Temporal Computational Objects: A Process for Dynamic Surface Generation

Kurt W. Swanson[1], Kenneth A. Brakke[2] and David E. Breen[3]

[1]Dept. of Mechanical Engineering, Drexel University, Philadelphia PA 19104 USA
[2]Dept. of Mathematics, Susquehanna University, Selinsgrove, PA 17870 USA
[3]Dept. of Computer Science, Drexel University, Philadelphia PA 19104 USA

Abstract. Physics-based modeling techniques have been shown to be advanced methods for creating high quality surfaces for application in CAGD. In this paper we propose a physics-based Temporal Computational Object (TCO) as a modeling process that simulates a liquid membrane to generate organic free-form models. This process compliments existing solid and elastic physics based modeling processes by simulating a common liquid phase of matter. The simulation consists of a mesh that undergoes motion in response to surface tension and other energies, resulting in a smooth, minimum energy surface. The surface is produced via a dynamic process that is directed and influenced by user-defined scalar and vector fields, boundary conditions and a variety of internal parameters and topologies. These surfaces include multiple free intersecting surfaces, volume-preserving surfaces and periodic surfaces. We provide a wide range of examples that show high quality physics-based surfaces.

1 Introduction

We describe a class of deformable surfaces that behave as a quasi-static liquid, referred to as Temporal Computational Objects (TCOs). A surface tension energy minimization process specific to liquid membranes drives the surface deformation. The user controls the deformation by subjecting the surface to a wide variety of forces and constraints. The focus of our work is to provide novel forms that are useful as artifacts in CAGD.

Variational surfaces, free-form design, surface smoothing, etc. have been important topics in CAGD. Roughly speaking, for a physical model the solution shape is characterized by a balance of internal and external forces. The use of energy minimization and optimization methods has been effective for finding this balance in CAGD applications. In a physics-based variational approach an elastic surface or curve is calculated that minimizes the energy of an elastic membrane while satisfying boundary conditions. Typically Willmore flow is used in geometric modeling of compact surfaces with fixed boundaries to minimize total curvatures, $\int_s \left(\kappa_1^2 + \kappa_2^2\right) dA$ where K_1 and K_2 are primary curvatures. This is a standard functional in variationally optimal surface modeling [1]. Models that employ

Willmore flow have internal forces that are a function of the Willmore gradient. The use of a single form of energy such as surface tension is significantly limiting in the type and range of useful surfaces. We have considered, and abandoned, an elastic membrane method of surface generation as being overly constraining. Instead, we describe the functional operations of liquid surface motion. This paper presents the first use of a liquid surface physics-based method for dynamic shape modeling.

The proposed TCO method provides a greater diversity of energies, topologies and boundary conditions. Liquid surfaces have been extensively studied in the past; including the works of Plateau, Scherk, Enneper, Costa, and Weaire-Phelan. For a TCO, the surface has an intrinsic contractile property that is influenced by an extrinsic environment that includes forces acting at a distance, such as gravity and other scalar and vector fields. The fields may act alone, they may increase and decrease, and there may be multiple sources of fields on a surface. TCO objects are assembled from multiple surfaces of dissimilar mean curvature produced by different boundaries and extrinsic environments. We employ use of a CAD environment to create boundary wires for unique surface patches. The TCO is dynamically generated and then reinserted into the CAD environment allowing TCOs to be used together with other CAD geometry. This paring of capabilities provides a robust process that has produced functional parts in an industrial setting. TCOs provide utility by enhancing the modeling capabilities of conventional CAD software systems through a process that creates and manipulates predominantly curved, minimum energy forms.

Fig. 1. Depicts a progression of change in surface representation for a catenoid. This surface is also a simulation of a liquid film bounded by two circular rings.

A TCO is constructed with a tessellated simplicial complex, union of triangles (called facets in this paper). Attributes may be ascribed to elements such as tension in such a manner as to change their interaction with the extrinsic environment. Volume, calculated by integration over a surface, may be specified as an attribute for a surface or series of facets. Highly complex, interconnected objects are created from very sparse and basic input data and constraints. During the deformation process the mesh is refined; essentially providing surface mobility. This very natural and physical property of a quasi-static liquid invariably generates a smooth surface. The surface will find an equilibrium state regardless of the initial conditions, provided the environment remains constant. The sample shown in Figure 1 shows a sparse initial boundary; an initial facet edge format; and finally a dense mesh. The figure demonstrates that the initial geometry is created in a standard CAD environment, moved into a TCO environment and then back to a CAD environment.

A TCO moves in response to its environment, it may be fixed or free (solid or liquid), the surface tension may vary locally and the enclosed volume may vary. All of these variables may change over time. During the evolution process the TCO shape will attempt to conform and adapt to its environment even as the environment changes. During the generation of the model, the observed transformation is one of growth and the state of the object is temporal. At some time during its evolution we may decide that the object is suitable for analysis and fabrication, and the entire shape is fixed. The general approach to creating a TCO involves a process of evolution by mean curvature, motivated by surface tension energy and other energies in the context of varifolds and geometric measure theory [2, 3].

2. Background

The TCO approach can be compared favorably to elastic membrane deformation methods that have been explored by Celniker and Gossard [4]. These approaches use an elastic deformation model that includes physical properties of bending and shear. Elastic deformation models have advantages over solid modeling in that many shapes can be created from character lines in what they term a 'ShapeWright Paradigm'. Within the paradigm character lines are skinned and then deformed with the objective function of the deformation being an energy reduction of the surface. The typical potential energy applied to the surface is pressure.

The original use of energy-based shape modeling is attributed to Schweikert [5] who introduced the method to Computer-Aided-Design with splines in tension. Nielson [6] elaborated on Schweikert's process of minimizing an energy function over the surface of an object. Subsequently, an energy-based process has been used to improve the ability to modify a surface by varying the parameters of the model [7, 8, 9, 10, 11, 12]. The common attribute of these methods is the use of a quasi-linear elastic model to deform the surface and/or the underlying curves. Welch and Witkin have expanded this basic work to facilitate user control of surfaces [13, 14, 15]. More recent developments include natural forces and time varying functions using D-NURBS [16, 17], and variational methods for subdivision surfaces [18, 19]. More recently minimal mean-curvature-variation flow [20] and discrete Willmore flows [1] have been introduced.

3 Surface Motion by Iteration

The TCO surface moves by vertex displacement. Each movement reduces energy while obeying any constraints. No changes in topology or triangulation are made during vertex motion iteration. (Other operations affect these attributes.) The method calculates the force at each vertex and moves the vertex in that direction, thus using a gradient descent method of minimization. The implementation of TCOs and the operations on them utilize the Surface Evolver software [2]. This capability was not created for physics modeling; instead the primary goal was the mathematical exploration of variational calculus of surfaces. The adaptation is unique in many ways.

We have expanded on this base system by including a CAD interface to create initial geometry. The sparse geometry consists of wire-frame loops that are translated to a TCO operating environment. We have enabled the generation and modification of innumerable geometries that are unavailable with the isolated systems – CAD or TCO. The introductions of multiple scalar and vector fields enable the modification of many classical surface forms such as Scherk's surface and catenoid surfaces. These modifications are re-introduced in a CAD environment with significant shape control utilizing a simulated liquid physics-based process.

3.1 Mesh Operation

Mesh operations are required to adjust the spacing and uniformity of the mesh for computational convenience. As a mesh expands or contracts, the mesh parameters such as aspect ratio become skewed and surface motion become difficult. Mesh smoothing such as Delaunay triangulation, vertex averaging, global mesh refinement, etc. serve to adjust the mesh to improve motion computation. The complete description of the process can be found in the reference material [2] if more information on basic mesh operations is desired.

3.2 TCO Data Structure

A TCO dataset has three fundamental parts: an Environment; a Sparse Initial Boundary Condition (SIBC); and an Evolution. These parts generally act together to refine, adjust and morph a surface or collection of surfaces. A basic and fundamental part of the TCO is the presumption that the surface is a liquid meniscus. In this capability, the surface has potential energy in the form of surface tension. Other extrinsic forces such as pressure, gravity, scalar and vector functions act at a distance to influence the surface form. There is a high degree of control of the surface since multiple fields can be specified. The intensity and location of the fields can be varied individually during the evolution. The initial topology of the surface and its fixed boundaries plays a dominant role. Used in balance, these three parts can achieve much more that either part alone.

The prescribed TCO method is a dynamicl physics-based process. A typical TCO will have fixed (solid) regions and liquid regions. The evolution governs the transformation between different property states and governs the application and intensity of forces that constitute the environment. In this manner a TCO achieves high levels of control. Below is listed a TCO data file that generates an ellipse-like shape. A simple graphic progression of a TCO shape is shown in Figure 2.

```
//ENVIRONMENT
STRING   // 1-dimensional "surface"
space_dimension 2
LINEAR
PARAMETER xa1= .5
PARAMETER ya1= 0
```

```
  QUANTITY edge_sint ENERGY method edge_scalar_integral
global
  scalar_integrand:  1/((x  -  xa1)^2  +  (y  -  ya1)^2)  +
1/((x + xa1)^2 + (y + ya1)^2)
  constraint 1   /* the abscissa */
  formula: x2 = 0
  //SIBC
  Vertices; 1 1.0 0.0; 2 0.0 1.0; 3 -1.0 0.0; 4 0.0 -
1.0
  Edges; 1 1 2; 2 2 3; 3 3 4; 4 4 1
  Faces; 1 1; 2 2; 3 3;4 4
  Bodies; 1 1 2 3 4
  //EVOLUTION
  evole := {
    edge_sint.modulus := .185 ; r;
    r; r; g 20; r; g 100 ;
    edge_divide 0.1;  g 10 ; V 3 ; g 10 ; V 3 ;
    edge_divide 0.05;  g 10 ; V 3 ; g 10 ; V 3 ;}
```

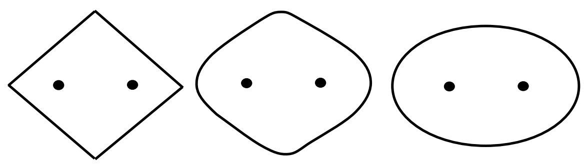

Fig. 2. A simple square SIBC morphs into an ellipse-like shape in this progression. Two point scalerand functions are depicted; any number of points can be used.

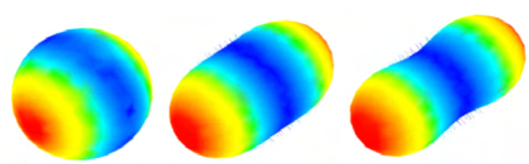

Fig. 3. This TCO begins as a sphere with two point type scalarands that are initially co-located and gradually move apart. The red areas are expanding regions and the blue regions are contracting

The TCO data file is loaded and evolved according to the commands listed in the 'Evolution' and results in the shapes depicted in Figure 2. A 1-D string model as well as a 2-D surface model can be generated. The origins of the scalar integrand (or scalarands) are depicted as points. The total field is a summation of the two point functions. This can be problematic for some computations, but it is not a serious difficulty. In many TCO's it is desirable to change the location of the origin and the intensity of the field during the evolution of the shape.

The TCO shown in Figure 3 shows the interaction of a surface and a changing environment. The two point scalarands gradually move apart and the surface responds to the movement by expanding and contracting to minimize the total energy of the surface. This interaction of intrinsic forces and an external scalar field can produce complex geometries such as branching tubes using a mobile point type scalar functional.

3.3 Transformation of Representation

It is essential to be able to move between software capabilities to create complex forms. The initial geometry depicted here was in all cases created in a CAD environment and then translated. The transformation allows the movement of data files between CAD based utilities in a *.iges format to a TCO *.fe format and to an *.stl format. Not having the transformation utilities would complicate the reproduction of the articles presented here, but it is not conceptually difficult. We do not describe these interfaces in detail here.

TCOs do not use a complex polynomial function to represent a surface; instead a TCO employs a simple union of triangles, or mesh. The internal TCO hierarchical structure is similar to an FEA data structure: point pairs form a line; a triplet of lines forms a facet; facets form surfaces; surfaces form volumes. There are additional attributes for elements such as surface tension. The formal listing of roles, attributes, values etc. is part of a more formal ontology definition, which is outside the scope of this paper.

The sequence of computation is depicted in Figure 4, where a coarse mesh is generated based on character lines created in a CAD environment. The fact that the mesh is coarse and the surface poorly defined is irrelevant. The initial surface is completely arbitrary except for a correct topological association of facets and the boundaries. The irregularities dissipate over time to form a uniform energy distribution over the entire mesh. Much of the prior literature, Celniker et al, has emphasized smooth continuous surfaces. With TCOs the challenge is to create surfaces that do not degenerate into a simple sphere or a smooth catenoid – additional energies have to be added.

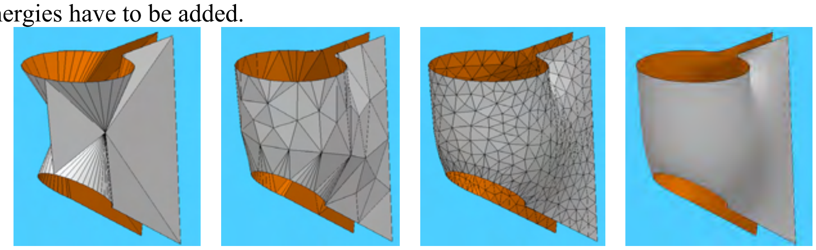

Fig. 4. The coarse mesh on the left is generated by a simple CAD wire frame. Over time the mesh is refined and additional energies are added to morph the object into the form on the right

There are two possible outcomes for the surface in Figure 4: one is to minimize the surface on the upper and lower loop and connect the two with a small vertical strip; however, here a scalarand provides the extra energy to keep the cylindrical nature of the part intact during generation. After generation the surface is transformed and merged along the boundaries of other sister surfaces in a CAD environment.

3.4 Scalar and Vector Fields

A TCO computes the trajectory of the surface vertices through space, iteratively, over time, with an objective function that minimizes energy. Scalar and vector functions are a class of functions within a TCO that are used to control the shape of the output surface and provide persistent stable computation. They expand the range of behaviors for a large system comprised of several million independent agents or cells. Individual facets can be regarded as cells of an automaton, each cell being interdependent with its neighboring cells and affected by an environment defined by functionals, pressure, gravity, etc. The effect is global the operations are local.

The application of ambient pressure is a relatively straightforward integration of a vector field over each facet and the summation of the forces to an adjoining vertex. A local pressure is more challenging. Applying a local distributed force to a TCO can be accomplished using an integrand or scalarand such that the total force decreases with the distance from the center of the force. This is a very natural function and examples are Newton's universal law of gravitation:

$$|F| = G\frac{m_1 \cdot m_2}{r^2} \quad . \tag{1}$$

Or Coulomb's law:

$$|F| = k_c \frac{q_1 \cdot q_2}{r^2} \quad . \tag{2}$$

Both of these expressions have singularities – they are undefined when the charges or masses are located at the same point and r^2 is zero.

$$f(x,y,z) = \frac{M}{(x-x_0)^2 + (y-y_0)^2 + (z-z_0)^2} \quad . \tag{3}$$

A typical point type scalarand is of the form shown above. The numerator is an intensity modulus *(M)* which determines the degree of local effect on the surface. Another important scalarand is a linear singularity, expressed as a direction vector. A linear singularity has an analog with electromagnetic force, where the field density is inversely proportional to a power of the distance to a line. Rather than explore this analog in great detail, the principal effort here is to write a function that will accomplish the same effect as a point scalarand with the difference being that the scalarand is linear. The scalarand is written as follows:

$$f(x,y,z) = M \Big/ \Big\{ \big((x-x_2)^2 + (y-y_2)^2 + (z-z_2)^2\big) - \big((x-x_2)*(x_1-x_2) + (y-y_2)*(y_1-y_2) + (z-z_2)*(z_1-z_2)\big)^2 \Big/ \big((x_1-x_2)^2(y_1-y_2)^2(z_1-z_2)^2\big) \Big\} \tag{4}$$

where (x_1,y_1,z_1) and (x_2,y_2,z_2) are two points on the line. Again, the modulus (M) is an intensity variable.

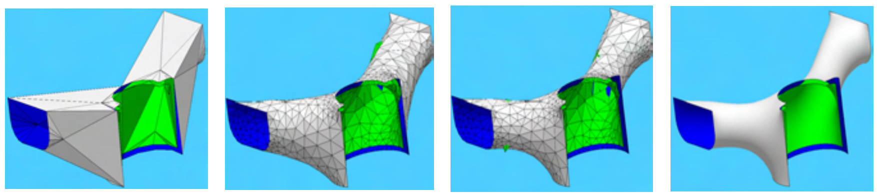

Fig. 5. In this depiction two surfaces are generated together, the green inside the blue. The interaction of the surface with the two linear scalarands is further modulated by increasing the surface tension of the green surface. In this way the surfaces will nest together

Fig. 6. Scherk's fundamental periodic surface has been deformed into a ring. In the center panel the bounding rings are coplanar, top and bottom, and in the right panel the lower edges have been thinned

The use of multiple scalarands provides a means of morphing fundamental surface form into more useful objects. Scherk's second surface was discovered in 1834 and is an example of a minimal surface. This surface can be created as a periodic surface. In Figure 6 a Scherk's periodic surface is deformed into a ring; as such, it is unstable. The holes will neck off and the surface degenerate. To stabilize the surface we introduce a point type scalarand to each of the holes and so maintain the basic form, though it no longer fundamentally a minimal surface. In the center panel the surface is deformed by moving the boundary rings. Finally on the right the lower portion of the ring is made thinner. The use of multiple scalarands provides persistent stable computation of the modified boundaries while preserving the basic topology.

4 TCO Object Assembly

The creation of objects in a TCO environment entails the separate generation of surfaces and assembling them into a single enclosure. It is standard CAD practice to join dissimilar surface forms, e.g. a cylinder and a plane used together form an enclosed volume – a rod. For a TCO there are no restrictions on the surface forms that can be used in combination, except those causing interpenetration. Each surface can be created as a surface patch and subsequently assembled. CAD serves as a basis environment for generation of character lines to ensure C1 continuity of boundaries. It is acceptable to have surfaces meet at right angles without smoothing if that is a functional requirement for the object. If a fillet is required then the character lines can be trimmed back and a fillet inserted. The joining process is referred to as 'equivalencing'; the goal is to join each facet edge with C1 continuity. Solid models may require C1, C2 and C3 continuity. Achieving this is not technically difficult, just time consuming.

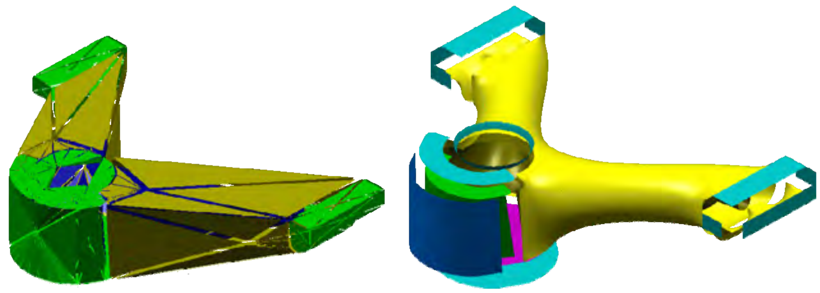

Fig. 7. Similar to the surface shown in Figure 5 this object is created from a series of dissimilar surface patches that are generated individually in a TCO environment

A TCO assembly is comprised of surface segments that are computed and then joined. The example shown in Figure 7 includes a set of dissimilar joined surfaces. On the left are the initial surfaces that match the character lines created in a CAD system. The right panel shows an exploded view of the same surfaces in their final minimum energy state. The multiple surfaces fit together neatly as they are located precisely in relation to each other to enable such an assembly.

The addition of fillets and other blending processes can be accomplished using a volume preserving functionality. The fillet that is shown in Figure 8 is created by trimming the character lines to make room for the surface. The surface is generated by a volume preserving functionality that results in a smooth transition between the adjoining surfaces. The adjoining surfaces are highly irregular, purposefully, but the transition surfaced accommodates this easily. Other edges of the TCO do not require this type of edge treatment and are not smoothed. A TCO assembly is comprised of surface segments that are computed and then assembled.

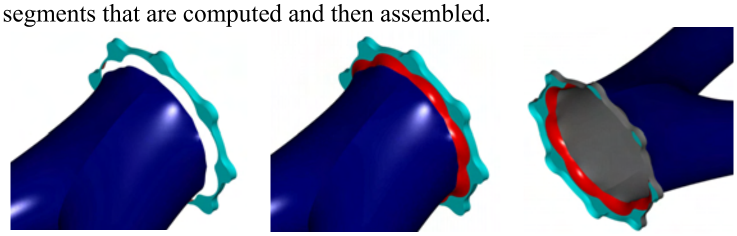

Fig. 8. The addition of a fillet is accomplished by trimming the edges of the character lines and inserting a TCO type surface generation.

5 Results and Conclusion

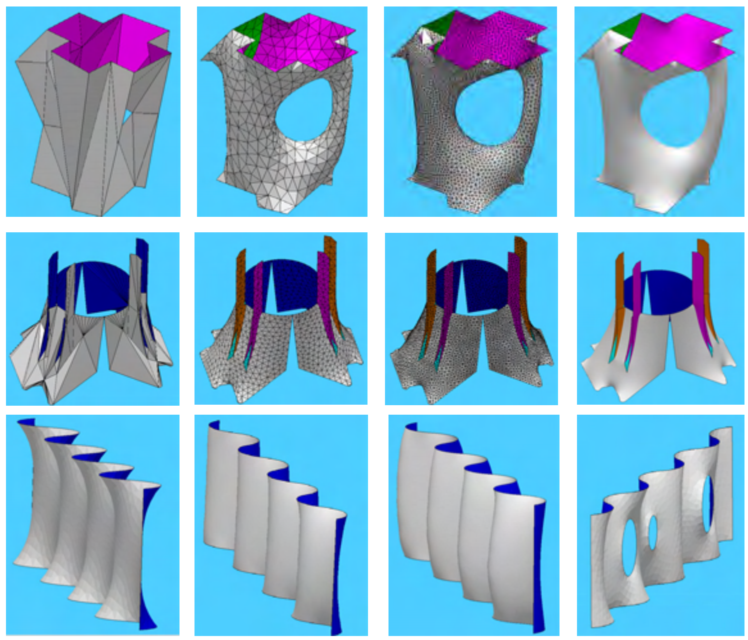

Fig. 9. The upper panel shows a surface with interpenetrations and irregular character lines. This surface is supported by multiple linear and point type scalarands. The middle row has multiple interacting surfaces. The lower series of pictures utilizes a sinusoidal scalarand. The intensity of the sinusoid is varied to deepen the sine wave. Also shown with multiple penetrations in the sine wave on the bottom right

Several TCO articles were generated and integrated into a CAD environment for analysis and review. The results are very favorable and TCOs provide a substantial improvement in creating objects with a preponderance of curved features. Seemingly complex features can be easily created with very sparse information. That information is acted on with a liquid surface physics-based model; extrinsic energies; and varying boundary conditions. The process is robust in creating objects considerably different from standard solid objects. Many of the common operations such as filleting and smoothing can be accomplished in a TCO. The use of multiple scalarands provides the ability to manipulate surfaces during evolution. Many surfaces, that would ordinarily be unstable, are made possible with the use of scalar functionals. The use of energy methods and a quasi-static fluid membrane provides some unique tools for geometry modeling. A large variety of forms were created with an almost limitless ability to deform the surfaces using extrinsic energies and intrinsic surface tension energies. These objects have exceptionally high quality surfaces and make excellent candidates for CAGD application.

References

[1] A. Bobenko, and P Schroder, Discrete Willmore Flow, Eurographics Symposium on Geometry Processing, 2005.
[2] Brakke, K., The motion of a surface by its mean curvature, Princeton University Press, Princeton NJ, 1977.
[3] Brakke, K, The Surface Evolver, Experimental Mathematics, 1, 1992, p. 141-165.
[4] Celniker, G, Gossard, D, Deformable curve and surface finite-elements for free-form shape design, (Proc. ACM SIGGRAPH) Computer Graphics, v.25 n.4, p.257-266, July 1991.
[5] Schweikert, D. G., An interpolation curve using a spline in tension, Journal of Math and Phys. No 45, p. 312-317, 1966.
[6] Nielson, G. M., Some piecewise polynomial alternatives to splines in tension, in Bamhill, RE and Riesenfeld, RF (eds) Computer Aided Geometric Design, Academic Press, 1974.
[7] Farin, G and Sapidis, N, Shape Representation of Sculpted Objects: the Fairness of Curves and Surfaces, Proceeding of Sea Grant Conference, MIT, October, 1988.
[8] Hagen, H., and Schulze, G., Automatic Smoothing with Geometric Surface Patches, Computer Aided Geometric Design Vol. 4, p. 231-236, 1987.
[9] Kjellander, J.A., Smoothing of bicubic parametric surfaces, Computer-Aided Design, Vol. 15, p. 288-293, 1983.
[10] Kallay, M and Ravani, B., Optimal twist vectors as a tool for interpolation a Network of curves with a minimal energy surface, CAGD, 1990.
[11] Nowack, H. and Reese, D., Design and fairing of ship surfaces, in Bamhill R.E. and Boehm, W. (eds), Surfaces in CAGD, N-Holland Amsterdam, p. 121-134, 1983.
[12] Sapidia, N. and Farin, G., Automatic fairing algorithm for B-spline curves, Computer-Aided Design, Vol. 22, No. 2 p. 121-129, March 1990.
[13] Welch, W. Witkin,A., Free-form shape design using triangulated surfaces, Proc. SIGGRAPH, p. 247-256, July 1994.
[14] Witkin, A.P., Heckbert P.S., Using particles to sample and control implicit surfaces, ACM SIGGRAPH 2005 Courses, 2005, Los Angeles, CA
[15] Welch, W. Witkin, A., Variational Surface Modeling, (Proc. ACM SIGGRAPH) Computer Graphics, 26, 2, p. 157-166, July 1992.
[16] Du, H; Qin, H; Free-Form Geometric Modeling by Integrating Parametric and Implicit PDEs Transactions on Visualization and Computer Graphics, Vol 13, Issue 3, May-June 2007 P. 549 - 561
[17] Qin, H.; Terzopoulos, D., D-NURBS: physics-based framework for geometric design, Visualization Computer Graphics, IEEE Trans,Vol 2,1,March 1996 P. 85 – 96
[18] Kobbelt, L and Schröder, P, A multiresolution framework for variational subdivision, ACM Transactions on Graphics, Vol. 17, No. 4, pp. 209-237, 1998.
[19] Kobbelt, L. Discrete fairing and variational subdivision for freeform surface design, The Visual Computer, Vol. 16, No. 3-4, pp. 142-158, 2000.
[20] G Xu and Q Zhang, G2 surface modeling using minimal mean-curvature-variation flow, Computer-Aided Design, Vol 39 p. 342-351, February 2007.