Size does matter - the use of fish motion for improving human swimming simulations

St. Pacholak[a*], Ch. Brücker[b*]

[a] Institute for Numerical Mathematics and Optimization, Technical University of Freiberg, Akademiestraße 6, 09596 Freiberg, Germany

[b] School of Mathematics, Computer Science and Aeronautics, City, University of London, 10 Northampton Square, EC1V 0HB London, UK

[*] corresponding author: christoph.bruecker@city.ac.uk

keywords: swimming kinematics, surface model, consistency, flexible backbone,

Abstract

In this study we evaluated the length and volume inconsistencies that occur when applying simple motion functions onto surface models of swimming fish and human, respectively. The larger the size (diameter and length) of the body, the larger are theses inconsistencies. To eliminate these errors, different surface correction algorithms are introduced. Additional terms are implemented to overcome structural insufficiencies through over-stretching and overlapping in sensible body parts. For human undulatory swimming these models combine the motion of rigid extremities with the smooth surface bending at joints. The advantages of this hybrid model can be crucial for Computational Fluid Dynamics, where deviation from natural behaviour results in unrealistic surface pressures and artificial vortex structures. The structural improvements are also promising for computational graphic programming, where skin-like stretching surfaces mediate a natural impression.

1. Introduction

The development of Computational Fluid Dynamics (CFD) has tremendously increased the possibilities to study aquatic locomotion. Various physiological factors involved can be detected from such simulations for instance in human swimming. However, in order to operate with realistic parameters and obtain useful results, a precise approximation of the body surface motion is needed. When comparing human undulatory swimming with fish swimming, similarities such as the undulation of an elongated body are obvious. Nevertheless, the almost stiff human body, which mainly bends at joints like shoulders, hips, knees and ankles, is in sharp contrast to the fully flexible fish body. This has consequences on the mathematical models applied to replicate the surface motion in detail.

For fish, there are several studies using rainbow trout (*Oncorhynchus mykiss)* in flow channels [12] to gain the motion parameters for undulatory swimming, even for swimming optimization behind an obstacle within the direction of stream [11]. We use these data herein for computational simulation of dynamic surface models with fully adjustable parameters, allowing to model a wide range of motions. In contrast to Loebbecke et al. [8], who applied automatic function determination by surface mesh manipulation programs, the proposed method does not require a new set of photo frames for every additional motion function. A similar approach was used by the research group of Borazjani [2] who created the fish geometry by using CT scans of a bluegill sunfish and interpolated the backbone at 20 nodes within a video frame. [4] and [6] also used a certain number of interpolation nodes to reconstruct the backbone bending between elliptic cross sections. The error in length

consistency of this method differs depending on the number and length of used splines. Another approach is the Lagrangian type mapping [13] to determine a deformation function of the surface points. However, it does not exactly preserve length and volume consistency, as the authors mentioned themselves. These consistencies should be fulfilled especially when calculating forces and torques [1] or using the digital surface models for CFD simulations. However, the error in length as well as volume conservation is not given in any of the studies.

Similar as in fish motion, these consistencies apply also for human swimming, especially in undulatory motions as e.g. in the underwater dolphin kick. Human undulatory swimming has been investigated by several research groups, with experimental works to determine joint momentum or Strouhal number [7]. Video footage of swim motions are employed to analyse special movements by interpolating surface points between these frames [8]. In contrast, Hochstein [5] presented a technique working with high speed camera systems and self-luminous markers labelling the main joints. The trajectories of these joints were fitted in MATLAB to obtain unique motion functions for each joint that are easily alterable. Pacholak et al. [10] included these results in a CFD analysis to determine drag forces, propulsion and unsteady structures within the flow. They used fully rigid human models, only bending at joints. Loebbecke [8] has introduced another approach based on semi-rigid backbones, when adding one or more auxiliary bending points within the trunk.

Ideally, for numerical simulations of the flow using methods like immersed boundary methods [9] or moving meshes [10] smooth surface models are required that behave naturally when performing bending motions. This paper presents a tentative solution by combining principles used in modelling human and fish swimming under the constrains of preserving length and volume consistency when moving. These conditions are automatically fulfilled in experiments, as in [3], but have to be considered additionally in computational modelling. Therefore, the objectives of the paper are the following:

- the assessment of volume and length constraints on the swimming motion, and
- the assessment of flexible, rigid or hybrid models of the body.

In the methodical section of this paper, we first present a length-consistent algorithm for undulatory swimming that can be applied to flexible bodies and their individual motion functions. Volume consistency is fulfilled automatically after correction of length. Secondly, an algorithm for rigid human undulatory swimming is presented and is improved through skin-stretching vectors to overcome structural insufficiencies as well as a correction term to ensure a constant volume over time. Here, length consistency is ascertained through translation and rotation of rigid body parts at discrete joints. Combining both algorithms into a hybrid model leads to more realistic replication of human swimming, where the backbone is fully flexible through spline interpolation over the natural curvature of the vertebral column. The presented algorithms are named volume consistent under the aspect, that no breathing is performed within the motion. The Result section contains sample calculations for trout and a female German competition swimmer with the according motion functions that were determined by other research groups. Finally, the results are discussed and conclusionsa are given to minimize the volume inconsistency error for individual motion functions.

2. Methods

Comparing the swim motion of fishes with human undulatory swimming shows significant differences in the motion style and degree of flexibility. While human are only able to bend their body at special joints like shoulders, hips and knees, fishes can curve nearly their whole body although it is supported all way long by the vertebral column. Therefore, mathematical models can be used to simulate a common continuous swimming without using photo frames as supporting points. However, length and volume consistency should be ensured which is subject of the following examination. If segments of the body can be treated as rigid extremities as in human swimming, the methods can be combined to generate a more realistic motion behaviour of the body surface.

The methods for analytical description of the motion of the surface points are listed in the following table:

Table 1: List of named methods as used further below in the text.

Body: / Methods:		flexible	chain with rigid bones	hybrid
Simple		Polynomial f(x,t)	Rigid motion	combined
Refined		Length-corrected	1) Skin-stretched 2) and Volume-corrected	combined

2.1. Motion of a continuous flexible body: fish locomotion

The continuous bending of the fish's backbone is generated through alternating contraction and relaxation of opposing muscle groups. In steady swimming, it can be approximated through a periodic function with symmetrical tail beats in lateral direction. We reduced the fish model to the vertebral column (skeleton line) and calculated the surface points with reference to their normal distance from the backbone. Up to now, the movement of the surface points has been determined by a simple polynomial function f(x,t) with varying extrema over time (simple method). However, in order to ensure the length consistency of an arbitrary backbone point *B*, the *x*-position of that point has to be corrected from $x_{p,0}$ to $x_{p,t}$ as well (Fig. 1).

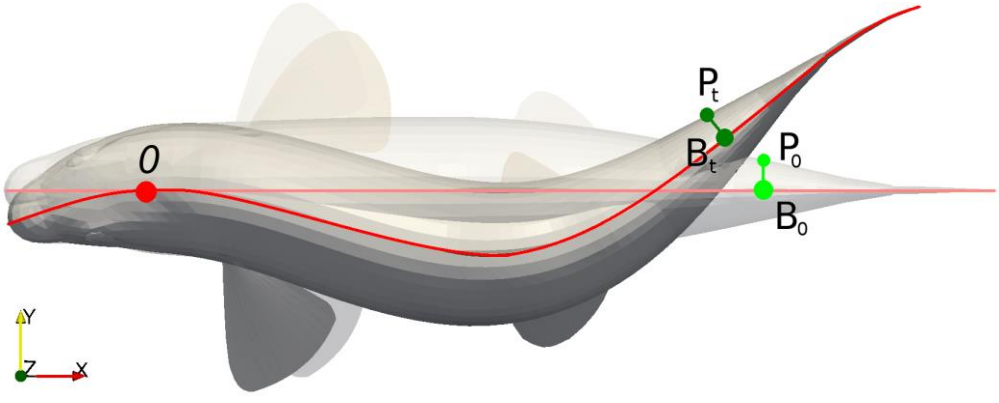

Figure 1: Fish motion. Dorsal view - Position changing of a surface point *P* when referring its position to the backbone with constant length from fix point *0* to *B* along the vertebral column (red line).

Therefore, we defined a reference point *0* near the fish's occiput where no changes of backbone position occurred (fix point). The length *l* from *0* to x_0 and x_t should stay constant:

$$x_0 = \int_0^{x_t} \sqrt{1 + f'(x)^2}\, dx\,. \tag{1}$$

Given that equation 1 can only be solved analytically for $f(x) = \sum_{i=0}^{n} c_i x^i$ if $n = \{0,1\}$, we substituted $g(x) = \sqrt{1 + f'(x)^2}$ and converted $g(x)$ into a Taylor series:

$$T_n(x) = \sum_{k=0}^{n} \frac{g^{(k)}(x_m)}{k!}(x - x_m)^k\,. \tag{2}$$

Aiming at a high accuracy with a low degree number *n*, half of the fish length $x_m = \frac{1}{2}(x_{max} + x_{min})$ was chosen as the center point. The coordinates x_{min} and x_{max} are the positions of the nose tip and tail tip in prone position, respectively. The missing derivatives $g^{(k)}$ can be calculated recursively out of $g^{(k-1)}$ for all $k > 0$ or when using the following explicit computation formula for *g*:

$$\forall n \geq 2: \quad g^{(n)}(x)\,g(x) = \frac{1}{2}\sum_{k=0}^{n}\binom{n}{k} f^{(k+1)}(x)\, f^{(n-k+1)}(x) \quad - \frac{1}{2}\sum_{k=1}^{n-1}\binom{n}{k} g^{(k)}(x)\, g^{(n-k)}(x) \tag{3}$$

The deduction and proof of equation 3 are shown in the Appendix.

Therefore, equation 1 simplifies after integration to

$$x_0 = \sum_{k=0}^{n} \frac{g^{(k)}(x_m)}{(k+1)!}\left((x_t - x_m)^{k+1} - (-x_m)^{k+1}\right). \tag{4}$$

Using the error formula

$$R_n(x) = \int_{x_m}^{x} \frac{(x-\zeta)^n}{n!} g^{(n+1)}(\zeta)\, d\zeta \tag{5}$$

leads to the needed number *n* of Taylor terms at a chosen accuracy. At last, equation 4 is solved for x_t with Newton's approximation method $x_{t,n+1} = x_{t,n} - \frac{h(x_{t,n})}{h'(x_{t,n})}$, where *h* is equation 4 in zero determination state. For the final solution, it has to be valid that $sgn(x_t) = sgn(x_0)$ to get the correct result of two existing real solutions.

After calculating the backbone bending $B_t(x_t; y_t)$, the new surface points $P_t(x_{p,t}; y_{p,t})$ are determined according to their distance from the backbone $y_{p,0}$ in prone position:

$$\begin{pmatrix} x_{p,t} \\ y_{p,t} \end{pmatrix} = \begin{pmatrix} x_t \\ y_t \end{pmatrix} + \frac{y_{p,0}}{g(x_t)} \begin{pmatrix} -f'(x_t) \\ 1 \end{pmatrix}. \tag{6}$$

Note that the above described method also ensures volume conservation as long as the bending curvature is not high in comparison to the total length of the body. Therefore the relative motion between neighbouring points on the surface is not significant. This is discussed further in the results.

2.2. Motion of chain with rigid bones: human undulatory swimming

In contrast to flexible fish motion, human undulatory swimming is performed in straight position with outstretched arms and legs that are only able to be bent at joints. In the following, major pivots like shoulders, hips and knees are significant (see figure 2). Additionally, ankles also play an important role in the dolphin kick. Furthermore, the minimal bending of elbows was neglected and the backbone was assumed to be rigid for this first approach. A co-moving observer in horizontal motion at the same speed as the swimmer sees the shoulder as the region where motion is only up and down in a harmonic manner with small amplitude [5]. Therefore, we chose the reference point at the location of the shoulders. The time- depending position of this reference point can be calculated as following:

$$x_{P_0} = 0, \quad y_{P_0} = 0, \quad z_{P_0} = A_z \cdot \sin(\omega t + \psi_z), \tag{7}$$

where *x*, *y*, *z* are the coordinates of the shoulder joint P_0, A_z is the amplitude of the up and down movement, ω and ψ_z are the angular velocity and phase shift at a given time *t*.

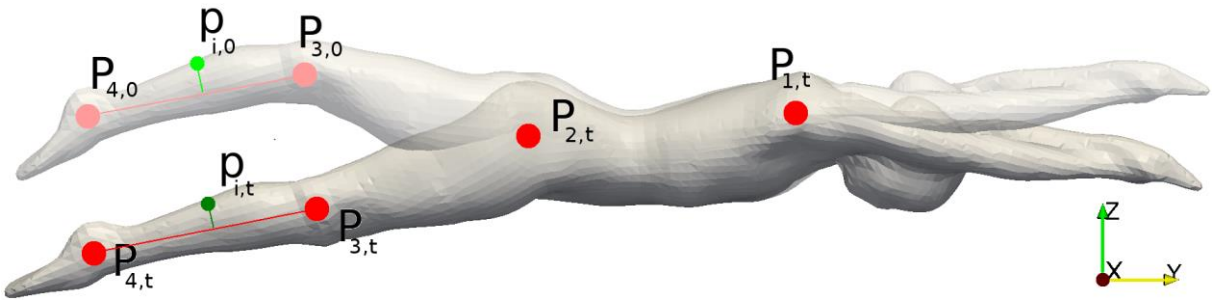

Figure 2: Human undulatory motion. Lateral view - Major pivot points (red) in human undulatory swimming (1 shoulder, 2 hips, 3 knees and 4 ankles) in prone position (index 0) and a time-dependent position (index *t*) to determine the new position $p_{i,t}$ of a surface point $p_{i,0}$ (green).

The motion of the other pivot points along the skeleton is then calculated by the given temporal information of the change of the rotation angles at the joints. The rotation angles around the *x*-axis are the absolute rotation angles α_i for general body part motion and the relative rotation angles θ_i . We use indices *0* and *t* to indicate whether the geometry is in prone position or in motion.

The pivot positions $P_{n,t}$ can be calculated iteratively out of prone position (pivot points with zero index $P_{n,0}$), starting with shoulder-nearest point (hips), followed by knees and finally ankles:

$$\vec{P}_{n,t} = R_x(\alpha)\left(\vec{P}_{n,0} - \vec{P}_{n-1,0}\right) + \vec{P}_{n-1,t} \quad \forall n > 0\,. \tag{8}$$

Therefore, the rotation matrix around *x*-axis $R_x(\alpha)$ is used, containing the absolute rotation angle α calculated with the relative angle θ:

$$\theta_{i,t} = \theta_{i,0} + A_i \cdot \sin(\omega t + \psi_i), \quad \alpha_{i,t} = \sum_{k=2}^{i} \theta_{k,t}\,. \tag{9}$$

As in equation 7, A_i is the amplitude in *z*-direction, ω is the angular velocity and ψ_i is the phase shift of a given pivot *i* at an arbitrary time *t*.

Out of the new pivot positions, the according surface points are calculated in three steps. The pure motion of translation and rotation (10) describes the relative position of the surface point within its body part. Surface deformations caused by overstretched surface cells are repaired with a skin-stretching term (11), where γ is the weighting factor of each surface point $\vec{p}_0 = (x_{p,0}; y_{p,0}; z_{p,0})$ to translate pivot-near points more than distant ones. $\vec{n}$ specifies the direction of smoothing and *dl* contains the according length. To fulfil the volume consistency condition, the surface points are corrected normally to the skeleton-line of the according body part (12).

rigid body:

$$\vec{p}_{simple} = R_x(\alpha_i)\left(\vec{p}_0 - \vec{P}_{n,0}\right) + \vec{P}_{n,t} \tag{10}$$

skin stretching:

$$\vec{p}_{skin} = \vec{p}_{simple} + k_1 R_x(\alpha_i)\gamma_0 dl_0 \vec{n}_0$$

with

$$\gamma_0 = 1 - 2\frac{y_{p,0} - y_{n,0}}{y_{n+1,0} - y_{n,0}}, \vec{n}_0 = \frac{\vec{P}_{n+1,0} - \vec{P}_{n,0}}{\left|\vec{P}_{n+1,0} - \vec{P}_{n,0}\right|} \tag{11}$$

and

$$dl_0 = \tan\left(\frac{\theta_i}{2}\right)\frac{(z_{n+1,0} - z_{n,0})(y_{n,0} - y_{p,0}) + (y_{n+1,0} - y_{n,0})(z_{p,0} - z_{n,0})}{\left|\vec{P}_{n+1,0} - \vec{P}_{n,0}\right|}$$

volume correction:

$$\vec{p}_{volume} = \vec{p}_{skin} + k_2 N |\gamma_t| dl_t \vec{n}_t$$

with

$$\gamma_t = 1 - 2\frac{y_{p,t} - y_{n,t}}{y_{n+1,t} - y_{n,t}}, \vec{n}_t = \frac{\vec{n}_s}{|\vec{n}_s|}, \vec{n}_s = \vec{e}_x \times \left(\vec{P}_{n+1,t} - \vec{P}_{n,t}\right) \tag{12}$$

and

$$dl_t = \cos\left(\frac{\theta_i}{2}\right)\frac{(z_{n+1,t} - z_{n,t})(y_{n,t} - y_{p,t}) + (y_{n+1,t} - y_{n,t})(z_{p,t} - z_{n,t})}{\left|\vec{P}_{n+1,t} - \vec{P}_{n,t}\right|}\,.$$

Inserting a local factor *k* into equation 11 and 12 allows us to define the influence of the smoothing algorithms at each pivot point. These factors determine the degree of smoothing

and have to be adjusted for each swimmer. Additionally, we implemented a global volume correction factor *N*, which was part of a parametric study shown in the Discussion section 4.

2.3. Combined hybrid motion

For a more realistic replication of the body motion in human swimming we combine the rigid body motion of the body extremities with a flexible backbone deformation. Therefore, we used the natural S-shape of the vertebral column and defined the seventh thoracic vertebra (7 TV = maximum dorsal backbone curvature) and the third lumbar vertebra (3 LV = maximum ventral backbone curvature) in prone position as our time-dependent reference points (figure 3) that can be seen as natural constrains.

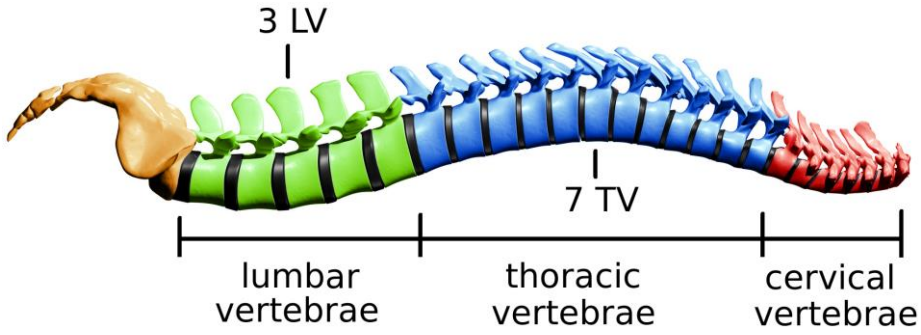

Figure 3: Human backbone. Human vertebral column divided into cervical (red), thoracic (blue) and lumbar vertebrae (green) with tagged seventh thoracic vertebra (7 TV) and third lumbar vertebra (3 LV) used as reference points for flexible backbone movement.

Torso bending was simulated with cubic spline interpolation through the shoulders (S), 7 TV, 3 LV and hips (H):

$$p_i = c_{i,3}x^3 + c_{i,2}x^2 + c_{i,1}x + c_{i,0} \quad \forall i = 1,2,3. \tag{13}$$

For the first cubic spline p_1, from shoulder to 7 TV, considering the maximum condition for 7 TV and that slope *m* at the shoulders has to be equal to rigid human motion, we got

$$C_1 = \begin{pmatrix} y_{7TV}^3 & y_{7TV}^2 & y_{7TV} & 1 \\ y_S^3 & y_S^2 & y_S & 1 \\ 3y_{7TV}^2 & 2y_{7TV} & 1 & 0 \\ 3y_S^2 & 2y_S & 1 & 0 \end{pmatrix}, \quad b_1 = \begin{pmatrix} z_{7TV} \\ z_S \\ 0 \\ m \end{pmatrix}. \tag{14}$$

The second spline p_2, from 7 TV to 3 LV, looks similar to p_1 but with different indices and unequal right-hand side *b*:

$$C_2 = \begin{pmatrix} y_{7TV}^3 & y_{7TV}^2 & y_{7TV} & 1 \\ y_{3LV}^3 & y_{3LV}^2 & y_{3LV} & 1 \\ 3y_{7TV}^2 & 2y_{7TV} & 1 & 0 \\ 3y_{3LV}^2 & 2y_{3LV} & 1 & 0 \end{pmatrix}, \quad b_2 = \begin{pmatrix} z_{7TV} \\ z_{3LV} \\ 0 \\ 0 \end{pmatrix}. \tag{15}$$

To fulfil the length consistency condition for the torso at every calculated time step, the third spline p_3 has a variable endpoint according to the length of p_1 and p_2, so that

$l_3 = l_{backbone} - (l_1 + l_2)$. Therefore, we used equation 1 to calculate the length of p_1 and p_2 and the entire equation set-up from flexible fish motion algorithm with length consistency to determine the endpoint $P_H\,(y_H; z_H)$ of the third spline, which is equivalent to the new position of the hips (H):

$$C_3 = \begin{pmatrix} y_H^3 & y_H^2 & y_H & 1 \\ y_{3LV}^3 & y_{3LV}^2 & y_{3LV} & 1 \\ 3y_H^2 & 2y_H & 1 & 0 \\ 3y_{3LV}^2 & 2y_{3LV} & 1 & 0 \end{pmatrix}, \quad b_3 = \begin{pmatrix} m \cdot y_H + z_S \\ z_{3LV} \\ m \\ 0 \end{pmatrix}. \tag{16}$$

The z-value of the endpoint has to be calculated by taking into account the motion of the shoulders as well as the rotation angle of the torso $m = \tan(\alpha_{torso})$.

When solving $C_i x = b_i$ for $i = 1,2,3$ to determine the coefficients $c_{i,j}$ for $j = 0,1,2,3$ of the three splines, we get a functional description of the complete backbone deformation and are able to calculate the new surface points of the torso. All remaining body parts (arms, upper legs, lower legs and feet) are rotated and translated as rigid bodies, as described in section 2.2. Skin-stretching and volume correction are performed in the same way as well.

3. Results

We used data from a rainbow trout in the first part and a female human swimmer in the second part to highlight differences between the algorithms without any length or volume consistency conditions and the refined methods which hold length and volume constant. In the third part, the hybrid model of human swimming is applied, where the backbone is flexible and the extremities behave rigid.

To test our algorithms we used digitized models of trout from the Online Toucan Virtual Museum of the Toucan Corporation Japan (http://www.toucan.co.jp) and the body scan of a female human swimmer from Pacholak et al. [10]. Both surface datasets were saved as stereo-lithography file (STL), so that each model consists of a finite number of triangles describing the subject's surface. The implementation of a sample motion function onto these STL-surfaces was realised with MATLAB 2014b (The Mathworks Inc. 1994-2015) reading the STL-data, modifying the coordinates of each surface point according to the motion function and rewriting them as a new STL-dataset [10]. Volume, surface area and length of the calculated models were determined with Meshmixer 2.9 (2011 Autodesk Inc., http://www.meshmixer.com) and visualised with ParaView 4.1.0 (Kitware, http://www.paraview.org).

3.1. Motion of a continuous flexible body

The typical motion function of the trout skeleton-line for undulatory swimming in cruising conditions was taken from [12]:

$$f(x) = (ax^2 + bx) \cdot \sin(cx + 2t\pi), \tag{17}$$

where t is the dimensionless time index within a motion cycle, $a = -0.023$, $b = 0.2573$ and $c = 15$ to fit the length of our trout with $l_{fish} = 0.40m$. With

motion function (17) we had a maximum tail deflection of about $\alpha \approx 21°$. The given motion function and the digitized data of the body contour of the trout are then used to determine the surface motion at each point of the STL surface with our algorithms.

In figure 4 significant time steps (I-IV) are shown during one motion cycle. We refer in the figures later to these time-steps.

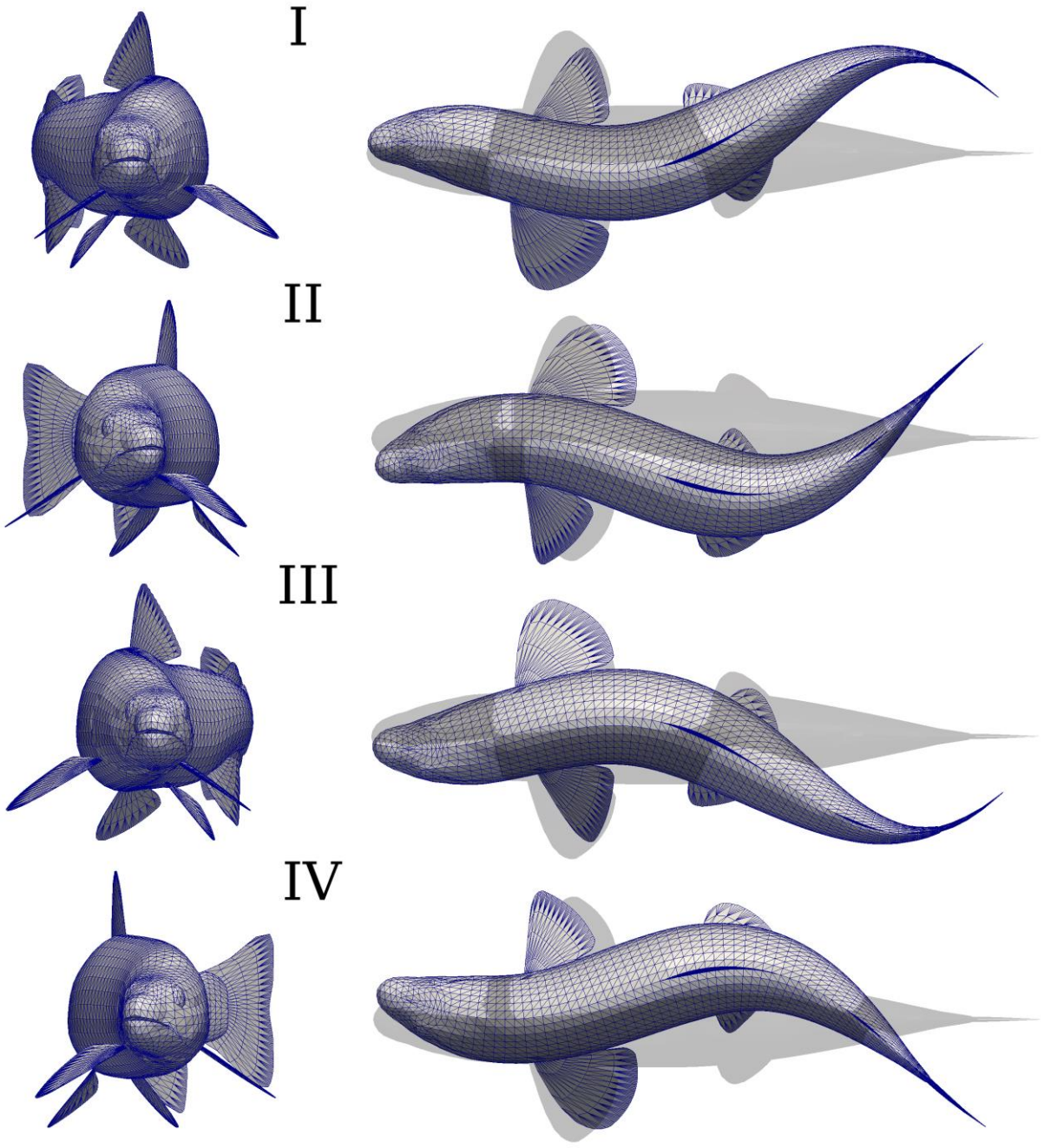

Figure 4: Fish motion cycle. One motion cycle of length-corrected trout swimming according to equation (17) at significant time steps: I) maximum tail deflection (right side), II) median tail go-through (from right to left), III) maximum tail deflection (left side) and IV) median tail go-through (from left to right) compared with the size of gliding trout (shade) in anterior and dorsal view.

Deviation in length and volume between the simple motion algorithm and the refined model, where the consistency conditions are fulfilled, is visually hard to estimate. Therefore, we plotted the results for length and volume referred to the initial geometry in figure 5A and 5B,

respectively. The blue graph indicates that the simple algorithm creates a length about 10 % higher in average and consequently a volume increase of about 5 %. Mean and maximum deviation of length and volume are shown in table 2.

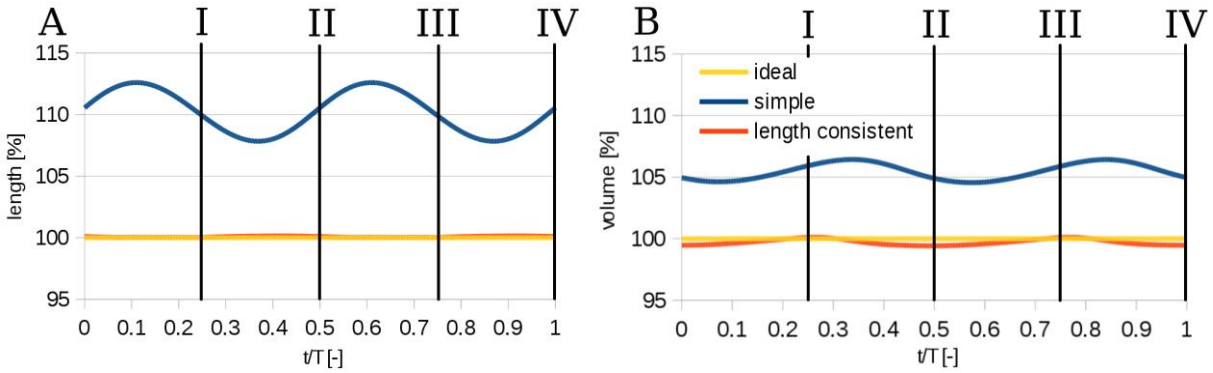

Figure 5: Time dependent length and volume of fish swimming. A) Time-dependent length and B) volume changing of trout over one motion cycle for the simple algorithm (blue), the refined model (red) and the ideal value (yellow) corresponding to the initial geometry. The Roman numbers (I-IV) indicate significant time steps as defined in figure 4.

Table 2: Average and maximum values of length error, volume error and surface area deviation over one motion cycle of trout for the simple algorithm and the refined model.

method	length error mean	length error max.	volume error mean	volume error max.	surface deviation mean	surface deviation max.
simple [%]	10.21	12.55	5.47	6.43	8.10	9.40
refined [%]	0.08	0.15	0.28	0.58	0.01	0.03

Additionally, we calculated the surface area at each time step to get a first impression of the time-dependent behaviour of symmetric motion functions applied onto symmetric surface models. These results can also be seen in table 1 and are plotted over one motion cycle in figure 5. Here again, the blue graph demonstrates a much higher surface area of about 8 % when using the simple algorithm.

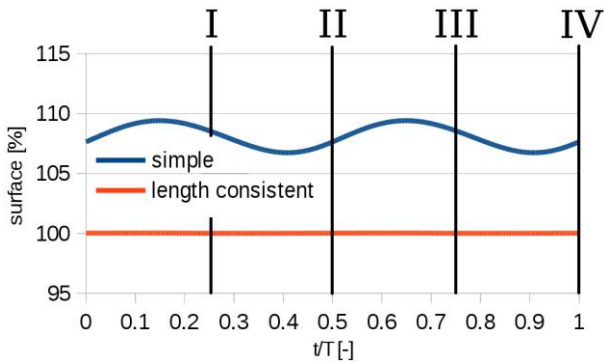

Figure 6: Time-dependent surface of human swimming. Time-dependent surface area deviation of trout over one motion cycle for the simple algorithm (blue) and the refined model (red) referred to the initial geometry. The Roman numbers (I-IV) indicate significant time steps as defined in figure 4.

3.2. Motion of chain with rigid bones

The undulatory underwater movement of a female swimmer is documented in [10]. To capture the motion function of the swimmer, the main joints of the female swimmer were labelled with self-luminous markers. These light dots were traced by a high-speed camera system and the trajectory was finally fitted with MATLAB to a unique function [5] for each joint (9). From these data the absolute and relative angles were captured for a complete stroke cycle and were fed into our algorithm. In Figure 7, the most significant time steps during one motion cycle, to which the following diagrams refer, are indicated by Roman numbers (I-IV). Because of rotation and translation, each body part (arms, upper body, upper legs, lower legs and feet) has always the same length independent of the used motion function. So, overall body length always stays constant. However, the volume of the body may differ between the simple and the refined algorithm.

When comparing the simple motion algorithm with the skin-stretching method (1st refinement, see table 1), the differences in volume are minor. The main effects of this natural skin-stretching apply for CFD where smooth surface models are needed (see Discussion section 4 for details). Visible differences between skin-stretching and volume-correction are small, but volume error shown in figure 8A and surface deviation plotted in 8B are still observable. The blue and red lines indicate that the simple and skin-stretching algorithms deliver a volume that is about 2 % too low in average. With the additional volume correction term (2nd refinement, see table 1) we could achieve a lower volume error of about 0.1 %. A collection of mean and maximum deviations for volume and surface results is presented in table 3 for all three versions of the rigid human swimming algorithm.

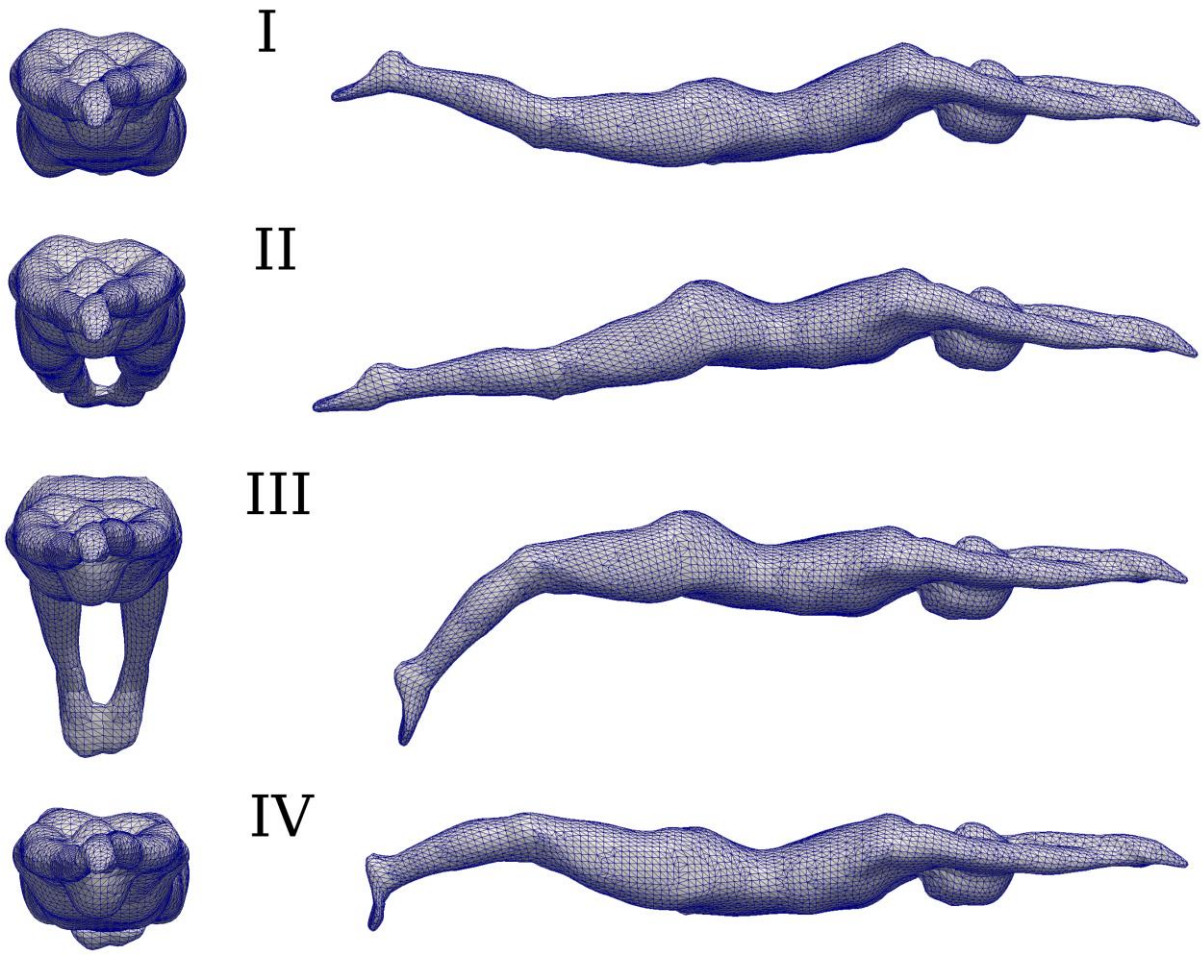

Figure 7: Human undulatory swimming cycle. One motion cycle of human swimming with rigid backbone at significant time steps: I) maximum feet deflection (dorsal), II) median feet go-through (from dorsal to ventral), III) maximum feet deflection (ventral) and IV) median feet go-through (from ventral to dorsal) in anterior and lateral view.

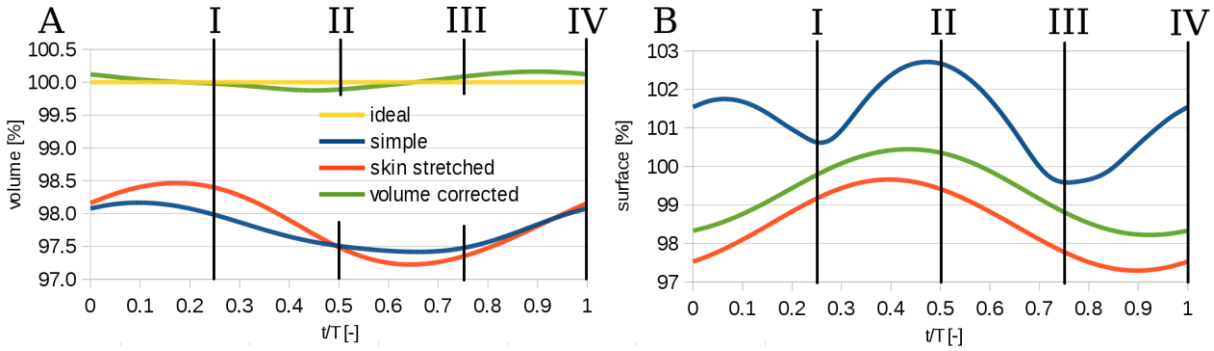

Figure 8: Time-dependent surface and volume of human swimming. A) Time-dependent volume and B) surface changing of a female human swimmer over one motion cycle for the simple algorithm (blue), the skin-stretching method (red) and the volume correction (green) referred to the initial geometry. This corresponds to the ideal volume (yellow) in A. The Roman numbers (I-IV) indicate significant time steps as defined in figure 7.

Table 3: Average and maximum values of volume error and surface area deviation over one motion cycle of a female human swimmer for the simple algorithm, the skin-stretched method and after the volume correction.

method	volume error mean	volume error max.	surface deviation mean	surface deviation max.
simple [%]	2.24	2.58	1.21	2.71
skin-stretched [%]	2.15	2.77	1.53	2.70
volume-corrected [%]	0.02	0.16	0.68	1.77

3.3. Combined hybrid motion

When combining the rigid human motion for the body extremities with the flexible bending of the vertebral column, we were able to obtain a more realistic surface motion in undulatory swimming when comparing to the video recordings in [5]. Therefore, we used for the joints the motion functions from rigid human swimming and defined the time-dependent position of the seventh thoracic vertebra and the third lumbar vertebra as following:

$$z_{7TV}(t) = A_{7TV}\sin(2t\pi + \phi_{7TV}) + z_{7TV,0} \quad + (my_{7TV} + z_S) \text{ and} \tag{18}$$

$$z_{3LV}(t) = A_{3LV}\sin(2t\pi + \phi_{3LV}) + z_{3LV,0} \quad + (my_{3LV} + z_S), \tag{19}$$

where the first term of these equations describes the time-dependent position of the vertebra around an offset z_0 (relative to the vertebral column) with the amplitude A and the phase shift ϕ. To transform the vertebra relative position into the global coordinate system taking the current position of the shoulders z_S as well as the torso slope m into account, we added the second term to equation 18 and 19. Amplitude, offset and phase shift were determined by recording swimmers in a swim channel with high-speed cameras through a side window [5].

In figure 9, the hybrid movement in anterior view and a close-up of the torso bending is visualised for the characteristic time steps. These are taken as reference in figure 10, where length and volume error as well as surface deviation over one motion cycle are drawn. As can be seen, the length error of the hybrid algorithm is zero (green) and the volume error (blue) shows an acceptably low value when adapting the volume correction factor N in equation 12. Additionally, we plotted the surface area referred to the initial geometry (red), to obtain a time-dependent deviation. Mean and maximum values of these curves are also presented in table 4 in comparison to the flexible and rigid motion.

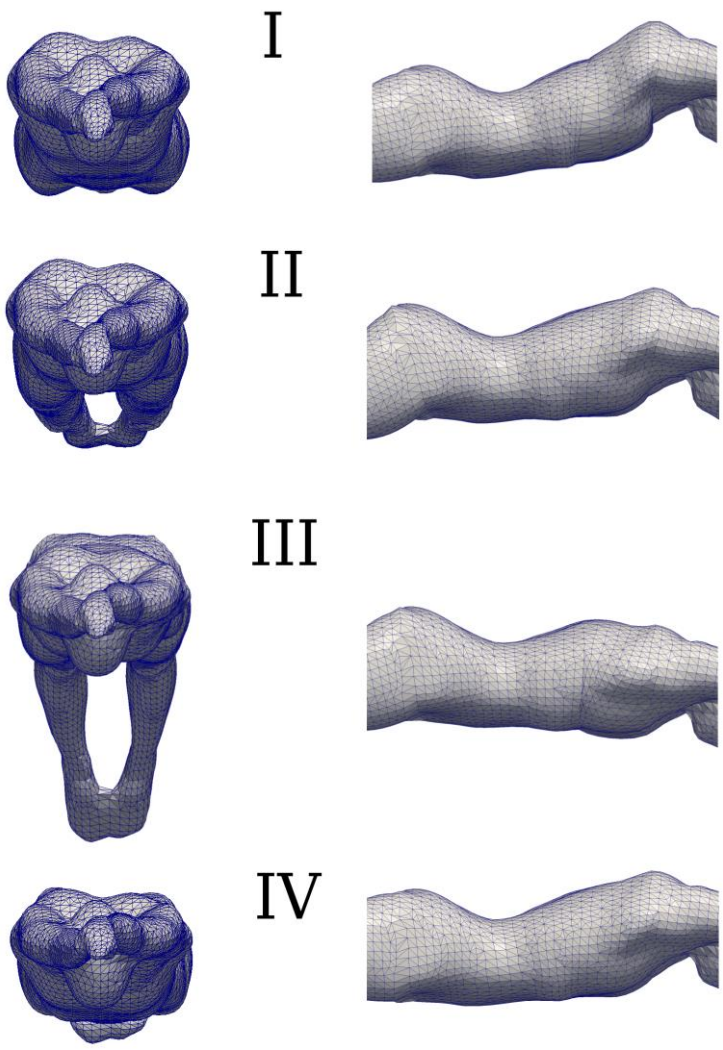

Figure 9: Hybrid swimming motion cycle. One motion cycle of combined hybrid swimming with flexible backbone at significant time steps: I) maximum feet deflection (dorsal), II) median feet go-through (from dorsal to ventral), III) maximum feet deflection (ventral) and IV) median feet go-through (from ventral to dorsal) in anterior view and a close-up of the trunk in lateral view.

Table 4: Average and maximum values of length and volume error as well as surface area deviation over one motion cycle of trout, the rigid human swimmer and the combined hybrid algorithm.

method	length error mean	length error max.	volume error mean	volume error max.	surface deviation mean	surface deviation max.
flexible [%]	0.08	0.15	0.28	0.58	0.01	0.03
rigid [%]	0.00	0.00	0.02	0.16	0.68	1.77
hybrid [%]	0.00	0.00	0.00	0.54	0.60	2.17

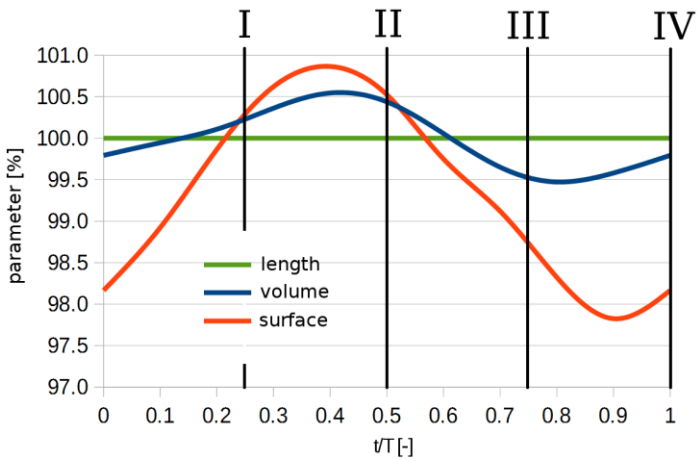

Figure 10: Time-dependent length, volume and surface of hybrid swimming. Time-dependent length (green), volume (blue) and surface deviation (red) of the human swimmer with flexible backbone over one motion cycle compared to the initial geometry. The Roman numbers (I-IV) indicate significant time steps as defined in figure 7.

4. Discussion

In this section, we will discuss some parametric adjustments to simplify and accelerate the computational calculations and achieve lower parameter errors. Especially when investigating the format of the geometries, saved as STL-files, great potential for speed-up was found.

4.1. Motion of a continuous flexible body

Surface data in STL-format consists of a finite number of triangles saved as a triple set of points with *x*-, *y*- and *z*-coordinates and the normal vector of this surface. For first approaches we read the STL-files and got an array of points in order of appearance. So, each point was present more than once in the point list. Keeping the point order has the advantage that no time-consuming surface re-triangulation is necessary after motion calculation. To speed-up the algorithm, we simplified the point list through deleting multiple appearances of every point and created an ID-list where the original order of the points was saved. So, number of points was reduced by a factor of 8 and a time saving of about 90 % could be reached. After calculation, the shortened point list was completed with the IDs of the deleted points to recreate the STL format.

The simple algorithm (no length consistency) produces 10 % mean and about 13 % maximum deviation from initial length for $l_{fish} = 0.40m$ and a maximum tail deflection of 21° (figure 5, table 1). So, if length is increasing over time, volume error based on the initial volume is also increasing significantly. Here, we had a mean enlargement of about 5 % and a maximum of 6%. Using these geometries of trout for CFD simulations would lead to wrong pressure and thrust values as well as vortex structures around the fish body. The deviations would increase when working with larger fishes like needlefish (*Belone*) or barracuda (*Sphyraena*), more complex motion functions or wider tail deflections. When applying our improved algorithm to fish swimming using the same motion parameters and species, we got a bending behaviour of the trout body with a mean length error of less then 0.1 % and a maximum error of 0.15 %. Hence, the mean volume deviation also decreased to 0.3 % with a maximum of 0.6 %. These values can be minimized further just by increasing the number of Taylor terms and consequently raising the accuracy. The number of terms needed has to be adjusted to fish length and the complexity of the used motion function.

Additionally, surface deviation relative to the initial surface area is shown in figure 6 and table 1. Just as the volume error, the mean surface deviation of 8 % and the maximum of 9 % results from length inconsistency of the simple algorithm. When considering a constant length, this error shrinks to a value of less than 0.03 %. Hence, this curve shows a better approximation of the time-dependent surface changing when using a symmetric motion function applied to a symmetric surface model. Nevertheless, the surface behaviour has to be investigated for accelerated swimming with non-periodic motion functions.

4.2. Motion of chain with rigid bones

In contrast to fish swimming, where surface smoothness is guaranteed for low tail deflections, the surface structure of human causes some undesirable irregularities at joints through strong bending, which make CFD simulations impossible. In figure 11, the most problematic areas of surface demolition through overlapping (hips in C and shoulders in E) as well as over-stretching (knees in B, hips in D and shoulders in E) are shown. The structure of feet in A is extremely disturbed because of their low thickness and high excursion. Finally, the ventral surface deformation in E would abort CFD calculations after a short time.

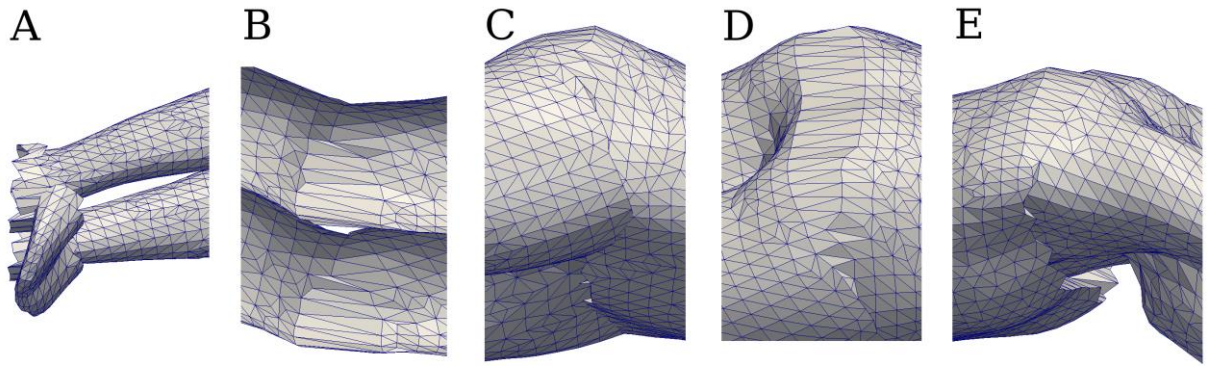

Figure 11: Surface deformation. Surface deformation irregularities of the simple motion algorithm in rigid human swimming of A) the feet, B) the knees C & D) and the hips and E) the shoulders.

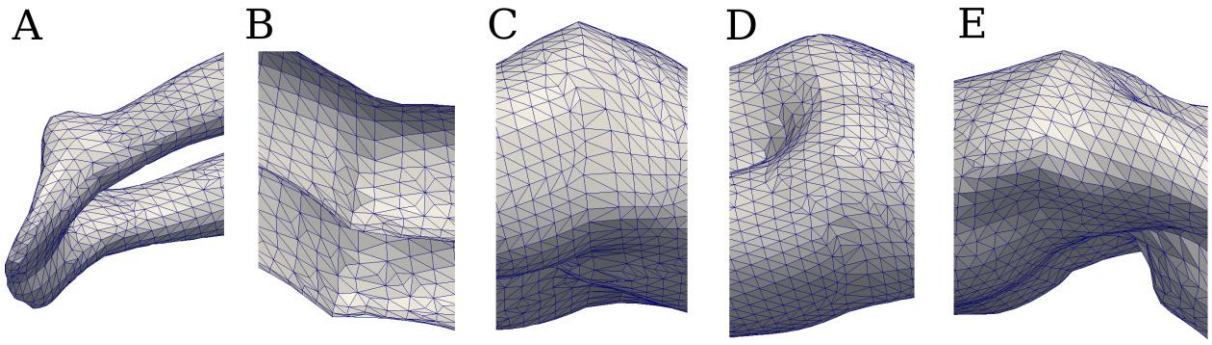

Figure 12: Surface repairing. Corrected surface deformation irregularities from figure 11 when applying the skin-stretching method in rigid human swimming at A) the feet, B) the knees C & D) and the hips and E) the shoulders.

Therefore, we decided to implement a surface deformation vector approximating the stretching of natural human skin during bending motions. Hence, pivot-near points were moved more intensely along the body surface than more distant ones to obtain a smooth surface and even grid structure. The improved close-ups of surface deformation from figure 11 are shown in figure 12 for the skin-stretching method. Now, even feet (A) and shoulders (E) display realistic surface deformation without sharp edges or grid destruction through

overlapping and over-bending. The slightly angled structures of the knees (B) represent the kneecaps that appear depending on the swimmer's body size and knee excursion.
After these necessary surface corrections, we investigated the consistency of length and volume over one motion cycle. Just as in fish swimming, we assume that the swimmer is not breathing and his volume stays constant over time. Unfortunately, through the bending we got a mean volume loss of about 2.2 % and maximum of more than 2.5 % even when skin-stretching algorithm was applied (figure 8, table 2). A length inconsistency was automatically excluded because of the kind of motion algorithm where the movement is simply accomplished through translation and rotation of the five body parts (arms, torso, upper legs, lower legs and feet) at collective pivot points. To find an appropriate volume correction, we performed a parametric study for N in equation 12 and plotted the results for several constants in figure 13. These results did not satisfy our requirements for volume errors less than 1 % even when reaching a mean error of about zero. Better results were achieved through a dynamic adjustment of N over one motion cycle when adding a negative resonance wave for N instead of a static value:

$$N = -A_N \sin\left(2t\pi + \phi_N\right) + N_0 \,, \tag{20}$$

where t is the dimensionless time index over one motion cycle, ϕ_N the phase shift of the resonance wave, A_N the amplitude and N_0 the offset that is equivalent to the curve N with a mean error of zero. After determining amplitude and offset, the phase shift was calculated through cross correlation. These three factors have to be adjusted depending on the swimmer's geometry and the according motion functions to minimize the volume error individually. To keep the structural integrity of small body parts like feet, we set k_2 in equation 12 to zero for the last body part, otherwise this segment would be deformed unnaturally.
The volume determined using the dynamic factor N is presented in figure 8A (green) as well as in table 2 and shows significantly lower mean (<0.02 %) and maximum (0.16 %) errors compared to the initial geometry. Surface deviation over time, plotted in figure 8B, exhibits a mean decrease of 0.68 % and a maximum decrease of 1.77 %. An explanation for the shrinking surface area may be the higher flexibility of human skin in contrast to fish skin.

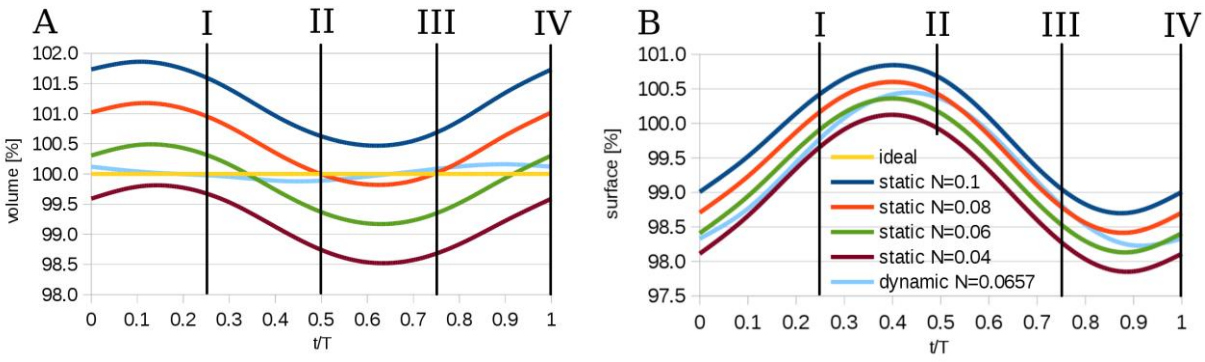

Figure 13: Parametric adjustments. Parametric study of A) volume error and B) surface deviation over one motion cycle at rigid human swimming with skin-stretching and volume correction for several static factors N and a dynamic adaptation (light blue). The Roman numbers (I-IV) indicate significant time steps as defined in figure 7.

4.3. Combined hybrid motion

The hybrid motion algorithm, where all body parts except the torso are moved as in rigid human swimming and the torso bending is simulated similar to fish motion, combines the advantages of both models regarding speed-up and error reduction. So, we could reduce the

calculation time by the factor of 20, when using the point list reformatting of fish bending and the dynamic number of Taylor terms. Additionally, we were able to keep a length consistency error of near zero as in rigid human swimming. The best results of fish, rigid human and hybrid swimming are compared in table 3 and show lower errors for the hybrid method than for rigid swimming. As for rigid human swimming, the parameters of the dynamic factor *N* in equation 20 (phase shift, amplitude and offset) had to be adjusted for this motion algorithm. The higher surface deviation may be caused by more dynamically and asymmetrically moving body points (torso). To visualize the differences of both motion algorithms in detail, we plotted the rigid motion with the shade of hybrid swimming in figure 14A, the hybrid swimming with the shade of rigid motion in B and the pure backbone trajectory from shoulders to hips of both methods in C.

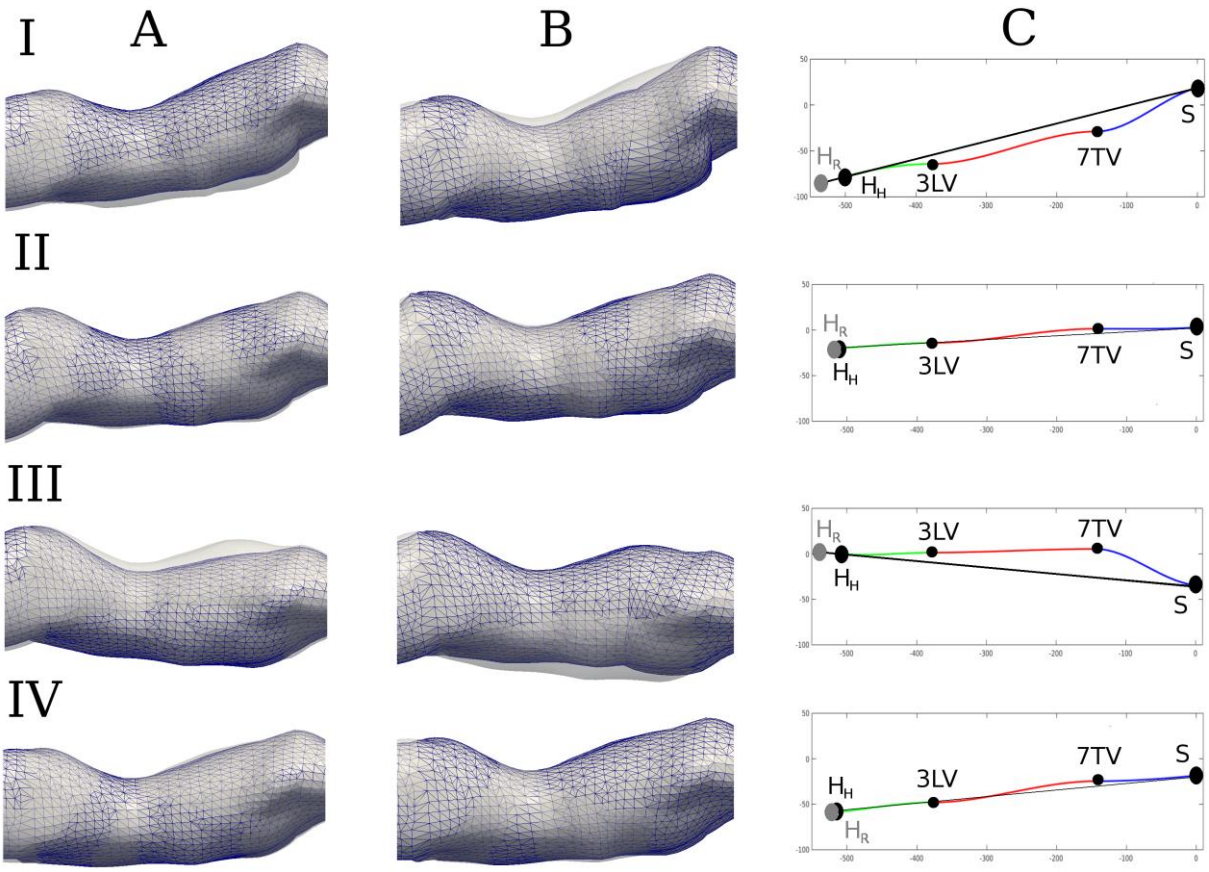

Figure 14: Motion cycle comparison. Torso close-up of human rigid and hybrid swimming models over one motion cycle. A) rigid torso with shade of the hybrid model; B) the flexible torso with shade of the rigid model; C) comparison of the backbone functions for rigid (black) and hybrid swimming (spline 1 - blue, spline 2 - red and spline 3 - green) from shoulder (S) over 7 TV and 3 LV to hips (H). Indices R and H indicate the position of hips when using rigid algorithm and hybrid method, respectively. The Roman numbers (I-IV) indicate significant time steps as defined in figure 7.

As can be seen, the differences between both methods are highest at time steps I and III, when maximum deflection of feet is reached. At dorsal maximum (I), the hybrid back shows a concave curvature, whereas the rigid model is slightly convex. In contrast, the hybrid algorithm generates a convex torso bending at time step III. These differences seem minimal, but when using CFD to investigate the vortex transport along the body surface, substantial changes in vortex energy and propulsion may occur.

5. Conclusion

In this study, we evaluated calculation results from simple motion algorithms, which are not considering consistency criteria for length and volume, applied onto well-investigated swimming subjects like rainbow trout and a human swimmer. Refined models were presented to overcome these inconsistencies and the results were compared against each other. In fish-type swimming with a continuous flexible body the results show that even low tail deflections of 21° led to length errors of more than 10 % and volume deviation of about 6 % if calculated with the simple models. To overcome these problems, we introduced an algorithm fulfilling the length consistency condition with user-defined errors that can be adjusted by increasing the number of Taylor terms. As a result, the volume error was minimized to an acceptable range.

In comparison to fish-type swimming of flexible bodies, human swimming is rather the motion of a chain of rigid bones that are covered with flexible skin. Because of the rigid elements, the length consistency is automatically fulfilled when simulating the motion of the chain with rotation and translations of the individual elements. However there is still a remaining inconsistency in volume of 2.5 %. More importantly, the simple algorithm caused serious structural problems along the surface near to the joints that did not only affect the visual impression of the moving geometry for computer graphics, but also prohibited further CFD applications where calculations of surface pressure and forces would become false or impossible due to artificial surface roughness on the surface mesh. These issues could be corrected through a skin-strechting correction which leads to a natural behaviour of the flexible skin. In addition, a further refinement with a dynamic volume-correction term minimizes the deviation of the volume to a decent value.

Finally, we combined the experiences and advantages of these methods to generate a human swimmer with a flexible backbone. This hybrid algorithm combining rigid segments with flexible ones required a consideration of length and volume consistency and parameter adjustments for the volume correction term. The resulting motion cycle looks much closer to the natural situation captured in video recordings of swimmer [5]

6. Appendix

For easier systematic illustration, all derivatives are styled as powers of Arabian numbers in brackets, $f(x) = f$ as well as $g(x) = g$.

Proof of equation 2 by the use of mathematical induction for $n \geq 2$:

$$
\begin{aligned}
&n = 0\text{:}\quad g \quad g = 1 + f^{(1)} f^{(1)} \rightarrow g = \sqrt{1 + f^2} \\
&n = 1\text{:}\quad g^{(1)} g = \mathbf{1} f^{(1)} f^{(2)} \\
&n = 2\text{:}\quad g^{(2)} g = \mathbf{1} f^{(1)} f^{(3)} + \mathbf{1} f^{(2)} f^{(2)} \qquad -\mathbf{1} g^{(1)} g^{(1)} \\
&n = 3\text{:}\quad g^{(3)} g = \mathbf{1} f^{(1)} f^{(4)} + \mathbf{3} f^{(2)} f^{(3)} \qquad -\mathbf{3} g^{(1)} g^{(2)} \\
&n = 4\text{:}\quad g^{(4)} g = \mathbf{1} f^{(1)} f^{(5)} + \mathbf{4} f^{(2)} f^{(4)} + \mathbf{3} f^{(3)} f^{(3)} \qquad -\mathbf{4} g^{(1)} g^{(3)} - \mathbf{3} g^{(2)} g^{(2)}
\end{aligned}
\tag{21}
$$

Focusing the bold numbers of the stepwise derivatives, led to the elements of Pascal's triangle divided by factor 2. From this, it follows that the *n*-th derivative is:

$$
g^{(n)} g = \frac{1}{2} \sum_{k=0}^{n} \binom{n}{k} f^{(k+1)} f^{(n-k+1)} \qquad - \frac{1}{2} \sum_{k=1}^{n-1} \binom{n}{k} g^{(k)} g^{(n-k)} \,. \tag{22}
$$

To prove the inductive step from $n \rightarrow n+1$, it's necessary to show that $g^{(n+1)} g + g^{(n)} g^{(1)} = \left(g^{(n)} g \right)'$, where:

$$
g^{(n+1)} g = \frac{1}{2} \sum_{k=0}^{n+1} \binom{n+1}{k} f^{(k+1)} f^{(n-k+2)} \quad - \frac{1}{2} \sum_{k=1}^{n} \binom{n+1}{k} g^{(k)} g^{(n-k+1)} \,, \tag{23}
$$

$$
g^{(n+1)} g + g^{(n)} g^{(1)} = \frac{1}{2} \sum_{k=0}^{n} \binom{n}{k} \left(f^{(k+2)} f^{(n-k+1)} + f^{(k+1)} f^{(n-k+2)} \right) - \frac{1}{2} \sum_{k=1}^{n-1} \binom{n}{k} \left(g^{(k+1)} g^{(n-k)} + g^{(k)} g^{(n-k+1)} \right).
$$

The derivatives of *f* and *g* are adjusted through index shifting:

$$
\begin{aligned}
&\sum_{k=0}^{n+1} \binom{n+1}{k} f^{(k+1)} f^{(n-k+2)} \quad - \sum_{k=1}^{n} \binom{n+1}{k} g^{(k)} g^{(n-k+1)} = \\
&\sum_{k=1}^{n+1} \binom{n}{k-1} f^{(k+1)} f^{(n-k+2)} + \sum_{k=0}^{n} \binom{n}{k} f^{(k+1)} f^{(n-k+2)} - \sum_{k=2}^{n} \binom{n}{k-1} g^{(k)} g^{(n-k+1)} - \sum_{k=1}^{n-1} \binom{n}{k} g^{(k)} g^{(n-k+1)} - 2 g^{(n)} g^{(1)}
\end{aligned}
\tag{24}
$$

and the sum borders are matched for combining through element ejection:

$$
\sum_{k=1}^{n} \left(\binom{n+1}{k} - \binom{n}{k-1} - \binom{n}{k} \right) f^{(k+1)} f^{(n-k+2)} = \sum_{k=2}^{n} \left(\binom{n+1}{k} - \binom{n}{k-1} - \binom{n}{k} \right) g^{(k)} g^{(n-k+1)} \,. \tag{25}
$$

Finally, the ejected elements of *f* and *g* nullify each other.

Focusing the binomial coefficients and substituting $k = p + 1$led to the Pascal rule:

$$\binom{n+1}{p+1}-\binom{n}{p}-\binom{n}{p+1}=0 \tag{26}$$

which proves that equation 22 is correct.

Annotation: Equation 22 is also valid for $n=1$, if we assume that sums with negative range (upper border less than lower border) are zero.

7. Acknowledgement

The authors acknowledge support from the Deutsche Forschungsgemeinschaft (DFG) in the beginning of this research subject. Funding of the position of Professor Christoph Bruecker as the BAE SYSTEMS Sir Richard Olver Chair in Aeronautical Engineering is gratefully acknowledged.

8. References

[1] M. Bergmann, A. Iollo, Modeling and simulation of fish-like swimming, Journal of Computational Physics, 230 (2011) 329–348.
[2] I. Borazjani, F. Sotiropoulos, E.D. Tytell, G.V. Lauder, Hydrodynamics of the bluegill sunfish C-start escape response: three-dimensional simulations and comparison with experimental data, Journal of Experimental Biology 215 (2012) 671–684.
[3] C. Brücker, H. Bleckmann, Vortex dynamics in the wake of a mechanical fish, Experiments in Fluids, 43 (2007) 799–810.
[4] M. Gazzola, P. Chatelain, W.M. van Rees, P. Koumoutsakos, Simulations of single and multiple swimmers with non-divergence free deforming geometries, Journal of Computational Physics, 230 (2011) 7093–7114.
[5] S. Hochstein, R. Blickhan, Vortex re-capturing and kinematics in human underwater undulatory swimming, Human Movement Science, 30 (2011) 998–1007.
[6] S. Kern, P. Koumoutsakos, Simulations of optimized anguilliform swimming, Journal of Experimental Biology 209 (2006) 4841–4857.
[7] A. von Loebbecke, R. Mittal, F. Fish, R. Mark, A comparison of the kinematics of the dolphin kick in humans and cetaceans, Human Movement Science, 28 (2009) 99–112.
[8] A. von Loebbecke, R. Mittal, R. Mark, J. Hahn, A computational method for analysis of underwater dolphin kick hydrodynamics in human swimming, Sports Biomechanics, 8 (2009) 60–77 .
[9] R. Mittal, H. Dong, M. Bozkurttas, A. von Loebbecke, Analysis of Flying and Swimming in Nature Using an Immersed Boundary Method, 36th AIAA Fluid Dynamics Conference and Exhibit, San Francisco, USA, (2006) 1–8.
[10] S. Pacholak, S. Hochstein, A. Rudert, C. Brücker, Unsteady flow phenomena in human undulatory swimming: a numerical approach, Sports Biomechanics, 13 (2014) 176–194.
[11] A. Przybilla, S. Kunze, A. Rudert, H. Bleckmann, C. Brücker, Entraining in trout: a behavioural and hydrodynamic analysis, The Journal of Experimental Biologie, 213 (2010) 2976–2986.
[12] S. Vaghefi, M. Abbaspour, Experimental Hydrodynamics Imaging of Trout in Steady Swimming, International Conference on Environment Science and Engineering, 32 (2012) 135–139.
[13] K.S. Yeo, S.J. Ang, C.Shu, Simulation of fish swimming and manoeuvring by an SVD-GFD method on a hybrid meshfree-Cartesian grid, Computers & Fluids, 39 (2010) 403–430.