INFOLIO 36

RIVISTA DEL DOTTORATO DI RICERCA IN ARCHITETTURA, ARTI E PIANIFICAZIONE
DELL'UNIVERSITÀ DEGLI STUDI DI PALERMO - DIPARTIMENTO DI ARCHITETTURA

UTOPIA E DISTOPIA NEL PROGETTO DIGITALE

IN FOLIO_36 | ISSN 1828-2482

2. STATO DELLE RICERCHE

The evolution of the digital curve: from shipbuilding spline to the diffusion of NURBS, subdivision surfaces and t-splines as tools for architectural design

Stato delle ricerche

Giuseppe Gallo, Fulvio Wirz

In this article, we will follow the history of splines, digital methods that have characterized the architectural design process since the last decade of the twentieth century. We will describe their evolutions since the beginning as analogical tools for boat construction in the eighteenth century, to its passage to digital within the automotive industry. Then the relationship with the aerospace industry, and the evolution which from digital animation, have led nurbs, subdivision surfaces and t-splines to become everyday computational tools within architecture workflow. Recent developments on t-splines, following their acquisition by Autodesk, will lead us to a reflection on the relationship between software houses, which develop tools, and architects, who are now dependent on them. Therefore, we will consider the need to develop open-source methods, as Pixar did with Opensubdiv, as knowledge and responsibility shared solution.

Keywords: Digital Design, Digital Architecture, Architectural Design, Parametric Design

Introduction

The word "spline" evokes in our minds the curved-linearity of many digital designed architectures, yet, despite the curve has always been a form used in architecture, and has enormously fuelled the architectural debate since the 1990s onward. Architecture designed employing splines is often described as architecture of curvilinearity, a phenomenon that began between the late 1980s and early 1990s, when Ghery designed the *Peix d'Or* built in Barcelona in 1992, and he looked for a system to allow the description and production of the sinuous form. He found it in the splines (Carpo 2018), integrated within the Catia software, never used before in architectural design. The spline, thanks to the increasing capacity of the software, and the availability of computing power at an affordable cost, will allow designers to manipulate curved lines directly on the screen, thanks to graphic interfaces such as vectors and control points. The splines freed architects from limits, constraints and problems of the exact description of a free-form curve through analogue systems, giving them the technical possibilities to develop formal research such as distortion, calligraphy, landform and explosion, developed over a decade within the Zaha Hadid studio architects (Schumacher 2016).

As Carpo (2013) acknowledges, the same notations of the curves, based on the calculation, become practically irrelevant to the design purposes, but two mathematical aspects of the digital environment dominated by the splines have had vast and lasting consequences on the design: continuity and variation.

The digital splines must, in fact, be continuous, for purely mathematical reasons, the curves can vary in shape within the limits chosen in the notation of a parametric function. Setting the limits for the variation of one or more parameters is crucial for the project, and is a choice that leads to the parametric definition, not of a unique geometry, but an entire family of curves, lines or surfaces. Where the architectural design process differs from previous currents is in the writings of a pupil of Eisenman, the American Greg Lynn (1993), who on the occasion of the volume Folding in Architecture expresses the concept of curvilinearity, taking it from purely architectural and theoretical considerations. Curvilinearity tools include the morphing effects used in the film industry, which, for the author, can play a role in the design practice. Lynn cites as an example the film Terminator 2, where an actor is fluidized into a substance that recalls the quality of liquid mercury, a transformation that allows a character to gain new possibilities of movement and greater ease in overcoming obstacles. This is technically possible thanks to digital tools, splines and subdivision surface, which make it possible to build intermediate forms between two stages, generating families of gradual transformations. The same deformation is also possible in architecture, where thanks to topological flexibility, we can generate space which is continuous and at the same time endowed with different intensities. A process where the forms are not superfluous, but the result of a curvilinear logic that seeks to internalize cultural and contextual forces within architectural geometry: forces are immediately translated into the form, generating unique geometries and more than ideal characteristics (Lynn 2013). Lynn was the first to use animation software within the design process, experimenting with the use of morphing and subdivision surfaces, methods that many protagonists of digital architecture will widely adopt to generate what Lynn defines as BLOB, Binary Large Objects. Architectures ideally inspired by the famous blob of American b-movies: viscous entities, at the same time alien and defined by the context in which they move (Lynn 1996). Methodological and design evolutions that would not have been possible without the technical capacity brought into the project by the family of splines, and which have widely contributed to developing the designers' attention towards parametric methods, fundamental not only in the first but also, and above all, in the second digital turn (Carpo 2017).

Lofting and splines in the shipbuilding industry

The spline is not born as a method for the realization of architectures, its path is much older than the digital revolution, and was born within the naval industry. Curved shapes have a hydrodynamic utility for a boat builder, so that curvature homogeneity of a hull surface indicates a boat quality: a homogeneous surface meets less resistance to movement in the water.

Until the end of 1600, building a boat is, however, mainly an artisan-ship practice, so the most adopted method for boat production in series involves the construction and use of life-size wooden models, needed to produce the keel and the ribs that define the supporting structure of the hull (Towsend 2017).

With the 700's, craft develops in science, drawings replace life-size models, and to reproduce the curved-linearity of the shipbuilding project, naval engineers start the new practice of "lofting": from the English loft or attic since the attic above the workshop was the only clear and dry floor large enough to accommodate this procedure. This practice involves the use of "splines" (Fig. 1) thin strips of metal or wood, which are fixed on a flat surface and blocked by the use of hooked weights in three or more determined points. Thanks to material physical properties and uniform distribution of tensions, splines allow reproducing a homogeneous and functional curve to the production of keel and ribs (Nowacki 2006). The boatbuilding industry uses lofting method widely for over two centuries, meanwhile, the curved line becomes of interest to designers within other fields, from the aerospace industry to automotive industry, where, for the reason of performance, curved shape is now exploited in any artefact.

Bezier/De Casteljau curves and the automotive industry

Yet in the 50s of 1900, designing a car bodywork to be mass-produced is a complex problem. To design cars such as the Citroën DS, presented with great success in 1955 (Barthes 1957) at Paris Motor Show, the team of Citroën designers mainly uses physical scale models, from which characteristic curved shapes are surveyed by points, then reproduced in real size, interpolating points of each curve with the help of flexible splines. The entire process of generating the final model is expensive in terms of time and skills, moreover, an often qualitative interpretation of the geometry leaves significant space for inconsistency, discrepancy and human error. The company overcomes the manual reproduction process and the relative errors towards the end of the 50s, with the adoption of the first analogical computers, when, to dialogue with the first Numerical Control, NC, machines, it adopts a numerical geometric language which allows producing regular shapes like lines, circles and parabolas, but not freehand forms necessary for car design. In 1959 Citroën hired Paul de Faget de Casteljau, with the task of dealing with the problem numerically, he will define the first computational curve, creating an intuitive method that allows drawing a curve (Fig. 2) not through the definition of many points over all its length, but manipulating a limited number of points placed near the curve (Farin, Farin 2002). Citroën tests and finally adopts the young mathematician method. The company will keep a strict industrial secret on the discovery, so much that we have to wait for 1966 for the first publication about the computational curve, on behalf of Paul Bézier, a Renault company engineer. Bézier had independently discovered a method as effective as that of de Casteljau, to whom the antecedence of the discovery was recognized only a few years later. Although they are impeccable geometric constructs, Bézier curves and surfaces have limited applicability, they do not allow, for example,

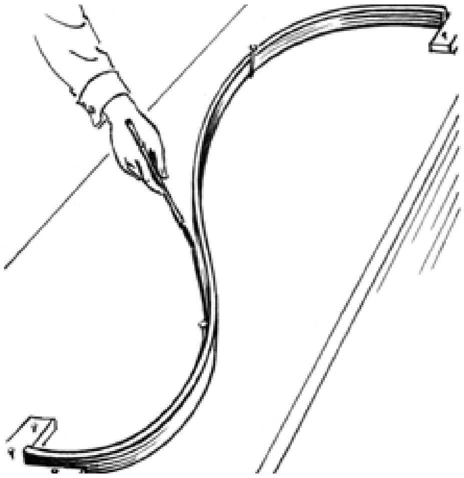

Fig. 1. A wooden spline (Pearson Scott Foresman. https://commons.wikimedia.org/wiki/File:Spline_(PSF).png, 02/01/2019).

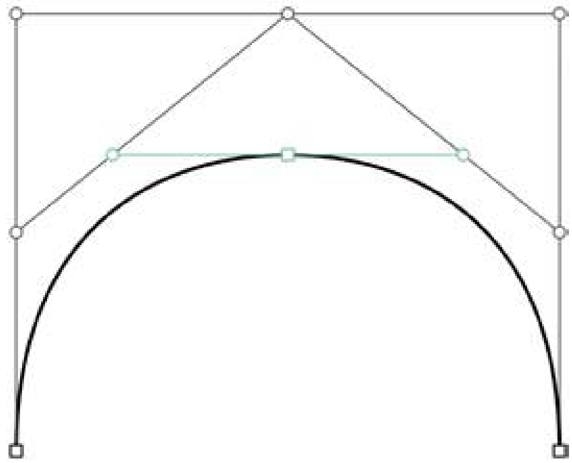

Fig. 2. A curve obtained with the De-Casteljau method. (Drawing by Giuseppe Gallo)

an accurate representation of conic sections, such as a simple circular arc. Another problem is that their algebraic degree increases along with the number of reference points, which greatly complicates numerical calculations (Ushakov 2011).

From b-splines to NURBS, General Motors and Boeing contributions

During the same period in which the French automotive industry is abuzz with the evolution of computational methods, US industries face the same needs differently. In 1978, Carl de Boor, a researcher at General Motors, invented a new recursive algorithm to evaluate a family of polynomial functions devised by the mathematician Isaac Jacob Schoenberg in 1946, the “basis spline” (Fig. 3) also known as b-spline (de Boor 1978). Many algorithms have a recursive structure: to solve a certain problem, they call themselves to handle sub-problems similar to the one at the beginning, namely: the problem is subdivided into sub-problems similar to the original one, but of a smaller size, the sub-problems are recursively solved, if the size of the sub-problems is sufficiently small, they are instead solved directly. The algorithm combines all solutions of the sub-problems to obtain the solution to the initial problem (Dijkstra 1960). The recursive analysis of the b-spline, therefore, makes it possible to further solve the problem of describing a curve, thanks to a system of sub-division of the same into sections of constant length. Once defined, the traits are superimposed and unified, their joining points are called “nodes”, these features guarantee the designer greater precision and local control by points.Mathematically, the b-spline was later discovered as a generalization of the Casteljau algorithm, including a wider range of geometries, including the Bézier curve as a special case (Farin et al. 2002). We find a further evolution in the history of splines inside Boeing, the American aerospace company, where, with the spread of the first computers in the 70s, each department had bought or developed software to meet specific design needs. Each software described distinct and common types of geometries with different methods, this prevented effective communication between departments. To overcome this systemic incompatibility, Boeing in 1979 appointed a group of mathematicians, asking them to create standardized representation methods for eleven different curves: from simple lines to b-splines. Mathematicians soon recognized as their goal to obtain a single method capable of coherently representing all 11 curves, they succeed in two steps: first, by making the Bézier curve rational, that is by giving to each single polygonal control points the possibility of varying in “strength” according to a weight ratio, and later incorporating the concept of Non-Uniform Spline, in which the spline nodes are not uniformly spaced apart (Piegl, Tiller 1995).
In 1981 the group of mathematicians will then find a mathematical solution in non-uniform rational basis splines, NURBS (Fig. 4) first defined by Ken Versprille, during his doctorate at the University of Syracuse, New York (Versprille 1975). In the same year, the initial graphics exchange standard IGES, a standard system promoted by a consortium of US industries and governmental bodies, following a proposal by Boeing designers, adopts NURBS. IGES is later recognized as a national standard by the American National Standards Institute ANSI (Brauner et al. 1981), becoming the common language of engineers and designers from the United States and therefore all over the world.

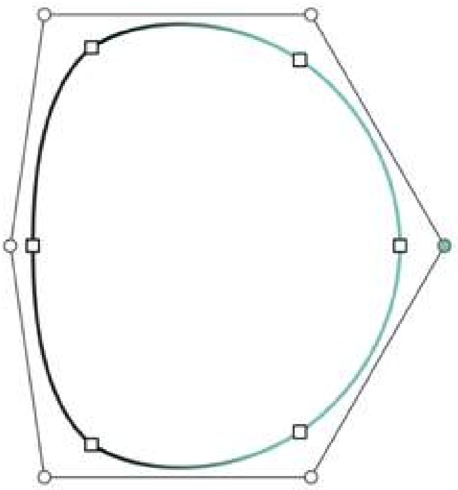

Fig. 3. A Basis Spline, the gradient shows the influence of the green control point on the curve. (Designed by Giuseppe Gallo)

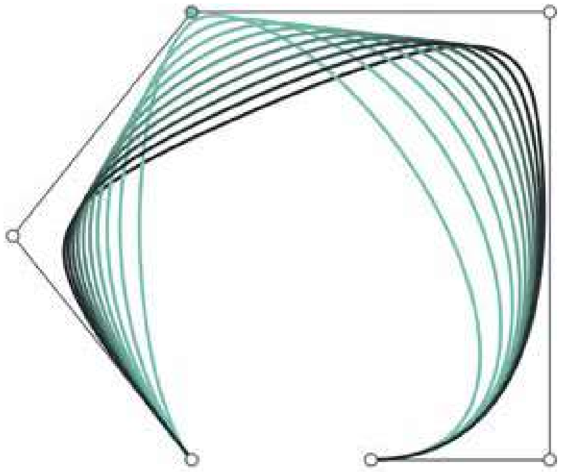

Fig. 4. A series of ten NURBS curves obtained varying the weight of the green control point, from 0.1 (black) to 1 (green). (Designed by Giuseppe Gallo)

Subdivision surfaces methods and the digital animation industry

With the progress of CAD, 3D modelling spread from the scientific world to industrial production and therefore to entertainment industry: in 1986 Steve Jobs acquired Lucas Film's Graphics Group, founding Pixar Animation Studios with Edwin Catmull and Alvy Ray Smith.
Pixar is today recognized as one of the leading pioneers of computer graphics imaging, rightly entered the history of cinema for having made Toy Story, the first feature film ever fully developed in computer graphics.
To produce the film, blockbuster and 1995 Oscar winner, the involved team had to work for 4 years, starting with researches on image rendering and animation (Henne, Hickel 1996), and then painstakingly modelling all the characters, objects and scenes using NURBS geometries. The modelling and animation of characters with NURBS surfaces proved to be a tough process, this is because NURBS surfaces are limited to a rectangular topology, such as distorted planes, cylinders, spheres and tori, but organic forms are rarely so simple.
Their topology is arbitrary, with many holes, folds and protrusions: recreating them with NURBS requires a piecemeal approach, is time-consuming and error-prone (De Rose et al. 1998). To solve the problems of organic forms modelling, in 1997 Pixar experimented the "subdivision surfaces method" (Fig. 5) an algorithm based on the research of Edwin Catmull and Jim Clark (Catmull, Clark 1978). They discovered, during their Ph.D, that it was possible to apply those rules that define b-splines surfaces, limited to a rectangular topology, also to groups of polygonal meshes with irregular topology, devising a geometric process that iteratively smooths the edges of the meshes, finally getting a smooth surface. Taking advantage of Catmull-Clark, Pixar can finally quickly obtain organic forms, these are however generated through a totally different process than the one performed with NURBS: the digital designer works first on a polygonal shape composed of Mesh, a multi-faceted version of the final one, then he launches the subdivision surfaces algorithm that approximates the shape of the meshes to obtain perfectly smooth surfaces. This method allows for solving the expensive continuity problems of 3d surfaces in the animation phase. The advantage is so great, that from then it will become the production standard for every Pixar film.

T-splines, opensubdiv and the spread of computational curves in architectural design

The practice of polygonal modelling has spread in all digital animation agencies today, less in engineering companies, where NURBS are preferred since they allow more precise control over curvatures which is a fundamental designer need for real object production. On the contrary, Animators, who are primarily concerned with how forms will be reproduced in motion, renounce part of their form control, in favour of artistic flexibility and speed offered by subdivision surfaces methods.Subdivision surfaces methods are also used in the architectural design field, many of the protagonists of digital architecture exploit subdivision surfaces in the conceptual design phase. It is no coincidence that they have entered the architecture world in Los Angeles, within the studio of Greg Lynn, who tested them for the production of architectural forms, using Autodesk Maya, a software created for digital animation (Lynn, 1999).

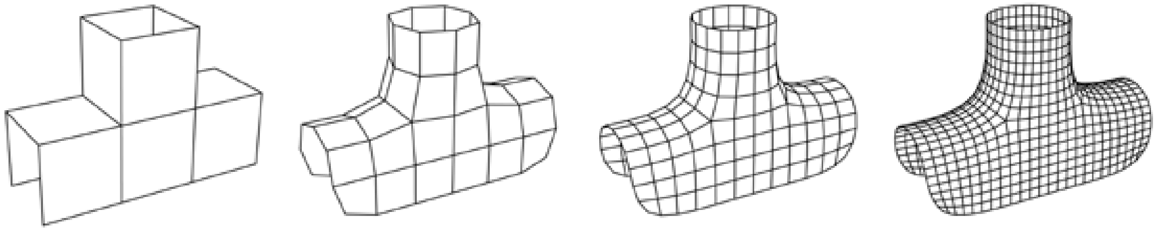

Fig. 5. Example of Catmull-Clark subdivision surfaces method, from the original shape (left) to the third level of application (right). (Designed by Giuseppe Gallo)

As evidence of their use within our field, we observe how subdivision surfaces are today available on several software packages used within contemporary architecture offices, starting from Autodesk Maya, up to Autodesk 3ds max, Blender, Catia 3dxperience and Rhinoceros, software that currently allows the definition of subdivision surfaces through third-party plug-ins and that from the version 7 will integrate a system for the creation and management of subdivision surfaces called Sub-D (Davidson 2020).
NURBS have meanwhile spread within the cad software in every field of design, and exactly the success of the CAD favoured the flourishing of specialized software houses. One of these, the T-spline Inc., was born in 2004 with the intention to market t-splines, a generalization of NURBS and Catmull-Clark, which allows creating articulated organic surfaces, controllable like NURBS, with an ease and speed until then reachable only with Subdivision Surfaces methods.
By definition NURBS must have a structure with a rectangular topology, on the contrary, t-splines can have an irregular topological structure, this thanks to the introduction of internal “T points”, connected not to 4 but 3 other points. This difference makes it possible to create complex organic surfaces that cannot be represented with a single NURBS surface, thus ensuring greater uniformity of the model (Sederberg et al. 2003).
To promote the spread of t-splines, the newborn company decides to develop two plug-ins to be used within the Rhinoceros 3D and SolidWorks modelling software. In 2011 Autodesk acquired T-spline Inc., the software corporation then eliminated the SolidWorks Plug-in (Mings, 2016) and in 2017 stopped updating and selling the t-spline Plug-in for Rhinoceros 3D, preventing the activation of new licenses: keeping t-splines exclusively on the Autodesk Fusion 360 software.
The impossibility of activating new licenses for Rhinoceros created a few problems within those architecture firms that use the t-splines in Rhinoceros version 5 to rationalize the shapes designed in the concept phase with subdivision surface methods through other software such as Autodesk Maya and Blender.
Adversely affecting the workflow for architects, that with the recent versions of Rhinoceros, from 6 onwards, are looking for alternative tools. As an example, in 2019, during a research period at the University of East London, we took part to several meetings of Zaha Hadid Architects BIM group, a team of specialists in digital modelling coordinated by Harry Ibbs. Within one of these meetings, the architects addressed, among other topics, the problem of t-splines, a method now fully structured within their workflow. To continue using it on Rhinoceros 3D, they asked McNeel software house for an *ad hoc* version 5 distribution of the program which would integrate some features of version 6.
The limited number of t-splines licenses owned by the studio, and their inability to activate new ones, has created many problems for the designers, who partially solved them thanks to the dialogue with the company that develops Rhinoceros 3D.
Within the meeting, they presented alternatives useful to replace the t-splines. Among these, the one that comes closest to the results obtained through the t-splines is Clayoo (2020), a plug-in for Grasshopper that allows creating clayoo objects, a method developed by the Spanish company TDM solutions and available to anyone who can afford a license.
The current t-splines problems, confirmed by Harry

Ibbs in an interview attached to Giuseppe Gallo's doctoral thesis, are common to several architectural firms and could have been avoided if these tools had been released with an open-source license, allowing users to view and modify source code of the software, participating the development, and making it collective knowledge. This is the direction chosen by Pixar since 2012, when the animation studio presented, in collaboration with Microsoft, Opensubdiv, (Nießner et al. 2012) a subdivision surfaces Open source library, which uses the efficiency of parallel computing to obtain faster organic surfaces starting from Meshes, and is currently available on some of the major 3D modelling software such as Autodesk Maya, Autodesk 3d Studio Max and Blender.

Conclusions

As we have seen, the history of splines, in its various ramifications, began in the 700s, and since then it has developed, following real needs within the design and industrial production, then in digital animation studios, and only after that, in architecture agencies, quickly becoming a fundamental tool within the largest firms in the world. Architects have largely exploited spline's capabilities, becoming dependent on these tools, without which some of the most celebrated architectures of the last 30 years would have never been realized.
Today, NURBS remains one of the best methods for the description of curves and surfaces within CAD software, through NURBS designers can mathematically represent both analytical geometric objects and freehand surfaces: rational weight control points and NURBS properties allow a practically infinite variation. At the same time, representation by NURBS requires more memory than other methods, especially for regular geometric figures, for example, representing a circle with NURBS is necessary to define seven reference points and ten nodes, instead of centre, normal and radius.
Where NURBS reveal their limits, showing itself as an awkward tool for complex organic shapes production, Sub-division methods easily allow obtaining organic and continuous surfaces that architects can further work rationalizing them for production within a NURBS modelling software. T-splines in particular, proved to be a powerful tool in surface rationalization, combining NURBS precision with subdivision surfaces speed. Unfortunately, the diffusion of t-splines seems to have come to a halt under the control of Autodesk, which became the owner of the technology and limited its exclusive use to one of its software, taking away from the market those t-splines plug-ins on which architects did and still rely, causing losses in terms of time and money.
This example shows the weight that digital tools and therefore specialized software houses have on our sector. The best solution to overcome this problem seems to be open source: sharing knowledge and responsibility in the development of digital tools, as already done by Pixar with the opensubdiv project.

Acknowledgements

We warmly thank Zaha Hadid Architects for their availability, especially Harry Ibbs for the invitation to attend the meetings of the BIM group and his participation in our series of interviews.

Giuseppe Gallo, Ph.D student
Dipartimento di Architettura
Università di Palermo
giuseppe.gallo03@unipa.it

Fulvio Wirz, Senior Lecturer
University of East London
f.wirz@uel.ac.uk

Opening image: Galaxy Soho building, designed by Zaha Hadid Architects, Beijing, China. Photo: Giuseppe Gallo (detail).

References

Barthes R. (1957), *Mythologies*, Seuil, Paris.

Brauner K., Bodnar I., Fournier E., Kelly J.C., Kennicott P., Liewald M., Moore D., Wellington J. (1981), *Digital Representation for Communication of Product Definition Data*, Proposed American National Standards, Engineering Drawing and Related Documentation Practices, Approved ANSI Standards, New York.

Catmull E., Clark J. (1978), Recursively Generated B-spline Surfaces on Arbitrary Topological Meshes, *Computer Aided Design*, New York, pp. 350-355.

Carpo M. (2017) *The second digital turn: design beyond intelligence*, MIT Press, Cambridge, Massachussets.

Carpo M. (2013) *Twenty Years of Digital Design. Digital Turn in Architecture 1992-2012*, Wiley.

De Boor C. (1978), *A practical guide to splines*, Springer, New York.

De Casteljau P. (1963), *Courbes et Surfaces à Pôles*, Citröen, Paris, p. 45.

Derose T. D., Kass M., Truong T. (1998), "Subdivision Surfaces in Character Animation", in *Computer Graphics Proceedings 1998*, Association for Computing Machinery, New York.

Dijkstra E.W. (1960), "Recursive programming", in *Numerische Mathematik*, Springer, Berlin, pp. 312-318.

Farin G.E., Farin G. (2002), *Curves and surfaces for CAGD: a practical guide*, Morgan Kaufmann, Burlington, pp. 216-220.

Farin G. E., Hoschek J., Myung-Soo K. (2002), *Handbook of Computer Aided Geometric Design*, Elsevier, Berlin.

Henne M., Hickel H. (1996), "The Making of Toy Story" in *Compcon*, IEEE, Santa Clara, pp. 463-468.

Lynn G. (1993), "Architectural Curvilinearity, The Folded, the Pliant and the Supple", in *Architectural Design*, 102, Wiley, pp.8-15.

Lynn G. (1996), "Blobs, or why tectonics is square and topology is groovy", in *Any magazine*,14, pp. 58-62.

Lynn G., Kelly T. (1999), *Animate form (Vol. 1)*, Princeton Architectural Press, New York.

Lynn G. (2013), *Folding in Architecture (1993),* Academy Editions, London.

Nießner M., Loop C., Meyer M., Derose T. (2012), "Feature-adaptive GPU rendering of Catmull-Clark subdivision surfaces", in *ACM Transactions on Graphics (TOG)*, New York, p. 6.

Nowacki H. (2006), *Developments in Fluid Mechanics Theory and Ship Design before Trafalgar,* Max Planck Institute for the History of Science, Berlin.

Piegl L., Tiller W. (1995), "Rational B-spline Curves and Surfaces, The NURBS Book", in *Monographs in Visual Communications*, Springer, Berlin.

Schoenberg I.J., (1946), "Contributions to the problem of approximation of equidistant data by analytic functions. Part B. On the problem of osculatory interpolation. A second class of analytic approximation formulae", in *Quarterly of Applied Mathematics*.

Sederberg T. W., Zheng J., Bakenov A., Nasri A. (2003), "T-splines and T-NURCCs", in *ACM Transactions on Graphics (TOG)*, New York, pp. 477-484.

Versprille K. J. (1975), *Computer-Aided Design Applications of the Rational B-Splines Approximation Form*, Syracuse University, New York.

Schumacher P. (2016), "Formalism and Formal Research" in *Arketipo*, 104.

Carpo M., (2018) "The Second Digital Turn. Talks at Google", (www.youtube.com/watch?v=UVerq5DSdKU, 20/07/2020).

Clayoo (2020) "Clayoo, organic modeling inside Rhino", (www.gemvision.com/clayoo, 23/07/2020).

Davidson S. (2020), "Rhino 7 Feature: Subdivision Geometry", (http://discourse.mcneel.com/t/rhino-7-feature-subdivision-geometry/96221, 28/07/2020)

Mings J. (2016), "Yes, Autodesk is Finally Ending t-splines", (www.solidsmack.com/cad/yes-autodesk-finally-ending-t-splines/, 03/04/2019).

Towsend A. (2017), "On the Spline: A Brief History of the Computational Curve", (www.alatown.com/spline-historyarchitecture 02/01/2019).

Ushakov D. (2011), "NURBS and CAD: 30 years together", (isicad.net/articles.php?article_num=14940, 02/01/2019).