iterations are used as input.
4. Compute the final transformation from all the CPs (transformed in the previous steps) using the spatial subdivision and apply it to the last intermediate morphed mesh.

2.5. Applications to the test case

For the current study, several morphing approaches were tested. Fig. 5 summarizes them in the form of flowcharts. The figure also defines the abbreviations that will be used to denote the individual morphing results in the following text: BMI#/[3S]/X, where # stands for either 20, 22.7 or 35, optionally followed by /3S for the models that used the 3S workflow. Finally, X denotes the technique used to maintain the continuity between subdivision cells: NE for neighborhood extensions, OV for overlapping and D for "direct", i.e. with neither NE nor OV.

First, the models without the 3S workflow were generated to obtain a basis for comparison (Fig. 5a). Then the 3S workflow was used with two iterations of subsampling (Fig. 5b). Both the skin and bones CPs are combined into a single set and treated the same way during morphing. In the first iteration, the 234 777 CPs were subsampled using 30 cells per width of the model along the X axis. This led to 39.77 mm cell size, 30x24x28 cells in the grid in X, Y, Z axes respectively and 2 005 CPs. This was the same for all test cases. A second subsampling iteration used 60 cells per width (60x49x56 cells in each test case). This resulted in 9 917 CPs for the BMI 35 case (cell size 20.17 mm), 8 490 for the BMI 22.7 case (cell size 19.89 mm) and 8 141 CPs for the BMI 20 case (cell size 19.92 mm). Then the last step was performed using all CPs and the spatial subdivision. For spatial subdivision, the splitting criterion was selected at 8 000 CPs per cell. The overlap was set to 6% for the overlapping strategy, meaning that after a cell is split in two, both halves are extended by 6% of their length along the splitting axis. For the neighborhood extension, the number of neighbors N was set to 30.

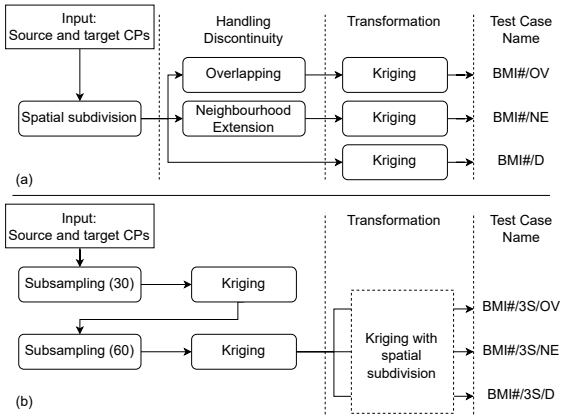

Fig. 5. Overview of the morphing processes. Flowchart (a) describes morphing with the spatial subdivision with its three variants (direct, overlapping, neighborhood extension), but without subsampling. Flowchart (b) describes morphing with the 3S workflow, i.e. with two steps of kriging with subsampling followed by the kriging with spatial subdivision, i.e. process (a). All variant were tested with three BMIs and names corresponding to all test cases are provided on the right of the figure, with '#' being a placeholder for one of the three BMIs: 20, 22.7, or 35.

The overlap percentage and number of neighbors was chosen in order to achieve similar computation times using both methods. Note that the overlap is applied when the cells are constructed, hence it is ensured that there will be no cell with more than the 8 000 CPs. In the neighborhood extension case, the neighborhood is extended after all the cells were built, because only then it is established which cells are neighboring to each cell. As a consequence, each cell might in the end have more than the 8 000 CPs.

3. Implementation and evaluation criteria

The PIPER software [26] was used to perform the kriging-based morphing. The implementation of kriging in PIPER is written in the C++ language, is optimized for multithreaded processors and uses the Eigen library [30] for linear algebra calculations. Tests were performed on a PC with Intel Xeon E5-1650 CPU (3.5 GHz, 6 cores), 32 GB RAM and Windows 7 64-bit operating system. The subsampling and spatial subdivision implementations were incorporated into PIPER.

In order to be able to start a simulation with an FE model, there must not be elements with negative volume. Only solid (3D) elements were concerned by the negative volume criteria, which represents 1 663 647 tetrahedral and hexahedral elements in the GHBMC M50-O. The morphed models without negative volume elements were subjected to an FE simulation of an abdomen impact by a solid bar corresponding to the test conditions in [31] using LS-Dyna (LSTC, CA).

While a mesh without negative volume elements is required to run a simulation in LS-Dyna, it is not a sufficient condition to ensure stress accuracy or normal termination of the simulation. The Scaled Jacobian of the elements has the best correlation to the accuracy of the partial differential equations solution used in FEM according to Knupp [32] and Gao et al. [33]. PIPER was first used to compute this metric for all the elements in the model and perform relative comparisons between models. The computation uses the Verdict library, which defines the Scaled Jacobian as the minimum determinant of the Jacobian matrix evaluated at each node of the element and its center, divided by the length of the three corresponding edges (or principal axes edges for the center) [34]. However, the Scaled Jacobian does not fully characterize the element quality nor solely determine the errors that would result from poor element quality. Because of the complexity of the issue, numerous element quality metrics and corresponding thresholds have been used in engineering practice to define requirements a mesh should meet, e.g. [35,36]. For this study, guidelines used during the development of the GHBMC models to define requirements (thresholds) for different quality metrics [36] were also used. Focusing on solid elements – which constitute most of the model – four of those metrics (Table 1) were computed with the Hypermesh (Altair) software to assess the effect of the morphing on the mesh and to check if the morphed models meet the thresholds. The Scaled Jacobian is also included in these metrics, but its definition differs from the one of Verdict already provided. The Scaled Jacobian measured in Hypermesh takes into account the number of integration points used in the element (Hypermesh 13.0 user manual, Altair) and not all nodes (as in Verdict). In further text, the acronyms SJH and SJV will be used for the Scaled Jacobian measured in Hypermesh and Verdict, respectively. Since the presented methodology aims to be used on personal computers for fast HBM morphing, the computation time required for the transformation was also evaluated.

4. Results

The different tested schemes had contrasted results in terms of element quality and computing cost. Results will be first presented for the BMI 35 target (Sections 4.1 and 4.2) to compare the impact of the subsampling and subdivision technique, and their combination. Results from the two thinning test cases will then be summarized in Section 4.3.

Table 1. Description of element quality metrics computed in Hypermesh (Altair) used in this study. The numerical range of each metric is specified by which value would an ideal element have and at which value does an element become degenerate. Note that the Aspect Ratio (AR) is theoretically unbounded, the element becomes "degenerate" when the smallest size is zero (AR becomes positive infinity). The guideline column specifies what values are considered acceptable by GHBMC guidelines for what portion of elements in the model

Metric Name and Description	Ideal	Dege-nerate	Guideline [36]
Scaled Jacobian: Ratio between the smallest and largest Jacobian determinant among the elements integration points.	1	< 0	> 0.3 for 100%
Tet Collapse: Minimum normalized distance between a node and opposite face, weighted by the area of the face.	1	< 0	> 0.2 for 100%
Aspect Ratio: Ratio between longest and shortest size of the element.	1	Large positive	< 8 for 99%
Skewness: Ninety degrees minus the minimum angle between vectors connecting opposite sides of the element.	0	90	< 70 for 99%

Section 4.4 will then show all element quality measurements for the resulting models morphed by the 3S workflow.

4.1. Spatial subdivision only for the BMI 35

The BMI35/D model had 3 449 negative volume elements. Using the neighborhood extension reduced that number to 80, i.e. for model BMI35/NE. Fig. 6 illustrates some of the artefacts near the boundary of subdivision cells with and without neighborhood extension. Neighborhood extension quadrupled the computation time (from 207 s to 830 s). The 6% overlap strategy led to slightly lower computation time (753 s), but a larger number of negative volume elements, 372.

4.2. Combination of subsampling and spatial subdivision for the BMI 35

The 3S workflow with neither overlapping nor neighborhood extension (BMI35/3S/D) resulted in a mesh with five negative volume elements. The total computation time was 310 s, of which 227 s was spent on the last step. Both the BMI35/3S/NE and BMI35/3S/OV models had no negative volume elements. The bone shape was conserved, as there was a CP for each node of the bone surfaces to constrain it. The mean distance between the bone shapes before and after morphing was 9.48·10-7 mm, with a maximum of 9.65·10-5 mm for neighborhood extension, 1.26·10-4 mm and a mean of 2.61·10-6 mm for overlapping. These differences were attributed to rounding errors.

Visually, the model skin surface also matched closely the skin target. There was not a one-to-one correspondence between the CPs and skin nodes as for bones. The distances between the BMI 35 skin surface and the target surface were computed using the Hausdorff distance function in Meshlab (ISTI – CNR, Italy). Fig. 7 shows the mesh-to-mesh distance as a color map on the skin surface along with an illustration of the causes for the largest differences, i.e. details of the GHBMC M50-O model not captured by the shape of the skin target. The mean distance was 0.72 mm (Root Mean Square (RMS) of 3.28) for BMI35/3S/NE, 0.73 mm mean (RMS of 3.28 mm) for BMI35/3S/OV. This is slightly higher than the initial distance between the shape of the GHMBC25.3 model and the baseline (BMI 25.34) target (mean 0.44 mm, RMS of 1.50, maximum 39.45 mm in the armpits). The largest differences for the BMI 35 shapes were observed on the feet (maximum 19.59 mm) and armpits (maximum 69.75 mm, same in neighborhood extension and overlapping). The actual mass of the resulting model was 115.36 kg, i.e. BMI 37.7.The added mass due to mass scaling was 127 g, i.e. similar to the baseline, which had 128 g added.

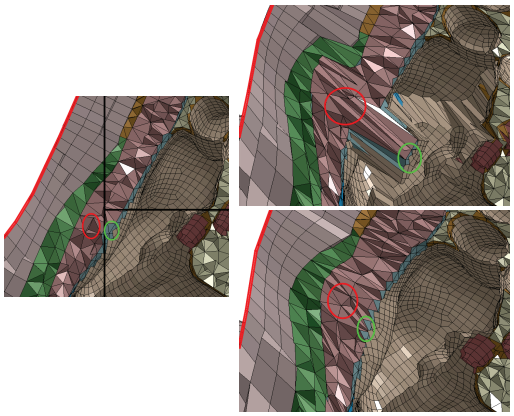

Fig. 6. Example of artefacts near cell boundaries when scaling towards the BMI 35. Detail of sagittal section through the abdomen. The top image shows the baseline model before morphing, including the cell boundaries, marked by thick black lines. The skin outline is marked by a thick red line. The green and red circles highlight the groups of nodes that share elements but fell into different cells. After transformation, without any measure for continuity (top right), the red nodes follow the skin while the green nodes stay with the abdomen, resulting in a large distortion of elements. With neighborhood extension (bottom right), the green nodes are influenced by neighboring cell, resulting in a smoother mesh. Other discontinuity artefacts near the box boundary can also be observed in the bottom left image.

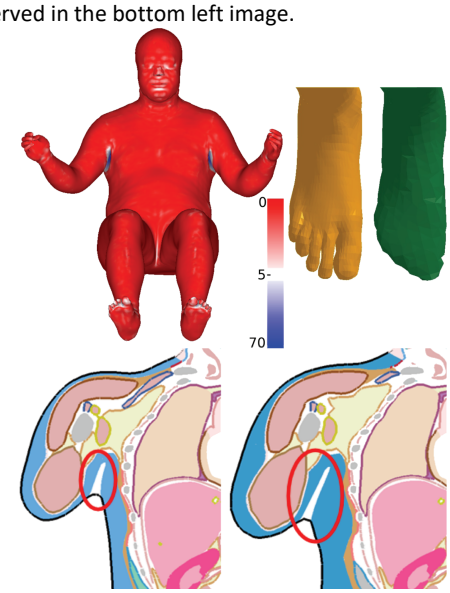

Fig. 7. Regions with the highest differences between the skin target and the model. Top left: mesh-to-mesh distance (in mm) of the BMI 35 test case between the target and the surface of models obtained by the 3S workflow with neighborhood extension. Vertices with distance lower than 5 mm (majority) are colored in shades of red, larger distances in blue. Top right: the level of details of the GHBMC M50-O foot (in orange) also differs from the target mesh (in green). Bottom: section near the right shoulder of the model (superficial flesh in blue) and the skin target (thick black line). Left is BMI 25.3 and right BMI 35. The armpit fold (marked by red ellipse) is not present on the target surface, which explains the high maximum distance between target skin and model.

Table 2 shows a detailed breakdown of computation times corresponding to the BMI35/3S/NE test case. For the steps performed with subsampling, the time to solve the kriging system is lower than the time needed to apply the transformation to the mesh. This is due to the low number of CPs used in these steps relatively to the total number of nodes to transform (1 255 225). However, in the last step, the higher algorithmic complexity of solving the system manifests itself, leading to a total transformation time of 914 s. The average number of iterations the nearest neighbor algorithm needs to "cover" one face of a cell (as described in section 2.3) was 3.82, with a median 3 and a maximum 18. This led to an additional 3.7 s of pre-processing time when spatial subdivision was employed (Step 4) and also additional 0.69 s for creating the subdivision grid. Thus the total pre-processing time rose to 5.02 s, which is still relatively small compared to the total computation time of the process (831.32 s).

These results are to compare with results without subsampling, i.e. 207 s for BMI35/D and 830 s BMI35/NE. This means that the subsampling contributed to additional 84 s of computation time, while the neighborhood extension added 623 s. The "Kriging solve" time (see Table 2) of the overlapping strategy was 728.77 s, i.e. almost identical to the neighborhood extension. However, the "Kriging apply" time rose to 119.09 s, since there are in total more nodes being updated as they are shared by multiple cells. As a result, the overall time of the third step rose to 872 s for the overlapping.

4.3. Thinning test cases

Similarly to the BMI 35 case, model of the BMI 22.7 with no negative volume elements was achieved only with the 3S workflow. The distance to the skin target of the resulting mesh of the BMI22.7/3S/NE was at maximum 31.39 mm, with mean 0.40 mm (RMS of 1.21) for the skin (Fig. 8, left).Corresponding maximum 1.23·10-4 mm and mean 9.43·10-7 mm were for the skeleton. The BMI22.7/3S/OV did not differ by more than 0.01 mm. The mass of the resulting model was 72.91 kg. Hence, the actual achieved BMI of the model was 23.8. The added mass due to mass scaling was 131 g (baseline model had 128 g added).

The application of the same approach to the BMI 20 led to 90 negative volume elements after the final step, although no negative volume elements appeared in the intermediate meshes. The negative volume elements were located in areas where the skin and bones get very close to each other, namely around ribcage and wrists. As Fig. 9 shows, reducing the distance between bones and skin in such areas causes compression of the elements in between them, and inversions in extreme cases.

Table 2. Computation times of the 3S process for the BMI 35 case and neighborhood extension. Legend: "Subsampling" is the time spent on the homogenous decimation of CPs. "Pre-processing, Subdivision" time is mainly spent on creating the subdivision grid and assigning CPs to its cells. "Kriging solve" time is spent to create the covariance matrix, including memory allocation, and to solve the system. "Kriging apply" is spent applying the transformation to the nodes. "Total" is the total time needed for the whole process, including post-processing operations done by the PIPER framework (writing output etc.). It is therefore slightly longer than the sum of the other times.

3S step	Steps 1-3		Step 4
Time (s) for BMI 35	2 008 CPs (Iteration 1)	9 912 CPs (Iteration 2)	234 777 CP
Subsampling	0.01	0.02	n/a
Pre-processing	0.59	0.63	5.02
Subdivision	n/a	n/a	
Kriging solve	0.50	16.96	772.14
Kriging apply	10.15	52.84	52.73
Total per step	13	69.94	831.32
Total (cumul.)	13	83	914

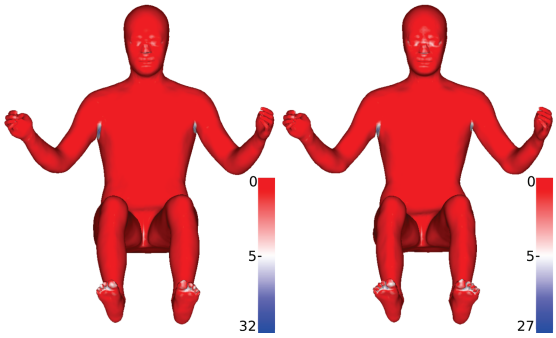

Fig. 8. Mesh-to-mesh distance (in mm) between the target and the surface of models obtained by the 3S workflow with neighborhood extension. Left: BMI 22.7, right: BMI 20. Vertices with distance less than 5 mm (majority) are colored in shades of red, larger distances in blue.

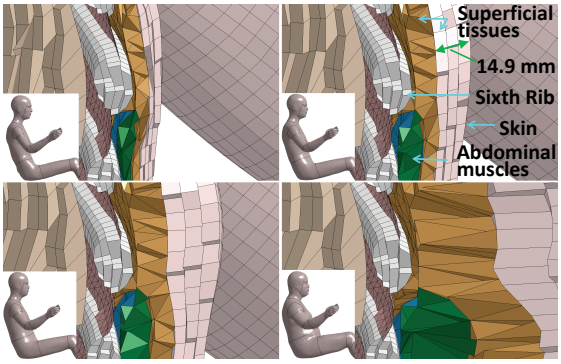

Fig. 9. A slice through the ribcage of the four created models: BMI 20 (top left), BMI 22.7 (top right), the baseline BMI 25.3 (bottom left) and BMI 35 (bottom right). The white cylinder-shaped parts are ribs, namely the sixth left rib in the middle of the image. The image illustrates the important changes in soft tissues thickness around the ribs. The full models corresponding to this image can be seen in the bottom left corners of each image.

After adjusting the target surface in the problematic regions, i.e. increasing the distance between bones and skin, a model with only four negative volume tetrahedral elements was produced. Those were removed in Ls-Prepost by the "reorder nodes" function. After that, the model successfully completed the same FE simulation of abdomen impact used for the BMI 35 test case. The BMI of the resulting model was 21.3 (mass 65.18 kg) instead of the targeted BMI 20. For the distance between resulting mesh and target, the mean was 0.42 mm for skin (RMS 1.24, maximum 26.82 mm, Fig. 8, right) and 8.89·10-7 mm for bones (9.31·10-5 mm maximum). The added mass due to mass scaling was 149 g (baseline model had 128 g added). The computation times were within 2% of the ones reported for the BMI 35 case (Table 2). The computation time is a function of the total number of CPs, the number of CPs per subdivision cell, and number of nodes to transform. While the number of nodes to transform is constant and that the same CPs and distributions are used in the first two iterations, small variations in CPs number and distribution are present in subsequent steps due to the different shapes of the models.

4.4. Element quality comparison

The Scaled Jacobian (SJH), Tet Collapse (TC), Aspect Ratio (AR) and Skewness (SK) metrics were measured only for the BMI#/3S/NE and BMI#/3S/OV models. Note that for all the results shown below, the BMI 20 case refers to the modified depenetrated target – see Section 4.3. Table 3 shows the results for SJH and TC. Although the number of elements below the guideline threshold was small, strict conformity (for 100% of elements) was not achieved, even for the baseline. For SJH, the number of non-conforming elements ranged from 0.01% (73 elements) in the baseline to 0.02-0.05% for the morphed model (worst cases: 378 elements with BMI20/3S/OV and 311 elements for BMI35/3S/OV). The percentage of non-conforming elements is higher for the TC, with the highest values reached for the BMI 20: 0.36% and 0.38% for NE and OV, respectively. All other models ranged between 0.09 and 0.012%, compared to 0.06% for the baseline. Table 4 shows a summary of the AR and SK results in terms of percentage of elements. The 99 percentile element threshold were met for all models. The baseline model had 0.08% and 0.10% of elements above the threshold for AR and SK, respectively. Both the BMI22.7/3S/NE and BMI22.7/3S/OV cases were close to these numbers with 0.12-0.13% for both metrics. The worst result was with the BMI 20 again, with 0.27% and 0.28% of elements above the threshold for SK (NE and OV, respectively). Tables with absolute numbers of elements for each metric can be found in Appendix D. Overall, the BMI 20 case had the highest number of low quality elements according to all metrics. The low quality elements were located in the layers of elements modelling soft tissue around bones, which are significantly thinned in this test case. Although the BMI 35 was better in this regard, it had the lowest amount of elements in the highest quality bin (Table 3 and Table 4) according to SJH, TC and AR among the three test cases. Models without negative volume elements were achieved only when the 3S workflow was used (Table 5).

Table 3. Tet Collapse (TC, for tetrahedral elements) and Scaled Jacobian (SJH, for other solid elements) for the baseline model and the three BMI models morphed by the 3S workflow with either overlapping (OV) or neighborhood extension (NE). Percentage of elements not satisfying the GHBMC threshold (left), high quality elements (right) and the remaining elements (middle).

	SJ 0 – 0.3	SJ 0.3-0.8	SJ 0.8 – 1	TC 0 – 0.2	TC 0.2 – 0.8	TC 0.8 – 1
Baseline	0.01	32.57	67.42	0.06	87.31	12.63
BMI20/3S/NE	0.03	32.72	67.24	0.36	88.25	11.40
BMI20/3S/OV	0.05	33.17	66.78	0.38	88.22	11.40
BMI22.7/3S/NE	0.02	32.52	67.46	0.09	87.50	12.41
BMI22.7/3S/OV	0.02	32.68	67.30	0.10	87.51	12.40
BMI35/3S/NE	0.04	33.66	66.30	0.12	89.10	10.78
BMI35/3S/OV	0.05	34.02	65.93	0.12	89.14	10.74

Table 4. Aspect Ratio (AR) and Skewness (SK) for the baseline and the three BMI models morphed by the 3S workflow with either overlapping (OV) or neighborhood extension (NE). Percentage of elements not satisfying the GHBMC threshold (left), high quality elements (right) and the remaining elements (middle).

	AR > 8	AR 2-8	AR 1-2	SK 70 - 90	SK 20 - 70	SK 0-20
BMI20/3S/NE	0.08	28.39	71.53	0.10	72.36	27.54
BMI20/3S/OV	0.18	33.84	65.98	0.27	75.92	23.81
BMI22.7/3S/NE	0.18	33.78	66.04	0.28	75.91	23.80
BMI22.7/3S/OV	0.12	31.49	68.39	0.12	73.42	26.46
BMI35/3S/NE	0.13	31.49	68.38	0.13	73.45	26.42
BMI35/3S/OV	0.13	36.38	63.49	0.21	74.74	25.06
BMI20/3S/NE	0.14	36.54	63.32	0.21	74.82	24.97

Table 5. Summary of the number of elements with negative volumes (NV) and negative Scaled Jacobian (SJv, using all nodes as integration points) for all the BMI test cases and with various morphing processes. Test cases are denoted as in Fig. 5: /OV, /NE and /D refers to morphing with neighborhood extension, overlapping and neither of those, respectively. /3S refers to using the 3S workflow, i.e. using two steps of subsampling before the final morphing.

	BMI 35		BMI 22.7		BMI 20	
	NVE	NSJV	NVE	NSJV	NVE	NSJV
BMI#/D	3449	5364	1019	2871	4239	8402
BMI#/OV	372	986	137	515	2716	5018
BMI#/NE	80	615	3	56	3	134
BMI#/3S/D	5	267	68	155	630	1464
BMI#/3S/OV	0	172	0	85	4	354
BMI#/3S/NE	0	140	0	52	4	162

The table also shows the number of elements with negative SJV (Appendix D contains histograms for the positive values for 3S models). The two metrics (negative volumes and SJV) show similar relative differences between the different models. Specifically, the neighborhood extension consistently performed better than the overlapping according to all metrics (including the ones measured in Hypermesh) but the differences were small in most cases in the 3S workflow (less than 1%). A more noticeable difference between the two techniques was observed on the number of negative volume elements and SJV when using the approach without subsampling: the BMI20/NE was close to success (only three negative volume elements, 134 negative SJV), while BMI20/OV produced 2716 negative volumes and 5018 elements with negative SJV.

The models obtained using the 3S workflow and the NE ran without error in a bar impact simulation. Illustrations can be found in Appendix B.

5. Discussion

By applying the proposed methodology, the initial HBM was successfully morphed to BMI 20, 22.7 and 35 by a large number of CPs. Since the subsampling and subdivision parameters were the same for the BMI 35, BMI 22.7 and BMI 20 test cases, the computation times to perform the transformation were almost identical. All models (from BMI 20 to 35) could be used in simulations that terminated without error, although the BMI 20 required correction for a few negative volume elements. Such issue was not present for BMI increase. This observation was previously made by Zhang et al. [8], who also had to locally violate the target shape for targets with BMI lower than the BMI of their baseline model. Although the BMI 35 imposed the largest deformation in terms of distance to the source model, the BMI 20 case had the highest amount of low quality elements according to all used metrics. The BMI 35 case degraded the aspect ratio of highest quality elements more than the other test cases, but this may be less critical than creating elements with a very low quality. This further suggest that the weight loss scenario is more difficult to achieve using morphing than the weight gain.

This BMI 20 case showed that large reductions of the thickness of superficial tissues becomes difficult with the current method without inverting elements. In the current study, exact targets were required, but without this requirement, the “nugget” parameter of kriging could be used. The nugget allows for smoother transformation in exchange of violating the target CPs position by a specified margin [13]. Using it for the skin could be considered to mitigate some of the transformation issues in thin regions. The local transformation smoothing presented in Janak et al. [12] may also be used to edit locally the transformation, for example by using a different nugget in specific parts of the model prone to element quality issues. Local remeshing could also be helpful to address element quality degradation due to morphing. Ideally, the need should be assessed based on the model stability and estimation of the stress errors. Such estimation may be challenging for such models as, beyond the element geometrical quality, errors may be affected by element and material formulations, boundary conditions and loading cases.

In all test cases, the morphed models ended with higher BMIs than the target, by about one point of BMI for the low BMI case and by close to three points for the high BMI case. Different reasons could affect this discrepancy. First, while a skeleton obtained from a statistical model would likely differ between the BMIs, the same skeleton was used for all BMIs (weight gain/loss scenario). This may be especially important for the low BMI for which skin depenetration was needed, pointing to a smaller skeleton for an average BMI 20. These may have contributed the mass overestimation in that case. Besides the skeleton, the soft tissue distribution (muscle vs. fat, thickness of subcutaneous fat) is likely affected by the BMI [37] but this was not accounted for in the target (muscles and subcutaneous tissues are morphed together), and the material density was not adjusted in the morphed models either. Since the density of muscles is higher than the density of adipose tissues (e.g. 1041 kg / m3 vs 916 kg / m3 according to Duck [38]), this might contribute to overestimation of the mass in the BMI 35 case. Furthermore, the GHBMC M50-O version 4.1 uses density of 1100 kg / m3 for adipose tissues, which is higher than what is reported in literature. Changing the density of the fat and muscle materials to 916 kg / m3 in the BMI 35 test case would lead to 12.6 kg mass decrease, which would lower the BMI below 34. Further work on the mass distribution would likely be needed should these models be used beyond the test case.

Two approaches, overlapping and neighborhood extension, were presented to mitigate the degradation of element quality on the boundaries of cells of the spatial subdivision. Both are based on sharing CPs with neighboring grid cells. The overlap approach does not guarantee that control points will be present in the overlap zone while the neighborhood extension does. However, this had limited effect in the test case, perhaps due to the even distribution and large density of CPs. Neighborhood extension performed marginally

better, which is best evidenced in the non-iterative scaling using all the CPs (80 negative volume elements vs. 372 in the BMI 35 test case, up to 3 vs 2716 in the BMI 20 case). With the 3S workflow, both approaches produced acceptable results, i.e. models with no negative volume elements and distances to the targets that were almost the same. The major drawback is that this was achieved while almost quadrupling the computational time compared to subdivision without additional measures. Without additional measure, a model with only five negative volume elements was produced for the BMI 35 test case, but 68 for the BMI 22.7. This suggests that the need for such costly strategies may be dependent on the transformation scenario. In addition, the implemented subdivision always splits the cells in half when above the threshold of 8000 CPs, regardless of the number of CPs in either half of the cell. As a result, some cells, 4 out of 57 in the case BMI35/3S, contained only tens of CPs. Other spatial subdivision schemes, e.g. k-d trees [39], are more computationally expensive to construct, but are likely to produce better-balanced partitioning of the points. This could decrease the overall computation time.

To optimize the number and location of CPs added from outside the current cell, further analysis of the deformation function near the boundaries could help. For example, Heaton et al. [40] did a hierarchical clustering based on a measure of spatial dissimilarity (applied for temperature data). The CPs subdivision may also be enhanced by some knowledge about the model. For example, Zhang et al. [8] manually divided their model by anatomical regions (arms, head torso, hips and legs), which allow limiting the effects disjoints regions of the body (e.g. torso vs. hands) have on each other, but still requires ensuring the continuity near the regions boundaries. Also, Auñón and Gómez-Hernández [41] derived blending functions for overlapping 2D Kriging regions that ensure C1 continuity of the final transformation function and it may be possible to extend their work to 3D. Finally, using alternative distance formulations that take into account curvature of the surface, e.g. based on geodesic distance [42,43], could help in rejecting CPs from the neighborhood that are close in terms of Euclidean distance, but lie on a different surface. Those approaches could be combined with the proposed methodology, although likely requiring some adjustments to the method.

The distance to the target surface for the second intermediate mesh (subsampling with input parameter 60) was already less than 1 mm (mean value). This raises the question of the number of control points really needed to capture the transformation of a human model. In the current test case, the respect of the bone target was assumed. Other scenarios for which an exact bone shape is expected could include the use of kriging for model positioning [12] or subject specific modelling based on high resolution imaging. However, for cases that allow some target violation, a subset of CPs might be sufficient. What magnitude of deviation from the target can be considered acceptable depends on individual applications. For example, Hu et al. [25] placed a particular emphasis on respecting the target mass of the morphed BMI instead of focusing on shape error. Zhang et al. [8] underlined the need to balance shape accuracy and mesh quality. In any case, it would be useful to have an automatic mechanism that would compare the distance to the target with a user-specified threshold and perform additional iterations only when needed. Also, evaluating automatically the mesh quality after each step (e.g. negative volume elements) could help automating the subsampling approach (e.g. revert the transformation and do a coarser pass first).

The test cases used in this study required manual work to pre-process and build the target surfaces. However, this was specific to the target building phase, and the transformation method itself is automated (box splitting, etc.). Regarding the applicability of the transformation method, while the test case used only one FE model, it is a model that is widely used and is believed to use similar principles as many other FE models in biomechanics. By making the method freely available in open source, it is also hoped that other evaluations and applications can be performed by other researchers. The method is theoretically applicable to a wide range of scenarios as its only inputs are the control points and the vertices to deform. Any applications currently using mesh morphing, for example in fluid dynamics [44], medicine [45,46] or composite development [47], may benefit from the presented techniques if a large number of control points have to be used.

In the current test case, the initial model transformation used only the skin in order to create an initial model it compatible with the available target shapes. This is of course a limitation due to the data available for test case. However, the control points can be used to express different types of targets than the ones used in this study. The skeleton could be also be morphed based on either a statistical model or patient-specific imagining data. Additional target could be added also for abdominal organs etc. One problem with such complex targets is that since they each usually come from different sources, they are not always

compatible with each other (e.g. [6,8]). The presented iterative approach can also be used to mitigate this problem: one could start from morphing by only the most important or precise target (e.g. skeleton) and then add the next target (e.g. skin) in the next iterations, with some nugget to allow for slight violation of the less important target.

6. Conclusions

An iterative approach able to morph a finite element model by kriging with a large (over 200,000) number of CPs was developed, implemented and applied. The main challenge of the proposed approach is maintaining continuity of the transformation on the boundaries of the subdivision grid, as large deformations tend to cause local artefacts on the boundaries. Subsampling used in conjunction with spatial subdivision allowed mitigating such discontinuities. The iterative approach was successfully applied to change the BMI of a GHBMC M50-O detailed model without changing its entire skeleton, with more difficulties for model mass loss than mass gain. The proposed approach should now be evaluated in other application scenarios.

7. Acknowledgements

Source code and executable for the presented methods are available online under an open source license on the repository of the PIPER project [26] (www.piper-project.org). The version used for the experiments presented in this paper is tagged “Kriging_JBE2020”.

References

[1] Euro NCAP, “Euro NCAP Pedestrian Test Protocol v8.4,” https://www.euroncap.com/en/for-engineers/protocols/vulnerable-road-user-vru-protection/ (accessed November 30, 2018).

[2] F.S. Gayzik, D.P. Moreno, C.P. Geer, S.D. Wuertzer, R.S. Martin, J.D. Stitzel, 2011, “Development of a Full Body CAD Dataset for Computational Modeling: A Multi-modality Approach,” Annals of Biomedical Engineering, 39(10), p. 2568.

[3] K. Shigeta, Y. Kitagawa, T. Yasuki, 2009, “Development of next generation human FE model capable of organ injury prediction,” Proceedings of the 21st Annual Enhanced Safety of Vehicles, Stuttgart, Germany.

[4] T.M. Rice, M. Zhu, 2013, “Driver obesity and the risk of fatal injury during traffic collisions,” Emergency Medicine Journal, 31(1), pp. 9–12.

[5] M.P. Reed, S.M. Ebert-Hamilton, J.D. Rupp, 2012, “Effects of Obesity on Seat Belt Fit,” Traffic Injury Prevention, 13(4), pp. 364–372.

[6] N.A. Vavalle, S.L. Schoell, A.A. Weaver, J.D. Stitzel, F.S. Gayzik, 2014, “Application of Radial Basis Function Methods in the Development of a 95th Percentile Male Seated FEA Model.,” Stapp Car Crash J, 58, pp. 361–384.

[7] J. Hu, A. Fanta, M.O. Neal, M.P. Reed, J.-T. Wang, 2016, “Vehicle Crash Simulations with Morphed GHBMC Human Models of Different Stature, BMI, and Age,” Proceedings of the 4th International Digital Human Modeling Conference, Montreal, Canada.

[8] K. Zhang, L. Cao, A. Fanta, M.P. Reed, M. Neal, J.-T. Wang, C.-H. Lin, J. Hu, 2017, “An automated method to morph finite element whole-body human models with a wide range of stature and body shape for both men and women,” Journal of Biomechanics, 60, pp. 253–260.

[9] E. Hwang, J. Hallman, K. Klein, J. Rupp, M. Reed, J. Hu, 2016, “Rapid Development of Diverse Human Body Models for Crash Simulations through Mesh Morphing,” SAE 2016 World Congress and Exhibition Technical Paper, Detroit, MI, USA.

[10] S. Schoell, A. Weaver, J. Urban, D. Jones, J. Stitzel, E. Hwang, M. Reed, J. Rupp, 2015, “Development and Validation of an Older Occupant Finite Element Model of a Mid-Sized Male for Investigation of Age-related Injury Risk,” Stapp Car Crash Journal, 59, pp. 359–383.

[11] P. Beillas, F. Berthet, 2017, “An investigation of human body model morphing for the assessment of abdomen responses to impact against a population of test subjects,” Traffic Injury Prevention, 18(sup1), pp. S142–S147.

[12] T. Janak, Y. Lafon, P. Petit, P. Beillas, 2018, “Transformation Smoothing to use after Positioning of Finite Element Human Body Models,” Proceedings of the 2018 IRCOBI Conference, Athens, Greece.

[13] E. Jolivet, Y. Lafon, P. Petit, P. Beillas, 2015, “Comparison of Kriging and Moving Least Square Methods to Change the Geometry of Human Body Models.,” Stapp Car Crash J, 59, pp. 337–357.

[14] F. Trochu, 1993, “A contouring program based on dual kriging interpolation,” Engineering with Computers, 9(3), pp. 160–177.

[15] G.H. Golub, C.F. Van Loan, Matrix Computations, 1996, Johns Hopkins University Press.

[16] R. Furrer, M.G. Genton, D. Nychka, 2006, “Covariance Tapering for Interpolation of Large Spatial Datasets,” Journal of Computational and Graphical Statistics, 15(3), pp. 502–523.

[17] L. Hartman, O. Hössjer, 2008, “Fast kriging of large data sets with Gaussian Markov random fields,” Computational Statistics & Data Analysis, 52(5), pp. 2331–2349.

[18] Y. Sun, B. Li, M.G. Genton, 2012, “Geostatistics for Large Datasets,” Advances and Challenges in Space-Time Modelling of Natural Events, pp. 55–77.

[19] N. Cressie, G. Johannesson, 2008, “Fixed rank kriging for very large spatial data sets,” Journal of the Royal Statistical Society: Series B (Statistical Methodology), 70(1), pp. 209–226.

[20] J.R. Bradley, N. Cressie, T. Shi, 2016, “A comparison of spatial predictors when datasets could be very large,” Statist. Surv., 10, pp. 100–131.

[21] M.L. Stein, 2014, “Limitations on low rank approximations for covariance matrices of spatial data,” Spatial Statistics, 8, pp. 1–19.

[22] T.C. Haas, 1990, “Lognormal and moving window methods of estimating acid deposition,” Journal of the American Statistical Association, 85, pp. 950–963.

[23] H. Sang, M. Jun, J.Z. Huang, 2011, “Covariance approximation for large multivariate spatial data sets with

an application to multiple climate model errors," Ann. Appl. Stat., 5(4), pp. 2519–2548.
[24] M. Katzfuss, W. Gong, 2017, "Multi-resolution approximations of Gaussian processes for large spatial datasets," ArXiv E-Prints 1710.08976.
[25] J. Hu, K. Zhang, A. Fanta, E. Hwang, M.P. Reed, 2017, "Effects of Male Stature and Body Shape on Thoracic Impact Response Using Parametric Finite Element Human Modeling," Proceedings of the 2017 IRCOBI Conference, Antwerp, Belgium.
[26] Piper-project, www.piper-project.org (accessed Feb. 7, 2020).
[27] M.P. Reed, U. Raschke, R. Tirumali, M.B. Parkinson, 2014, "Developing and Implementing Parametric Human Body Shape Models in Ergonomics Software," Proceedings of the 3rd International Digital Human Modeling Symposium, Tokyo, Japan.
[28] UMTRI, "Human Shapes," humanshape.org (accessed September 1, 2017).
[29] S. Yamazaki, M. Kouchi, M. Mochimaru, 2013, "Markerless landmark localization on body shape scans by non-rigid model fitting," 2nd Digital Human Modeling Symposium.
[30] "Eigen," eigen.tuxfamily.org (accessed Dec. 7, 2018).
[31] W.N. Hardy, L.W. Schneider, S.W. Rouhana, 2001, "Abdominal impact response to rigid-bar, seatbelt, and airbag loading.," Stapp Car Crash J, 45, pp. 1–32.
[32] Knupp Patrick M., 2000, "Achieving finite element mesh quality via optimization of the Jacobian matrix norm and associated quantities. Part II—A framework for volume mesh optimization and the condition number of the Jacobian matrix," International Journal for Numerical Methods in Engineering, 48(8), pp. 1165–1185.
[33] X. Gao, J. Huang, K. Xu, Z. Pan, Z. Deng, G. Chen, 2017, "Evaluating Hex-mesh Quality Metrics via Correlation Analysis," Computer Graphics Forum, 36(5), pp. 105–116.
[34] C. Stimpson, C. Ernst, P. Knupp, P. Pébay, D. Thompson, 2007, "The Verdict Library Reference Manual."
[35] LS-DYNA AWG, "Modeling Guidelines Document," https://awg.lstc.com/tiki-index.php?page=MGD (accessed July 2, 2020).
[36] D. Schwartz, 2015, "Development of a Computationally Efficient Full Human Body Finite Element Model," Master thesis, Virginia Tech – Wake Forest Univ.
[37] S.A. Holcombe, S.C. Wang, 2014, "Subcutaneous fat distribution in the human Torso," 2014 IRCOBI Conference Proceedings - International Research Council on the Biomechanics of Injury, pp. 389–396.
[38] F.A. Duck, Physical Properties of Tissue: A Comprehensive Reference Book, 1990, Academic Press.
[39] R.A. Brown, 2015, "Building a Balanced k-d Tree in O(kn log n) Time," Journal of Computer Graphics Techniques (JCGT), 4(1), pp. 50–68.
[40] M.J. Heaton, A. Datta, A.O. Finley, R. Furrer, J. Guinness, R. Guhaniyogi, F. Gerber, R.B. Gramacy, D. Hammerling, M. Katzfuss, F. Lindgren, D.W. Nychka, F. Sun, A. Zammit-Mangion, 2019, "A Case Study Competition Among Methods for Analyzing Large Spatial Data," Journal of Agricultural, Biological and Environmental Statistics, 24(3), pp. 398–425.
[41] J. Auñón, J.J. Gómez-Hernández, 2000, "Dual Kriging with Local Neighborhoods: Application to the Representation of Surfaces," Mathematical Geology, 32(1), pp. 69–85.
[42] X. Wan, S. Liu, J.X. Chen, X. Jin, 2012, "Geodesic Distance-Based Realistic Facial Animation Using RBF Interpolation," Computing in Science Engineering, 14(5), pp. 49–55.
[43] Y. Lipman, R.M. Rustamov, T.A. Funkhouser, 2010, "Biharmonic Distance," ACM Trans. Graph., 29(3), p. 27:1–27:11.
[44] M.E. Biancolini, I.M. Viola, M. Riotte, 2014, "Sails trim optimisation using CFD and RBF mesh morphing," Computers & Fluids, 93, pp. 46–60.
[45] S.-X. Zheng, J. Li, Q.-F. Sun, 2011, "A novel 3D morphing approach for tooth occlusal surface reconstruction," Computer-Aided Design, 43(3), pp. 293–302.
[46] L. Grassi, N. Hraiech, E. Schileo, M. Ansaloni, M. Rochette, M. Viceconti, 2011, "Evaluation of the generality and accuracy of a new mesh morphing procedure for the human femur," Medical Engineering & Physics, 33(1), pp. 112–120.
[47] G. Seon, Y. Nikishkov, A. Makeev, B. Shonkwiler, 2016, "Mesh morphing methodology for strength predictions in composites," Composite Structures, 140, pp. 612–620.

Appendix A: Kriging Background

The following text will briefly present the interpolation method used in this study. As has been noted before, the method can be seen as either an RBF or kriging interpolation, with the difference being only in the way the method is derived: RBF interpolation is derived as a superposition of basis functions [1], while kriging is derived as a stochastic estimation of a random function [2]. In the context of interpolating positions or displacements of 3D points, both definitions end up formulating the same function. Since the latter provides perhaps a more easy to understand insight on the foundations of the interpolation function, it will be used in the following text.
The stochastic approach to interpolation sees the known values $u(X_i)$ of the CPs as realizations of an unknown random function. To find the values of this function for other points, statistical analysis of the known dataset can then be employed to find the "most likely" positions for the new points. The criterion to achieve the "most likely" interpolation values is realized by minimizing the variance of the estimation, while assuming there is no bias between the interpolation function and the actual function, which is formulated as Equation (A.1) (*E* signifies the expected value)*:*

$$Min(E[u(X) - u^*(X)]^2), while\ E[u(X)] = E[u^*(X)] \tag{A.1}$$

Furthermore, Kriging is based on the assumption that the interpolated function $u(X)$ can be expressed as a sum of two terms – drift $a(X)$ and fluctuation $b(X)$. The drift represents the linear part of the transformation, while the fluctuation represents the local differences around each CP. In terms of the statistical description, this can be expressed by assuming that $E[u(X)] = a(X)$ and $E[b(X)] = 0$.
The two aforementioned conditions lead to a construction of the following interpolation function (A.2). Please refer to the appendix in [2] for a detailed derivation.

$$u(X) = \sum_{j=1}^{M} a_j D_j + \sum_{i=1}^{N} cov(X_i, X) \cdot b_i \tag{A.2}$$

By solving equation (A.2), the obtained coefficients *a* and *b* can be used to interpolate the function *u* at arbitrary location. This function, commonly referred to as Dual Kriging, can be formalized in a matrix form as shown in equation (A.3):

$$\begin{bmatrix} F_{N,N} & D_N \\ D_N^T & 0 \end{bmatrix} \begin{bmatrix} b \\ a \end{bmatrix} = \begin{bmatrix} U \\ 0 \end{bmatrix} \tag{A.3}$$

The product of the $F_{N,N}$ submatrix and the coefficients *b* describes the fluctuation part $b(X)$. As [2] shows, it is equal to the covariance between the CPs, i.e. $F_{i,j} = E[u(X_i)u(X_j)]$. The product of the D_N submatrix and the coefficients *a* describe the drift part, usually realized by a linear polynomial $p(X)$ that describes the affine part of the transformation, i.e. $p(X) = a_1 + a_2x + a_3y + +a_4z$, where *x, y, z* are the coordinates of *X*. Again, see [2] for an evaluation and discussion of several different choices of the drift part formulation.
Note that the definition of the submatrix $F_{N,N}$ implies that the diagonal elements are equal to the variance of $u(X_i)$, which is zero, since the values in CPs are known exactly. However, by assigning a non-zero negative value as the variance in CPs, one can express an uncertainty of the sampled values. The interpolation then effectively changes into a regression. This technique is usually called the "nugget effect", based on its first application in mining planning [3].
This form is called the dual form of kriging, or simply Dual Kriging (DK). The advantage of this formulation is that after the equation (A.3) is solved, which is the most costly operation as it requires inverting the left side matrix, any number of points in space can be interpolated using the transformation – it is a global interpolation method.
The assumption used for choosing the covariance function is that there is a spatial correlation between the measured values of function $u(X)$, i.e. functions that have increasing values with increasing distance (the higher the value, the lower is the correlation). The functions usually used as covariance functions are various spline functions ($cov_{ij} = \|X_i - X_j\|^d$), mainly thin-plate spline ($cov_{ij} = \|X_i - X_j\|^2 \cdot \ln(\|X_i - X_j\| + \varepsilon), \varepsilon \ll \min_{i \neq j}(cov_{ij})$). However, as [4] shown, there are no significant differences in the output quality when the interpolation is applied to HBM. The biharmonic 3D transformation resulting from the Euclidean distance as the covariance function ($cov_{ij} = \|X_i - X_j\|$) is usually sufficient.

Appendix B: Finite Element model Simulation

Models morphed by the 3S workflow (with neighbourhood extension) were subjected to an impact simulation with loading conditions from Hardy et al (2001) (6 m/s, mid abdomen impact). Figure D.1 shows a still image from the simulation after the bar has impacted the abdomen. All simulations terminated successfully. As an example, the elapsed time needed to simulate 120ms for the BMI35 model was 12 hours and 46 minutes on a 64 cores cluster (Ls Dyna R7.1.3, single precision, Open MPI).

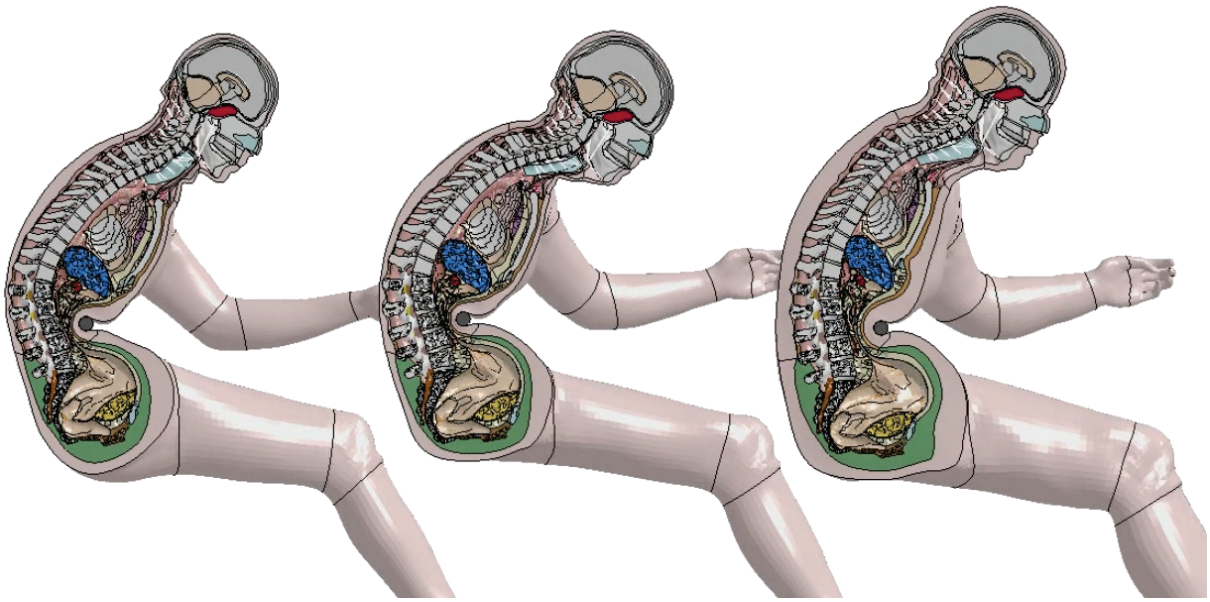

Figure D.1: Image from the simulations performed on the BMI 20, 25 and 35 models (from left to right) transformed using the proposed method (60 ms after contact).

Appendix C: Target pre-processing

Target surfaces obtained from humanshape.org required various post-processing steps: closing several holes in the mesh, resampling it to obtain a better distribution of vertices, and some manual adjustments in order to avoid penetrations with the skeleton (BMI 20 and 22.7 cases) or correct some postural differences (arms and legs on the BMI 35). Figure D.2 shows the target for BMI 20 before and after these modifications, superimposed with the skeleton. Notice the numerous penetration of the skin by the skeleton. The assumption of the used workflow is that the skeleton does not change with BMI but that is a simplification that is likely not true in reality. Figure D.3 shows the BMI 35 target before and after manual corrections.

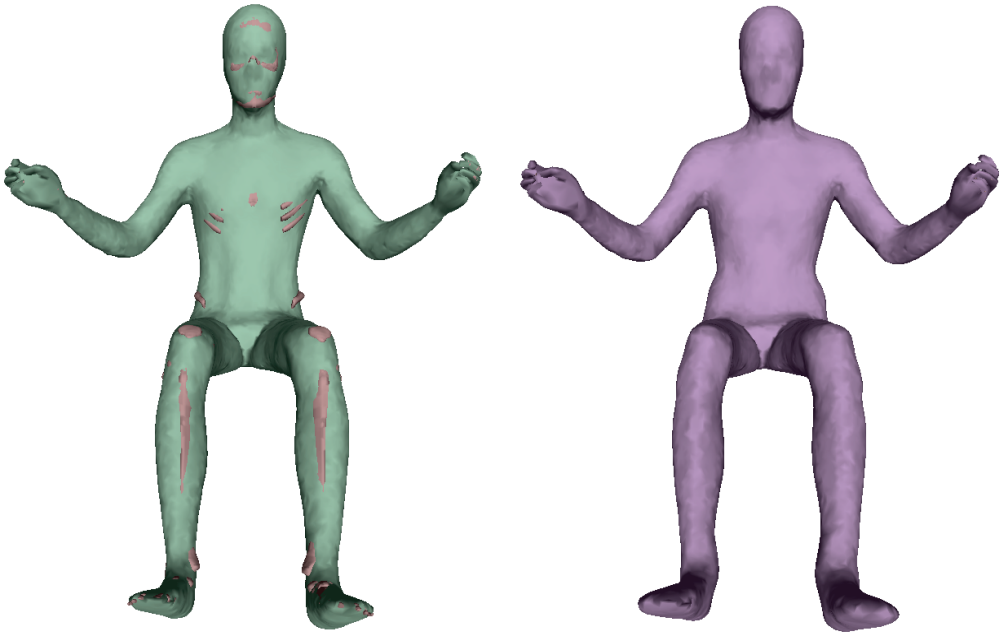

Figure D.2: Target for the BMI 20 test case, before (left) and after (right) manual adjustments. The skeleton is clearly visible in many parts of the model for the initial case, but no longer shows after the manual adjustments.

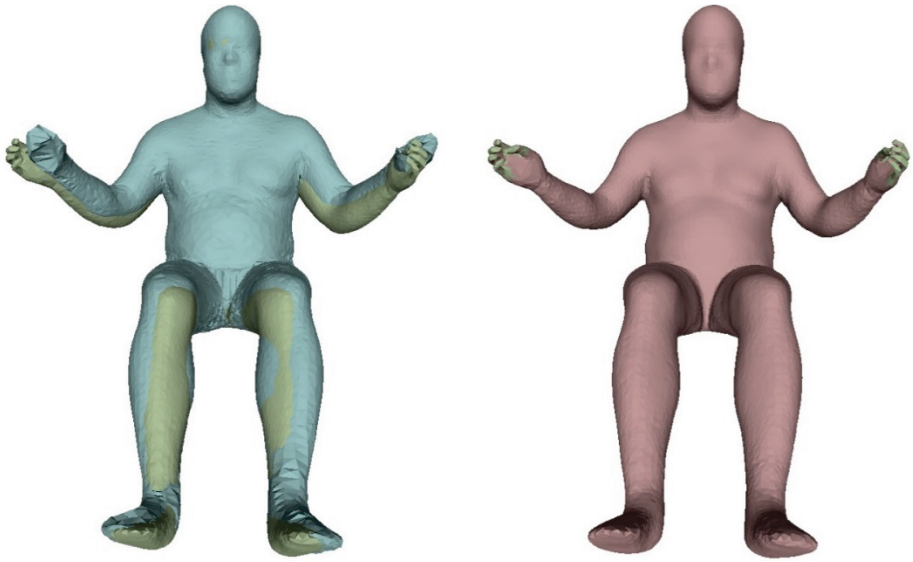

Figure D.3: Target for the BMI 35 test case, before (left) and after (right) manual adjustments. The skin of the baseline model is shown as the green mesh: on the left, the target is clearly not encompassing the entire baseline mesh, while on the right it does (the hands are shared with the baseline model as the BMI 35 lacked them).

Appendix D: Element quality – detailed results

Table D.1: Number of non-tetrahedral solid elements by range of Scaled Jacobian (SJH) for the baseline model and the six models created using the 3S workflow, i.e. each BMI with the neighborhood extension (NE) or with overlapping (OV).

Model : / SJH :	0.0 – 0.3	0.3 – 0.4	0.4 – 0.5	0.5 – 0.6	0.6 – 0.7	0.7 – 0.8	0.8 – 0.9	0.9 – 1
Baseline	73	6965	24099	33644	56152	122447	271775	228797
BMI20/3S/NE	231	7256	23952	34416	56902	121861	273479	225855
BMI20/3S/OV	378	7395	24161	34790	57687	123730	274953	220858
BMI22.7/3S/NE	124	6994	24109	33856	56376	121595	273259	227639
BMI22.7/3S/OV	161	7024	24225	33838	56724	122323	274490	225167
BMI35/3S/NE	251	7498	24113	34215	58119	127551	277945	214260
BMI35/3S/OV	311	7561	24283	34636	58836	128886	278487	210952

Table D.2: Number of tetrahedral elements by range of Tet Collapse for the baseline model and the six models created using the 3S workflow, i.e. each BMI with the neighborhood extension (NE) or with overlapping (OV).

Model : / TC :	0 – 0.2	0.2 – 0.8	0.8 – 1
Baseline	522	789807	114290
BMI20/3S/NE	3225	798284	103110
BMI20/3S/OV	3438	798071	103110
BMI22.7/3S/NE	836	791560	112223
BMI22.7/3S/OV	862	791596	112161
BMI35/3S/NE	1099	805992	97528
BMI35/3S/OV	1116	806367	97136

Table D.3: Number of solid elements by range of Aspect Ratio for the baseline model and the six models created using the 3S workflow, i.e. each BMI with the neighborhood extension (NE) or with overlapping (OV).

Model : / AR :	1-2	2-3	3-4	4-5	5-8.0	8.0-10	10-15	>15
Baseline	1189938	355366	77747	24279	14940	995	326	56
BMI20/3S/NE	1098609	422682	92638	28350	18390	1911	995	72
BMI20/3S/OV	1097719	423040	92913	28488	18503	1932	981	71
BMI22.7/3S/NE	1137677	398001	84894	24645	16314	1749	323	44
BMI22.7/3S/OV	1137701	397664	85021	24787	16328	1758	335	53
BMI35/3S/NE	1056198	448604	104361	33003	19253	1742	452	34
BMI35/3S/OV	1053477	450186	104915	33444	19356	1767	464	38

Table D.4: Number of solid elements by range of Skewness for the baseline model and the six models created using the 3S workflow, i.e. each BMI with the neighborhood extension (NE) or with overlapping (OV).

Model : / SK :	0 – 10	10 –20	20 – 60	60 – 70	70 – 90
Baseline	109494	348647	1186310	17522	1674
BMI20/3S/NE	80786	315407	1230072	32901	4481
BMI20/3S/OV	80556	315474	1229944	33009	4664
BMI22.7/3S/NE	100518	339615	1200339	21126	2049
BMI22.7/3S/OV	100336	339265	1200776	21183	2087
BMI35/3S/NE	88581	328313	1214696	28636	3421
BMI35/3S/OV	87928	327535	1215785	28917	3482

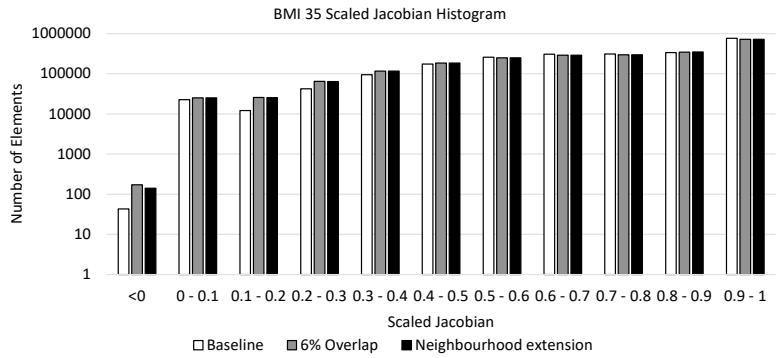

Figure D.4: Histogram of SJV for the BMI 35 test case, comparing the baseline model with the BMI35/3S/NE and BMI35/3S/OV models on a logarithmic scale.

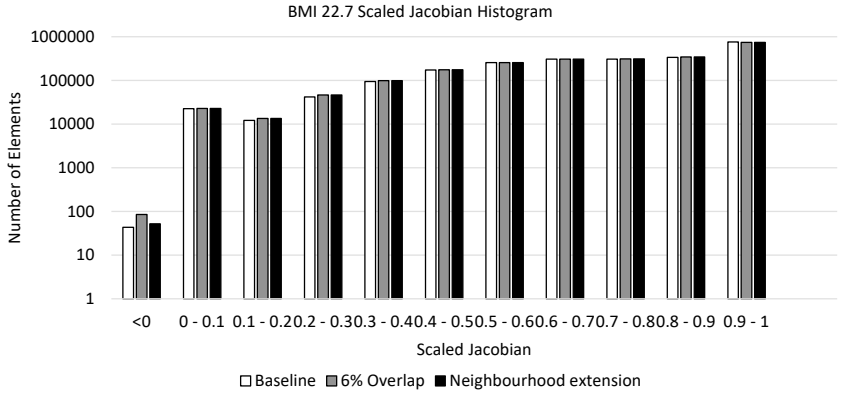

Figure D.5: Histogram of SJV for the BMI 22.7 test case, comparing the baseline model with the BMI22.7/3S/NE and BMI22.7/3S/OV models on a logarithmic scale.

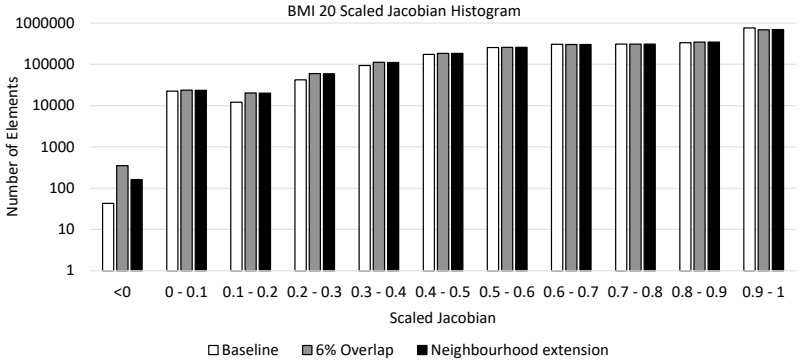

Figure D.6: Histogram of SJV for the BMI 20 test case, comparing the baseline model with the BMI20/3S/NE and BMI20/3S/OV models on a logarithmic scale.

References

[1] B. Baxter, The Interpolation Theory of Radial Basis Functions, 1992, University of Cambridge.

[2] F. Trochu, 1993, “A contouring program based on dual kriging interpolation,” Engineering with Computers, 9(3), pp. 160–177.

[3] D.G. Krige, 1951, “A Statistical Approach to Some Basic Mine Valuation Problems on the Witwatersrand,” Journal of the Chemical, Metallurgical and Mining Society of South Africa, 52(6), pp. 119–139.

[4] E. Jolivet, Y. Lafon, P. Petit, P. Beillas, 2015, “Comparison of Kriging and Moving Least Square Methods to Change the Geometry of Human Body Models.,” Stapp Car Crash J, 59, pp. 337–357.

[5] W.N. Hardy, L.W. Schneider, S.W. Rouhana, 2001, “Abdominal impact response to rigid-bar, seatbelt, and airbag loading.,” Stapp Car Crash J, 45, pp. 1–32.