INTERNATIONAL DESIGN CONFERENCE - DESIGN 2004
Dubrovnik, May 18 - 20, 2004.

DESIGN 2004

SHAPE MODIFICATION OF SCULPTED GEOMETRIC MODELS OF ARBITRARY TOPOLOGY

B. N. Veelo

Keywords: computer aided design, curve interpolation, transfinite surface patches, arbitrary topology, free-form surfaces, deformation, modification, manipulation

1. Introduction

In the conceptual design of a sculpted object, sketched curves often represent the geometric features of the design. When a computer model of the design is needed, it is an advantage if the geometric modeller allows the designer to continue to think in terms of these feature curves. This has been accomplished by a modelling methodology in which the concepts of wire-frame modelling and solid modelling are merged, and consequently is denoted by the hybrid representation (H-rep) concept [Koelman 1999]. The H-rep defines smooth sculpted models by means of an arbitrary network[1] of intersecting curves, which are interpolated by transfinite patches with an arbitrary number of sides.

In Koelman´s implementation (see also [Koelman et al 2001] and [Koelman 2003]) the designer starts with an initial model, defined by a minimal set of curves. The shape of the surface patches is completely derived from the shape of the curves. In other words, the only means to control the shape of the model is through manipulation of curves. Thus, the first step in the design process is to modify the existing curves to their correct shape, by traditional control point manipulation. When the surface is visualised at this stage, the designer will likely not be satisfied with the shape of the surface patches. They are still large and the defining curves are too far apart to describe every detail that is envisioned for the design. The solution is to add more curves to the network, effectively splitting up patches into smaller ones. New curves can be generated automatically by intersecting the model with a user-defined plane, or by projecting a separate curve onto the surface. After addition of a new curve, the shape of newly split patches can be modified by manipulating the curve. By this process, the shape of a sculpted model evolves from a course definition to a detailed definition, until the designer is satisfied with the result.

There is a downside to this modelling methodology. As the design progresses and more curves are added to the model, more of its shape is rigidly defined. The more curves present, the smaller the surface patches, and the more local shape manipulations become. The presence of more curves also means that the number of intersections that a curve has with other curves, increases. As a result, during curve manipulation, there is a higher risk that the curve is pulled away from these intersections, causing serious defects in the surface. Currently, no mechanism is implemented that prevents this, or that resolves the incompatibilities. Although it is possible to restore the intersections manually, by adjusting all affected curves to the changes, this is a lengthy, iterative task.

Consequently, making design changes that affect larger surface areas of the model, is discouraged at late design stages. One could say that designing sculpted shapes this way, in practice is a one-dimensional process because the design has a preference to evolve only in one direction. This paper proposes a simple and efficient method for shape manipulation of a dense curve network that is not strictly local and does not destroy the consistency and the geometric continuity of the network. This makes it possible to migrate directly from one shape variation to the other, by which the design process becomes, in our way of speaking, multi-dimensional.

2. Background

In the computer-aided design (CAD) of free-form or sculpted shapes, so-called Non-Uniform Rational B-Spline (NURBS) surfaces enjoy great popularity. NURBS surface patches derive their geometry from control points that they approximate. Whether their popularity is justified, is debatable. The challenge of designing sculpted shapes boils down to two main problems. These are the problem of geometric continuity and the problem of control.

A single NURBS surface patch without degenerate sides[2] fits only well to deformations of the square, the cylinder and the torus—of which only the torus can describe the boundary of a solid. All other geometries, thus including most subjects of design, are denoted as having arbitrary topology. For a composition of (non-degenerate) patches to describe shapes of arbitrary topology, the patches must be allowed to be organised in an arbitrary manner, where the number of patches that mutually connect with one of their corners is not always four. Maintaining geometric continuity, i.e., a smooth composite surface, is especially difficult at these irregular points.

The problem of control is implied by the fact that NURBS surface patches approximate a regular grid of control points. The problem becomes eminent, e.g., when more control points are needed locally, in order to define some local detail in a surface patch. Since extra control points can only be added in complete rows or columns, they also appear in regions where they are not wanted, because they make achieving surface fairness[3] more difficult.

Transfinite surface patches, which derive their geometry from curves that they interpolate, do not suffer from a control problem, as the patch does not care how its bounding curves are defined. For adjacent transfinite patches to connect with tangent plane continuity, tangent information is required along the curves, which is represented by so-called tangent ribbons. In order to model arbitrary topology, the curve network must also be arbitrary, i.e., without regularity requirements. Jensen et al [1991] were the first to develop a technique for the generation of tangent ribbons on such networks by using a boundary representation (B-rep)[4], which is a data structure used primarily in solid modelling. Their field of application was automotive styling. Van Dijk [1994] took this to conceptual industrial design and Michelsen [1995] to naval architecture.

Although having developed an alternative to NURBS surface modelling without the associated problems, it remained a challenge to keep the curve network simultaneously smooth and consistent as a surface representation. By integrating a curve fitting and fairing algorithm, Koelman [1999] was able to improve that situation, and produced an implementation for the design of (the exterior of) ship hulls, at production quality. In addition, he removed the need for the user to worry about surface patches, by using the B-rep to its full potential.

In Figure 1, the hybrid nature of the H-rep is clearly visible. The nodes in the B-rep refer to intersection points in the wire-frame for their geometry. The edges refer to the curve sections in-between the intersection points and the faces refer to the *n*-sided patches that can be generated to fill the openings in the wire-frame. Tangent ribbons are also partly indicated.

3. Related Research

The problem of the global shape of a model getting fixated by the definition of details is not specific to the H-rep and its precedents. It also exists in systems that are based on approximation of control points, although the consequences are not as dramatic as the surface defects that can arise in an H-rep. If a surface region is to be modified for which a larger number of control points need to be shifted, this must be done in a way that preserves the coherence between the control points, so that both the global shape remains fair and the detailed surface features are not damaged. This has been addressed by the integration of physics based properties [Terzopoulos and Qin, 1994], [Léon and Trompette, 1995] and hierarchical refinement [Forsey and Bartels, 1988]. These references do not explicitly consider

arbitrary topology however. The hierarchical refinement principle has also been proposed for the so-called surface splines [Gonzales-Ochoa and Peters, 1999], which approximate an arbitrary mesh of control points. Contrary to plain NURBS surfaces, the result may be a viable alternative to the H-rep.
Free-form deformation (FFD) [Sederberg and Parry, 1986] is a technique to reshape a geometric model indirectly by warping the space in which it is defined. FFD is independent of the model definition, and thus competes with the method presented here.

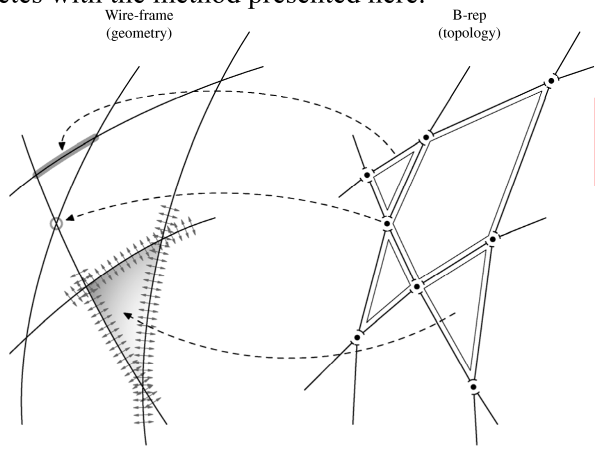

Figure 1. The hybrid representation with references between topology and geometry

4. Manipulation of Sets of Data Points

We will state our problem as follows. "Given a certain region on a surface that interpolates a network of curves, manipulate all curves in that region simultaneously, in a way that does not destroy the consistency of the network and does not introduce unwanted geometric discontinuities".
We note that the details behind the process of adding a new curve consist of tracing a string of data points over the surface, as a sampling of the intersection curve or the projected curve. Then the fitting/fairing algorithm is invoked to generate a curve through these points, which is added to the model. During manipulation of curves (and thus the surface) these data points can be made to move with the curve, so that they indeed remain positioned on the surface. If we assume for the moment that all data points are persistent, meaning that they remain in existence throughout the modelling process, then the complete model can be regenerated from the data points and the B-rep alone[5], with the help of the fitting/fairing algorithm. Thus, we can reformulate the problem as, "Given a point set belonging to a consistent H-rep, shift a selection of points to a new position, so that the distance and the direction of the shift of each individual point varies smoothly over the set".
We will now assert our assumption. Data points that are associated with intersections between curves are persistent, because they represent the geometry of node elements in the B-rep. Currently, other data points are not persistent, as they serve no purpose after the creation of a curve. Nevertheless, in a dense curve network, there will likely be enough intersections (and thus persistent data points) to record the shape of the curves. A simple heuristic can verify this, e.g., by checking whether the number and distribution of data points belonging to a curve stands in proportion to the number and distribution of control points of the curve. If the verification fails, extra data points can be inserted at low computational cost.

4.1 Shift Vectors

Let us declare $\mathbf{s}_i$ to be the *shift vector* for a data point i, i.e., the difference between the position of that point after and before the shape modification. We will define this shift vector as the vector sum of the sample of one or more three-dimensional vector fields.
A vector field is primarily defined by a *selection field* j of varying intensity, which is concentrated around a point, a curve or a surface, which we will call the *base* of the selection field. This base may be part of the model, or be dedicated to support the selection field. The intensity f_j of the field will be

unity at its base, decreasing smoothly with increasing distance d to the base, and level off to zero at a distance r_j to the base, which we will call the *extent* of the selection field. If the base is singular, r_j is a constant; but if the selection field is based on a curve (or surface), r_j may be a function of the curve parameter (or surface parameters)[6]. In a similar fashion, we will define a vector on the base, whose length and direction may be a function of the base parameters. We will call this vector the *typical shift vector* of the selection field, denoted by $\mathbf{S}_j$.

In addition to a selection field, one or more *deselection fields*, enumerated by k, may take part in the definition of a vector field. Deselection fields reverse the effect of the selection field. Their definition is similar to the definition of selection fields, except that they lack a typical shift vector and their intensity g_k is opposite to the intensity of selection fields: unity outside their extent, smoothly decreasing in proportion to the distance d to the base inside their extent, and levelling off to zero at their base.

The vector field is then defined as the typical shift vector, evaluated on the closest point on the base, multiplied by the selection field intensity and the deselection field intensities. Especially for deselection fields it is interesting to have them act differently on the x, y and z coordinates of the vectors in the field, and thus we will redefine field intensities as diagonal matrix functions:

$$\mathbf{f}_j(d) \equiv \begin{bmatrix} f_{j,x}(d/r_j) & 0 & 0 \\ 0 & f_{j,y}(d/r_j) & 0 \\ 0 & 0 & f_{j,z}(d/r_j) \end{bmatrix} \text{ and } \mathbf{g}_k(d) \equiv \begin{bmatrix} g_{k,x}(d/r_k) & 0 & 0 \\ 0 & g_{k,y}(d/r_k) & 0 \\ 0 & 0 & g_{k,z}(d/r_k) \end{bmatrix}, \quad (1)$$

in which we have normalized the support of the intensity functions with respect to the extent of the respective field. If we then say $g_{k,y}(d/r_k) \equiv 0$ for a deselection field based on the plane $y = 0$, we accomplish that data points in that plane will only shift in that plane and not away from it, regardless of the direction of the typical shift vector. This is advantageous if the design is symmetrical around $y = 0$ and only one half of it is being modelled.

The definition of the shift vector can now be formalised as

$$\mathbf{s}_i \equiv \sum_j \left(\prod_k \left(\mathbf{g}_k(d_{i,k})\right) \mathbf{f}_j(d_{i,j}) \mathbf{S}_j \right). \quad (2)$$

Here $d_{i,j}$ denotes the shortest distance through space from the data point i to the closest point on the base of selection field j, and $\mathbf{S}_j$ and $\mathbf{f}_j$ are evaluated at that position on the base. Analogously, $d_{i,k}$ denotes the shortest distance through space from the data point i to the closest point on the base of deselection field k, and g_k is evaluated at that position on the base. What remains is to find suitable definitions for the selection functions f and g, and for the typical shift vector $\mathbf{S}$.

4.2 Selection Functions

Any function that behaves as described will give useful results. For more control of the shape of the resulting modification, one may want to vary the shape of the selection function over the base of the selection, as a function of the base parameters. This is possible in the following definition of a cubic piecewise polynomial, in which a parameter $\kappa \in [0,1]$ defines how fast the function falls off.

$$f\left(\frac{d_{i,j}}{r_j}\right) \equiv f(u) \equiv \begin{cases} \dfrac{\kappa^2 + \kappa u^2 (u-3) + u^3}{\kappa^2} & \text{if} \quad 0 \le u < \kappa \\ \dfrac{(u-1)^3}{\kappa - 1} & \text{if} \quad \kappa \le u < 1 \\ 0 & \text{if} \quad 1 \le u. \end{cases} \quad (3)$$

in which u has been substituted for $d_{i,j}/r_j$ for simplicity. This function, plotted in Figure 2, is derived from the Cox-deBoor recursive definition of B-spline basis functions. The selection function of deselection fields can simply be defined as $g \equiv 1 - f$.

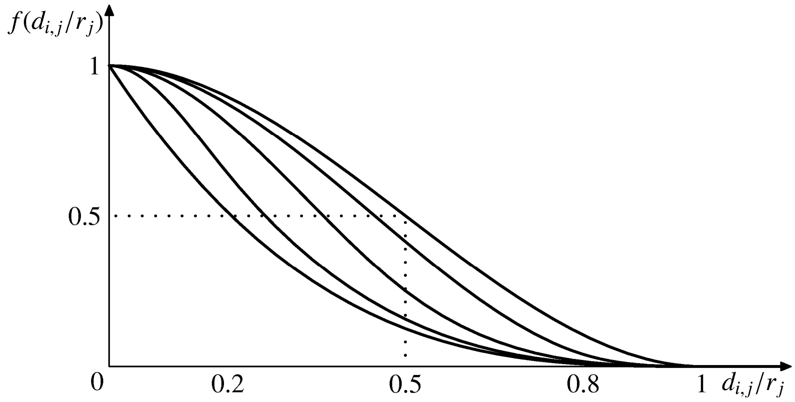

Figure 2. The selection function defined by equation (3). Smaller values of κ make the function fall off faster. Plotted are κ=1.0, which is point-symmetric about (0.5,0.5), κ=0.8, κ=0.5, κ=0.2 and κ=0.0

4.3 Typical Shift Vector

A selection field with a singular base can very well be based on a data point on the surface. It will be natural to take the surface normal at that point as the typical shift vector, scaled up or down if necessary.

For selection fields that are based on a curve, a powerful modelling tool results if the typical shift vector can be varied along the curve. Put simply, the shift vector can be defined as the difference between the base curve, say $\mathbf{c}(t)$, and an other curve, say $\hat{\mathbf{c}}(\hat{t})$. If $\mathbf{c}(t)$ is a curve on the surface prior to the shape modification, then $\hat{\mathbf{c}}(\hat{t})$ is exactly what the model will look like at this location, after the modification. Thus, designers will be able to manipulate feature curves, or even completely redesign them, while they will be able to control how the other curves (and thus the surface) in their vicinity adapts to the changes with the parameters r and κ. In addition, they will be able to protect other feature curves during the modification, by basing a deselection field on them.

To make this principle work as expected, it needs to be somewhat refined. As $\hat{\mathbf{c}}(\hat{t})$ may be completely different from $\mathbf{c}(t)$, their parameterisation may not be similar. In other words, when two particles are considered, one travelling down each curve at proportional increments of $\hat{t}$ and t, the variation in velocity of the two particles may not be parallel. The effect on the typical shift vector will be that it changes direction more often than necessary. We will remedy this by evaluating the curves with respect to arc length. In addition, if $\mathbf{c}(t)$ is a feature curve, the designer may not want to vary the complete curve, and $\hat{\mathbf{c}}(\hat{t})$ may partly coincide with $\mathbf{c}(t)$. But due to the different lengths of $\mathbf{c}(t)$ and $\hat{\mathbf{c}}(\hat{t})$, the typical shift vector may still have non-zero length in these parts, which is not intended. To counter this, we must evaluate the curves only over the curve sections that actually have different geometries.

Say that the curves $\mathbf{c}(t)$ and $\hat{\mathbf{c}}(\hat{t})$ differ from each other for $t \in [t_a, t_b]$ and $\hat{t} \in [\hat{t}_c, \hat{t}_d]$, with $t_{begin} \le t_a < t_b \le t_{end}$ and $\hat{t}_{begin} \le \hat{t}_c < \hat{t}_d \le \hat{t}_{end}$. For other parameter values, the curves coincide, although not necessarily for equal parameter values. The exact value of t_a, t_b, $\hat{t}_c$ and $\hat{t}_d$ can be determined by analysis of the control points and knot vectors of the curves. For a formal definition of the shift vector, we need a mapping $m : t \mapsto \hat{t}$. For a certain parameter value t_i, $m(t_i)$ must produce a $\hat{t}_j$ so that the arc lengths of the curve sections on either side of these parameter values are proportional:

$$\frac{\int_{t_a}^{t_i} |\dot{\mathbf{c}}(t)| \mathrm{d}t}{\int_{t_i}^{t_b} |\dot{\mathbf{c}}(t)| \mathrm{d}t} = \frac{\int_{\hat{t}_c}^{\hat{t}_j} \left|\dot{\hat{\mathbf{c}}}(\hat{t})\right| \mathrm{d}\hat{t}}{\int_{\hat{t}_j}^{\hat{t}_d} \left|\dot{\hat{\mathbf{c}}}(\hat{t})\right| \mathrm{d}\hat{t}} \tag{4}$$

in which the arc length of a curve is defined as the integral of the length of the first derivative of that curve with respect to the curve parameter. Because m is inefficient to be evaluated directly, one should first assess whether arc length evaluation is at all worthwhile, by comparing internal knot spacings and control point distances of $\mathbf{c}(t)$ and $\hat{\mathbf{c}}(\hat{t})$, looking for large discrepancies. If so, the map m may be approximated by evaluating $m(t)$ at distinct values of t_i, and fitting a polynomial, say $\hat{m}(t)$, through the mapped values $\hat{t}_j$.

Now we are able to define the typical shift vector **S** as the difference between relating positions on the two curves according to arc length, expressed as a function of the curve parameter t:

$$\mathbf{S}(t) \equiv \hat{\mathbf{c}}(\hat{m}(t)) - \mathbf{c}(t) . \tag{5}$$

4.4 Unintended Selections

The proposed method for shape modification is obviously simple, as we are not regarding the surface of the model at all, and only consider data points and their shortest distance to selection bases. The advantage is speed. Shift vectors can be computed quickly enough to visualise them in real time, while the designer manipulates the modification parameters. They give a sufficient indication of how the shape will be modified once the parameters are accepted. Therefore, the presented method for shape modification is interactive to a great extent.

However, in some situations this approach can be too simple. For instance, when modifying an area on the upper side of a thin wing. Because the data points on the under side are close to the upper side, they may be selected unintentionally. Even though this may be prevented by careful definition of deselection fields, there is an alternative that can be automated, which involves putting the B-rep to good use.

Let us define the *root node* of a selection field as the node that references the data point that is closest to the base of the selection field. If there are several nodes that qualify, any one of these will do. The algorithm listed in Table 1 will only shift data points that form a contiguous selection that is rooted at the base, and prune away isolated sets of selected data points that are separated from the main selection by more than one surface patch.

5. Finishing Up

Once the fitting/fairing algorithm has re-interpolated the curves over the shifted data points, the H-rep has become a consistent and smooth modification from the original, by which we have succeeded in our objective. But if the original primarily consisted of plane curves, such as is customary in the design of ship hulls, these may no longer be planar after the modification.

Table 1. Pseudo-code of an algorithm that only shifts contiguous sets of data points

```
Let A be an empty set, capable of containing node references
Mark all nodes and faces in the B-rep as unconsidered
Mark the root node as considered and add it to A
While A is not empty {
        Compute s_i for a node i ∈ A
        If |s_i| > 0, then {
                For all faces j that are adjacent to node i and that are still unconsidered {
                        Mark j as considered
                        For all nodes k that are adjacent to j and that are still unconsidered {
                                Mark k as considered and add it to A
                        }
                }
                shift node i
        }
        remove i from A
}
```

Plane curves can be restored by intersecting the modified model with the planes in which the curves were originally defined, and adding the intersection curves to the model. These new curves take over the definition of the modified shape, by which the old curves become redundant and may be removed.

6. Application Example

Figure 3 shows how the foreship of a frigate has been made slightly narrower. Note that the system has no difficulties with the knuckle line that is present in this region.

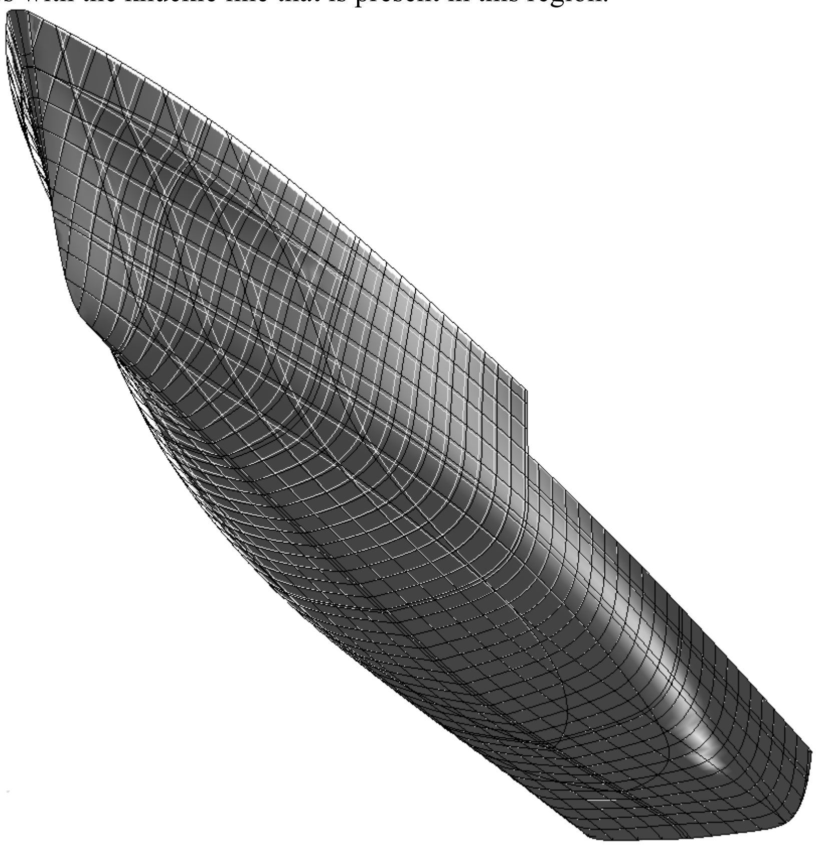

Figure 3. Modification of the hull of a frigate (shaded surface and light grey wire-frame) together with the original shape (black wire-frame overlay)

7. Conclusions

A simple method for the modification of a network of intersecting curves was presented, which preserves the consistency of the network, the fairness of the surface and local surface features. It has been found that feature curves may be redesigned explicitly, regardless of the detail in a design. This can be regarded as an advantage over the competing method of free-form deformation.

Notes

[1]The network is arbitrary except for the requirement that curves start and stop at another curve.

[2]A degenerate side is a side that is collapsed to a single point. The resulting singularity makes maintaining geometric continuity difficult.

[3]We differentiate between smoothness and fairness. Smoothness is about geometric continuity, a mathematical quality. A curve is smooth if it has tangential continuity, and even smoother if it also has curvature continuity. Fairness is an aesthetic quality. A curve is fair if it has no unwanted variations in curvature, including inflections, but it may have intentional geometric discontinuities, such as a cusp or the connection between a circular arc and a straight line.

[4]A B-rep data structure describes the boundary of a solid, consisting of the three types of topological elements of "node", "edge" and "face". References exist in the data structure such that for each element, its neighbouring elements can be determined. The orientation of the faces, i.e., which side is facing inward or outward, is implicitly defined by the ordering of references.

[5]Actually, this is a slight simplification, as Koelmans implementation supports curves to be composed of shorter curve sections. The end-conditions of a curve section may be dictated by the adjacent section in a master/slave fashion. This information can easily be conserved throughout a surface modification.

[6]In order to guarantee that vectors in the field vary smoothly, we must require that the radius of curvature of the selection base is at least as large as its extent, everywhere. Neither may two distinct parts of the same base come closer than the sum of the extents at those parts. I.e., selection fields must not self-intersect.

References

van Dijk, C. G., "Interactive modeling of transfinite surfaces with sketched design curves", Ph.D. thesis, Delft University of Technology, the Netherlands, 1994.

Forsey, D. R. and Bartels, R. H., "Hierarchical B-spline refinement", Proceedings of the 15th annual conference on Computer graphics and interactive techniques, ACM Press, 1988, pp. 205–212.

Gonzalez-Ochoa, C. and Peters, J., "Localized-hierarchy surface splines (LeSS)", Proceedings of the 1999 symposium on Interactive 3D graphics, ACM Press, 1999, pp. 7–15. http://doi.acm.org/10.1145/300523.300524.

Jensen, T. W., Petersen, C. S. and Watkins, M. A., "Practical curves and surfaces for a geometric modeler", Computer Aided Geometric Design, vol. 8, 1991, pp. 357–369.

Koelman, H., Horv´ath, I. and Aalbers, A., "Hybrid representation of the shape of ship hulls", International Shipbuilding Progress, vol. 48, no. 3, 2001, pp. 247–269.

Koelman, H. J., "Computer Support for Design, Engineering and Prototyping of the Shape of Ship Hulls", Ph.D. thesis, Delft University of Technology, the Netherlands, 1999. Published by SARC BV, sarc@sarc.nl.

Koelman, H. J., "Exploring the H-rep Ship Hull Modelling Concept", Bertram, V., editor, Proceedings of the 2nd International Conference on Computer Applications and Information Technology in the Maritime Industries (COMPIT'03), Hamburg, Germany, 2003, pp. 140–153. Also available at http://www.sarc.nl/compit03.pdf.

L´eon, J. and Trompette, P., "A new approach towards free-form surfaces control", Computer Aided Geometric Design, vol. 12, no. 4, 1995, pp. 395–416.

Michelsen, J., "A Free-Form Geometric Modelling Approach with Ship Design Applications", Ph.D. thesis, Techn. Univ. of Denmark, Dept. of Naval Architecture and Ocean Engineering, Lyngby, 1995.

Sederberg, T. and Parry, S., "Free-form deformation of solid geometric models", ACM SIGGRAPH '86 Proceedings, vol. 20, no. 4, 1986, pp. 151–160.

Terzopoulos, D. and Qin, H., "Dynamic NURBS with geometric constraints for interactive sculpting", ACM Transactions on Graphics, vol. 13, no. 2, 1994, pp. 103–136. URL http://citeseer.nj.nec.com/terzopoulos94dynamic.html and http://mrl.nyu.edu/˜dt/papers/tog94/tog94.pdf .

B.N. Veelo, MSc BEng
NTNU — Norwegian University of Science and Technology, Department of Engineering Design and Materials
N-7491 Trondheim, Norway
Telephone: +47 73930757, Telefax: +47 73930756
E-mail: Bastiaan.N.Veelo@ntnu.no