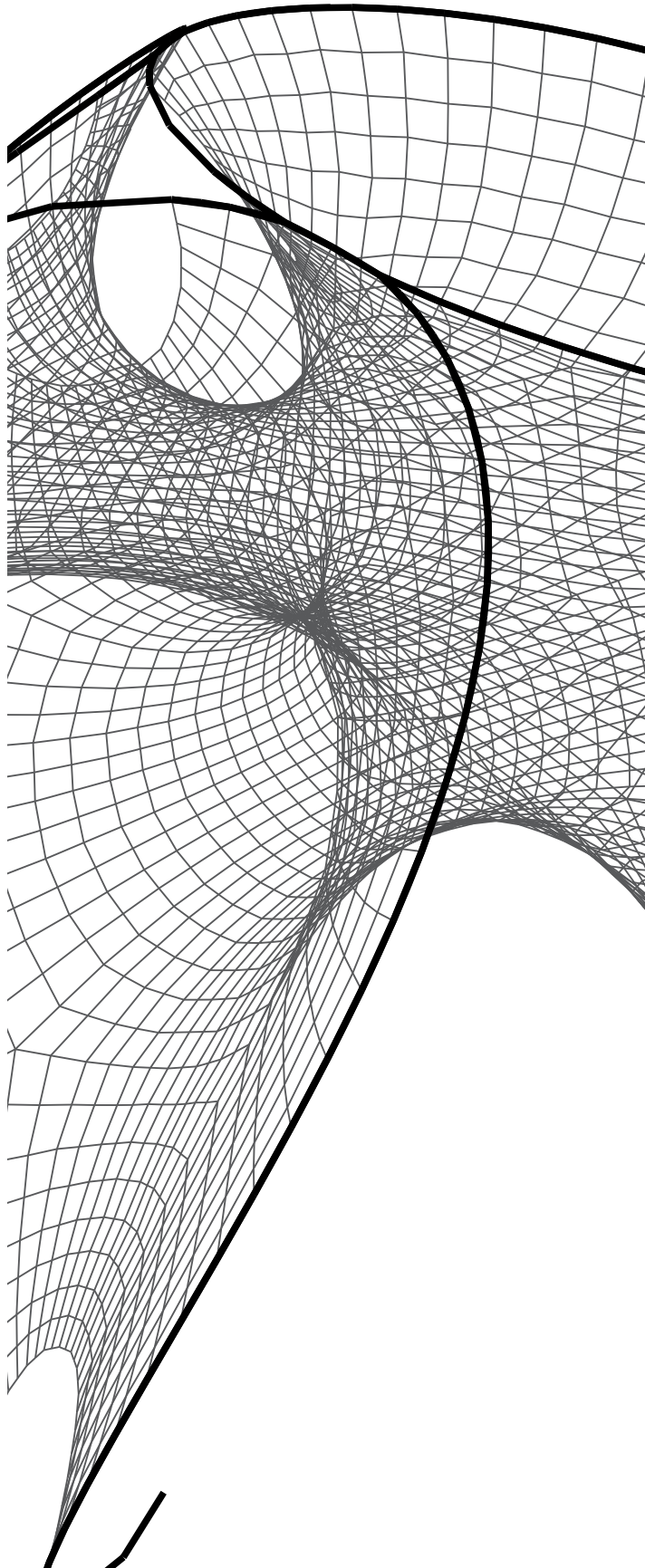
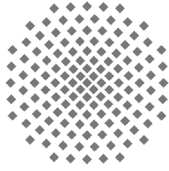


Seiichi Suzuki

Topology-Driven Form-Finding

Interactive computational modelling
of bending-active and textile hybrid structures
through active-topology based real-time physics
simulations, and its emerging design potentials





Forschungsberichte

itke

aus dem Institut für Tragkonstruktionen
und Konstruktives Entwerfen,
Universität Stuttgart

Herausgeber:
Professor Dr.-Ing. Jan Knippers

Institut für Tragkonstruktionen und Konstruktives Entwerfen:
Forschungsbericht 46

Seiichi Suzuki:

Topology-Driven Form-Finding: Interactive computational modeling of bending-active and textile hybrid structures through active-topology based real-time physics simulations, and its emerging design potentials

Stuttgart, März 2020

ISBN 978-3-922302-46-9

D 93

© Institut für Tragkonstruktionen
und Konstruktives Entwerfen
Universität Stuttgart
Keplerstraße 11
D-70174 Stuttgart



Alle Rechte, insbesondere der Übersetzung, bleiben vorbehalten.
Vervielfältigung jeglicher Art, auch auszugsweise, ist nicht gestattet.

TOPOLOGY-DRIVEN FORM-FINDING

Interactive computational modelling of bending-active and textile hybrid structures through active-topology based real-time physics simulations, and its emerging design potentials

Von der Fakultät Architektur und Stadtplanung der Universität Stuttgart
zur Erlangung der Würde eines Doktor-Ingenieurs
(Dr.-Ing.) genehmigte Abhandlung

Vorgelegt von
Seiichi Suzuki
aus Quito

Hauptberichter: Prof. Dr.-Ing. Jan Knippers
Mitberichter: Prof. Achim Menges

Tag der mündlichen Prüfung: 13. February 2020

Institut für Tragkonstruktionen und Konstruktives Entwerfen der
Universität Stuttgart, 2020

Abstract

The inherent relation between aesthetic qualities and structural efficiency inherent on bending-active and textile hybrid structures is associated with a vast range of design opportunities for generating innovative architectural solutions. Exploring those opportunities is a non-trivial task demanding to expand outside current geometric modeling paradigms and develop an insightful and methodological simulation-based design practice. Considering that dynamic methods, such as Particle Systems (PS) and Dynamic Relaxation (DR), have become an important research trend with major advances only focused on speeding up numerical convergence, the main focus of this work is to develop a comprehensive approach for enabling topologic transformations with real-time physics named topology-driven form-finding that, during conceptual stages, can serve to extend design spaces by improving modeling freedom.

A general introduction to bending-active and textile hybrid structures is presented in chapter 2. The following chapter then serves to introduce basic principles of PS and DR methods with an overview of mechanical formulations and integration schemes used throughout this work. Chapter 4 introduces the theoretical framework regarding the development of a generic topologic model with geometric and mechanic embeddings for the implementation of PS models supporting topologic transformations on the fly. Chapter 5 proceeds with the presentation of implementations along with a description of design and modeling implications. Different case studies in which the conceptual design development has been conducted via topology-driven form-finding approaches are presented in chapter 6. Finally, this thesis concludes with a discussion of methods and models developed, and a number of recommendations for future research.

Kurzfassung

Die inhärente Beziehung zwischen ästhetischen Qualitäten und statischer Effizienz von Hybridstrukturen aus biegeaktiven Elementen und textilen Membranelementen ist mit einer großen Bandbreite an Gestaltungsmöglichkeiten zur Generierung innovativer architektonischer Lösungen verbunden. Die Erforschung dieser Möglichkeiten ist eine nicht triviale Aufgabe, die es erfordert, über die aktuellen geometrischen Modellierungsparadigmen hinauszugehen und eine aufschlussreiche und methodische simulationsbasierte Entwurfspraxis zu entwickeln. In Anbetracht der Tatsache, dass dynamische Methoden wie Partikelsysteme (PS) und Dynamische Relaxation (DR) zu einem wichtigen Forschungstrend geworden sind, bei dem größere Fortschritte nur auf die Beschleunigung der numerischen Konvergenz ausgerichtet sind, liegt der Schwerpunkt dieser Arbeit auf der Entwicklung eines umfassenden Ansatzes zur Ermöglichung von topologischen Transformationen mit Echtzeit-Physik, der als Topologie-gesteuerte Formfindung bezeichnet wird und der in der Konzeptphase dazu dienen kann, die Entwurfsräume durch Verbesserung der Modellierungsfreiheit zu erweitern.

Eine allgemeine Einführung in biegeaktive und textile Hybridstrukturen wird in Kapitel 2 vorgestellt. Das folgende Kapitel dient dann der Einführung in die Grundprinzipien der PS- und DR-Methoden mit einem Überblick über die mechanischen Formulierungen und Integrationsschemata, die in dieser Arbeit verwendet werden. Kapitel 4 stellt den theoretischen Rahmen für die Entwicklung eines generischen topologischen Modells mit geometrischen und mechanischen Einbettungen für die Implementierung von PS-Modellen vor, die spontane topologische Transformationen in Echtzeit unterstützen. Kapitel

5 führt mit der Darstellung von Implementierungen und einer Beschreibung der Auswirkungen auf Design und Modellierung fort. Verschiedene Fallstudien, bei denen die Entwicklung des konzeptuellen Designs mit Hilfe von Topologie-gesteuerten Formfindungsansätzen durchgeführt wurde, werden in Kapitel 6 vorgestellt. Schließlich schließt diese Arbeit mit einer Diskussion der entwickelten Methoden und Modelle und einer Reihe von Empfehlungen für nächste Schritte in diesem Forschungsgebiet.

Acknowledgements

I would like to express my deepest gratitude to my supervisor Prof. Dr.-Ing. Jan Knippers for guiding me throughout this journey and for giving me the opportunity to join his team at ITKE. This research would have never been made possible without his invaluable support, patience and advice. To him, I owe the recognition of stepping with courage inside my mind and helping me reformulate my questions and direct my research. I would also like to express my gratitude to my co-advisor Prof. Achim Menges, head of ICD, for sharing his experience and knowledge. It was a great privilege for me to work on a collaborative environment as unique as the one created by both institutes.

I would like to extend my gratitude to my good friends and colleagues at ITKE, specially to Axel Körner, Valentin Koslowski and Florian Jonas for all the great moments that I lived with them. Thank you, Petra Heim and Michaela Denzel, for your invaluable support throughout these years. A special thanks to my good friend and colleague Dr.-Ing. Jian-Min Li with whom I have shared fantastic conversations. I wish to also extend my gratitude to former students Mathias Maierhofer and Valentina Soana for creating new opportunities for the application of my work.

Finally, I would like to thank all my family for their support from the very beginning of my journey. To my dear grandmas, Fanny and Amada, thank you for your unconditional love and support. I miss you both every day. Thank you to my dear wife, Veronica, for her love and patience throughout this crazy journey and for her genuine support with my demanding work. Thank you to my beloved mother, Alba, to whom I owe everything. There are never enough ways to express my gratitude to her.

Table of Contents

ABSTRACT	V
KURZFASSUNG	VII
ACKNOWLEDGEMENTS	XI
LIST OF ABBREVIATIONS	XIX
1 INTRODUCTION	21
1.1 Background	23
1.2 Motivation	24
1.3 Research Problem and Methodology	26
1.4 Structure	29
1.5 Limitations	29
2 FUNDAMENTALS	33
2.1 Bending-Active Structures	36
2.1.1 Background	36
2.1.2 description	38
2.2 Textile Hybrid Structures	40
2.2.1 Background	40
2.2.2 Description	42
2.3 Materials	44
2.3.1 Polymer-based Composites	44
2.3.2 Bamboo	45
2.3.3 Textiles	46
2.4 Design strategies	47

2.4.1	Empirical	47
2.4.2	Experimental	48
2.4.3	Analytical	49
2.4.4	Numerical	50
2.5	Review of projects	51
2.5.1	ICD/ITKE Research Pavilion 2010	52
2.5.2	Umbrella Marrakech	53
2.5.3	Textile Hybrid M1	55
2.5.4	Material Equilibria Installation	57
2.5.5	The Tower	58
3	PHYSICALLY-BASED MODELING	61
3.1	Dynamic Methods	64
3.1.1	Design tools	66
3.2	Dynamic Relaxation (DR)	69
3.2.1	Description	69
3.2.2	Evolution of the method	70
3.2.3	Node structure	71
3.2.4	Forces	72
	3.2.4.1 <i>Translation</i>	72
	3.2.4.2 <i>Rotation</i>	73
3.2.5	Time Integration for Translation	74
	3.2.5.1 <i>Leapfrog Method</i>	74
	3.2.5.2 <i>Runge-Kutta 4th Method</i>	75
3.2.6	Time Integration for Rotation	76
	3.2.6.1 <i>Leapfrog Method</i>	77
	3.2.6.2 <i>Runge-Kutta 4th Method</i>	78
3.2.7	Damping Techniques	78
3.2.8	Elements	80
	3.2.8.1 <i>Elastic Bar</i>	81
	3.2.8.2 <i>Constant Strain Triangle</i>	82
	3.2.8.3 <i>Beam 3DOF</i>	84
	3.2.8.4 <i>Beam 6DOF</i>	85
3.3	Particle Systems (PS)	87
3.3.1	Description	87
3.3.2	Evolution of the method	89
3.3.3	Particle Structure	91
3.3.4	Potential Energies	92
3.3.5	Variational Implicit Euler (VIE)	93

3.4	Remarks	97
4	TOPOLOGY-DRIVEN FORM-FINDING	99
4.1	Generative Approaches for Form-Finding¹	101
4.1.1	Geometric-Driven Approach	103
4.1.1.1	<i>Definition</i>	103
4.1.1.2	<i>Examples</i>	105
4.1.2	Topology-Driven Approach	106
4.1.2.1	<i>Definition</i>	106
4.1.2.2	<i>Examples</i>	108
4.1.2.3	<i>Related Examples in other fields of research</i>	109
4.2	A Generic Topologic Model For Topology-Driven Form-finding	111
4.2.1	Topologic Models	111
4.2.2	Combinatorial Maps	115
4.2.2.1	<i>n-Gmaps</i>	117
4.2.3	Orbits	120
4.2.4	Embeddings	122
4.2.5	Subgraphs	125
4.2.6	Vertex State	128
4.2.7	Graph Grammars	129
4.2.7.1	<i>Ruleset for Particles</i>	129
4.2.7.2	<i>Ruleset for Cables</i>	132
4.2.7.3	<i>Ruleset for Beams</i>	133
4.2.7.4	<i>Ruleset for Tensile Surfaces</i>	137
5	COMPUTATIONAL IMPLEMENTATIONS	143
5.1	Object-Oriented Particle System	146
5.2	Data Structures	149
5.2.1	Pointer-Based Half-Edge Data Structure	149
5.2.2	Array-Based Half-Facet Data Structure	152
5.3	Discretization Techniques	155
5.4	Developments	158
5.4.1	ElasticSpace	158
5.4.1.1	<i>Implementation</i>	159
5.4.1.2	<i>Numerical Solver</i>	161
5.4.2	Iguana	163
5.4.2.1	<i>Implementation</i>	163
5.4.2.2	<i>Numerical Solver</i>	166

5.5	Topologic Implications in Numerical Models	168
5.5.1	Assisted Decision-Making ²	169
5.5.2	Selective Control of Interactivity and Accuracy	172
5.5.2.1	<i>Multi-Scalar Spline-Beams³</i>	173
5.5.2.2	<i>Multi-scalar Tensile Patches⁴</i>	175
5.5.2.3	<i>Multi-Resolution Tensile Patches</i>	177
6	CASE STUDIES	183
6.1	Bending-Active Demonstrator 2017-18	185
6.1.1	Description	185
6.1.2	Topology-Driven Approach	187
6.2	AAVS Madrid 2016⁵	191
6.2.1	Description	191
6.2.2	Topology-Driven Approach	192
6.3	Textile hybrid Models	195
6.3.1	BAT01 ⁶	195
6.3.1.1	<i>Description</i>	195
6.3.1.2	<i>Topology-Driven Approach</i>	196
6.3.2	BAT02	200
6.3.2.1	<i>Description</i>	200
6.3.2.2	<i>Topology-Driven Approach</i>	201
6.4	IAAC-GSS 2018 Computational Bamboo⁷	205
6.4.1	Description	205
6.4.2	Topology-Driven approach	207
6.5	Self-Choreographic Networks⁸	211
6.5.1	Description	211
6.5.2	Topology-Driven Approach	212
7	CONCLUSIONS AND FUTURE DIRECTIONS	217
	APPENDIX A	225
	BIBLIOGRAPHY	229
	IMAGE INDEX	247
	PROJECT ACKNOWLEDGEMENTS	251

List of Abbreviations

AHF	Array-Based Half-Facet Data Structure
CAD	Computer-Aided Design
CBD	Constraint-Based Dynamics
CFRP	Carbon Fiber Reinforced Polymer
DOF	Degrees of Freedom
DR	Dynamic Relaxation
FBD	Force-Based Dynamics
FEM	Finite Element Model
FRP	Fiber Reinforced Polymer
GFRP	Glass Fiber Reinforced Polymer
GUI	Graphical User Interface
NURBS	Non-Uniform Rational B-Spline
PHE	Pointer-Based Half-Edge Data Structure
PS	Particle System
VIE	Variational Implicit Euler

1

INTRODUCTION

1 INTRODUCTION

1.1 BACKGROUND

Computer-based technologies and composites materials have brought a revitalized interest on the research of structures whose form and performance capacities are inevitably associated. Bending-active and textile hybrid structures are among the most relevant examples of this. Here, problems are mainly associated with the complexity of their design process since their shape cannot be predicted with standard geometric modeling techniques and needs to be form-found, either experimentally or numerically. Current efforts on the use of computer simulations as generative tools are mainly focused on improving the rate of numerical convergence without major considerations for extending modeling capabilities. Fast simulations allow to conduct multiple design iterations, but a limited user-model interactivity is still reported due to the incapacity to modify topological conditions on the fly. This lack of interactivity has a great impact during earlier design stages where main ideas and concepts come to light.

The work presented in this dissertation is placed at the intersection of a large research body regarding physically-based modeling and the research on bending-active and textile hybrid structures. The term physically-based modeling is a generic term widely used in the field of computational graphics to denote the simulation of physically plausible behaviors of real-world objects and may not be confused with the purpose of scientific modeling. In this context, our interest is the study of modeling capabilities and design opportunities that can be created by allowing designers to directly interact with the topological properties of physically-based models during simulation runtime. The aim is to provide designers with more modeling freedom during conceptual stages

to facilitate the exploration of design spaces derived from the elastic properties of these structures. A more flexible and robust combination of data structures, topologic models and numerical techniques is therefore required to support different geometric, topologic and mechanic operations that permit to transform, in a reliable way, the structure of the numerical model in response to user inputs. These alternative solutions fully oriented towards design applications need to also find an equilibrium in the consumption of computational resources where flexibility is improved by means of keeping the mechanical accuracy within certain ranges of “soft” tolerances. This condition contradicts traditional numerical implementations derived from engineering applications that are entirely focused on reducing interactivity for optimizing the use of resources for solving highly complex and detailed mechanical problems.

1.2 MOTIVATION

Nowadays, real-time physically-based models are commonly used to assist the design process of bending-active and textile hybrid structures during conceptual design stages. However, developing structural shapes from material behaviors keeps being a complex task regardless of the introduction of these “fast” simulations. Simulation technology has been developed to conduct analysis and validation routines on topologic models that are well-defined in advance. Hence, a numerical model of any type is always described by topologic model embedding geometric and mechanic information. By executing the simulation, it is implicitly assumed that the topologic model is solved and the unique problem to address is entirely geometric. However, introducing a wrong set of initial conditions (geometric, topologic and mechanic) will end with an incorrect geometric output not satisfying performative conditions or design intentions.

To address major changes on the design, the numerical integration must be stopped for introducing topologic changes before relaunching the process with a new set of initial conditions that can finally led towards a more desirable geometry. This means that designers need to intuitively modify the topologic properties of the numerical model on an “undeformed” state by just envisaging

what could be the potential impact on the “deformed” state. It may be argued that breaking the simulation to introduce changes compromises the design process and, eventually, constrains the design space of solutions. Moreover, new practitioners might perceive a loss of control of the entire modeling process since the explicit manipulation of the geometry is replaced by a hidden numerical procedure that automatically develops the geometry. Because focus is on processes and not directly on the form itself, it takes some practice for designers to gradually adopt this new design mode. For this reason, preparing and running numerical simulations tends to be a time-consuming practice that strongly reduces user-model interactivity and compromises creativity.

Being able to fully exploit the architectural potentials of bending-active and textile hybrid structures entirely depends on the capability of designers to work on the global geometry, the topologic arrangement and the mechanic properties of its constituent. While numerical form-finding deals with geometry and materials, it is assumed that designer’s expertise can always generate a relatively good set of topologic conditions to start the simulation. The use of fast simulations can help to gain this expertise through a process of trial and error. However, it is still difficult to transfer results from deformed geometric models to topologic modifications on undeformed geometries.

This limitation is more evident in the exploration of non-regular topologic assemblies producing highly intricate elastic deformations. Working with such levels of complexities require a highly creative and intuitive practice that is commonly solved through the use of analogue models. In fact, scaled-physical models are still used as exploratory design tools to provide the required topologic data for building a numerical model. It is however no less paradoxical that a great amount of manual computation is still required to create complex simulations that ultimately assist in the construction of these highly innovative structures.

As the exploration of these elastic structures becomes more and more integrated within architectural design agendas and related research questions are continuously rising, there

is a growing necessity to introduce more design-oriented simulations for conceptual design stages. It is in this context that the integration of the topologic problem within the scope of numerical simulations play a key role because it permits to blur the separation between undeformed and deformed geometries. Eventually, this new condition may facilitate the appearance of novel forms of user-model interactions with enhanced modeling capabilities for an intuitive exploration of complex design spaces.

1.3 RESEARCH PROBLEM AND METHODOLOGY

The general assumption from where this research is built upon is that numerical form-finding processes were directly adapted from engineering to design applications by only relying on speeding up numerical convergence without taking further considerations of different modeling requirements appearing during conceptual design stages. It is also assumed that, for this reason, analogue form-finding practices build on scaled models keeps being the primary source of creativeness for solving complex design problems, especially in the extended design space of bending-active and textile hybrid systems. In response to these limitations, this research proposes to go beyond the common duality between speed and accuracy of numerical simulations in order to put more emphasis on the concept of flexibility for addressing specific modeling tasks. In this sense, flexibility denotes the capabilities of numerical models to support multiple types of interactions for modifying their properties according to different requirements occurring along the design process.

Our main hypothesis is that *during conceptual design stages, a greater emphasize on the flexibility of numerical models can generate more intuitive and design-oriented solutions to expand modeling capabilities of traditional numerical form-finding processes.*

Overcoming the above mentioned limitations do not only permit to enhance the interactivity of numerical models, but, most important, it can develop an insightful and methodological computational design approach allowing new practitioners to acquire an easy-to-follow holistic understanding for deriving

structural forms from material behaviors. Therefore, the principal objective of this research is to construct a greater basis of knowledge regarding the specific use of “fast” simulations as exploratory design tools in architectural practices. Our focus is entirely based on the use of highly reliable and flexible numerical models based on particles systems permitting designers to directly interact with their topological, geometrical and mechanical properties for the design of bending-active and textile hybrid structures.

The main questions associated with this research relates to *how numerical models based on particles systems can support the transformation of their topologic, geometric and mechanical properties during time integration without creating numerical instabilities or affecting the interactivity of the design process; and what are the design opportunities that can be enabled from those extended modeling capabilities.*

Accordingly, the specific research problematics that need to be addressed relates to the definition of a comprehensive topologic model for describing the entire connectivity of numerical models; the control of the corresponding topologic transformations with regards to mechanical formulations and the stability of numerical integration; and the use of such topologic models to promote the exploration of innovative concepts associated with static and kinetic behaviours, and non-standard topologies of bending-active and textile hybrid structures.

To do this, our aim is to first categorize the different modeling approaches for exploratory form-finding that can be built from the use of numerical simulations based on particle systems and, then, address the introduction of topologic problems by taking advantage of previous work conducted on the field of graph theory, combinatorial maps and data structures (Figure 1.1). To our knowledge, the latter problem has not been fully covered within studies of numerical form-finding, but the concept of introducing topologic changes on real-time simulations has recently become an important topic in computational graphics. For this reason, this dissertation will also include multiple references regarding the subjects of soft-body animations, surgical simulators, computational geometry and physically-

based models. On this basis, the conceptual development and computational implementation of a generic topologic approach will be introduced for extending modelling capabilities of particle models when used as exploratory form-finding in the context of bending-active and textile hybrid structures.

The use of more comprehensive topologic models within numerical techniques aims to improve the rate of user-model interactivity and the ludic characteristics of the design process which can ultimately help to bridge the gap between physical and digital form-finding practices. A series of experiments with digital models and physical prototypes have been conducted to evaluate the design implications of such form-finding processes and validate the different tools developed throughout this research. These experiments have also served to gain a better understanding of the design potentialities of form-finding with dynamic topologies for the development of novel typologies.

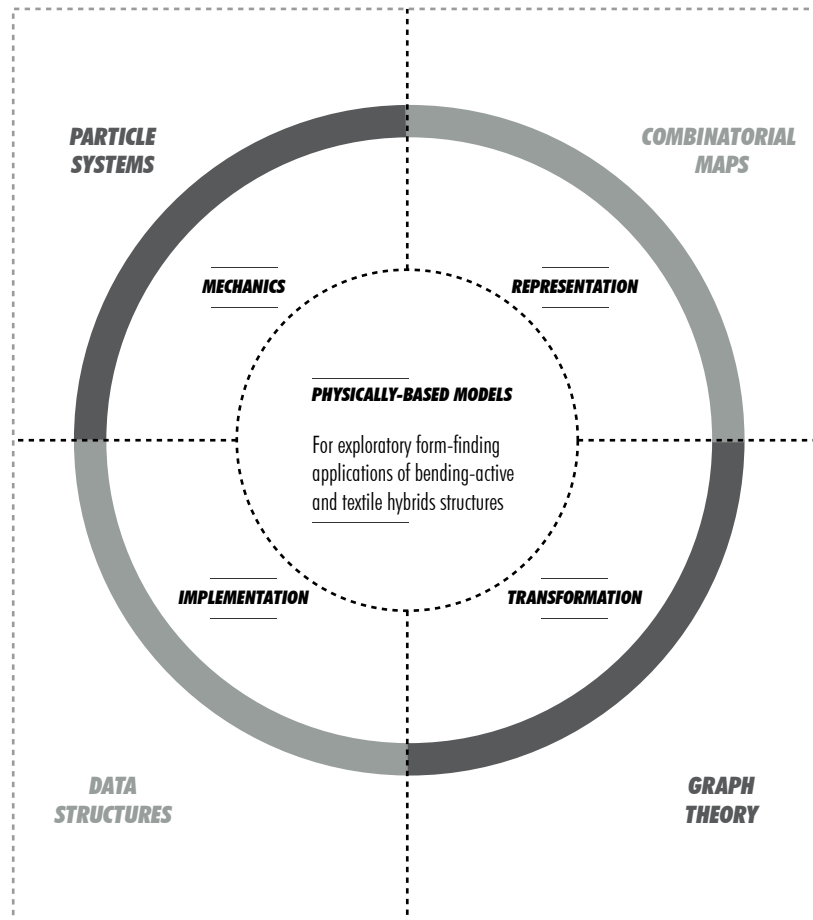


Fig. 1.1

Related topics constituting the core of this research

1.4 STRUCTURE

This dissertation consists of five chapters excluding introduction and conclusion, and a collection of appendices. Chapter 2 “Fundamentals” introduces the fundamental concepts and principles of bending-active and textile hybrid structures. This includes a brief historical overview, the types of materials required, the techniques for form-finding and the most relevant research project from where the main problematic of this dissertation has been formulated. Chapter 3 “Physically-Based Modeling” makes an introduction on the use of real-time numerical simulations for form-finding applications in current architectural and engineering practices. Then, it describes the principles, formulations and techniques of the Dynamic Relaxation (DR) Method and the method based on Particle Systems (PS). Chapter 3 “Topology-driven Form-Finding” focuses on categorizing the different approaches for conducting explorative form-finding process using real-time numerical simulations. On this basis, it describes the conceptual development of a generic topologic model embedding geometric and mechanic information for enabling topology modeling with real-time physics. Chapter 5 “Computational Implementation” describe the implementation of such framework using an Object-Oriented Programming approach in a way that the code could be easily reusable, extendable, controllable and user-friendly. In addition to this, it shows the different modeling and design implications that can be derived from it. Chapter 6 “Case Studies” present a series of prototypes that were conceived to evaluate the different computational implementations as well as the design opportunities resulting from transforming the topologic properties of numerical models on the fly. Each prototype shows different ways of using such extended modeling characteristics.

1.5 LIMITATIONS

Note that this research doesn’t attempt to, neither can, provide a solution for all form-finding problems involving bending-active and textile hybrid structures but rather presents an alternative to expand the design space from where solutions can be found. The numerical approach presented in this dissertation is entirely oriented towards conceptual design applications in earlier

design stages, and it is not designed to replace later analysis processes. Moreover, it shall not be expected to present new mechanic formulations of elements within the research context of bending-active and tensile form-active structures. On the contrary, it builds on existing formulations and combines them to develop novel approaches for computational implementations that are fully oriented to extend modeling capabilities of real-time simulations used during the conceptual design stage of these structures. For this reason, a detailed overview of all the mechanic formulations used in this dissertation will be presented, but, for a more complete validation, we suggest the reader to review the original literature accompanying each description.

2

FUNDAMENTALS

2 FUNDAMENTALS

The concept of form generation in architecture relates to the creation of unspoken associations emerging from ordering and organizing inhabitable spaces within functional forms satisfying material, structural and aesthetical criteria. Hence, it should be seen as a design process that facilitates the exploration of multiple aesthetic and functional concepts while, at the same time, allowing an early evaluation of different performative criteria. However, performative aspects tend to be excluded from computer modeling techniques used in architecture since they are mainly built on geometrical schemes lacking material qualities and physical behaviors. This oversimplification was intended in order to enhance the creative process by presenting a computational model in a more meaningful and understandable way. These non-physical modeling schemes for Computer-Aided Design (CAD) technology have been developed to expand the catalogue of geometric spaces that a designer can work with. Each type of modeling scheme provides different opportunities in terms of interactivity, intuition, visualization and manipulation of the design space. Through an appropriate scheme selection, as for example Non-Uniform Rational B-Splines (NURBS) surfaces or Boundary Representation (BREP) models, more freedom in the design process can be gained. These types of modeling schemes have reinforced the long-running design paradigm in which material systems are used as a receptive “substratum” for a physical shape that is fully conceived in isolation by designer’s mind (DeLanda 2002).

The lack of optimization principles during the generation of form has strongly divided the design process into different stages of modeling, analysis and fabrication. Under these conditions, it is assumed that predefined geometries can always

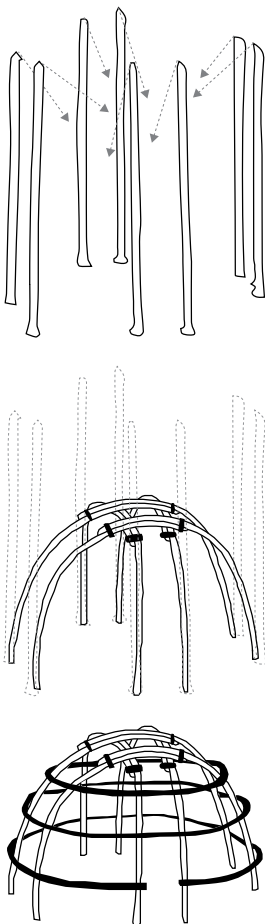


Fig. 2.1

Burnham Pavilion.

be superimposed over a rigid structural typology with a well-defined set of primary structural forces and clear materialization capabilities (Figure 2.1). The required performance is then attained by using stiffer materials or increasing its solid volume, and not through inexpensive operations for optimizing its geometry.

An alternative solution to this problem is condensed within the research on bending-active and textile hybrid structures. Lightweight design principles are oriented towards the development of more sustainable architectures that can self-regulate and coexist with a reduced impact on their surroundings. Architects and engineers have been intrigued, over the last years, by the vast variety of aesthetic and material qualities exhibited by these structures. The key aspect of these types of structures is to attain maximal performances by means of a minimal use of material resources and energy. This is originated by a distinctive load bearing mechanism that is essentially determined by the geometrical shape of the structure. However, the problem is that this geometry can only be approximated through a shape optimization process called form-finding that necessarily requires a paradigm shift on computer modeling techniques used in architecture. In the following, an introduction to both types of structural systems will be presented.



2.1 BENDING-ACTIVE STRUCTURES

2.1.1 BACKGROUND

Historically, the use of active-bending as a building technique to produce curved elements is not entirely new in architecture. Across different cultures and regions where mainly softwoods are available and timber technology hasn't been fully deployed, the intuitive use of elastic deformations and a practical knowledge of traditional building materials have constituted the fundamental principles for the construction of vernacular architectures (Lienhard et al. 2013c). These empirical constructions that are typically formed by curved arches or shell structures, make use of material elastic deformations as an efficient building technique to face local economic and fabrications constraints in the construction of permanent and temporary shelters. There is a vast

Fig. 2.2

Bender tent construction principle.

evidence reported in the Near East of Asia from at least Sumerian times till today, on the use of active bending as a construction technique for local developments on barrel-vault “bender” tents and huts (Cribb 1991). Such construction types are shaped from tying together palm ribs or reed into vertical bundles that are later bent to form arches and then lashed together with other horizontal elements following the same construction principles (Figure 2.2). Africa, South America and, specially, South-East Asia have also a long tradition on the use of the elastic properties of bamboo for the development of local architecture (Gass et al. 1985).

In the mid-twentieth century, the growing necessity to rapidly build lighter and economical housings in the United States had motivated Richard Buckminster Fuller to conduct several studies on the idea of geodesic domes for architecture that can be efficiently produced with the same technology used by the aircraft industry during wartime. In this context, the Plydome experiments (Figure 2.3) were among the firsts to approach the methodical utilization of elastic material properties to construct more economically and technologically efficient shelters. By 1960s, the strong interest of Frei Otto and his teams on the elastic properties of materials has introduced the research on elastic gridshells. This type of freeform structure that is formed out of a connected assembly of rods or laths, achieves its final geometrical configuration through the elastic deformation of its constituent during the erection process.

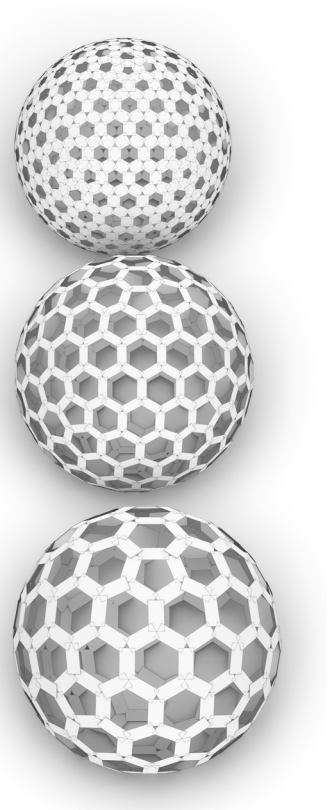


Fig. 2.3
Plydome construction principle.



Fig. 2.4
Multihalle Garden Pavilion.

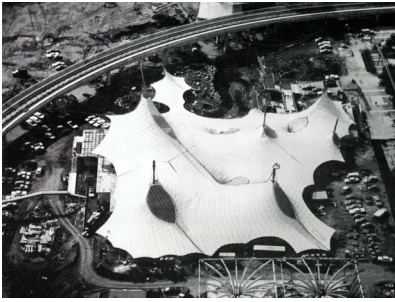


Fig. 2.5

German Pavilion 1967.



Fig. 2.6

Japan Pavilion in Hannover.



Fig. 2.7

Weald and Downland Gridshell.



Fig. 2.8

Ephemeral Cathedral of Créteil.



Fig. 2.9

ICD/ITKE Research Pavilion 2015-16.

By 1975, the construction of the Multihalle Garden Pavilion (Figure 2.4), a 7500 m² wooden elastic gridshell for the Mannheim Bundesgartenschau, was completed. An innovative experimental approach that will be detailed in the following section, was developed to conduct the design of this structure. This pioneer lattice structure that still exists today, is the result of almost a decade of research that started with the designs of the Essen Pavilion in 1962 and the German Pavilion on the occasion of the International and Universal Exposition in Montreal in 1967 (Figure 2.5). The Mannheim Multihalle has, by its own, originated a new research agenda on elastic structures and, ultimately, has inspired the construction of modern elastic gridshells like the Japan Pavilion in 2000 (Figure 2.6), the Weald and Downland Gridshell in 2002 (Figure 2.7) or the Ephemeral Cathedral of Créteil in 2013 (Figure 2.8), as well as few experimental and research-oriented structures like the ICD/ITKE Research Pavilion 2010 and 2015 (Figure 2.9).

2.1.2 DESCRIPTION

The term bending-active was first used in Knippers et al. (Knippers et al. 2011) to describe a special type of section-active structure in which the elastic deformation of its loadbearing elements is used to develop curved geometry and system stiffness. Active bending defines a controlled forming approach, also found in vernacular architectures of different cultures, to self-stabilize a structural system by means of elastic deformations inducing curvature on flat components (Lienhard 2014). Recent developments have shown the greater opportunities that bending-active systems offer for exploring innovative static and kinetic architectural solutions due to the intrinsic relation among aesthetic qualities, shape adaptability and structural efficiency.

Here, the distinguished characteristic is the utilization of structural elements with slender cross-sections that allow to keep materials below its elastic limits when a deforming load is applied. This condition not only makes bending-active systems an extremely lightweight construct but also implies that if the deforming load is removed, those elements can return to its original size and flat configuration facilitating the development

of compliant mechanisms and deployable systems. The inherent contradiction of such slenderness and lightness is that the derived flexibility normally compete against the capacity to withstand loads (La Magna 2017; Gengnagel 2013). Under these conditions, the fundamental mechanism to control its structural performance is through an appropriate geometry or geometrical assembly. However, the task of predicting such geometrical shapes and complex arrangements through pure geometric modeling is nearly impossible because the static equilibrium of the structure is reached on the deformed state. This condition necessarily requires the utilization of analogue and numerical methods, or a combination of both, for finding an appropriate geometrical shape satisfying most of the initial design intentions through local adjustments on forces, boundary conditions and material properties given its reciprocal relationships.

Among the many different advantages that are derived from the characteristic lightness and flexibility of bending-active systems, it stands out the large design space and extended materialization capabilities that such systems offer for generating complex doubly curved geometries with different loadbearing capacities through a smart combination of simple planar elements. The most important point here does not only relate to aesthetics or performative criteria but also to the potential economic and environmental impact that this premise might have in the near future of the building industry. Efforts on this subject may generate alternative solutions to replace the massive mold customization and high-end machinery required for materializing complex freeform geometries. This may allow the use of more affordable and local manufacturing techniques to produce multiple elements with different curvatures.

On the other hand, it has been also found that such inherent flexibility characterizing bending-active systems constitutes an important mechanism to explore novel solutions for adaptive structures. From a kinetic standpoint, the concept of compliant mechanisms is originated from the principle of the reversibility of elastic deformations. Compared with a mechanical solution, this permits to massively reduce the number of elements and moveable joints required to accomplish the same task. While several small-



Fig. 2.10

Kinetic façade of the Thematic Pavilion at the Expo 2012.

to medium-scale projects have been realized on this subject in multiple disciplines going from aeronautics to industrial design, few efforts have been conducted within a large-scale architectural context. The kinetic façade of the Thematic Pavilion at the Expo 2012 in Yeosu is the most up-to-date example on the use of active bending for kinetic architecture (Figure 2.10). This example of a kinetic façade consists of 108 individually controlled glass fiber reinforced polymers (GFRP) lamellas that are supported at both, top and bottom, edges of the façade for covering a total length of 140 m with multiple heights (Rutzinger et al. 2012). This fully functional build example served to highlight the versatility of bending-active systems to develop remarkable solution for complex architectural problems that were formerly difficult to approach due to the mechanical, material and fabrication implications of the design task.

2.2 TEXTILE HYBRID STRUCTURES

2.2.1 BACKGROUND

Despite that the use of textiles as building materials has a millenary tradition in architecture which is exemplified in the roof of Romans stadiums and theaters, the introduction of synthetic materials from the second-half of the twentieth century has created a renewed interest and importance on the research of tensile structures due to the availability of stronger and more durable textiles (Knippers et al. 2011). By 1967, Heino Engel published the first edition of his book *Structure Systems* where he introduced the term *tensile form-active* to define those flexible systems of non-rigid materials that are only subject to in-plane membrane forces of tensile nature and where the loadbearing capacity is directly derived from a suitable equilibrium shape (Engel 2013). The ability to withstand loads is therefore determined by the form, the materials and the boundary conditions of the structure. This class of lightweight system that includes cables, cable-nets, pre-stressed membranes and air-supported membrane structures, has been originated from the old idea of tent-like constructions of nomadic people to develop contemporary large span architectural solutions with a minimum material usage.

Frei Otto and his team were among the firsts to extensively contribute on the construction of innovative tensile structures for evaluating their material qualities but also to comprehend the difficulties rising within their design processes. Otto's vision was to build lightweight structures with high strength by making an efficient use of modern materials like synthetic fabrics or steel cables. By 1955, he built a four-point tent structure for the roof of the music pavilion at the German Horticulture show in Kassel (Figure 2.11) which has, for the first time, demonstrated the principle of anticlastic curvature or saddle-shape on pre-stressed textile surfaces (Knippers et al. 2011). In the years that followed, he continued to work on different commission for the construction of medium to large span tensile structures opening the modern era of textile constructions. Particularly important was the sculptural Dance Pavilion in Cologne in 1957 exploring the idea of ridges and valleys supported by poles (Figure 2.12), the Entrance Arc in Cologne in 1957 based on the concept of an arch-supported structure (Figure 2.13), and the German Pavilion at Expo '67 in Montreal. The latter being his first experience on the realization of a large-span project that later aided the construction of the roof of the Olympic Stadium of Munich in 1972 (Figure 2.14) (Otto and Glaeser 1972).

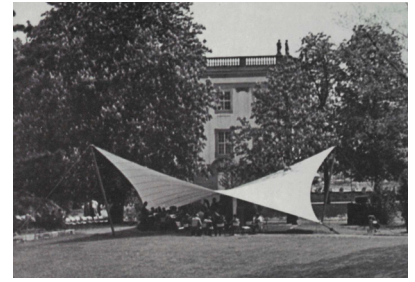


Fig. 2.11

Four-point tent structure by Frei Otto.



Fig. 2.12

Dance Pavilion at the Federal Garden Exhibition.

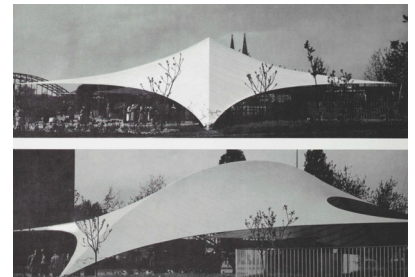


Fig. 2.13

Entrance Arc in Cologne.

Fig. 2.14

Olympic Stadium of Munich.

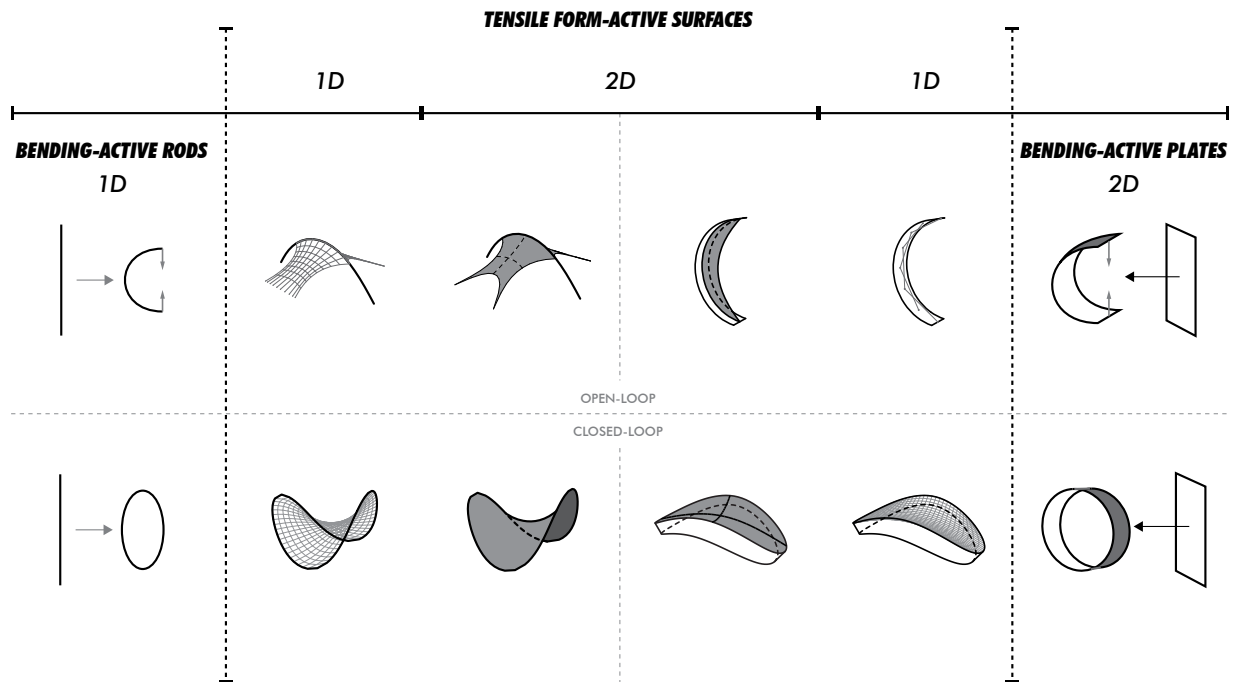
2.2.2 DESCRIPTION

The intriguing characteristic of textile architectures lies in the structural capacity to withstand different loading conditions by means of shape adaptation. This condition is offering designers an extended creative freedom for the exploration of architectural solutions that can adapt to different environments and functional requirements. However, it is also true that a critical point in its design relates to the way these structures are supported and on how such support conditions affect the final shape. On this subject, the integration of elastic beam elements within a pre-stressed membrane permits to design a self-tensioned solution where tension forces can be reduced and free corner points created (Lienhard and Knippers 2015; Lienhard 2014). In this type of hybrid constructs, stiffness is gained through reciprocal stress compensation and elastic deformations of both parental systems. That is, membranes are prestressed through the elastic deformation of bending-active elements whose shapes are in turn controlled by the deformation of membranes. The main idea is to reduce the amount of external supports in the structure without changing the equilibrium form of the unloaded pretensioned membrane. However, the influence of integrating elastically bent elements on prestressed membranes cannot be generalized and the structural behavior might not always be improved (Gengnagel 2013). Through an appropriate configuration, both coupled systems must interact in a beneficial way which means that the stresses introduced on the membrane through the elastic deformation of the slender beams have to be internally solved with minimum or null reaction forces projected towards the ground or surroundings. This principle has been widely used for small-scale applications as the high-tech industry of expeditions tents (Figure 2.15) where the entire design process is based on the empirical analysis of full-scale prototypes with a reduced influence of computational models (Hilleberg Group 2018). It was not before 1992 that the idea of these hybrid systems for large-scale application started to be explored with the development of the design concept of the Oleada main entrance structure for the Seville Expo (Adriaenssens and Barnes 2001).



Fig. 2.15
Hilleberg Atlas expedition tent.

The term textile hybrid is here adopted to define this class

**Fig. 2.16**

Generic types of combinations between bending-active and tensile form-active components on the basis of to their dimensions.

of structural systems shaped from the combination of tensile form- and bending-active elements. Other studies have termed the same type of construct as spliced spline stressed membranes (Adriaenssens 2008), hybrid structures (LaFuente Hernandez and Gengnagel 2012; Gengnagel 2013), bending-active tension structures (van Mele et al. 2013), bending-active tensile membrane hybrid structures (Holden Deleuran et al. 2015). The term was originally proposed to describe the coupling of elastic beam elements with pre-stressed membrane surfaces (Lienhard 2014; Lienhard and Knippers 2015), but the varied dimensionality of its basic building blocks might offer various types of combinatorial logics to build different types of systems (Figure 2.16). Any one or two-dimensional element of the first parental system can be combined with another element of the second parental system with equal or dissimilar dimensionality. This coupling generates a three-dimensional component that can shape a structure by itself, or that can be used to assemble a large system of components. Furthermore, the characteristic modularity, flexibility and lightness of its constituents permits to address the design of these hybrid structures within multiple geometrical configurations and load case scenarios. This also implies designing for one or



Fig. 2.17

Pultrusion process.

multiple stages of static equilibrium that can usually go from fully closed compact configuration for transportation to fully open static configuration able to carry loads. In the last few years, the growing number of efforts conducted for the study of textile hybrid systems has opened different challenges specially for the utilization of numerical form-finding methods within architectural and engineering practices.

2.3 MATERIALS

2.3.1 POLYMER-BASED COMPOSITES

In the research of bending-active structures, the use of low-density materials with a ratio of high yield strength and low bending stiffness is a requirement. Modern synthetic materials like fiber reinforced polymers (FRP) offer the capability to calibrate these required ranges of mechanical properties in response to specific design objectives. Different studies have been conducted on the appropriate selection of materials for active bending purposes (Kotelnikova-Weiler et al. 2013; Lienhard 2014). With rapid advances on composite technologies and a growing interest on performance-based design methodologies among all industrial fields, the principle of active bending is today largely applied on the development of a wide range of finished industrial products going from high performance sport equipment like pole vaults, camping tents or fishing rods, to daily objects like umbrellas or car sunshades. Semi-finished FRP products that are currently employed in the research of bending-active systems, like pultruded rods, have gradually become more economically accessible due to industrial manufacturing process. Nevertheless, the access to such special types of products is still difficult in certain areas mainly because of the high-end machinery required for its production (Figure 2.17). More important is that the environmental impact of polymers regarding its production methods and recycling processes may indicate the necessity to rediscover the potential use of natural material within modern structural systems by means of advanced computer-based technologies. This condition has occasioned new efforts for the utilization of traditional building materials exhibiting comparable properties but processed with local production processes.

2.3.2 BAMBOO

Following Ashby's selection method, Lienhard (Lienhard 2014) considered traditional building materials, like different types of timbers and metals, to be adequate for active bending. It is interesting to observe that bamboo is the only natural renewed material that can be comparable to FRP. Bamboo is a grass plant with a hard outer layer and an inner soft part that makes it strong as wood but much more flexible and lighter. Considering that bamboo is a fast-growing natural resource with minimum environmental impact, its use as building material can be classified as sustainable. The material has a high resistance to tension and an average bending strength twice as strong as most convention structural timber (Minke 2016). For this reason, it has been classified as an ideal material for bending-active and even elastic kinetics (Lienhard 2014). In this context, different considerations need to be taken when employing the elastic properties of bamboo. In its natural form, the minimum bending radius of canes requires larger spans for increasing the geometric curvature of the structure. In case that more flexibility is required, canes can be splitted into laths by introducing cuts parallel to the direction of fibers. The challenge in both cases is to control structural and geometrical behaviors derived from the use of a natural material with heterogeneous properties.

Frei Otto and his multidisciplinary research team at the University of Stuttgart utilized the generic term curved compression arches to describe those structures that are shaped from elastically bent bamboo (Gass et al. 1985). Further studies were conducted by Gernot Minke at the University of Kassel where different domed-like and hyperbolic structural solutions based on the experimental use of elastically bent bamboo laths were developed in Guatemala and Colombia with the collaboration of local universities (Minke 2016). By the end of 2006, two experimental elastic gridshells utilizing bamboo laths were developed in Germany by Auwi Stübbe and his students (Minke 2016) that later served as inspiration for the development of the collaborative Cocoon project led by the Aarhus School of Architecture (Figure 2.18) (Hansen and Kim 2016). In both cases, the design was entirely driven by experimental studies on scaled

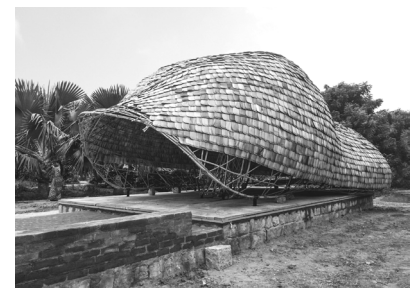


Fig. 2.18

Cocoon project in South India.



Fig. 2.19
ZCS Bamboo Pavilion.

models that, in the case of the Cocoon project, also served to inform the construction of a computational model based on T-Splines. From a digital perspective, special attention demands the ZCS Bamboo Pavilion and the textile hybrid amphitheater in Xingu, Brazil. Built in 2015 as a large-span elastic gridshell in Hong Kong, the ZCS Bamboo pavilion (Figure 2.19) was developed from exhaustive experimental studies on different scaled physical models that were used to gradually calibrate numerical models based on particle methods (Crolla 2017). Both projects employed elastically bent bamboo rods and showcased strategies for addressing the use natural materials within numerical strategies.

2.3.3 TEXTILES

Owing to the geometrical complexity that textile membranes can develop, the selection of the material used for its manufacturing has a direct impact in the final features of the structure as well as in its erection process. Among the most important factors that influence the selection of the material within the current state of the research of textile hybrid systems is the prestressing of the membrane, the generation of cutting patterns and the capacity of the structure to withstand external loads. One group of projects in which the common denominator is the relative simplicity or standardization of the global geometry has reported the use of PVC-coated polyester fabrics (Hernández and Gengnagel 2014; Lienhard 2014). These materials are the most common option for tensile structures due to its high mechanical strength and low cost (Knippers et al. 2011). However, the difficulties for its utilization rises from the necessity to develop very precise cutting patterns and additional equipment for tensioning due to its strength. The second group of projects exposes a clear focus on the exploration of more complex geometrical arrangements (Holden Deleuran et al. 2015; Ahlquist et al. 2013) . For such efforts, the use of uncoated knitted fabrics has been more appropriated. The main advantage of these highly stretchy materials is that they don't require an accurate cutting pattern which is certainly more difficult to generate for complex geometries since wrinkles on the surface can always be eliminated by tensioning more the fabric. While it is true that this type of fabrics compromises the loadbearing capacities of

membrane structures and more studies needs to be conducted on this subject, they are facilitating the exploration of further geometrical and functional possibilities of textile architectures. It is also interesting to note that novel studies on deployability and self-formation has been conducted on this basis (Aldinger et al. 2018; Ahlquist et al. 2014).

2.4 DESIGN STRATEGIES

By studying the variety of built examples, a categorization of bending-active systems has been proposed based on the different approaches adopted for its design. These design approaches named behavior-based, geometry-based and integral were defined according to the type of form-finding technique or modeling strategy employed to derive its shape (Lienhard et al. 2013c). Following a chronological organization, modeling strategies are classified as empirical, analytical, experimental and, finally, numerical. In addition to this, it may be possible to make a distinction between digital strategies like numerical ones, and those that are mainly based on analogue procedures such as empirical, experimental or analytical. This additional layer of categorization aims to differentiate the multiple ways of interactions characterizing each design strategy.

2.4.1 EMPIRICAL

The behavioral-based approach is entirely established on the use of empirical techniques to derive the geometry of the structure. It is evident that in this design approach the use of materials elastic behavior is instinctively applied and the know-how on limitations is gradually acquired through experience. Empirical strategies are therefore the most intuitive but also rudimentary design mode of bending-active structures. Despite that the explicit manipulation of materials under real conditions result on a better understanding of materialization capabilities, the lack of an effective planning completely constraint the design space of possibilities. For this reason, there is evidence that the same empirical practice of active bending often generates similar shapes even across very different regions.

2.4.2 EXPERIMENTAL

The geometric-based approach may be divided into two subtypes. The first variant is associated with the use of experimental strategies. These strategies rely on the construction of scaled physical models to approximate the geometrical shape of a structure under simulated realistic conditions. These types of analogue modeling techniques permit the exploration of structural forms by means of a direct material manipulation. By the late 19th century, Gaudi included, from the very beginning of the design process the use of physical models for studying the static equilibrium of funicular shapes on the basis of Hooke's analogy of hanging chain models (Huerta 2006). Gaudi's vision not only helped to stimulate the study of the influence of form over structure but, most important, it brought a revitalized know-how on the use of materials and physical behaviors within the design process. Since mid-20th century, Heinz Isler gained renown for the development of numerous thin concrete shell projects and, also, for the innovative use of experimental solutions for the design of them (Chilton and Isler 2000). With that in mind, Isler's most important contribution was on the development of hanging cloth reversed methods for form-finding. Around the same period, Frei Otto and his multidisciplinary team started to investigate the manifold of forms that a structure can take as a function of material properties and the flow of forces acting on it. This is to date the most comprehensive study regarding the use of experimental techniques for architecture. Those efforts were established on the basis of material driven processes by making an extensive use of physical models and experiments to investigate the inherent relationship among form, force and mass (Gaß 2016). Different techniques exploring the generation of form through self-formation processes were presented to solve specific architectural design task (Gaß 1990). The principle behind these methods is to identify the most important design parameters for controlling the fundamental relationships between the form of a structure and the forces acting on it.

Special attention demands the experimental strategies applied in the design of the Multihalle in Mannheim. The form of this elastic gridshell was developed through varied explorations

and refinements on hanging chain models. After several studies, a very accurate 1:100 model (Figure 2.20) was built to conduct photogrammetric measurements. The result of these measurements served to build a numerical model by using the recently developed Force Density method of Klaus Linkwitz to better approximate the final geometry. More experimental studies were also conducted to investigate the buckling capacities of the structure through pin-jointed models made of Perspex strips where nails, used as little weights, were attached to simulate loading conditions (Figure 2.21) (Liddell 2015).

Considering that physical models can be easily and cheaply constructed, experimental strategies have become the perfect medium to generate design concepts and explore novel solutions. Moreover, designers can rapidly modify boundary conditions as well as the connectivity of the model to directly produce geometrical results and develop more desirable solutions. This represents a great improvement in relation to empirical strategies since a methodological approach is introduced for the development of structural forms based on inherent materials behaviors. Physical models create a playful modeling interface permitting a highly intuitive design process. This implies that the empirical knowledge on material, forms and structures normally acquired through a craft-making process of learning by doing can be in part gained through analytical observations conducted with the aid of scaled models. However, it is normally the case that the knowledge gained through the use of scaled physical models cannot be directly transferred to large scale applications due to the presence of inaccuracies and principles only working in one particular scale. By the end of the 20th century, Otto's work had already instigated a paradigm shift within architecture design practices.

2.4.3 ANALYTICAL

The second variant of the geometric-based approach is based on analytical strategies. In a period prior to major advances on the field of numerical methods, analytical strategies have been an important contribution in the research of active bending. These strategies were established upon some known studies or through

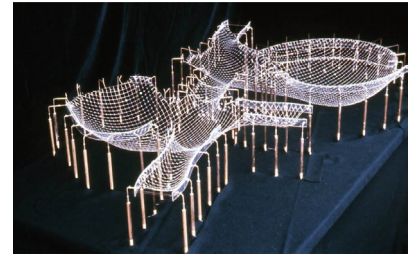


Fig. 2.20

Final hanging chain model of the Multihalle in Mannheim.

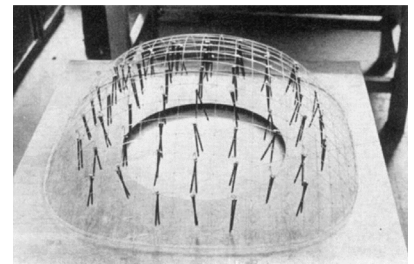


Fig. 2.21

Early studies on load test models using nails.

basic assumptions that, in some cases, required empirical results to better match a good solution. The basic principle is to construct an abstract representation of an idealized model that could be used to understand and accurately predict the behavior of a system. Increasing the complexity of the model however directly creates more difficult calculations that rapidly becomes impossible to solve by hand. An example of the latter was the Plydome experiments conducted by Buckminster Fuller. These experiments focused on fitting flat plates of plywood material through elastic deformation on the double curved surface of a sphere for shaping a dome-like structure (Schleicher et al. 2015; La Magna et al. 2016). The strategy consists of approximating the geometry of a sphere through a regular polyhedron in which polyhedron-edges are used to position each individual plate. By pulling together the overlapping areas between two adjacent plates, individual plates bend in its own plane while the global arrangement generates the desired double curvature of the sphere. Fuller's development has demonstrated the potential of analytical strategies, but it has also proved its limitations. Matching an appropriate solution was always carried through small modifications on the calculations that were intuitively introduced by Fuller based on experience. More important is that basic assumptions like the pre-existence of a basic polyhedral shape, always constrain the design space. This also implies a less intuitive design process in which designers are always working in more complex abstract levels.

2.4.4 NUMERICAL

The integral approach is entirely articulated on the basis of numerical form-finding procedures. Current development on simulation technology have permitted to fully control the generation of forms that are based on the elastic behavior of materials. Compared to analogue strategies, numerical models permit larger design spaces for the exploration of more complex solutions with a relatively limited user-model interactivity according to the type of method adopted. In this context, numerical models are considered as abstract mathematical representation that are used to model the behavior of a physical object over time. Regardless the type of method, the construction of a numerical model tends to be a tedious and difficult task because it implies

working on three abstract layers linked with topologic, geometric and mechanic properties.

A large number of efforts have been conducted for the utilization of diverse types of numerical models within the many different stages required in the design of bending-active and textile hybrids systems. In this context, the ICD/ITKE 2010 Research Pavilion was the first built example entirely form-found with the contracting cable approach for Finite Element Method (FEM) simulations that, almost a decade later, has become the common practice in the research of bending-active structures. The problem originated from the necessity to start FEM simulations from planar configurations and the difficulties of predefining the displacement paths of supports to put the system in place given the intricacy of the structure (Lienhard et al. 2014). The same approach was then applied for the development of the ICD/ITKE Textile Hybrid M1 project where an important contribution was equally reported regarding the use of PS as a fast technique for conducting exploratory form-finding procedures (Lienhard et al. 2013a). More recently, the formulation of the form-conversion approach (La Magna 2017) opened a new research field for the use of numerical models in the design of bending-active structures. This has established a core differentiation between those strategies having a bottom-up organization where target geometries are unknown but topologic configurations are known (i.e. form-finding), and a top-down organization where the target geometries are known but topologic configurations are unknown (i.e. form-conversion).

2.5 REVIEW OF PROJECTS

For the last years, a rapidly growing number of developments have been conducted on different subjects related to active-bending, a review of this is presented in (Lienhard and Gengnagel 2018). The following section presents several research projects based on active bending in which the geometrical and topological complexity is analyzed in relation to the use of specific numerical strategies. To do this, this research adopts the concept proposed by Dominique Chu that defines complexity as the difficulty to model things (Chu 2011). Complexity is then

considered as a property of the modeling process rather than a property of the model itself, like great part of the literature suggests. The practical use of analyzing complexity in this context is that it helps to understand what made difficult to model the specific characteristics of these projects. The selection of projects is chronologically organized and attempts to highlight the growing importance of topological configurations in the design of bending-active structures. Each project represents a starting point for important developments on novel computational processes.

2.5.1 ICD/ITKE RESEARCH PAVILION 2010



Fig. 2.22

ICD/ITKE Research Pavilion 2010.

The research on active bending had a new departure point with the development of the ICD/ITKE Research Pavilion 2010 (Figure 2.22). This structure was designed from a predefined half-torus geometry that was approximated through an interconnected radial configuration of thin birch plywood strips. A particular aspect of this structure was its great stiffness derived from a global connection logic created from local bent and tensioned regions along each of strips (Fleischmann et al. 2012). This way, a system of self-equilibrated arches was developed by using elements only subjected to in-plane bending moments. Weak points on the structure were avoided by breaking the regularity on the connectivity of the strips. This necessarily required the development of exploratory procedures for topologic modeling.

The fact that the target geometry was a half-torus and the discretization logic was radial greatly facilitated the design process. Even if this target geometry can be easily modeled by revolving an arc about a coplanar axis, the early integration of material behaviors and, eventually, its use on the exploration of non-regular connections among strips represented a complex modeling problem. At that time numerical strategies for bending-active structures were not so common neither in FEM nor PS. Therefore, to conduct such design explorations, a rigorous geometric model lacking physical behaviors was developed based on extensive studies on scaled physical models. Physical models served to intuitively acquire all the required knowledge of material behavior for the development of a geometrical method to model the bent shape of each strip. From a given polyline aligned with the radial direction of the half-torus, a B-Spline curve was constructed to approximate the geometry of the bent parts (Fleischmann et al. 2011). The resultant accuracy of such geometric approximation was entirely associated with the use of strips that were solely bent in their own planes. FE-models were introduced only at later stages of the design process when a mature topologic solution was already found. The final geometry of the structure was then form-found via FE-simulations by using the contracting cable approach.

The development of a unique multi-scalar design mode based on the combined use of physical, geometric and FE- models was of great importance among the many contributions of this project. In this case, the use of a pure geometric approach with embedded behavioral information certainly improved the capacity to explore design spaces beyond experimental limits by permitting to successfully break the regularity on the connectivity among the strips. This is the result of an enhanced modeling interactivity that permitted to explore multiple topologic arrangements and produce, at the same time, fast geometrical approximations.

2.5.2 UMBRELLA MARRAKECH

The Umbrella Marrakech was among the firsts realized textile hybrid structure developed after the breaking point established by the ICD/ITKE Research Pavilion 2010 (Figure

2.23). The project was developed in 2011 by HFT Stuttgart in collaboration with ITKE to be permanently mounted in the schoolyard of an architectural school in Marrakech. This permanent structure was designed as a membrane roof integrating three elastically bent GFRP rods in each of its free edges for minimizing the anchoring forces to the surroundings. The simple funnel-shaped geometry of the structure linked with a highly regular topological arrangement of its constituents made initial form-finding conditions for FE-simulations easily to find with physical models. This relative simplicity also served to explore the capabilities that FE-simulations have to conduct different form-finding procedures associated with specific design problems of textile hybrid structures. Such studies helped to define three FEM form-finding approaches that were categorized as additive, successive and simultaneous. In this case, the geometric and topologic simplicity of the Umbrella Marrakech fitted best with the simultaneous form-finding approach in which the numerical process parallelly solved the geometry of the tensile form- and bending-active elements (Lienhard et al. 2013b).



Fig. 2.23

Umbrella Marrakech.

The development of this project has largely contributed to strengthen the importance and validate the mechanical accuracy of FE-simulations. It was in fact reported that the maximum deflection difference with point loads between the real structure and the FE-model was minimal (Lienhard 2014). Furthermore, it has also contributed to understand the versatility of FE-simulations for addressing different design problems appearing in the research of textile hybrid structures. However, the extended geometrical

possibilities rising from the combination of tensile membranes with integrated networks of elastic rods were not fully explored. In fact, further exploratory modeling strategies like the geometric model developed for the ICD/ITKE Research Pavilion 2010 were out of the scope of this project. This was in part associated with a late introduction of the idea of bending-active beams within an already mature design of a membrane structure (Lienhard 2014). In any case, FE-simulations are not well suited for conducting conceptual design explorations.

2.5.3 TEXTILE HYBRID M1

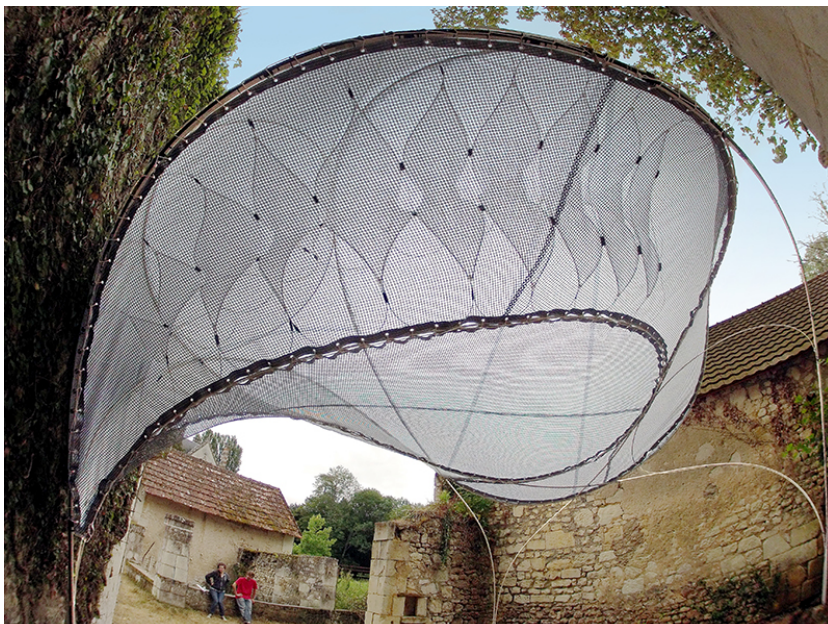


Fig. 2.24
ICD/ITKE La Tour M1 project.

The Textile Hybrid M1 project was developed in 2012 as a temporary structure at La Tour de l'Architecte in Monthoiron, France (Figure 2.24). Among all the reviewed projects, this structure constitutes the most relevant project for this dissertation since it regroups most important factors necessitating the activation of topology modeling during exploratory form-finding. The Textile Hybrid M1 was developed at ICD/ITKE to explore the extended aesthetical and functional potentialities emerging from the exploration of different topological configurations of tensile form- and bending-active elements. The structure was hierarchically designed with a primary and a secondary system. The primary system was constructed from a non-standard network of elastically bent GFRP rods combined with surface

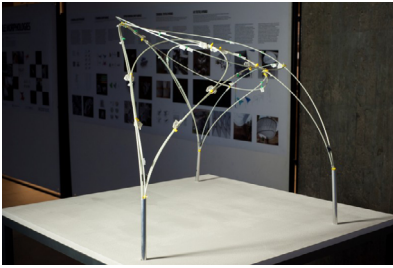


Fig. 2.25

Physical model of the primary structure.

membranes and open-weave meshes to shape a hybrid structural system (Lienhard et al. 2013b). The secondary system consisted of a linear arrangement of small textile hybrid modules placed inside the primary structure to provide additional stiffness and improve illumination qualities (Lienhard et al. 2013b).

The Textile Hybrid M1 clearly exemplifies how the full exploration of the extended design space of these hybrid systems may rapidly originate more complex geometries and non-standard topologic articulations. Despite the low number of elements in this structure, the intricate relationships between them added to the multiple types of interactions made this structure highly complex to solve. A strategy to simultaneously form-find both parental system during early design stages was required. However, the global complexity of the structure made numerical simulations not apt to solve this problem. To address this, a multi-scalar modeling procedure was proposed on the basis of physical models (Figure 2.25). The primary structure was designed from extensive experimental studies on physical models exploring potential geometries and topological configurations that later served to construct the FE-model (Lienhard et al. 2013a). This model was then used to numerically form-find the exact geometry of the structure and determine its viability. Contrary to the Umbrella Marrakesh, the complexity of this structure required the use of the successive FEM form-finding approach. Here, bending-active elements were first form-found to then generate all the membrane meshes before finally run the form-finding process of the coupled system (Lienhard et al. 2013b).

On the other hand, the secondary system was equally designed on the basis of extensive studies on physical models but with the variant of using PS as the numerical form-finding method. These efforts were among the firsts to practically address the use of PS as an exploratory design tool on the research of textile hybrid systems. It must be noted that this secondary structure was developed as a continuation of the Deep Surface Experiments, previously conducted at ICD, about the use of PS for the exploration of complex multi-dimensional arrangements of single textile modules with clear boundaries that were initially form-found with physical models (Ahlquist et al. 2013). Based

on these studies, PS simulations for the secondary structure were constructed on the same basis of the successive FEM form-finding approach. The main reason for this was because the complex interaction of form- and bending-active elements exhibited in the Textile Hybrid M1 limited the adoption of a simultaneous form-finding approach either with PS or FEM (Ahlquist and Menges 2013). The problem on the design consisted on the necessity to simultaneously solve the geometry and the topological arrangement of the structure, without forgetting that both parts depend on each other. Therefore, bending-active elements on the interconnected array of textile hybrid modules were first solved to then generate the tensile surfaces and finally conduct the coupled form-finding of the secondary system. This design mode employing different types of numerical simulations was entirely constructed on the basis of scaled physical models. Those models permitted to create an exhaustive topologic and geometrical database that not only served to construct a complete mechanical description of the primary structure but also to conduct exploratory design studies of the secondary structure.

2.5.4 MATERIAL EQUILIBRIA INSTALLATION



Fig. 2.26
Material Equilibria installation.

The Material Equilibria was developed in 2012 at ICD as a semi-toroidal indoor textile hybrid installation (Figure 2.26). The geometry of this structure approximated a parabolic Dupin cyclide in which the topologic organization of its constituents was relatively straightforward like in the Umbrella Marrakech.

Contrary to the Textile Hybrid M1 project, this condition allowed to explore the capabilities of PS-simulations to conduct form-finding processes on the basis of a simultaneous approach (Ahlquist et al. 2013). A simultaneous form-finding approach was required in order to facilitate the geometric exploration of the Material Equilibria installation while taking in consideration specific design constraints like the necessity to fit in an interior space with a maximum volume and minimum number of support points (Ahlquist et al. 2013).

The main contribution of this project was the development of a simultaneous form-finding workflow using PS. In fact, it was among the first studies addressing the development of a highly interactive form-finding tool for the design of textile hybrid structures in which fast geometrical variations can be produced without altering topological conditions. It may be argued that its applicability is limited to simple topologic conditions or well-known topologies that need to be previously defined on the basis of physical models. It is also important to note that the Material Equilibria originated the current interest on knitting technology as an alternative solution for the materialization of textile hybrid systems due to the capacity of this technology to directly associate digital information with manufacturing logics.

2.5.5 THE TOWER

The tower was a collaborative project developed in 2015 at CITA, KET and Fibrenamics. This textile hybrid structure was constructed from a stacked net of elastically bent GFRP rods restrained by a knitted membrane (Figure 2.27). The distinctive aspect in this structure was its vertical configuration that was associated with specific loading conditions. For this reason, tension cables were introduced to pull the restraining membrane towards the central axis of the structure to better withstand wind loads (Thomsen et al. 2015). Similar to the Textile Hybrid M1, a multi-scalar design approach was also developed to explore the intricate relationships of all the constituents of the structure. It may be also argued that the Tower has some fundamental points in common with the design methodology originated from the Deep Surface Experiments.

Exhaustive experimental studies on physical models were carried to understand and define the geometry and topology of an initial module. From here, great part of the project was oriented towards the exploration of large topological arrangements of such basic modules. Exploratory studies on module aggregations were then conducted on the basis of a parametric model that was used to construct a PS-model and produce fast numerical iterations. In fact, one of the main goals was to address the complex design of the textile hybrid structure through a highly interactive modeling process informed by material behaviors. In this case, different parameters controlling the geometrical aspect of the module, the number of modules per floor and the number of floors in the system were introduced to fully control the generation of all the required topologic and geometrical information (Holden Deleuran et al. 2015).

Similar to other projects, the fast convergence rate of PS-simulations allowed to generate multiple geometric variations on a single simulation run. Modifications on the topologic structure of the PS model were in turn introduced by breaking and restarting the numerical process. Once a mature solution was found, all the geometric and topological information derived from the PS-model was used to inform the construction of a FE-model and analyze the structure. To simplify this connection, stresses on rods calculated on the relaxed PS-model were superimposed on the FE-model (Holden Deleuran et al. 2015).



Fig. 2.27
Hybrid Tower.

3

PHYSICALLY-BASED MODELING

3 PHYSICALLY-BASED MODELING

From the previous chapter, we can recap that the research on bending-active and textile hybrid structures has been established on the basis of a simulation-based paradigm in which building performance and materialization constraints constitutes the guiding principles for designing. Designers are then focused on exploring formation processes regulated by quantifiable aspects rather than on the form itself (Kolarevic 2003; Menges 2010). Through the integration of numerical simulations within earlier design stages, such performative criteria can be quickly evaluated to assist the decision-making for the modification of the geometry (Oxman et al. 2007). On this subject, the growing interest to further explore the complex interrelation between their form and structure is gradually requiring more computational avenues to interact with physical behaviors especially during conceptual design stages.

The concept of physically-based modeling was coined by the computer graphic community to describe all the different types of modeling strategies that incorporate physical behaviors through numerical simulations. By 1999, Greg Lynn already argued for the adoption of animation tools to introduce force and motion dynamics for the generation of form within architectural design processes (Lynn 1999). Physically-based modeling consists of a large collection of algorithms and models that can be used interchangeably in different fields related to computer graphics for an automatic creation of complex shape and motions. In fact, such numerical models have been utilized for different applications including rigid and no-rigid bodies, fractures and fluids. The main focus has been the development of realistic animations by adjusting numerical models to specific purposes

usually associated with interactive applications. It should be pointed out that there are many issues regarding the reusability, extensibility, usability, heterogeneity and control in physically-based models that make them sufficiently different from scientific models (Barzel 1992). Typical numerical simulations for scientific modeling are developed to validate and analyze well-defined models in which mechanics are realistically formulated and accuracy is critical. Because of this, they tend to be highly difficult to set and expensive to compute, most of the time requiring hours or days to produce results. These numerical models are not appropriate for conducting exploratory design studies due to the lack of fast feedbacks to iteratively assist the modification of models.

The use of physically-based models for form-finding is a relatively new design approach that is placed at the intersection of architecture and engineering. Efforts have been developed from special requirements and with specific goals for each application in which a key aspect has been to find an appropriately balance between accuracy and interactivity. The large framework of physically-based modeling can also offer different perspectives to find solutions for more specific problems that are continuously appearing on the research of bending-active structures. Special attention then demands the extensive work conducted on the field of deformable bodies. This implies that comparable problems encountered in other fields of studies, not necessarily related to architecture or engineering practices, can serve to better understand modeling capabilities with real-time physics and construct alternative design-oriented solutions.

3.1 DYNAMIC METHODS

Nowadays, the design community has largely focused on the study of dynamic methods with special attention to DR and PS for the construction of physically-based models. A detailed overview of both methods will follow in the next sections of this chapter. For the sake of simplicity, dynamic methods may be considered as all types of numerical techniques based on discrete sets of interconnected, or disconnected, point masses called particles, or sometimes nodes, permitting to describe the dynamic

behavior of a system over a finite period of time regardless the type of dynamic schemes and numerical integration adopted. A major reason to do this is, firstly, because of the persistent difficulties to establish differences among numerical methods that are under continuous development without a common notation and mathematical scheme. Secondly, it is because of the lack of clarity regarding their aptitudes to support specific modeling requirements appearing along the entire design process. This is in spite of important efforts that have been conducted for developing comparative (Veenendaal and Block 2011; Veenendaal and Block 2012; Veenendaal and Block 2014) and non-comparatives (Li et al. 2015; Lewis 2008; Tibert and Pellegrino 2003) reviews on this subject.

Addressed through this unified principle, dynamic methods can be more roughly differentiate based on the type of integration, either explicit or implicit, and the type of dynamics, either force- or position- based. Explicit methods utilize small time-steps with a higher number of iterations to find a solution. The computational cost of each iteration is relatively small because only local systems of linear equations need to be solved. Implicit solvers support larger time-steps to reduce the number of iterations, but the computational cost of each iteration is higher. Moreover, two main dynamic schemes can be also recognized and used to differentiate methods. Force-based dynamics are generally derived from continuum mechanical theory but, in general, explicit integration is limited to be conditionally stable. The other case is constraint-based dynamics where the great advantage is that integration is unconditionally stable with numerical formulations generally constructed on the basis of potential energies satisfying geometric constraints.

Although there is a long history of developments regarding dynamic methods for scientific applications in engineering and basic sciences, a radical diversification of its usage occurred when they were introduced within related fields of computer graphics due to the growing necessity of fast and realistic simulations for special effects, animations and games. Compared to stiffness matrix methods like FEM, the great advantage of dynamic methods is that they can easily handle large elastic

deformations by means of models that are far more intuitive to conceptualize, ease to implement, efficient to compute and, even, apt for parallelization. It is also true that the disadvantage is still associated with a reduction of the mechanical accuracy of models when real-time applications are required and the lack of physical representation of certain parameters. However, this doesn't necessarily represent a problem during early design stages where user-model interactivity is more desirable than mechanical accuracy.

3.1.1 DESIGN TOOLS

Along the last two decades, a fastest growing number of specialized design tools built on dynamic methods have been developed for different design environments utilized by architects, designers and engineers (Figure 3.1). The courtyard roof of the British Museum completed in 2000 was a pioneer project on this subject where a custom-built DR tool was developed to equalize the distribution of a triangular structural grid approximating the

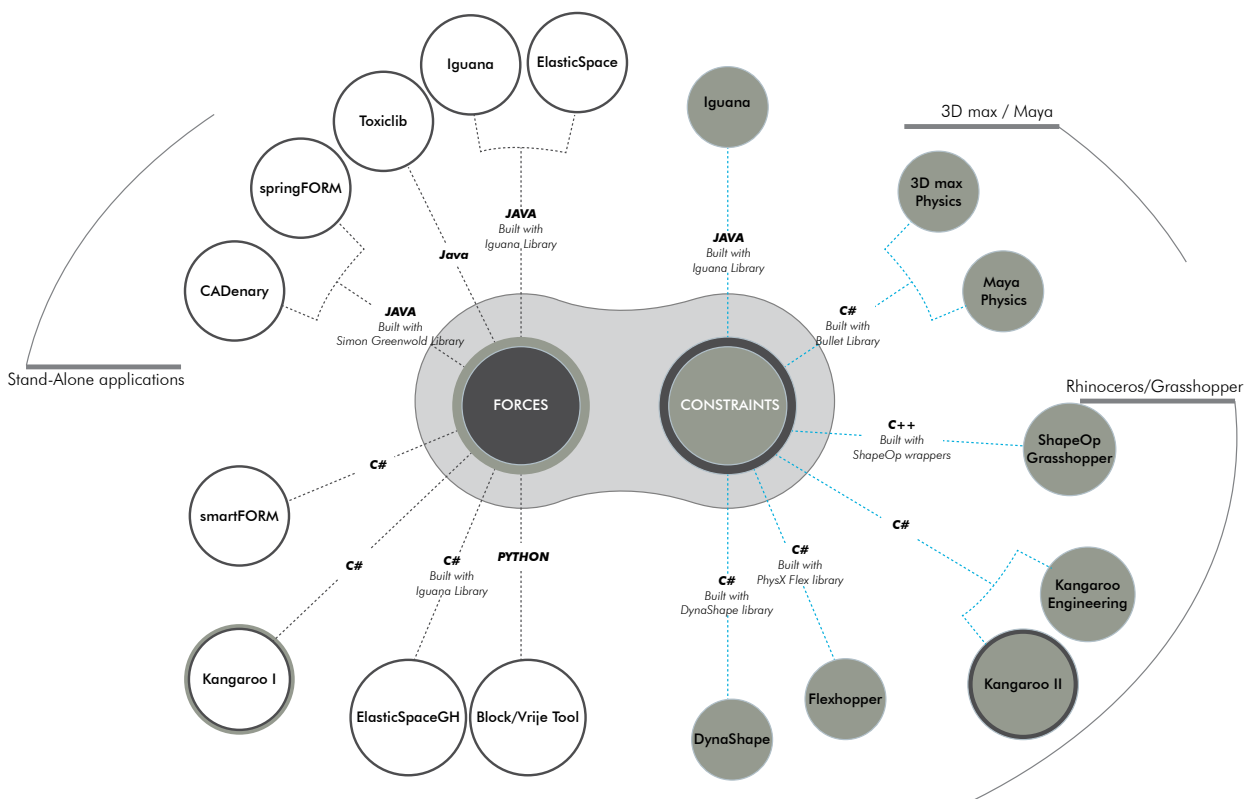


Fig. 3.1

Overview of available numerical form-finding tools for exploratory design studies.

geometry of the roof that was predefined from a mathematical surface (Williams 2001). Since the beginnings of the 2000s, Kirk Martini has extensively contributed on the research regarding the use of PS for teaching and solving solutions of structural engineering problems in highly interactive manner (Martini 2001, 2013, 2005). By 2002, a student project for a computer graphic class at MIT also reported initial efforts on the use of PS for the development of an interactive form-finding application for funicular structures (Chak et al. 2002). Further efforts on the development of this tool were conducted by Axel Killian, an original member of this student project, for a hanging model workshop conducted at MIT in the Spring of 2004 (Killian 2004). This design tool called CADenary was entirely rewritten with Processing which, at that time, was a new programming language developed by Casey Reas and Ben Fry at the MIT Media Lab for creating visual and interactive media. CADenary used a physics library specially developed for this workshop by Simon Greenwold (Killian 2006) that can still be accessed in (Greenwold 2012). Greenwold's library implements a PS method using force dynamics and implicit solvers. Then, Killian and Ochsendorf presented in 2005 one of the most important paper for the research on the use of PS for form-finding applications (Killian and Ochsendorf 2005).

Based on Greenwold's library, Sean Ahlquist and Moritz Fleischmann presented in 2008 different studies exploring the design spaces of cable-net systems by means of PS (Ahlquist and Fleischmann 2008; Fleischmann and Ahlquist 2009). Ahlquist extended the application of PS to the design of more complex tensile systems where their intricate behavior can be computed from simple principles (Ahlquist and Menges 2011). This ultimately ended with the development of the Deep Surface experiments. By 2013, Ahlquist reported the development of springFORM (Ahlquist and Menges 2013), a form-finding environment developed in Processing with Greenwold's library to address the design of tensile structures and textile hybrids. At the same time, Van Mele et al. (van Mele et al. 2013) presented a form-finding tool for textile hybrid systems based on DR that implemented the same simplified mechanism to calculate forces due to bending and a force density formulation for tensile

elements. In this case, the tool was developed as a plugin for the commercial CAD package Rhinoceros.

Special attention demands the case of Kangaroo Physics which was developed around the same time by Daniel Piker. Kangaroo is to date the most popular tool within the large community of designers using Rhinoceros for integrating physical behaviors through fast simulations within the modeling process. Piker described Kangaroo as a “physic engine” directly embedded in the parametric modeling environment of Rhinoceros-Grasshopper permitting an interactive exploration of geometrical shapes through simulated behaviors based on material properties and applied forces (Piker 2013). However, it should be taken into account that the numerical method implemented in Kangaroo changed from the beginnings to the present time. While the first release (K1) was developed on the basis of force dynamics by using DR, the second release (K2) was built on the basis of constraint-based dynamics through PS models where geometrical constraints are used to directly update the position of particles. Over the last years, multiple researchers have reported different efforts conducted on the basis of Kangaroo for addressing specific form-finding problems (Deleuran et al. 2016; Schmeck and Gengnagel 2016; Holden Deleuran et al. 2015; Vestartas et al. 2018). It is also particularly interesting to note that other studies have focused on extending the capabilities of K2. Cecilie Brandt-Olsen developed K2Engineering (Brandt-Olsen 2016) as an open source plugin to analysis results and replace the use of fictitious coefficients with real mechanical properties in K2. On this basis, efforts have also been focused on the use of K2 and K2Engineering for educational purposes by means of augmented reality applications (Quinn et al. 2016).

The numerical approach adopted in K2 has been derived from the work of Bouaziz et al. (Bouaziz et al. 2012; Bouaziz et al. 2014) regarding the integration of more rigorous continuum mechanic principles within classic constraint-based schemas. Based on these studies, an open source C++ physics library called ShapeOp was developed with automatic bindings for different programming languages. Efforts have also been reported regarding the direct implementation of a python binding

of ShapeOp for Grasshopper (Deuss et al. 2015). Moreover, custom implementation of Nvidia GPU physic libraries Flex and PhysX used for visual effects have recently been reported for the Grasshopper environment (Felbrich 2017; Kao and Nguyen 2018). Another physic library used in commercial games and 3D modeling environments, like Blender or Cinema 4D, is Bullet Physics which can also be implemented in Processing through JBullet (Dvorak 2008). Nevertheless, the most common open source library used these days in Processing for different design tasks involving the integration of real-time physics is ToxiClibs (Schmidt 2007). Finally, an important contribution for the design community using Autodesk Dynamo has been the development of DynaShape by Long Nguyen based on PS and constraint-based dynamics (Nguyen 2017)

Since 2010, there is a fastest-growing interest on the development of custom modeling tools integrating real-time physical behaviors for addressing specific design tasks. Among all these important contributions, Kangaroo and springFORM were among the firsts tools entirely oriented to exploratory design applications that implemented the simplified beam mechanism with three degrees of freedom (DOF) per node proposed by Adriaenssens and Barnes (Adriaenssens and Barnes 2001). Thus, the firsts permitting to address explorative design studies of bending-active structures and hybrid by-products.

3.2 DYNAMIC RELAXATION (DR)

3.2.1 DESCRIPTION

Dynamic Relaxation (DR) is a numerical method for solving systems of ordinary differential equations (ODE) using an explicit time integration scheme. The method was originally formulated from the analogy of tidal flow computations in 1965 by Alistair Day for the analysis of frames and plate structures (Day 1965), and later for tension structures (Day and Bunce 1969). By mid 1970s, Barnes had already improved the method to treat complicated nonlinear equilibrium problems of cable and membrane structures (Barnes 1977). From then on, a wide variety of studies have been conducted on the method and its potential

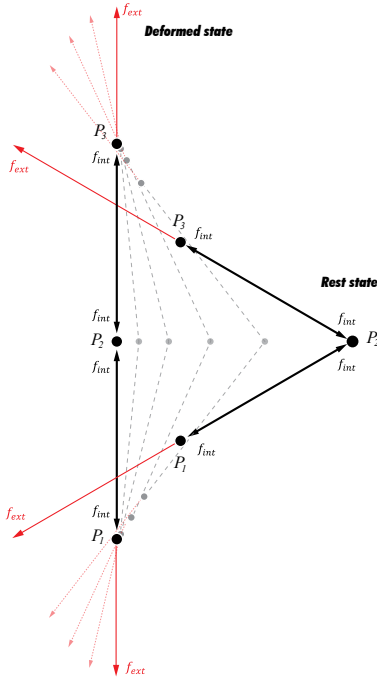


Fig. 3.2

Internal forces of two elastic bars computed through the links of a connected subset of nodes.

application to tensile structures (Lewis 2003; Lewis and Gosling 1993; Wakefield 1980), reciprocal structures (Doutte and Baverel 2009), tensegrity (Tibert and Pellegrino 2003), active bending (Adriaenssens and Barnes 2001; Li and Knippers 2012; D'Amico et al. 2016; Senatore and Piker 2015; Barnes et al. 2013), molecular dynamics (Pan et al. 2004) and surgical simulation (Joldes et al. 2009, 2011).

Through a dynamic analysis using fictitious masses and damping techniques, DR is focused on finding a static equilibrium of a structure exhibiting material and geometric nonlinearities. The method requires the discretization of a continuum into a finite set of nodes with lumped masses that are interconnected via elements. Each element corresponds to a topological subset of connected nodes in which forces are calculated through its links (Figure 3.2). These calculations only depend on the position of nodes and are governed by Hooke's law. Applying those forces into their corresponding nodes causes motion and, eventually, activates the dynamic behavior of the entire discrete system. In this context, DR traces step by step the motion of nodes under the action of forces through an explicit time-integration scheme based on small time increments. The dynamic system oscillates until attaining a steady-state in which all the forces on nodes are balanced under the influence of nodal masses and damping techniques. Because the aim is to attain the static equilibrium state without giving much importance to the dynamic behavior, masses may be fictitious in order to improve the rate of convergence. Under these conditions, DR permits to start the form-finding process from inaccurate geometric conditions that are progressively refined throughout the numerical integration. Clear topological relationships among nodes are always required at each time-step for defining elements and calculating forces.

3.2.2 EVOLUTION OF THE METHOD

Typical well-established implementations of DR for form-finding have been developed to address the design of form-active structures with greater emphasize to tensile structures. In this sense, the method was entirely focused on the generation of equilibrium geometries with no shear or bending moments,

but with uniformly distributed in-plane tension or compression stresses. For this reason, most of the implementations only required the use of translational degrees of freedom per node. It is within the context of active bending that the introduction of rotational degrees of freedom has become important to handle other mechanical problems of elastic members like biaxial bending and torsional effects. An important contribution on this subject was presented in the dissertation of a former ITKE colleague, Dr. Jian-Min Li (Li 2017).

Nowadays, a growing number of efforts are focusing on the introduction of finite element formulations in DR for which co-rotational approaches are required. Co-rotational formulations are commonly used for beams and plates finite element formulations. The main idea is to describe the motion of an element by separating rigid body motions from strain causing local deformations on the element. In general, the integration of co-rotational approaches in DR implies more complex mathematics for the formulation of elements than those utilized in traditional implementation with only translational degrees of freedom. It also requires taking additional considerations for defining the structure of nodes, the equations of motion and the time integration. Because of this, it seems more appropriate at this moment to present DR on the basis of the full rigid body kinematics of nodes (translational and rotational). The following sections will introduce the method through the previously mentioned subjects.

3.2.3 *NODE STRUCTURE*

The motion of each node is governed by Newton's second law and can be decoupled into a translational and a rotational part. To calculate translational motions, each node has variables for describing its position \vec{x} , velocity \vec{v} and residual force \vec{F} that are stored via three-dimensional vectors. The nodal mass for each coordinate direction can be also described through individual values but, for simplicity, a single scalar value M is normally used for all directions. Conceptually, a node moves through a three-dimensional space with a certain velocity caused by a constant acceleration that is derived from all forces acting on the node at a certain point in time.

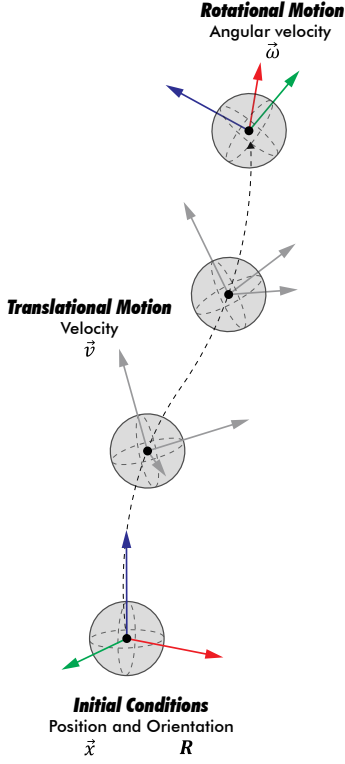


Fig. 3.3

Internal forces of two elastic bars computed through the links of a connected subset of nodes.

$$\vec{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad \vec{v} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \quad \vec{F} = \begin{bmatrix} F_1 \\ F_2 \\ F_3 \end{bmatrix}$$

In addition to this, variables for orientations, angular velocities, residual torques, and moment of inertia may be included to calculate rotational motions. In such cases, each node needs to have attached a local coordinate frame for describing its orientation (Figure 3.3). Node orientation \mathbf{R} can be considered to be a 3x3 matrix describing the three local rotational axes of the node. Analogous to translational motion, a node can rotate about any of those local axes with certain angular velocity caused by a constant angular acceleration that is derived from all the torques acting on the node at a certain point in time. Angular velocity and residual torques are considered to be three-dimensional rotational vectors. The moment of inertia, or angular mass, may be assumed to be isotropic when DR is only focused on the static equilibrium. So, it can be described through a single scalar value for the sake of simplicity.

$$\vec{\omega} = \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} \quad \vec{T} = \begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix} \quad \mathbf{R} = \begin{bmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{bmatrix}$$

Following this structure, internal forces and torques at each node can be locally computed through vector forms in which the number of operations is determined by the degrees of freedom of the node. This way the motion of a node can be treated independently and is only affected by what happens directly around it. Because of this, DR does not rely on solving problems associated with the assemblage of a global stiffness matrix which makes it easier to implement and maintain. Moreover, vector operations are relatively faster to calculate even if a large number of iterations is required to better approximate the solution.

3.2.4 FORCES

3.2.4.1 Translation

In the discretized system, the translational motion of a node based on Newtonian dynamics is governed by

$$(3.1) \quad \vec{F}_i^t = M_i^t \cdot \vec{a}_i^t$$

where \vec{F}_i^t denotes the translational residual force, M_i^t the lumped mass and \vec{a}_i^t the acceleration of the node in global coordinate system at t time. The residual force is the sum of all internal and external forces acting on a node at a given time (Figure 3.4). Thus,

$$(3.2) \quad \vec{F}_i^t = \sum \vec{f}_{int,i}^t + \sum \vec{f}_{ext,i}^t$$

Rearranging equation 3.1 to solve for acceleration, we get:

$$(3.3) \quad \vec{a}_i^t = \frac{\vec{F}_i^t}{M_i^t}$$

To ensure numerical stability, the lumped mass of a node needs to be adjusted at each time step based on the greatest direct stiffness S_i^t calculated on a node.

$$(3.4) \quad M_i^t = \frac{\Delta t^2}{2} \cdot S_i^t$$

3.2.4.2 Rotation

The equivalent rotational motion of a node based on Newton's second law of motion is governed by:

$$(3.5) \quad \vec{T}_i^t = I_i^t \cdot \vec{\alpha}_i^t + \vec{\omega}_i^t \times (I_i^t \cdot \vec{\omega}_i^t)$$

where \vec{T}_i^t denotes the resultant residual torque, I_i^t the moment of inertia and $\vec{\alpha}_i^t$ the angular acceleration of the node in global coordinate systems at t time. The calculation of rotational forces can be simplified by applying an isotropic inertia to the system (Li 2017) that reduces equation 3.5 to:

$$(3.6) \quad \vec{T}_i^t = I_i^t \cdot \vec{\alpha}_i^t$$

In a similar manner, the residual torque is the sum of all internal $\vec{\tau}_{int}$ and external torques $\vec{\tau}_{ext}$ acting on a node i .

$$(3.7) \quad \vec{T}_i^t = \sum \vec{\tau}_{int,i}^t + \sum \vec{\tau}_{ext,i}^t$$

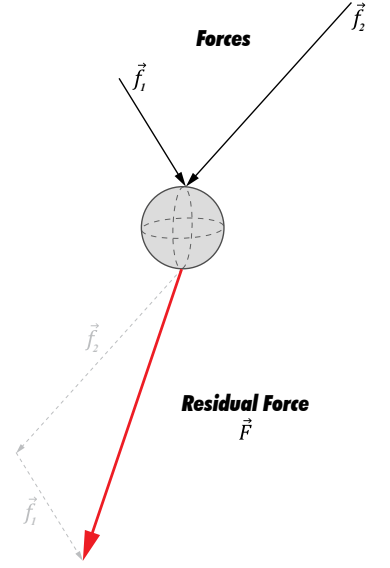


Fig. 3.4

Residual force as the sum of all internal and external forces acting on a particle.

Rearranging equation 3.6 for angular acceleration, we get:

$$\vec{\alpha}_i^t = \frac{\vec{T}_i^t}{I_i^t} \quad (3.8)$$

3.2.5 TIME INTEGRATION FOR TRANSLATION

In its classical form, DR focuses on computing the translational motion of individual nodes under the action of specific forces that make the totality of nodes to act as a unique system. The basic principle is to discretize this motion into a finite set of time-steps that are solved through an explicit time integration scheme. This means that the unknown nodal variables at the next time-step are calculated from the known variables of the current step. This way, the general procedure for each time-step is to solve nodal accelerations by calculating forces and masses via predefined elements. By integrating these accelerations, nodal velocities are solved and, by integrating velocities, the updated nodal positions are finally found. As a result, explicit time integration is said to be conditionally stable because stability and convergence is only guaranteed with small time increments. The two types of explicit integration schemes that are commonly used in DR are presented in this section.

3.2.5.1 Leapfrog Method

From the beginnings, many studies have reported the use of the Explicit Central Difference algorithm (Barnes 1999; Li and Knippers 2012; Adriaenssens et al. 2014) also known as the Leapfrog method, which will be the term used for the continuation of this dissertation, to integrate Newton's Equations of motions in DR. This implies that derivatives of positions are evaluated at mid-points of a given time interval. Thus, expressing acceleration and velocity in central difference form give the following equations:

$$\vec{a}_i^t = \frac{\vec{v}_i^{t+\Delta t/2} - \vec{v}_i^{t-\Delta t/2}}{\Delta t} \quad (3.9)$$

$$\vec{v}_i^{t+\Delta t/2} = \frac{\vec{x}_i^{t+\Delta t} - \vec{x}_i^t}{\Delta t} \quad (3.10)$$

Substituting equation 3.3 in equation 3.9 and rearranging

both equations gives the recurrent equations for calculating velocities and updating positions:

$$(3.11) \quad \vec{v}_i^{t+\Delta t/2} = \vec{v}_i^{t-\Delta t/2} + \frac{\Delta t}{M_i^t} \cdot \vec{F}_i^t$$

$$(3.12) \quad \vec{x}_i^{t+\Delta t} = \vec{x}_i^t + \Delta t \cdot \vec{v}_i^{t+\Delta t/2}$$

As a general remark, the Leapfrog integration is a second-order method that solves positions and velocities at asynchronous points in time. The method is not self-starting and requires forces to be calculated two times per each time-step.

An alternative algorithm that does not require any type of precomputation is the Symplectic-Euler integration (Senatore and Piker 2015). The method is categorized as semi-implicit because accelerations are expressed in forward difference and velocities in backward difference. Under these conditions, nodal positions are updated with the velocities of the next time-step $t+\Delta t$ which are in turn solved by using the forces of the current time-step t . The Symplectic-Euler method is a first-order method with a global error truncation $O(\Delta t)$ producing less accurate result when compared with the second-order Leapfrog method with a global error truncation of $O(\Delta t^2)$.

3.2.5.2 Runge-Kutta 4th Method

Higher order time integrations schemes may also be used with DR to obtain more accurate approximations of the result. One of the most popular high-order schemes is the fourth order Runge-Kutta method with a global error truncation of $O(\Delta t^4)$. The procedure consists of calculating four estimates of the nodal acceleration and velocity at different point of the interval defined by the given time-step: one at the beginning, two at the mid-points and one at the end (Figure 3.5). Those estimates are then averaged with specific weights and used to finally update the position and velocity at the next time step. This gives the following set of equations (Voesenek 2008):

$$(3.13) \quad \vec{v}_i^{t+\Delta t} = \vec{v}_i^t + \frac{\Delta t}{6} (\vec{K}_{T,1} + 2 \cdot \vec{K}_{T,2} + 2 \cdot \vec{K}_{T,3} + \vec{K}_{T,4})$$

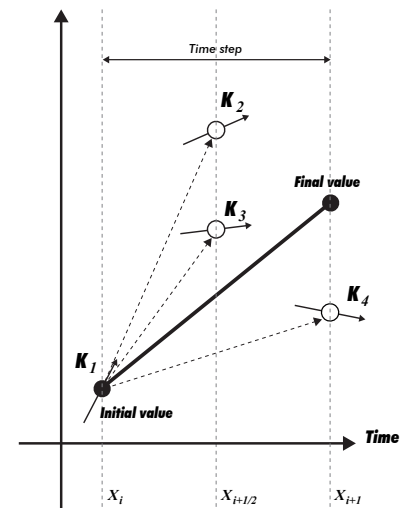


Fig. 3.5
Evaluation points at each time step of the Runge-Kutta 4th method.

$$\vec{x}_i^{t+\Delta t} = \vec{x}_i^t + \frac{\Delta t}{6}(\vec{L}_{T,1} + 2 \cdot \vec{L}_{T,2} + 2 \cdot \vec{L}_{T,3} + \vec{L}_{T,4}) \quad (3.14)$$

The different variables $\vec{K}_{T,j}$ and $\vec{L}_{T,j}$ are the corresponding estimates of the accelerations and velocities. The basic strategy is to approximate the acceleration as a function of the nodal position at the required time of the given interval by calculating the respective residual forces. Using Euler's equation, this acceleration is then used to update the subsequent velocity. It is therefore possible to approximate all the estimates through an incremental and alternately procedure. Thus,

$$\begin{aligned} \vec{L}_{T,1} &= \vec{v}^t & ; & & \vec{K}_{T,1} &= \vec{a}(\vec{x}) \\ \vec{L}_{T,2} &= \vec{v} + \frac{\Delta t}{2} \cdot \vec{K}_{T,1} & ; & & \vec{K}_{T,2} &= \vec{a}\left(\vec{x} + \frac{\Delta t}{2} \cdot \vec{L}_{T,1}\right) \\ \vec{L}_{T,3} &= \vec{v} + \frac{\Delta t}{2} \cdot \vec{K}_{T,2} & ; & & \vec{K}_{T,3} &= \vec{a}\left(\vec{x} + \frac{\Delta t}{2} \cdot \vec{L}_{T,2}\right) \\ \vec{L}_{T,4} &= \vec{v} + \Delta t \cdot \vec{K}_{T,3} & ; & & \vec{K}_{T,4} &= \vec{a}\left(\vec{x} + \Delta t \cdot \vec{L}_{T,3}\right) \end{aligned} \quad (3.15)$$

An RK4 iteration is more expensive to compute than a Leapfrog iteration in which only two evaluations are used per iteration. Special attention should be taken to this additional computational cost since it can rapidly compromise the development of real-time applications according to the size of the discrete model.

3.2.6 TIME INTEGRATION FOR ROTATION

The co-rotational formulation in DR is built on the accumulation of small rotational transformations on nodal orientations under the action of torques during a period of time to solve the rotational motion of a node. Considering that rotation is part of the special orthogonal group $SO(3)$, a specific type of parameterization is required to be numerically represented (Lázaro et al. 2016). A common solution for this problem is the Rodrigues formula through which a rotational vector \vec{r} is used to calculate the linear transformation matrix \mathbf{R}_T to perform a rotation. In this sense, this parameterization describes a rotation on the basis of a vector in which the magnitude represents the

angle $\theta = |\vec{r}|$ and its corresponding unit vector $\hat{e} = \vec{r}/|\vec{r}|$ defines the axis of rotation. The relationship between the rotational vector \vec{r} and the transformation matrix \mathbf{R}_T is then given by:

$$(3.16) \quad \mathbf{R}_T(\vec{r}) = \begin{bmatrix} \cos \theta + e_x^2(1 - \cos \theta) & e_x e_y(1 - \cos \theta) - e_z \sin \theta & e_x e_z(1 - \cos \theta) + e_y \sin \theta \\ e_x e_y(1 - \cos \theta) + e_z \sin \theta & \cos \theta + e_y^2(1 - \cos \theta) & e_y e_z(1 - \cos \theta) - e_x \sin \theta \\ e_x e_z(1 - \cos \theta) - e_y \sin \theta & e_z e_y(1 - \cos \theta) + e_x \sin \theta & \cos \theta + e_z^2(1 - \cos \theta) \end{bmatrix}$$

In DR, this rotational vector can be approximated in a finite difference form based on the angular velocity $\vec{\omega}_i$ of the i th node at a given time-step. Therefore, if the rotational vector \vec{r} is known and expressed in global coordinates, it is possible to solve the nodal orientation at the next time-step $\mathbf{R}_i^{t+\Delta t}$ by establishing a relation between the current orientation \mathbf{R}_i^t and the transformation matrix \mathbf{R}_T . Thus, for updating nodal orientations, we get

$$(3.17) \quad \mathbf{R}_i^{t+\Delta t} = \mathbf{R}_T(\vec{r}) \cdot \mathbf{R}_i^t$$

3.2.6.1 Leapfrog Method

The Leapfrog method can also be used to integrate the equations of motion for rotation. In doing so, the rotational vector is approximated by evaluating the angular velocity at the midpoint of a given time interval as suggested by Li (Li 2017). Thus, expressed in a central difference form, we get:

$$(3.18) \quad \vec{r} = \int_t^{t+\Delta t} \vec{\omega}_i \cdot dt \approx \vec{\omega}_i^{t+\frac{\Delta t}{2}} \cdot \Delta t$$

Substituting equation 3.18 in equation 3.17 to include the mean angular velocity term, gives the recurrent equation for updating orientations:

$$(3.19) \quad \mathbf{R}_i^{t+\Delta t} = \mathbf{R}_T(\Delta t \cdot \vec{\omega}_i^{t+\frac{\Delta t}{2}}) \cdot \mathbf{R}_i^t$$

Similar to the translation motion, the angular velocity can be updated in terms of angular accelerations as:

$$(3.20) \quad \vec{\omega}_i^{t+\Delta t/2} = \vec{\omega}_i^{t-\Delta t/2} + \Delta t \cdot \vec{\alpha}_i^t$$

Substituting equation 3.8 in equation 3.20 gives the recurrent equations for calculating angular velocities:

$$\vec{\omega}_i^{t+\Delta t/2} = \vec{\omega}_i^{t-\Delta t/2} + \frac{\Delta t}{I_i^t} \cdot \vec{T}_i^t \quad (3.21)$$

3.2.6.2 Runge-Kutta 4th Method

For the Runge-Kutta 4th integration, the rotational vector is approximated by the weighted average of the corresponding angular velocities at the given points of the interval defined by the time-step.

$$\vec{r} \approx \frac{\Delta t}{6} (\vec{L}_{R,1} + 2 \cdot \vec{L}_{R,2} + 2 \cdot \vec{L}_{R,3} + \vec{L}_{R,4}) \quad (3.22)$$

Substituting equation 3.22 in equation 3.17 to include the weighted average of the estimates, gives the recurrent equation for updating orientations:

$$\mathbf{R}_i^{t+\Delta t} = \mathbf{R}_T \left(\frac{\Delta t}{6} (\vec{L}_{R,1} + 2 \cdot \vec{L}_{R,2} + 2 \cdot \vec{L}_{R,3} + \vec{L}_{R,4}) \right) \cdot \mathbf{R}_i^t \quad (3.23)$$

The angular velocity can then be updated in terms of the weighted average of the angular accelerations evaluated at the same points of the time interval that are defined as follow:

$$\vec{\omega}_i^{t+\Delta t} = \vec{\omega}_i^t + \frac{\Delta t}{6} (\vec{K}_{R,1} + 2 \cdot \vec{K}_{R,2} + 2 \cdot \vec{K}_{R,3} + \vec{K}_{R,4}) \quad (3.24)$$

Similar to the procedure applied for the translational motion, the required estimates for the angular accelerations are calculated as a function of the orientations at the required time of the given interval by calculating the respective residual torques. These estimates are used to solve the following angular velocity.

$$\begin{aligned} \vec{L}_{R,1} &= \vec{\omega}^t & ; & & \vec{K}_{R,1} &= \vec{\alpha}(\mathbf{R}_i^t) \\ \vec{L}_{R,2} &= \vec{\omega} + \frac{\Delta t}{2} \cdot \vec{K}_{R,1} & ; & & \vec{K}_{R,2} &= \vec{\alpha} \left(\mathbf{R}_T \left(\frac{\Delta t}{2} \cdot \vec{L}_{R,1} \right) \cdot \mathbf{R}_i^t \right) \\ \vec{L}_{R,3} &= \vec{\omega} + \frac{\Delta t}{2} \cdot \vec{K}_{R,2} & ; & & \vec{K}_{R,3} &= \vec{\alpha} \left(\mathbf{R}_T \left(\frac{\Delta t}{2} \cdot \vec{L}_{R,2} \right) \cdot \mathbf{R}_i^t \right) \\ \vec{L}_{R,4} &= \vec{\omega} + \Delta t \cdot \vec{K}_{R,3} & ; & & \vec{K}_{R,4} &= \vec{\alpha}(\mathbf{R}_T(\Delta t \cdot \vec{L}_{R,3}) \cdot \mathbf{R}_i^t) \end{aligned} \quad (3.25)$$

3.2.7 DAMPING TECHNIQUES

DR has two main approaches to damp the movement of

nodes in order to attain a steady state solution. In its more classical form, the method uses viscous damping to slow-down the motion of a node by introducing drag constants $A = (1 - C/2)/(1 + C/2)$ and $B = (1 + A)/2$ on the recurrent equations for updating velocity (Adriaenssens et al. 2014). The variable C refers to a constant damping factor for the entire structure. Taking the case of the Leapfrog method, equations 3.11 and equations 3.21 can be rearranged so that:

$$(3.26) \quad \vec{v}_i^{t+\Delta t/2} = A \cdot \vec{v}_i^{t-\Delta t/2} + B \cdot \frac{\Delta t}{M_i} \cdot \vec{F}_i^t$$

$$(3.27) \quad \vec{\omega}_i^{t+\Delta t/2} = A \cdot \vec{\omega}_i^{t-\Delta t/2} + B \cdot \frac{\Delta t}{I_i} \cdot \vec{T}_i^t$$

An alternative solution for viscous damping is the use of kinetic damping in which all the nodal velocities are reset to zero when a peak of kinetic energy is detected. The technique of kinetic damping relaxation was proposed by Cundall for problems on unstable rock mechanics (Cundall 1976) and then extended by Barnes to address form-finding problems of tensile structures (Barnes 1988). Given the absence of viscous damping, the process of detecting peaks and resetting velocities must be repeated until a steady-state solution is reached. Based on the Leapfrog method, the total kinetic energy KE is calculated by:

$$(3.28) \quad KE = \sum_{i=1}^n M_i^t \cdot |\vec{v}_i^{t+\Delta t/2}| + \sum_{i=1}^n I_i^t \cdot |\vec{\omega}_i^{t+\Delta t/2}|$$

The simplest way to proceed, when a kinetic peak is detected at time, is to stop the process for resetting velocities and then restart from the updated nodal positions (Du Peloux 2017). This is sufficiently convenient when translation and rotational motions are computed in order to simplify the damping process and avoid additional iterations. It should be noted that when a peak is detected, the true kinetic energy peak has already been passed at some earlier time t^* . Because the time-step used is relatively small to avoid instabilities, it can be assumed that the updated positions are relatively closed to the positions at time t^* . However, for systems in which only translational motion is calculated, the effect of the kinetic damping can be maximized by estimating nodal positions at the time when the true kinetic peak

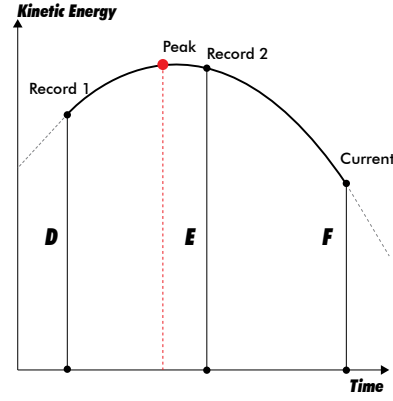


Fig. 3.6
Evaluation points for kinetic damping.

occurred. Through a parabolic interpolation using the current (F) and a record of the two previous values (D and E) of the total kinetic energy (Figure 3.6), the elapsed time t^* since the energy peak can be calculated in terms of differences between previous $G = D - E$ and current $H = E - F$ kinetic energies (Barnes 1999).

$$\delta t^* = \Delta t \cdot \frac{H}{H - G} = \Delta t \cdot q \quad (3.29)$$

Nodal positions are then corrected to the interpolated position at time t^* by:

$$\vec{x}_i^{t^*} = \vec{x}_i^{t+\Delta t} - \Delta t \cdot \vec{v}_i^{t+\Delta t/2} - \delta t^* \cdot \vec{v}_i^{t-\Delta t/2} \quad (3.30)$$

This equation can be rewritten using equations 3.11, 3.12 and 3.29 so that:

$$\vec{x}_i^{t^*} = \vec{x}_i^{t+\Delta t} - \Delta t \cdot (1 + q) \cdot \vec{v}_i^{t+\Delta t/2} + \frac{\Delta t^2}{2} \cdot q \cdot \frac{\vec{F}_i^t}{M_i^t} \quad (3.31)$$

3.2.8 ELEMENTS

At this point, it is clear that the basis of DR is the motion of nodes caused by the action of internal and external forces that are applied during short time intervals. However, we didn't describe how those forces were calculated. While external forces are relatively simpler to define, internal forces are calculated via elements that are formulated on the basis of mechanical principles. As a general concept, an element is an ordered set of nodes in which different forces may be calculated to produce a grouped nodal motion that approximates the physical behavior of a particular mechanism. This way, elements permit to direct the way nodes relate to each other and constitute the fundamental mechanisms to hold everything together. It is through the combination of nodal motions and multiple elements formulations that the dynamic behavior of a global system can be traced until reaching a steady state. In general, there are many types of linear and surface elements with specific topologic dimensionalities that are commonly used for form-finding applications in DR. In the following, an overview of the different elements used on the research of bending-active and textile hybrid system is presented.

3.2.8.1 Elastic Bar

For form-finding and analysis, the behavior of a cable-net structure is modeled through linear elements. A cable-net structure is conceived as a discrete network assembly in which each link is associated with an elastic bar element permitting the calculation of forces. More specifically, an elastic bar is classified as a linear element that keeps two nodes connected by means of axial forces which, in the case of tensile structures, is only tension. According to Barnes (Barnes 1999), the tension T_m of an elastic bar m connecting a node i with an adjacent node j at a time t is calculated by:

$$(3.32) \quad T_m^t = T_m + K_m \cdot (L_m^t - L_m)$$

where L_m^t is the current length of the element and L_m is its initial length. In form-finding, the factors T_m and K_m are the respective geometric and elastic stiffnesses of the elastic bar that are used by designers to control the equilibrium geometry and the distribution of prestress. In the case of load analysis, T_m is the tension derived from the form-found geometry and the elastic stiffness is defined by $K_m = EA/L_m$ where E is the Young's modulus and A the cross-sectional area of the element. On this basis, the internal axial force \vec{f}_m of the elastic bar connecting to i is resolved by:

$$(3.33) \quad \vec{f}_{m,i}^t = \frac{T_m^t}{L_m^t} \cdot (\vec{x}_j^t - \vec{x}_i^t)$$

Hence, the residual force acting on the same node is calculated by adding the axial forces of all the connected elastic bars and external forces as:

$$(3.34) \quad \vec{F}_i^t = \sum_m \vec{f}_{m,i}^t + \sum \vec{f}_{ext,i}^t$$

To define the nodal mass M_i^t of the node, the greatest direct stiffness is obtained by:

$$(3.35) \quad S_i^t = \sum_m \left(\frac{EA}{L_m} + g \cdot \frac{T_m}{L_m^t} \right)$$

where g is a constant factor that may be used to calibrate the contribution of the geometric stiffness when defining the nodal mass.

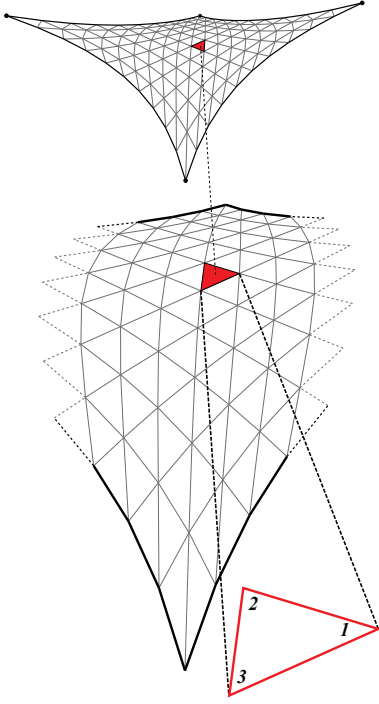


Fig. 3.7
Triangular mesh structure of a textile membrane.

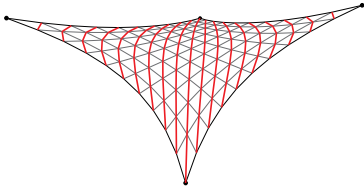


Fig. 3.8
Geodesic paths.

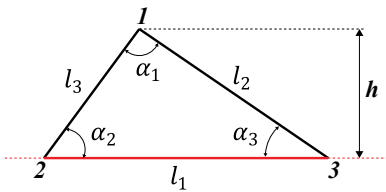


Fig. 3.9
Parameters of the triangular element.

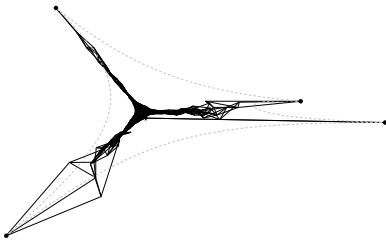


Fig. 3.10
Soap film behavior without boundaries.

3.2.8.2 Constant Strain Triangle

Based on the analogy of a soap film, the surface tension of a prestressed membrane structure can be modeled with three-nodal triangular surface elements called constant strain triangle (CST) in which tension forces between nodes are calculated from the links of the element. To do this, a triangular mesh is used to describe the discrete configuration of the membrane structure (Figure 3.7). In this thesis, it is more convenient to assume that, at any time, the first side of each triangular element is always aligned with a warp-control line following a geodesic path that is associated with the warp direction of the textile membrane (Figure 3.8). This condition permits to simplify the equations required for computing forces but imposes additional constraints for the generation of the initial mesh. In form-finding, tension forces in triangular elements are calculated from prescribed stresses in the warp σ_x and weft σ_y direction. This process is commonly known as stress controlled form-finding. This means that designers can set different stresses in both directions in order to control the shape of the membrane. In the special case of setting equal warp and weft stresses, a true minimal surface can be achieved with uniform stress distribution on the surface. Using Barnes's equations (Barnes 1999), the tensions T associated with each side l of the triangular element due to direct stresses are calculated with:

$$\begin{aligned} T_1 &= \frac{h}{2} (\sigma_x - \sigma_y) + \frac{\sigma_y l_1}{2 \tan \alpha_1} \\ T_2 &= \frac{\sigma_y l_2}{2 \tan \alpha_2} \quad ; \quad T_3 = \frac{\sigma_y l_3}{2 \tan \alpha_3} \end{aligned} \quad (3.36)$$

where all the terms in equation 3.36 are defined on the basis of the deformed state of the triangular element as in figure 3.9.

At this point, it is important to note that, similar to the soap film analogy, modeling the behavior of a membrane with this triangular element always requires a closed boundary to avoid that the surface shrinks until disappearing (Figure 3.10). This is achieved by adding constraints on model's boundary such as elastic bars to simulate edge-cables, anchored nodes to determine

fixations or a combination of both. Another important aspect is that warp-control lines, also called G-Strings (Geodesic strings), require additional tension forces to preserve its geodesic path while triangles deform. It should be noted that these additional forces only act on the plane of the surface and have no effects in the global geometry. They only prevent nodes to be randomly distributed throughout the surface and that a mesh may not be developed without a clear weft and warp directionality (Figure 3.11).

On the other hand, for analysis, tensions need to be calculated on the basis of stress/strain relationships from the already form-found geometry. To do this, strains from the prestressed state are calculated by:

$$(3.37) \quad \begin{aligned} \epsilon_x &= \frac{l_1^t}{l_1} - 1 \quad ; \quad \epsilon_y = \frac{h^t}{h} - 1 \\ \gamma &= \frac{1}{\tan \alpha_2^t} - \frac{1}{\tan \alpha_2} \cdot \left(\frac{1 + \epsilon_x}{1 + \epsilon_y} \right) \end{aligned}$$

From strains, direct and shear stresses can be obtained by:

$$(3.38) \quad \begin{aligned} \sigma_x &= d_{xx} \cdot \epsilon_x + d_{xy} \cdot \epsilon_y \\ \sigma_y &= d_{yx} \cdot \epsilon_x + d_{yy} \cdot \epsilon_y \\ \tau &= G \cdot \gamma \end{aligned}$$

where d_{xx} , d_{yy} , d_{yx} and d_{xy} are the orthotropic material constants.

Shear stiffness of textiles is relatively low and is neglected during form-finding. However, for load analysis tension forces due to shear stresses may be also calculated on the links to avoid the distortion of the mesh. According to Barnes (Barnes 1999), these additional tensions T due to shear stresses and associated with each side l of the triangular element are calculated by:

$$(3.39) \quad \begin{aligned} T_1 &= \frac{\tau l_1}{2} - \frac{\tau h}{2 \tan \alpha_2} \\ T_2 &= \frac{\tau l_2}{2} \quad ; \quad T_3 = \frac{\tau l_3}{2} \end{aligned}$$

Concerning numerical stability, the greatest stiffness for each node is calculated according to equation 3.35. In form-

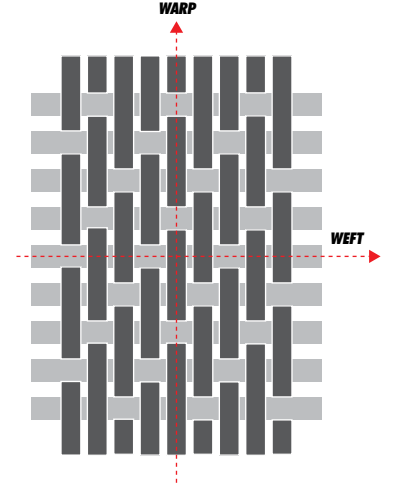


Fig. 3.11

Weft and warp directionality of a textile fabric.

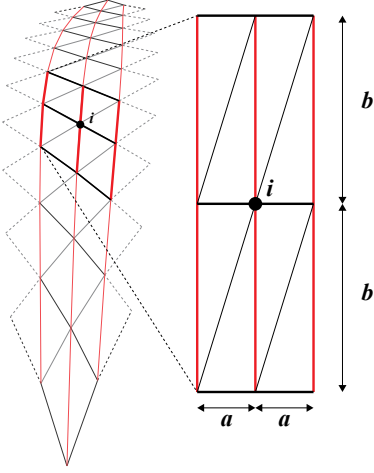


Fig. 3.12
Parameters of the deformed triangle to calculate the corresponding geometric stiffness.

finding, the geometric stiffness of a single triangle element at a node i is:

$$\frac{T}{L} = a \cdot \sigma_x \cdot \frac{2}{b} + b \cdot \sigma_y \cdot \frac{2}{a} \quad (3.40)$$

where all the terms in equation 3.40 are defined on the basis of the deformed state of the triangular element as in (Figure 3.12). In addition to this, the geometric and elastic stiffness of the rest of elements connecting to this node must be added.

3.2.8.3 Beam 3DOF

The most straightforward approach to numerical model the form and behavior of linear elastic elements incorporating bending moments and shear forces is through the use of spline beams. Originated from the work of Pian (Pian et al. 1967) regarding the dynamic buckling of two dimensional structures, the formulation of the spline beam element was proposed by Barnes and Adriaenssens (Barnes 1988; Barnes 1999; Adriaenssens and Barnes 2001) for three-dimensional modeling problems of elastic gridshells and battened membranes. A spline beam is therefore defined as a linear element created from a set of sequentially connected nodes in which the bending effect is calculated by only using translational degrees of freedom. In its simplest form, this scheme requires a set of three consecutive nodes i , j and k that are connected with two links a and b , as shown in (Figure 3.13). It is assumed that the deformation of these two adjacent links provokes the nodes to lie on a circular arc built on the reference plane ijk . Accordingly, the radius of curvature of this arc and its bending moment can be defined by

$$R = \frac{L_c}{2 \sin \alpha} \quad ; \quad M = \frac{EI}{R} \quad (3.41)$$

where L_c is the distance between nodes ik , E the Young's modulus, I the second moment of area and α the angle difference between the initial and deformed state. On this basis, axial forces on the links are calculated by using equation 3.33 and shear forces according to the bending moment M at node j are calculated with:

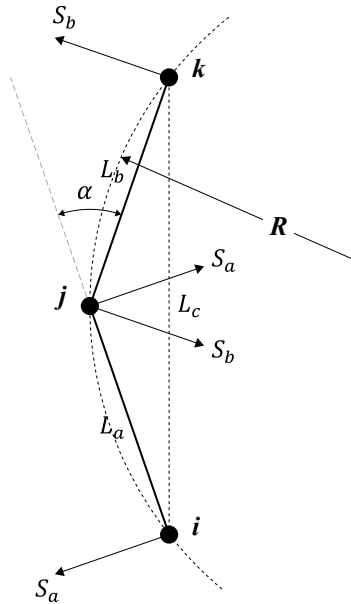


Fig. 3.13
Parameters of a simplified beam with 3DOF per node.

$$(3.42) \quad S_a = \frac{2EI \sin \alpha}{L_a L_c} \quad ; \quad S_b = \frac{2EI \sin \alpha}{L_b L_c}$$

where L_a and L_b are the distances between nodes ij and jk . Shear forces S_a and S_b act normal to the links a and b in the reference plane ijk and are applied to node ij and jk respectively (Barnes et al. 2013). To incorporate the stiffness component due to bending for calculating the mass of node j , an additional term is added to equation 3.35

$$(3.43) \quad S_j^t = \sum_m \left(\frac{EA}{L_m} + g \cdot \frac{T_m}{L_m^t} \right) + \sum_{a,b} \left(\frac{4EI}{L^3} \right)$$

This element may be classified as a simplified formulation for beam elements permitting faster calculations and reducing the complexity for implementation. The accuracy of the model and the smoothness of the curvature only depend on the finite set of consecutive nodes and, by consequence, on the number of links constituting the spline beam element. The scheme only takes into account in-plane bending effects on tubular elements with symmetric cross-sections. For more general cases where torsional and twisting effects play a more significant role in form-finding and, certainly, for analysis, rotational degrees of freedom need to be activated.

3.2.8.4 Beam 6DOF

The introduction of rotational degrees of freedom for beam elements can make the form-finding process more complicated but the solution will be more accurate because, under these conditions, twisting and torsional effects can be taken into account. An Euler-Bernoulli beam element formulation for DR was proposed by Li (Li 2017) in which internal forces are computed through bend-ends positions and orientations. In this model, a beam element is composed of two beam-ends that are directly associated with the positions of two nodes. It is important to note that the orientation of a beam-end will not coincide with the orientation of its corresponding node when multiple beams are coupled at this node. For this reason, each beam-end is defined with a specific orientation that co-rotates as a rigid-body with the position of the corresponding node. By defining specific initial beam-ends orientations, the pre-stress of an initial straight joint can be created (The procedure to calculate the initial beam-end orientations is described in Appendix A)

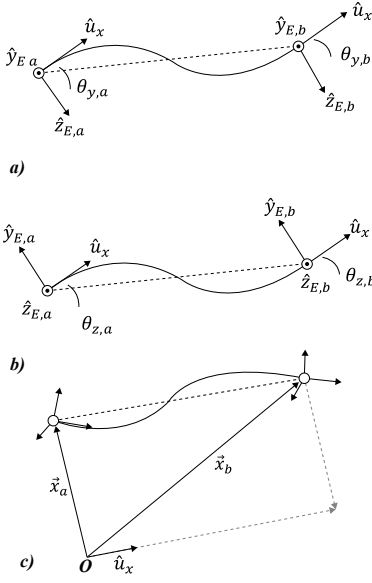


Fig. 3.14

Definition of a) included angles $\theta_{y,a}$ and $\theta_{y,b}$, b) included angles $\theta_{z,a}$ and $\theta_{z,b}$ and the x-direction \hat{u}_x of the orientation of the beam element.

To calculate internal forces, the orientations of the two beam-ends a and b need to be converted from global coordinates to local element coordinates. To do this transformation, it is necessary to define the orientation of the beam element. This orientation is calculated from the positions and orientations of both beam-ends in which the y-directions \hat{u}_y and z-directions \hat{u}_z are computed from the average of the two beam-ends orientations and the x-direction \hat{u}_x is always parallel to center line of the beam (The procedure to calculate the beam orientation is described in Appendix A). On this basis, it is possible to calculate the included angles that are used to compute bending moments, shears and axial forces on a deformed beam element. Their corresponding geometric definitions are shown in (Figure 3.14) and are calculated by (Li and Knippers 2012):

$$\theta_{z,j} = \hat{u}_x \cdot \hat{y}_{E,j} \quad ; \quad j = a, b$$

$$\theta_{y,j} = -\hat{u}_x \cdot \hat{z}_{E,j} \quad ; \quad j = a, b \quad (3.44)$$

$$\theta_{x,b} - \theta_{x,a} = (\hat{y}_{E,a} \cdot \hat{z}_{E,b} - \hat{y}_{E,b} \cdot \hat{z}_{E,a})$$

$$\hat{u}_x = \frac{(\vec{x}_b - \vec{x}_a)}{|\vec{x}_b - \vec{x}_a|} \quad (3.45)$$

Hence, the equations for computing internal forces and torques as functions of included angles and extended lengths are (Li 2017):

$$\begin{aligned} f_{x,a} &= \frac{EA \cdot (L^t - L)}{L} \quad ; \quad f_{x,b} = -f_{x,a} \\ f_{y,a} &= \frac{-6EI_z}{L^2 \cdot (\theta_{z,a} + \theta_{z,b})} \quad ; \quad f_{y,b} = -f_{y,a} \\ f_{z,a} &= \frac{6EI_y}{L^2 \cdot (\theta_{y,a} + \theta_{y,b})} \quad ; \quad f_{z,b} = -f_{z,a} \\ \tau_{x,a} &= \frac{GJ}{L \cdot (\theta_{x,b} - \theta_{x,a})} \quad ; \quad \tau_{x,b} = -\tau_{x,a} \\ \tau_{y,a} &= \frac{-2EI_y}{L \cdot (2\theta_{y,a} + \theta_{y,b})} \quad ; \quad \tau_{y,b} = \frac{-2EI_y}{L \cdot (\theta_{y,a} + 2\theta_{y,b})} \\ \tau_{z,a} &= \frac{-2EI_z}{L \cdot (2\theta_{z,a} + \theta_{z,b})} \quad ; \quad \tau_{z,b} = \frac{-2EI_z}{L \cdot (\theta_{z,a} + 2\theta_{z,b})} \end{aligned} \quad (3.46)$$

where E is the Young's modulus, G the shear modulus, A the cross-sectional area, I_y the second moment of area around the y-axis, I_z the second moment of area around the z-axis and L the initial length of the beam.

For the translation motion of a node i , the corresponding beam-end a has three different direct stiffness values associated with each direction. Because we are only considering lumped masses, numerical stability is secured by using the maximum direct stiffness value k among the three as:

$$(3.47) \quad k_a = \max \left\{ \frac{E_a A_a}{L_a}, \frac{12E_a I_{z,a}}{L_a^3}, \frac{12E_a I_{y,a}}{L_a^3} \right\}$$

By summing all the largest direct stiffnesses of beam-ends associated with node i at a time t , we get the direct stiffness of the node for calculating its nodal mass:

$$(3.48) \quad S_i^t = \sum_a k_a$$

For the rotational motion, the corresponding beam-end has also three different rotational stiffness values associated with each direction. The scalar inertia is then set to the largest rotational stiffness value k_τ among the three as:

$$(3.49) \quad k_{\tau,a} = \max \left\{ \frac{G_a J_a}{L_a}, \frac{4E_a I_{y,a}}{L_a}, \frac{4E_a I_{z,a}}{L_a} \right\}$$

Hence, the rotational stiffness of the node for calculating its nodal inertia:

$$(3.50) \quad I_i^t = \sum_a k_{\tau,a}$$

3.3 PARTICLE SYSTEMS (PS)

3.3.1 DESCRIPTION

Particle-Systems (PS) is one of the most effective and widely adopted physically-based method for modeling discrete, rigid and deformable bodies in computer graphics. Similar to DR, the method is also focused on finding a steady state solution for

a dynamic problem involving discrete models in which nodes are called particles. The main difference is that PS supports the use of more schemes for time integration and also different dynamics to produce the motion of particles. For this reason, DR may be perceived as a closed form of PS. The method was originally proposed by Reeves in 1983 (Reeves 1983) to model the dynamic behavior of fire, gasses and liquids over finite periods of time based on the idea of using the motion of a disconnected set of particles and their interaction with a predefined environment. Craig Reynolds (Reynolds 1987) was among the firsts to conceive the idea of coupled particle system in which particles interact with each other to produce some kind of emergent behavior. By introducing new terminology, he described how complex aggregate behaviors can be produced from the motion and interactions of a large number of small independent particles (Breen et al. 1994; Eberhardt et al. 1996). By 1987, Terzopoulos et al. (Terzopoulos et al. 1987) extended the method to model the physical behavior of deformable bodies. Here, the key point was to connect particles by means of “springs” that served to incorporate simplified principles of elasticity and simulate behaviors like tension and rigidity. This specific type of connected particle model is also referred as Particle-Spring or Mass-Spring System.

Throughout the years, PS has become highly popular among all the related research fields of computer graphic. In this context, special attention is dedicated to the simulation of deformable bodies, such as cloth hair and elastics in the animation and gaming industry. For this reason, multiple efforts are continuously conducted to expand the catalogue of resourceful strategies to produce the motion and interaction of particles and simulate a wide range of realistic physical behaviors. Those strategies can go from simplified approaches fully oriented towards interactivity to highly accurate version based on complex mechanism properly derived from finite element formulations. However, it was not until the beginnings of the 2000s that different efforts (Killian and Ochsendorf 2005; Martini 2001) were conducted on this method to develop interactive structural design tools. From then on, PS simulations have been gradually integrated within computational design practices to facilitate the exploration of ideas and concepts of new structural typologies,

especially during early design stages.

3.3.2 *EVOLUTION OF THE METHOD*

Contrary to DR, PS is not entirely associated with a specific type of time integration scheme. The initial work of Terzopoulos et al. (Terzopoulos and Fleischer 1988a, 1988b) with connected particle models was formulated on the basis of implicit time integration schemes. Further important contributions (Volino et al. 1995; Volino et al. 1996) were mainly associated with explicit solvers requiring small time-steps to secure numerical stability, in a similar way to DR. It was not until 1998 that Baraff and Witkin (Baraff and Witkin 1998) reintroduced the principles of implicit solvers for the research on deformable bodies like cloth. The use of implicit solvers was proposed as an alternative solution to reduce the computational cost of calculations when a large number of particles was required to realistically simulate certain types of physical behaviors such as wrinkles and folds on cloth.

The implementation of implicit solvers is not as straightforward as explicit ones mainly because the given functions associated with the motion of particles are evaluated with unknown derivatives of a later state. To compute those unknowns, a large system of equations, generally non-linear, needs to be solved by using a matrix or iterative technique at each time step. This key aspect permits to remove the instability problems exhibited in explicit integrations and makes the solution stable for arbitrary large values of the time-step. With a larger time-step, the equilibrium condition can be reached with a reduced number of iterations, but each iteration is more expensive to compute due to the additional layer of equations. Under these conditions, the appropriateness of implicit or explicit time integration schemes largely depends on the type of modeling problem. A strategy to improve the convergence rate and numerical stability of explicit schemes is to decrease the strength of forces (fictitious forces) that leads towards a type of bouncy behavior. The unconditional stability of implicit methods makes them a better option for the numerical solution of stiff systems.

Another important aspect that should be considered is the

different types of dynamics that are used to produce the motion of particles. This is associated with the rate of accuracy and interactivity that simulations need to have among the different fields of application. Until recently, common approaches for PS have been developed on the basis of force dynamics that, based on Newton's second law of motion, directly relates the accumulation of internal and external forces to accelerations by using a mass term. This permits to compute forces from mathematical formulations that are rigorously derived from continuum mechanical principles. By using a time integration scheme, the resulting accelerations are used to update the velocities and positions of all the particles. These types of dynamics, called Force-Based Dynamics (FBD), are mainly utilized for the development of models in which mechanical accuracy plays a significant role for the simulation.

On the other hand, Constraint-Based Dynamics (CBD) have been proposed to gain more control over the simulation and facilitate the manipulation of the model. These conditions are of special interest for the development of interactive environments and real-time applications in the gaming industry. High-end products like PhysX, Nvidia Flex, Bullet and Maya nCloth have adopted this method due to its simplicity and versatility. The idea of using constraint functions was already addressed in the work of (Baraff and Witkin 1998). Recently, Müller et al. (Müller et al. 2007) proposed a method, named position-based, that bypasses acceleration and velocity layers to directly modify the positions of particles on the basis of previous positions and estimated projections calculated through geometrical constraint functions. This method is to some extent related to the general Variational Implicit Euler (VIE) scheme that uses potential energies instead of forces (Liu et al. 2013). Accordingly, numerical stability is determined by a constraint function used to compute a potential energy and not by the size of the time step. In the initial formulation proposed by Müller et al. (Müller et al. 2007), the system of constraint is solved one after each other using a Gauss-Seidel-type of iteration. The main drawback of this formulation is that stiffness and deformations are affected by the number of iterations and by the order of solving the system of constraints (Bouaziz et al. 2014). In a general case, the great advantage of CBD is that it permits to speed up and unconditionally stabilize

the numerical process but limits accuracy to a visual plausibility (Bender et al. 2014; Bender et al. 2012). Such limited accuracy is associated with the lack of continuous-mechanic principles for defining constraint functions which is an issue that has been recently addressed by Bouaziz et al. (Bouaziz et al. 2014). The following sections will introduce a general overview of this method.

3.3.3 PARTICLE STRUCTURE

The structure of a particle is decomposed into a translational and a rotational part. In this dissertation, both parts are presented in a general local form to allow forces and constraints to be used simultaneously and avoiding the assemblage of global matrices. The general translational structure of a particle is composed by a scalar value M defining its mass and two vector variables in three-dimensional space for describing its position \vec{x} and velocity \vec{v} . To solve translational forces, a resultant vector \vec{F} is included to accumulate all the internal and external vector forces that are applied on the particle. In the literature of PS, such resultant vector is referred as force accumulator which is equivalent to the residual force term in DR. Note that a particle has the same structure of a DR node if its motion is only provoked by forces. On the other hand, translational constraints are solved with a resultant vector \vec{P}_T that accumulates all the local projections derived from the constraints functions that are linked to the particle.

$$\vec{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad \vec{v} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$$

$$\vec{F} = \begin{bmatrix} F_1 \\ F_2 \\ F_3 \end{bmatrix} \quad \vec{P}_T = \begin{bmatrix} P_{T,1} \\ P_{T,2} \\ P_{T,3} \end{bmatrix}$$

The same principle is applied for the rotational part. Its structure is composed of a local coordinate frame describing its orientation through a 3x3 matrix R and a rotational three-dimensional vector describing the angular velocity $\vec{\omega}$. Torques and weighted rotational constraints are then accumulated in two additional rotational vectors \vec{T} and \vec{P}_R . Two scalar variables are also included to store the moment of inertia I and the sum of all

the weights of the constraints W_R .

$$\mathbf{R} = \begin{bmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{bmatrix} \quad \vec{\omega} = \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix}$$

$$\vec{T} = \begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix} \quad P_R = \begin{bmatrix} P_{R,1} \\ P_{R,2} \\ P_{R,3} \end{bmatrix}$$

This structure permits to solve the entire motion of a particle through a local procedure that only uses vector operations. This means that all the required operations can be treated independently. Furthermore, this particle structure is proposed to permit the simultaneous or individual use of forces and potential-energies.

3.3.4 POTENTIAL ENERGIES

By comparing the literature originated from computational graphics and engineering fields, it is to some extent difficult to establish the difference between PS and DR. Veenendaal and Block (Veenendaal and Block 2011) categorized both methods as dynamics and proposed to differentiate them based on the use of elements formulations and time integration schemes. Through a broader perspective, we can in fact consider PS as a generalized form of DR permitting to not only uses forces and explicit solvers but, also, potential-energies and implicit solvers. In the previous section, the basic formulations of force elements for conducting form-finding processes of bending-active and textile hybrid structures have been presented. Therefore, to not over-extend this chapter, their formulations as potential-energies are not going to be re-addressed one-by-one because these ones can be easily derived by describing the general relation between a force and a potential energy (Figure 3.15).

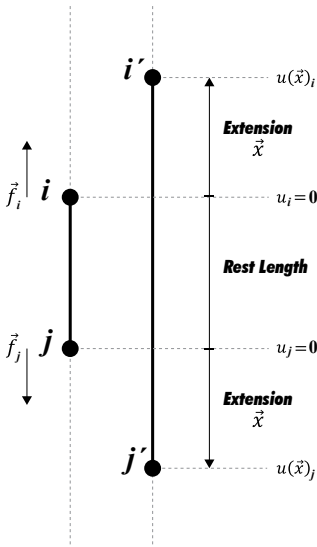


Fig. 3.15
Elastic potential energy

From the previous section, we known that the calculations of forces only depend on the position of particles and are associated with link elements governed by Hooke's law. They are used to cause the deformation of an elastic object in which case they produce an elastic potential energy. Hence, a force can be

defined in a generic way as:

$$(3.51) \quad \vec{f} = -k \cdot \vec{x}$$

where $-k$ and \vec{x} are the two main parts of the force relating to its stiffness and the elongation between the rest and deformed state. By applying this force on a given particle, the stored elastic potential energy $u(\vec{x})$ is a function of the elongation such as:

$$(3.52) \quad u(\vec{x}) = \frac{1}{2} \cdot k \cdot |\vec{x}|^2$$

The elongation is computed from a known particle position, at the current time step, and an unknown position at the next time step that could be in turn estimated through a constraint function. A description of a generic constraint function will come in the following.

3.3.5 VARIATIONAL IMPLICIT EULER (VIE)

The implicit Euler integration scheme, also known as Backward Euler, is defined by the following recurrent equations to update positions and velocities:

$$(3.53) \quad \vec{x}_i^{t+\Delta t} = \vec{x}_i^t + \Delta t \cdot \vec{v}_i^{t+\Delta t}$$

$$(3.54) \quad \vec{v}_i^{t+\Delta t} = \vec{v}_i^t + \frac{\Delta t}{M_i^{t+\Delta t}} \cdot \left(\sum_{k=1}^m \vec{f}_{int}(\vec{x}_i^{t+\Delta t})_k + \sum_{j=1}^n \vec{f}_{ext}(\vec{x}_i^{t+\Delta t})_j \right)$$

where m is the total number of internal forces and n the total number of external forces acting on a given particle i . Here, we have conveniently separated the residual force term and expressed forces as a function of positions in order to relate them with the negative gradient of an elastic potential energy function such as:

$$(3.55) \quad \vec{f}_{int} = -\nabla u(\vec{x})$$

Substituting equation 3.55 and 3.54 into equation 3.53 and reorganizing we obtain:

$$(3.56) \quad M_i^{t+\Delta t} (\vec{x}_i^{t+\Delta t} - \vec{x}_i^t - \Delta t \cdot \vec{v}_i^t - \frac{\Delta t^2}{M_i^{t+\Delta t}} \cdot \sum_{j=1}^n \vec{f}_{ext}(\vec{x}_i^{t+\Delta t})_j) = \Delta t^2 \cdot \sum_{k=1}^m -\nabla u(\vec{x}_i^{t+\Delta t})_k$$

Equation 3.56 is nonlinear and equivalent to a discretized version of Newton's second law of motion. To solve this equation, Baraff and Witkin (Baraff and Witkin 1998) proposed to apply a Taylor series expansion of internal forces at the known position \vec{x}_i^t . In doing so, each iteration requires to solve a large system of equations that, due to the high computational costs, can eventually compromise the development of real-time applications. Because of this, recent contributions (Bouaziz et al. 2014; Liu et al. 2013; Martin et al. 2011) proposed an alternative solution referred as Variational Implicit Euler (VIE) through which the non-linear equation is solved by finding the critical points of the following equation:

$$g(\vec{x}_i^{t+\Delta t}) = \frac{1}{2} \cdot M_i^{t+\Delta t} \cdot (\vec{x}_i^{t+\Delta t} - \vec{x}_0)^2 + \Delta t^2 \cdot \sum_{k=1}^m u(\vec{x}_i^{t+\Delta t})_k \quad (3.57)$$

To simplify notation, we have defined $\vec{x}_0 = \vec{x}_i^t + \Delta t \cdot \vec{v}_i^t + \Delta t^2 / M_i^{t+\Delta t} \cdot \Sigma \vec{f}_{ext}(\vec{x}_i^{t+\Delta t})$. Equation 3.57 is effectively the gradient of function g equated to zero ($\vec{\nabla} g(\vec{x}_i^{t+\Delta t}) = 0$). This leads to an optimization problem that looks for minimizing function g over $\vec{x}_i^{t+\Delta t}$. The fundamental principle is then to apply a block coordinate descend strategy, also known as alternating optimization technique (Sorkin and Alexa 2007), on the basis of potential energies. This is a two-step technique consisting of a local step in which energy functions are minimized and a global step that combines all the results and find the best compromise between them.

The process starts with an initial guess of $\vec{x}_i^{t+\Delta t}$ that is estimated by minimizing the first part of equation 3.57. This estimation guarantees that the solution will follow its momentum after each iteration (Bouaziz et al. 2014). Thus, the initial guess of $\vec{x}_i^{t+\Delta t}$ is defined by:

$$\min_{\vec{x}_i^{t+\Delta t}} \frac{1}{2} \cdot M_i^{t+\Delta t} \cdot (\vec{x}_i^{t+\Delta t} - \vec{x}_0)^2 \quad (3.58)$$

Using the approach proposed by Bouaziz et al. (Bouaziz et al. 2012), a potential energy function is formulated on the basis of distance function of the form $d(\vec{x}_i^{t+\Delta t}, \vec{p}_k)$ that measures the least

amount of change that requires the estimated position $\vec{x}_i^{t+\Delta t}$ to satisfy a given constraint C_k . The concept of a constraint relates to the idea of attaching a fixed or moving point to the given particle in order to satisfy a goal condition (Baraff and Witkin 1998). Constraints were initially used to implement contact mechanisms that anticipate intersections within particle models, but today they constitute the central basis of the VIE method.

A constrain function has the form $C(\vec{x}_1, \dots, \vec{x}_n)$ and relates to the strain E of a given element by measuring the deformation from its current state, defined by the corresponding local set of estimated positions $\{\vec{x}_1, \dots, \vec{x}_n\}$, to its rest state. The constraint is then satisfied if the measured strain is equal to zero ($E = 0$) or within certain accepted ranges ($min < E < max$). Once satisfied, there is no potential energy stored within the particles that is associated to the given constraint. Otherwise, the constraint uses those positions and strain value to estimate new positions, referred as projections, where the given goal condition would be satisfied. Each particle has then a corresponding projection \vec{p}_k attached to it until the given constraint C_k is satisfied. In such cases, the potential energy is defined by:

$$(3.59) \quad u(\vec{x}_i^{t+\Delta t})_k = \frac{1}{2} \cdot K_k \cdot |\vec{p}_k - \vec{x}_i^{t+\Delta t}|^2$$

At the local step, all the potential energies governing the motion of the particle are derived by solving the corresponding constraint functions. Each iteration weakly decreases the magnitude of those energies because the particle is gradually moving towards an equilibrium or goal position. Once all the potential energies are calculated, the results are globally combined and then minimized. The total sum of potential energies $U(\vec{x}_i^{t+\Delta t})$ acting on a particle is equivalent to the sum of weighted squared distances measured between the estimated position and the corresponding projections. Thus,

$$U(\vec{x}_i^{t+\Delta t}) = \sum_{k=1}^m \frac{1}{2} \cdot K_k \cdot |\vec{p}_k - \vec{x}_i^{t+\Delta t}|^2 \quad (3.60)$$

It should be noted that the estimated position was fixed at

the local step in order to compute the corresponding projections. Therefore, at the global step, projections are in turn fixed to calculate an optimum position that minimizes the weighted sum of squared distances. To do this, we first equate the gradient of equation 3.60 to zero such as:

$$\vec{\nabla}U(\vec{x}_i^{t+\Delta t}) = \sum_{k=1}^m K_k \cdot |\vec{p}_k - \vec{x}_i^{t+\Delta t}| = 0 \quad (3.61)$$

By expanding equation 3.61 and reorganizing, we obtain:

$$\vec{x}_i^{t+\Delta t} \cdot \sum_{k=1}^m K_k = \sum_{k=1}^m K_k \cdot \vec{p}_k \quad (3.62)$$

Finally, we can use the following equation to solve the optimum position:

$$\vec{x}_i^{t+\Delta t} = \frac{\sum_{k=1}^m K_k \cdot \vec{p}_k}{\sum_{k=1}^m K_k} \quad (3.63)$$

Knowing the new positions, we can update the velocities by:

$$\vec{v}_i^{t+\Delta t} = \frac{\vec{x}_i^{t+\Delta t} - \vec{x}_i^t}{\Delta t} \quad (3.64)$$

The procedure of this algorithm is summarized in (Figure 3.16). The core idea of the method is to update the position of particles by directly estimating valid positions that do not overshoot the equilibrium condition.

```

foreach particle  $i$  do
     $\vec{x}_0 = \vec{x}_i^t + \Delta t \cdot \vec{v}_i^t + \Delta t^2 / M_i^{t+\Delta t} \cdot \sum \vec{f}_{ext}(\vec{x}_i^{t+\Delta t})$ 
     $\vec{x}_i^{t+\Delta t} = \vec{x}_0$ 
    loop
        foreach constraint  $k$  do
             $\vec{p}_k = \text{ProjectConstraints}(C_k, \vec{x}_i^{t+\Delta t})$ 
        end
         $\vec{x}_i^{t+\Delta t} = \text{ComputeOptimumPosition}(\vec{p}_1, \vec{p}_2, \vec{p}_3, \dots)$ 
    end
     $\vec{v}_i^{t+\Delta t} = (\vec{x}_i^{t+\Delta t} - \vec{x}_i^t) / \Delta t$ 
end

```

Fig. 3.16

Variational Implicit Euler algorithm.

3.4 REMARKS

PS and DR are defined as two types of dynamic methods that share close similarities between each other. For the sake of simplicity, we consider that PS is in fact a generalized form of DR because the formulation of the problem is exactly the same in both cases. That is, subdividing a continuous system into a finite set of discrete units, called particle or nodes, where masses are concentrated, and motion produced. Through a time-integration process based on Newton's laws, the discrete model oscillates until finding a steady-state under the influence of damping. The main difference only lies in PS supporting more schemes for time-integration and dynamics to produce motion. Based on this logic, a DR model is equivalent to a PS model that uses an explicit time-integration scheme and a motion dynamic based on forces. Therefore, to maintain consistency and avoid misunderstandings with new terminology arising, we will only refer to PS terminology for the continuation of this dissertation by assuming the above-mentioned relationship between PS and DR.

4

TOPOLOGY-DRIVEN FORM-FINDING

4 TOPOLOGY-DRIVEN FORM-FINDING

The most useful feature of PS models is the capacity to model geometric and material non-linearities at highly interactive rates. This means that physical behaviors may not be only treated as visualization data coming from a pre-executed simulation, as in the case of FE-models, but as an interactive medium to progressively explore design intentions. For this reason, the use of PS models has become important during conceptual stages where fast responses of physically-plausible behaviors are preferred over accurate mechanical descriptions that, due to their high computational costs, can eventually constraint the exploration of design spaces. At first sight, this reduced accuracy may contradict traditional approaches for scientific modeling that are entirely focused on analysis and verification routines. However, the derived complexity of form-finding problems normally requires dividing the design process into an exploratory stage to facilitate an iterative exploration of equilibrium shapes and an analysis stage entirely focused on the validation of mature solutions. Under these conditions, PS models can't replace FE-models, or vice-versa, because both models are utilized for different modeling purposes. A thoughtful consideration and a careful decision are demanded regarding when to introduce a numerical model and based on which requirements, knowing that each design stage addresses different problematics and each technique is oriented to solve different problems.

4.1 GENERATIVE APPROACHES FOR FORM-FINDING¹

In the context of bending-active and textile hybrid structures, the great advantage of PS models is that they can

¹Based on pre-published article (Suzuki et al. 2017): “A comparative overview of generative approaches for computational form-finding of bending-active tensile structures”.

generate an appropriate mechanical description while maintaining an appropriate rate of interactivity between designers and digital models. Since the fundamental purpose is not analysis, efforts have focused on the development of more flexible and user-friendly workflows resulting on new kinds of form-finding practices adapted for expert and non-expert users. A PS model is basically built from a massive combination of geometric and mechanic properties embedded within a topologic model that directly determines the behavior of a system composed by particles and elements. The major challenge on the construction of a PS model directly relates to how well defined are those properties and topologic conditions to satisfy a given design intention. As an example, an alteration of the geometric properties of an element affects the motion of its particles, causing a change in the overall dynamic behavior of the system. This indicates that all equilibrium states will depend on the geometric properties of the system.

Geometric properties are those properties that are not preserved under the continuous deformations of the PS model during numerical integration. That is, it deals with quantifiable measurements describing the relative positions of particles in the Euclidean space, their sizes or masses and the properties of elements such as stiffness values, angles, areas or lengths, among others. On the other hand, changing the topologic properties of a system affects the number of particles and/or elements, causing again a change in its overall dynamic behavior and resulting equilibrium states. The topologic properties of a PS model are preserved despite the continuous deformations of the system during numerical integration. They basically define the set of particles constituting the model and the multiple ways those particles relate to each other. These relationships are the essential mechanism to determine the type of forces or constraints acting on particles and causing the motion that will eventually lead towards the deformation of the entire model. Therefore, it is important for designers to have a clear understanding of these topological properties and also on how to modify them dynamically. As a result, depending on the management and modification of both types of properties during time-integration, two main families of generative approaches for form-finding can be defined.

4.1.1 GEOMETRIC-DRIVEN APPROACH

4.1.1.1 Definition

Generative design approaches for numerical form-finding are suggested to be categorized as geometric-driven when the variation of equilibrium shapes is the result of dynamic alterations of geometric properties within a persistent topological model. From a topological point-of-view, geometric-driven approaches result on purely deterministic procedures outputting the same topologic model from any given input of geometric conditions. In other words, an equilibrium condition is an optimized geometric variation of particle's positions in which, during numerical integration, the connectivity of the entire system was not modified (Figure 4.1). These types of numerical models are implemented through adjacency matrices for describing their topological properties. Those properties are usually considered to be static or quasi static in order to improve the convergence rate of numerical solutions. For this reason, they are just indexed in computer's memory without explicitly encoding all the topological relationships among particles. By having a fast rate of numerical convergence, it is assumed that topological modifications can be treated separately from numerical integration without affecting the interactivity of the design process. This implies that a relatively good knowledge of the design problem is always required prior the construction of the numerical model. In certain cases,

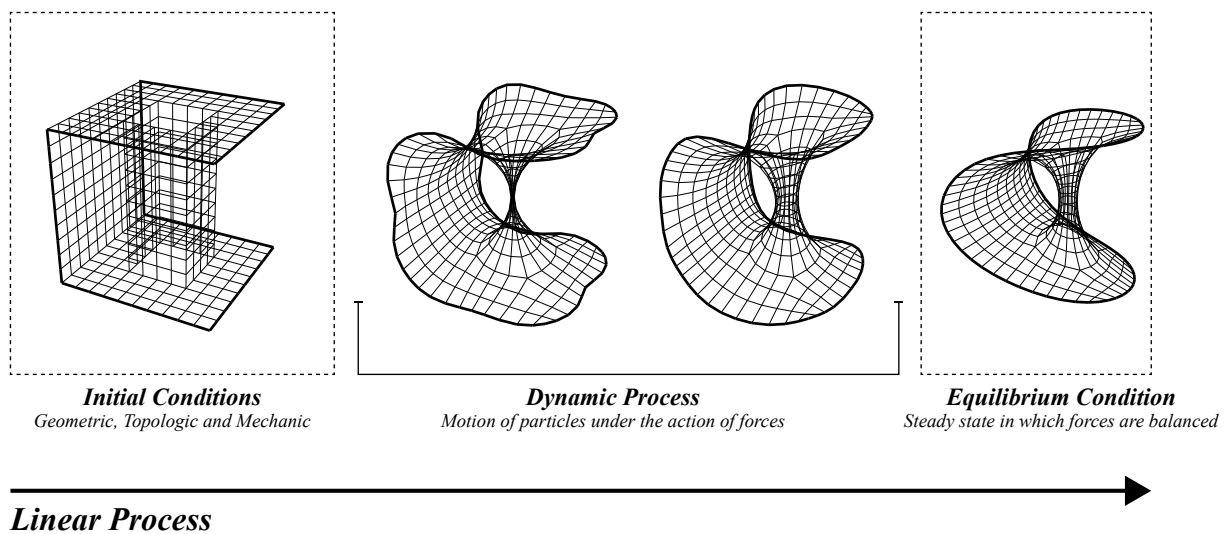


Fig. 4.1

Typical numerical form-finding workflow.

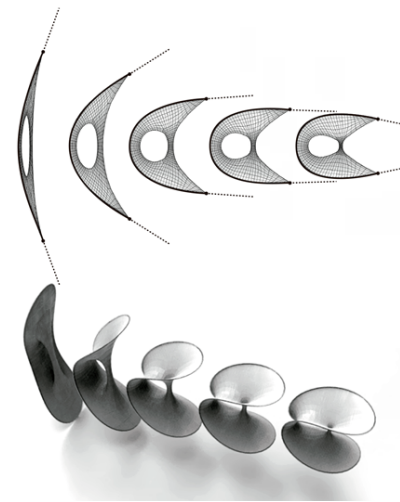


Fig. 4.2
Geometric variation.

this knowledge can certainly come from designer's intuition. However, in other cases, it needs to be gained through experience by conducting experimental studies on scaled physical models. It is at this point that the non-comparable intuitive and interactive characteristics of analogue models continue to play a fundamental role in contemporary computational design practices.

Nowadays, geometric-driven approaches are among the most popular strategies for generative form-finding. In general, they follow a similar workflow of the one adopted for scientific modeling but relies on speeding up numerical convergence to create fast iterations (Figure 4.2). The main issue is the necessity to break and re-initialize the algorithm for changing topological conditions when design intentions are not satisfied. Accordingly, geometric-driven approaches may be categorized as one-directional processes defined by independent modeling and simulation stages without recursive procedures (i.e. output geometries cannot be reconverted to input models of the same numerical process) and limited capabilities for data storage modification (Figure 4.3).

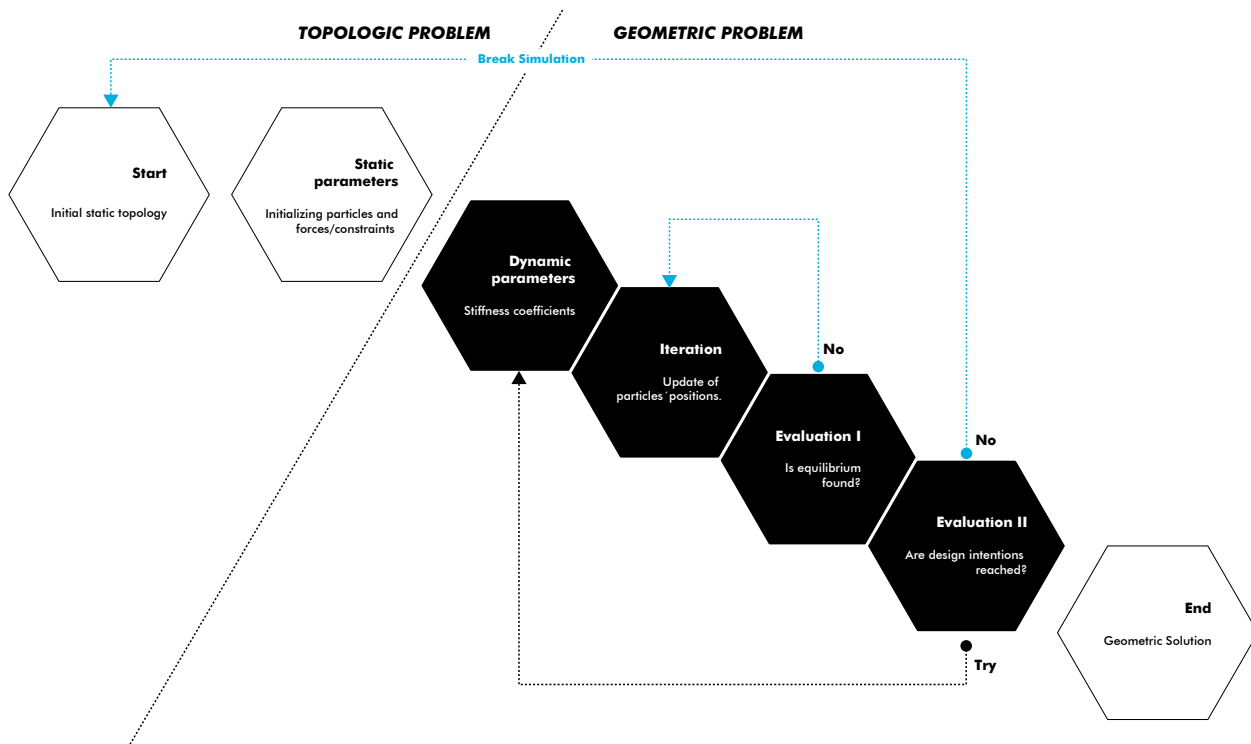


Fig. 4.3
Geometric-Driven Form-Finding Approach

It can be observed that geometric-driven approaches are more appropriate for form-finding processes dealing with topologic conditions with a certain regularity or a low complexity that can be intuitively solved by the designer. However, this condition is particularly problematic when highly intricate material relationships are involved, and designer's intuition cannot provide a valid solution. This is because topological operations need to be carried on undeformed geometries where its consequences on the deformed state are rather difficult to predict. An example of this is the ICD/ITKE M1 La Tour Project in France where the problem of finding a valid topology for the primary structure required exhaustive studies on physical models (Lienhard et al. 2013a). This project is one of the most interesting examples for showcasing the potentialities of textile hybrid systems to break standard topology of membrane and gridshell structures and develop complex geometrical shapes from a limited combination of simple components. Paradoxically, it also exemplifies that, under these conditions, additional amounts of manual calculations are still required for dealing with highest degrees of topological complexity.

4.1.1.2 Examples

So far, most relevant examples of this type of form-finding approach are associated with the use of Kangaroo (K1 and K2) without extended scripted functionalities. Such processes allow to conduct studies for variegating structural shapes by altering parameters controlling material and physical properties. In this case, real-time interactions with digital models allow designers to have similar intuitive feelings as with analogue models. A basic form-finding process in Kangaroo is defined by four separated stages being modeling, discretization, parameters initialization and simulation. During simulation runtime, geometric properties may be easily altered but topologic properties cannot be transformed in a straightforward manner. Through the development of a custom script using the Kangaroo solver, it was determined that particles and goals are sequentially stored on computer's memory by index assignation within dynamic arrays. In other words, the topologic model is stored as an incidence graph. This type of data structure permits to easily to add and delete items at the

end of the sequence. However, it is rather difficult to conduct the same operations on other positions of the array without a comprehensive process for updating indexes. Altering the index sequence will inevitably require updating connectivity's data that, in some circumstances, make operations for adding and deleting objects complex and unsafe. Lastly, the fact that Kangaroo is embedded within Grasshopper results on form-finding process organized as a data-flow diagram with strict dependencies, and where recursive procedures have always been a challenge to implement. This is a well-known cost of pure data-flow models derived from visual programming languages where key recursive routines need to be tackled through complex implementations of control-flows (Mosconi and Porta 2000).

4.1.2 TOPOLOGY-DRIVEN APPROACH

4.1.2.1 Definition

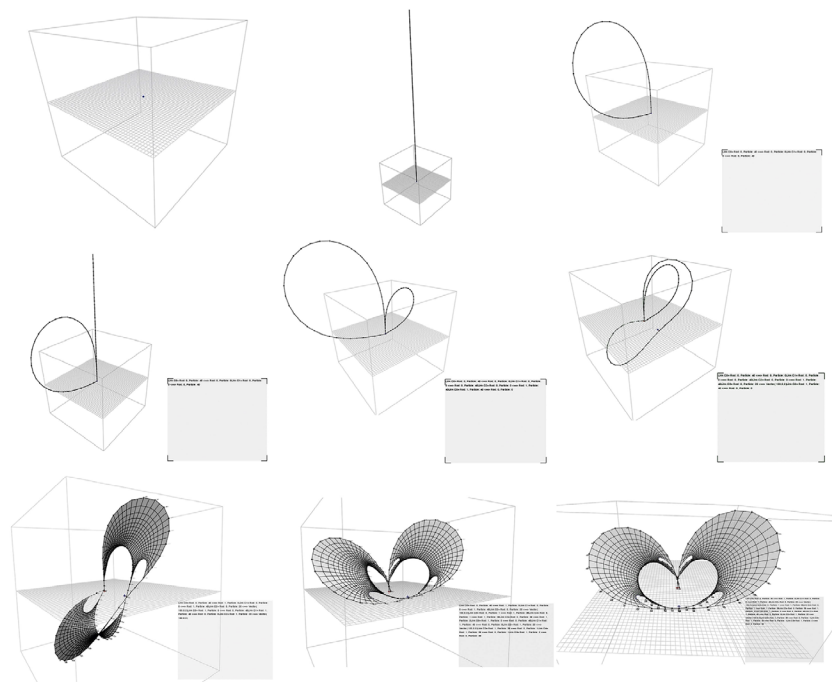


Fig. 4.4

Topologic differentiation.

Topology-driven approaches for numerical form-finding based on PS models may be considered as extended and more generative versions of geometric-driven approaches. Added to model variation, the key idea is to introduce the concept of model differentiation by permitting the dynamic alterations of topological properties. Hence, an equilibrium condition is an

optimized geometric variation of particle's positions resulting from topologic properties that can gradually change during time-integration (Figure 4.4). In other words, a radical increase of modeling flexibility and design space freedom. The main idea of topology-driven approaches is synthesized within fundamental operations for adding, deleting or modifying topological properties on the fly. In geometric-driven approaches, these operations are treated separately and necessarily requires breaking the simulation. For this reason, topology-driven approaches may be categorized as circular processes (Figure 4.5) where problems regarding interactivity, data organization, decision-making and discretization need to be solved in parallel and recursively.

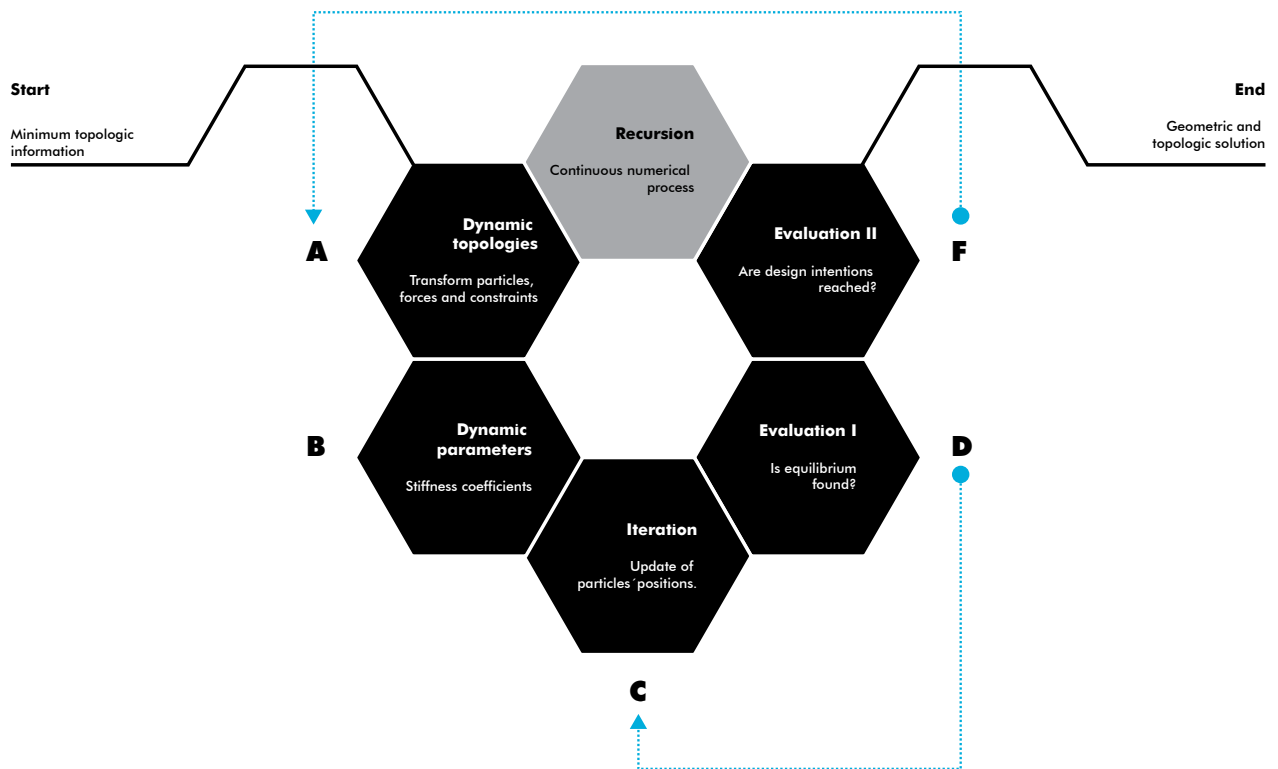


Fig. 4.5
Topology-Driven Form-Finding.

The main advantage of these approaches is that they give back designers a complete freedom for exploring as many solutions as their creativity permit. This is because they enhance the interactivity of PS models by replicating the empirical and experimental experience of playing with physical materials to derive equilibrium shapes. Designers can gradually shape a PS model by easily adding or removing parts of the topologic model

and controlling the corresponding geometric properties. It is even possible to start the construction of a PS model without any specific pre-requisite because geometric and topologic properties can be solved simultaneously in order to increase the specificity of the designed object. This means that the numerical process is more intuitive because it can carry out fast physical calculations and, at the same time, dynamically alter the design at will. Such modeling characteristics can then be treated in a way that design concepts can easily emerge, or further ideas develop. Eventually, this condition results on computational models fully oriented towards design process that combines the playfulness of analogue models and the geometric informed complexity of numerical ones. Moreover, this extended modeling interactivity can play a significant role in education for introducing new practitioners to the realm of simulation-based design.

A useful feature of PS methods that was previously mention, is that they can be solved in a local vector form without the assemblage of matrices. At this point, this does not only facilitate computational implementations, but also permits to modify the topological properties of the PS model without having to reformulate the entire numerical process. However, the implementation of dynamic topologies is still a complicated task that normally requires a complete restructuration of the form-finding workflow. This problem can be addressed by reconstructing the entire topological model with each modification, or by letting it evolve over time. Note that the numerical process needs to remain stable under all circumstances. The continuous reconstruction of topological models can easily derive into error-prone operations that will eventually produce numerical instabilities. On the other hand, letting the topologic model evolve over time is best suited to address the problem of form-finding with dynamic topologies. The complexity in these types of topologic models is that they need to be able to store and describe changes on the connectivity without the imperative necessity of reconstructing the entire data structure.

4.1.2.2 Examples

From a design perspective simulating real-time topologic

alterations during form-finding has not been fully explored. Few efforts have been reported on this subject mainly because there is still some debate regarding the loss of computational performance when increasing the flexibility of simulations. Ahlquist et al. (Ahlquist et al. 2015) presented a study for form-finding cable-net systems in which the topology of PS models, built with Greenwold's library (Greenwold 2012), is altered via explicit user-model interactions, or automated through the use of evolutionary algorithms. By analyzing the open source code of the PS library, a better understanding was obtained on how those topologic operations are carried during time-integration. Following an Object-Oriented Paradigm (OOP), the library stores particle and force objects through index associations within fixed sized arrays in computer's memory where adding and deleting elements requires special functions for updating indexes and data connectivity. Once an array is full, the entire data structure is reconstructed for increasing its initial capacity. This process describes the basic reconstruction of a topologic model that was initially implemented to static or quasi-static properties. Another study proposed the development of a computational pipeline based on K2 to support topological modifications during the form-finding of textile hybrid structures (Deleuran et al. 2016; Quinn et al. 2016). In this case, topologic modifications are supported by using an auxiliary adjacency graph, called NetworkX (NetworkX Developers 2019), to continuously reconstruct the entire database of the physical solver. Finally, Mesh Machine (Piker 2014) is another example built on the idea of topologic modifications during time-integration. Based on the numerical solver of Kangaroo I, this Grasshopper tool was developed for dynamically changing the edge topology of a given triangular mesh with a half-edge data structure in order to best satisfy a set of geometric conditions. Here, topological properties are stored through a more complex data-structure that facilitates continuous adjacency queries.

4.1.2.3 Related Examples in other fields of research

The introduction of topologic alterations during simulation runtime has not been fully addressed within form-finding or structural analysis applications. However, it is a well-established

research area in different fields of computational graphics related to the animation and gaming industry and, particularly, for medical applications. In fact, it constitutes the fundamental mechanism for simulating tearing, cutting, fracture and explosion on deformable bodies, among many other effects. A state-of-art report of techniques for cutting deformable bodies was presented in (Wu et al. 2015). Throughout these years, surgical simulators have become among the most important references on the combination of topologic modeling and real-time physics with highly interactive applications like the ones presented in (Mitchell et al. 2015; Chentanez et al. 2009). Major efforts are focused on the development of computational models for training purposes that simulate complex surgical routines in real-time. The major concern is to interactively deform and cut a virtual organ and, eventually, feeling its reaction (Delingette et al. 1999). Another possibility is surgery planning where the interactive characteristic is less important than the geometric accuracy of the deformed model. These conditions go back to the problem of equilibrating the rate of interactivity and mechanical accuracy of computational models according to application requirements.

For offline simulations where real-time interactivity is not required, a specialized FEM technique for simulating cuts in deformable bodies, called Extended Finite Element Method (XFEM), has recently gained in importance. XFEM was originally formulated to model cracks and crack growth without remeshing by introducing local enrichment functions that separates the standard FE-model from the crack representation (Mos et al. 1999). The main drawback is the computational cost of the numerical process that, added to the pre-processing stage required for building a FEM model, makes the method incompatible with the possibility of introducing cuts in real-time. However, the required pre-processing stage of FEM models added to the computational costs of the numerical solution prevent the possibility to introduce cuts in real-time (Ganovelli et al. 2000). For this reason, PS are still the most popular methods for the development of real-time surgical simulators. With fastest developments, these types of computational models have been naturally conducted into more immersive spaces for improving the efficacy of training processes. The state-of-art of these simulators with applications in virtual,

augmented and mixed realities (Buckley et al. 2012; Lahanas et al. 2016) clearly exemplifies that the separation between analogue and digital training practices can be progressively reduced.

4.2 A GENERIC TOPOLOGIC MODEL FOR TOPOLOGY-DRIVEN FORM-FINDING

Considering previous studies that have been conducted for the numerical form-finding of bending-active and textile hybrid structure, a generic PS model for explorative form-finding processes enabling topologic modifications on the fly is still required. In the late seventies, Barnes (Barnes 1977) already mentioned that DR is a suitable method to support dynamic topologic transformations. However, it should be noted that this characteristic cannot be easily activated within traditional numerical schemes because a more detailed description and capability of analysis of all topologic properties would be required along the entire integration process. To our knowledge, no form-finding model have been proposed with enough topologic information for enabling various types of modifications during simulation runtime. Therefore, the use of an appropriate topologic model constitutes a powerful instrument for such purposes and is the most fundamental prerequisite for the concept of topology-driven form-finding.

4.2.1 TOPOLOGIC MODELS

Topologic models play a central role for the construction of PS models. They permit to represent the discrete assembly of particles and their structural relationships (i.e. connectivity) as an n-dimensional subdivision shaped from topologic elements with different dimensionalities called cells, along with neighborhood relationships. In computer-modeling terminology, this is commonly referred as a mesh structure. The main idea is to link the topologic model with mechanic and geometric properties in order to finally obtain a comprehensive topologic definition of a PS model. Cells are categorized as vertices in zero-dimension, edges in one-dimension, faces in two-dimension and volumes in three-dimensions (Figure 4.6). The highest dimensionality of a cell is determined by the given dimensionality of the topological

model. Incidence relationships can be formed by considering that cells of higher dimension are always made from cells of lower dimensions. For instance, an edge is said to be incident to the face that contains it, in which case the face is also incident to the edge. A cell-tuple is in turn defined as an ordered sequence of cells with decreasing dimensionalities such that a path for going from the cell with the highest dimension to a vertex is created (Kraemer et al. 2014). Adjacency relationships are in turn constructed by comparing the existence of common cells within different cell-tuples. This means that two cells are adjacent if they are incident to the same cell.

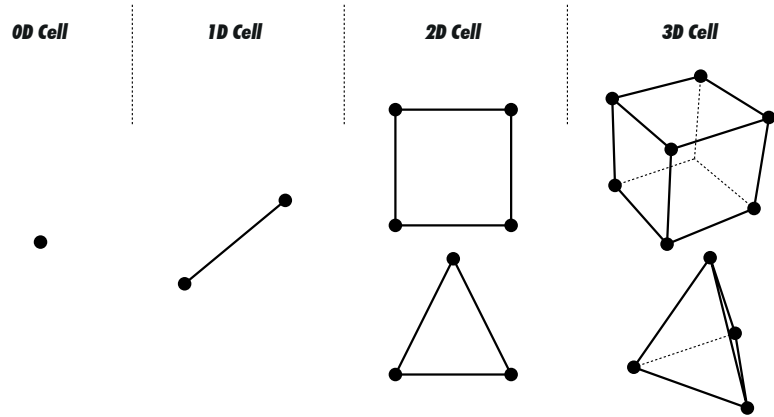


Fig. 4.6

Cells of a three-dimensional cellular decomposition.

In order to select an appropriate representation of the topologic model, it is useful to first understand the types of objects to be modeled. In the scope of this dissertation, PS models are used for explorative form-finding processes of bending-active and textile hybrid structures based on one-dimensional bending-active elements and, at most, two-dimensional tensile elements. This means that the cellular decomposition needs to be represented by cells of up-to two-dimensions. In addition to this, we restrict the shape of two-dimensional cells to be quadrangular, named cubical simplices in combinatorial topology literature, to easily match the mechanical characteristics of textile fabrics. For certain types of designs, the assembly of cells eventually results on a quasi-manifold representation (Figure 4.7a). Note that quasi-manifolds can be defined using combinatorial logics on a discrete structure while this is not possible for manifolds (Damiani and Lienhardt 2015). A quasi-manifold assembly of an n -dimensional

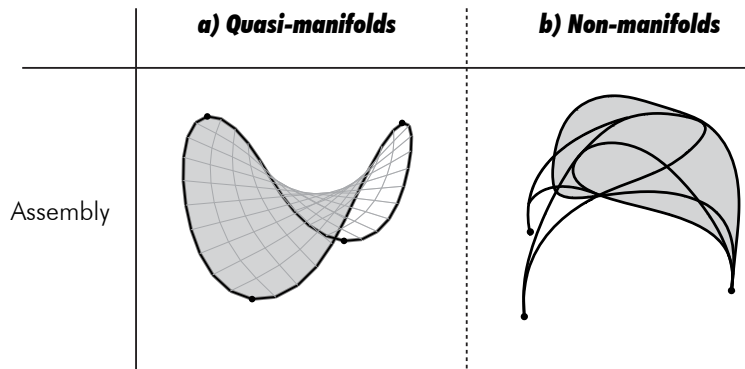
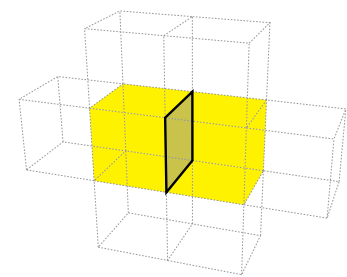


Fig. 4.7

Types of cellular decompositions.

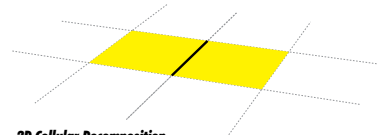
topologic model is constructed by first identifying the $(n-1)$ -dimensional cells constituting the boundaries of n -dimensional cells, and then guaranteeing that each of the $(n-1)$ -dimensional cell is incident to at most two n -dimensional cells (Figure 4.8). More precisely, a two-dimensional quasi-manifold has interior vertices with neighborhoods homeomorphic to a disc (closed two-dimensional polygon) and boundary vertices with neighborhoods homeomorphic to half disc (Damiani and Lienhardt 2015). Two topologic models are said to be homeomorphic, or topologic equivalent, when a one-to-one mapping exists between both models. Nevertheless, in most of the cases, quasi-manifolds assemblies may not be best suited to match design intents. For instance, the topologic model of the ICD/ITKE M1 La tour builds on a non-manifold (i.e. simplicial complex in algebraic topology terminology) assembly of cells (Figure 4.7b). This condition is especially evident when using multiple mechanical elements with different dimensionalities. A non-manifold assembly can be intuitively defined as an assembly type of cells that do not respect the above mentioned properties of quasi-manifolds. It is therefore necessarily for form-finding to have the possibility to represent both, quasi-manifolds as well as non-manifolds models.

The computational implementation of a topologic model is carried through a specialized form of data structure that determines the type of assembly as well as the way to store, organize, access and manipulate topological data. Data structures can be considered as the central interface for retrieving topologic information in order to know the current state of the PS model



3D Cellular Decomposition

2-dimensional cell (face) is incident to two 3-dimensional cells (volumes)



2D Cellular Decomposition

1-dimensional cell (edge) is incident to two 2-dimensional cells (faces)

Fig. 4.8

Quasi-manifold conditions.

and assist its construction. In this context, the topologic properties of PS models are normally implemented via incidence graphs (Figure 4.9). An incidence graph is a representation of a directed graph where a vertex correspond to a cell and edges are associated with the incidence relationships between those cells (Damiani and Lienhardt 2015). The correspondence with the PS model is then established by considering particles equivalent to vertices and edges equivalent to mechanical elements. The general case is the use of mechanical elements that always relate two particles together (e.g. springs or elastic bars). Even higher dimensional elements like CST can be represented as pseudo-system of elastic bars.

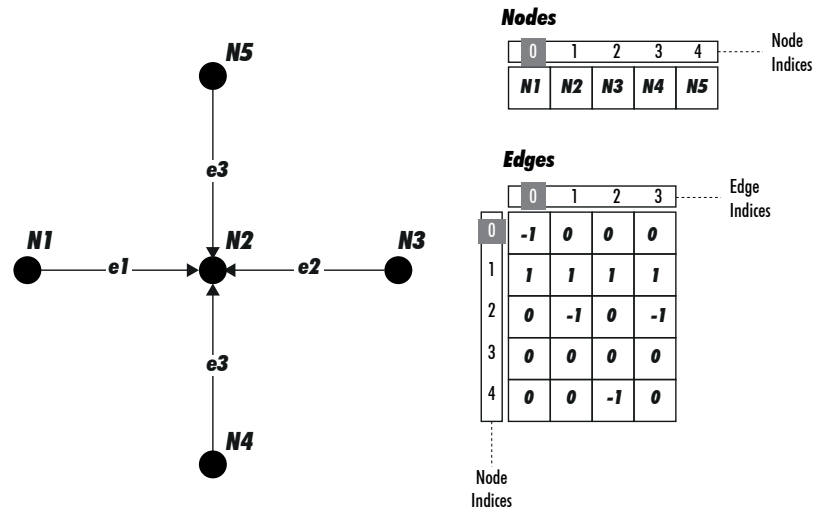


Fig. 4.9
Incidence graph..

In practice, an incidence graph is an n -by- m matrix implemented via an array-based data structure where n and m are the corresponding number of vertices and edges. The entries of the matrix are defined as -1 if the vertex is the start of the edge, 1 if it is the end and 0 for other cases. They permit a fast and easy access to those entries when the matrix is kept constant over numerical integration. The problem is that they do not explicitly encode enough information to solve all the incidence and adjacency relationships, in which case additional information of higher dimensional cells, like the decomposition of faces into edges, would be required. Under these conditions, a complete topologic description would require a lot of memory due to the large amount of information derived from the number of particles, mechanical elements and topologic relationships. Incidence

graphs create a less flexible topologic model, but with a greater availability of computational resources reserved for solving the different mathematical operations at each time integration step (e.g. computation of forces, integration of equations of motion, visualization).

Fenves and Branin (Fenves and Branin 1963) were among the firsts to proposed the use of incidence graphs for the design and analysis of structures like plane frames, plane grids and space frames. Their use for form-finding problems of tensile structures was later addressed by Linkwitz and Schek (Schek 1974; Linkwitz and Veenendaal 2014) with the development of the Force Densities Method and by Barnes (Barnes 1977) with DR. Even today, major advances on the research of PS methods are still built on the use of incidence graphs. The problem of altering the topologic properties of incidence graphs is that data structures need to be reconstructed without creating inconsistencies on the connectivity. This process is highly complex and error-prone because most of the neighborhood relationships cannot be answered during time integration. Because of this, we can conclude that incidence graphs may lack the adequate robustness for dynamically handling topologic changes during simulation runtime.

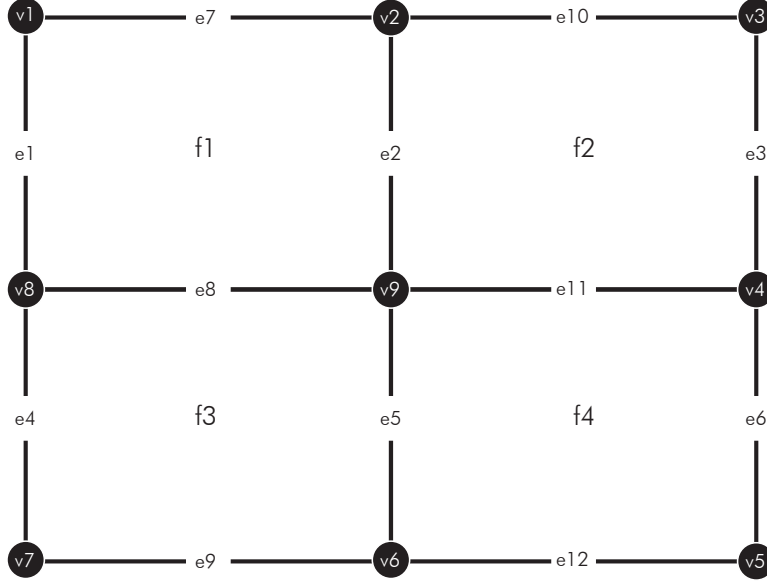
4.2.2 COMBINATORIAL MAPS

Combinatorial maps are ordered models in which the representation of a cellular structure is established on more elementary elements than cells so that all neighborhood information can be retrieved at any time and with minimal information. For this reason, combinatorial maps may be considered as the most appropriate type of topologic model to be associated with the idea of topology-driven form-finding. Several studies regarding surgical simulators and cloth animation are already arguing for the adoption of these complex topological models for guaranteeing the consistency of topological properties when changes need to be interactively introduced during simulation runtime (Ben Salah et al. 2017; Flechon et al. 2013; Luciani et al. 2014). Combinatorial maps have been widely used in the field of computational geometry and, more precisely, for the development of modeling

scheme based on boundary representations (B-REP).

Among the different types of combinatorial maps, special attention is dedicated to n -dimensional maps (n -maps), n -dimensional generalized maps (n -Gmaps) and n -dimensional closed chain of maps (n -chains). The concept of n -maps was initially proposed by Edmonds (Edmonds 1960) as a type of mathematical structure for describing two-dimensional orientable surfaces using combinatorial logics based on the idea of embedding a graph on a surface. Its generalization (n -Gmaps) was proposed to deal with non-orientable surfaces and, by the end of 1980s, both types of combinatorial maps were already extended for the representation of n -dimensional quasi-manifolds. Furthermore, n -chains have been proposed for dealing with non-manifold representations in which sets of i -dimensional cells can be built with more flexible topologic constraints by identifying cells of any dimension (Lienhardt et al. 2004)

The universal principle in combinatorial maps is to gradually decompose all the cells of the subdivision until obtained the simplest “atomic” elements, called darts, while, at the same time, keeping track of all the finite set of the relations between them. The different relations between those darts serves to describe all the cells of the model as well as their incidence and adjacency relationships. Even if each type of combinatorial map (n -maps, n -Gmaps and c -maps) contains different topologic constraints, they all share a common decomposition logic. To simplify the explanation of the proposed PS model for topology-driven form-finding, it is easier to use the generic definition of n -Gmaps and assume that, on this basis, simple modifications can be introduced to attain more flexible topologic constraints. This is because an n -chain is always constructed on the basis of an n -Gmap. N -Gmaps are preferred because of their homogenous algebraic definition with respect to any dimension permitting to simplify the definition of topologic operations (Meseure et al. 2010). Therefore, the aim of this chapter is to provide a conceptual description regarding the coupling of maps and mechanical schemes for building an appropriate PS model for topology-driven form-finding. Specific implementations are going to be presented in the next chapter.

4.2.2.1 *n*-Gmaps**Fig. 4.10**

An example of a two-dimensional cellular decomposition.

An n -dimensional Gmap ($n \geq 0$) is defined by $G = (D, \alpha_n, \dots, \alpha_0)$ where D is a finite set of abstract elements called darts that map cells together and α_i is an involution function defined on the basis of adjacency relationships, such that $\forall i, 0 < i < n$. Based on the description of Lienhardt et al. (Lienhardt et al. 2011), we introduce these abstract notions by describing the construction of a Gmap. The two-dimensional cellular decomposition on figure 4.10 is composed by four faces (f1, f2, f3 and f4) that are joined together by common edges (e2, e11, e5 and e8). An n -Gmap is constructed by first splitting those common edges so that each face contains one part of the corresponding common edge. Each pair of split-edges is linked with a two-dimensional relationship α_2 , called involution function, permitting to associate the face containing the split edge with the corresponding common edge (Figure 4.11a). All the faces are shaped from individual edges that are joined together by common vertices. A similar process is then applied to split a common vertex in order and create two vertices that are linked by a one-dimensional relationship (involution α_1). Note that two-dimensional relationships needs to be also updated for these new basic elements (Figure 4.11b). At this point, each edge connects two distinct vertices creating a zero-dimensional relationship (involution α_0) (Figure 4.11c). The split vertices

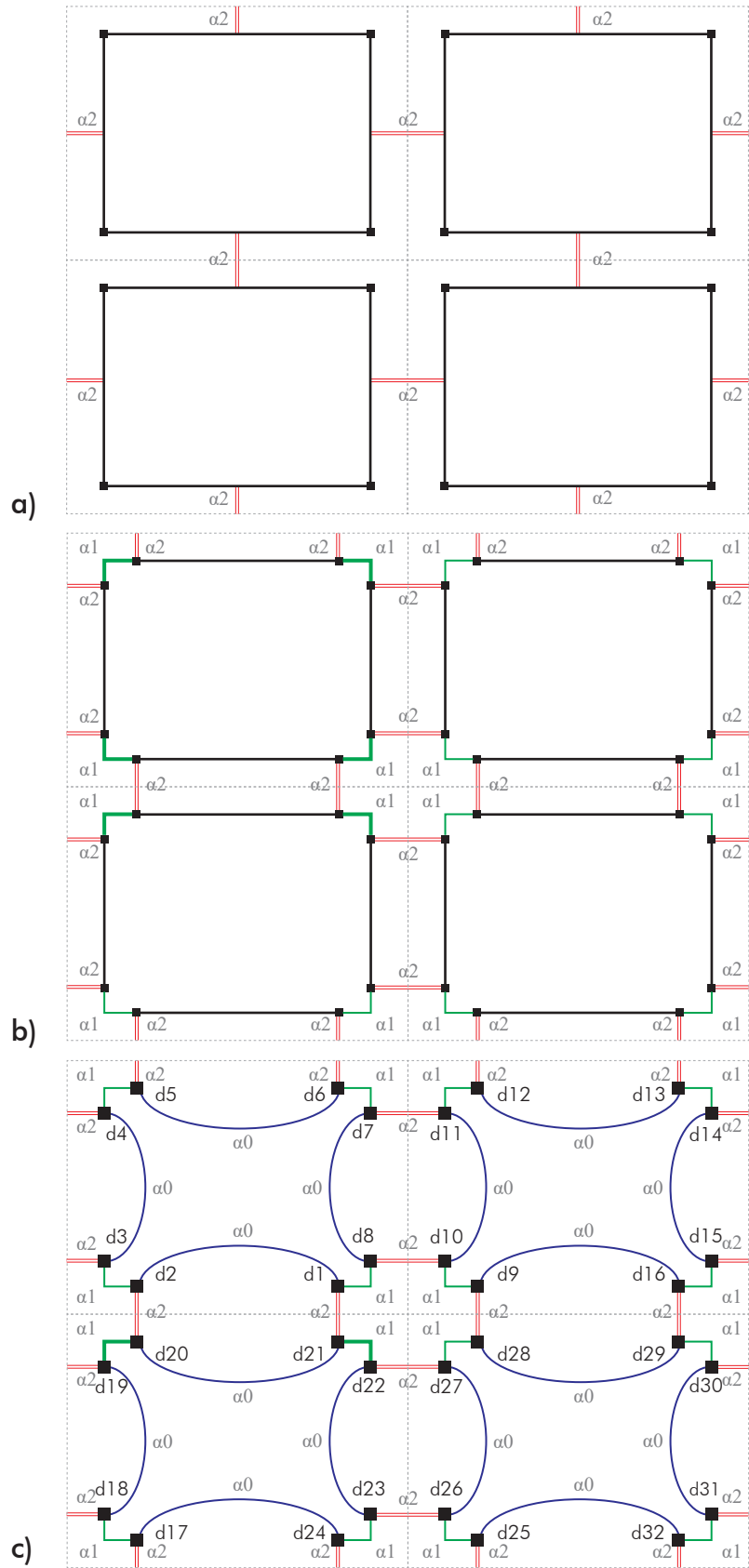


Fig. 4.11

A two-dimensional combinatorial map of figure 4.10.
a) Two-dimensional relationships, b) One-dimensional relationships, c) Zero-dimensional relationships

obtained at the end of the process are the fundamental elements of the n -Gmap which have been previously named as darts. Hence, an involution of a dart d can be generally defined as a function $x_i(x_i(d)) = d$, for all $d \in D$, where the subscript corresponds to the dimensionality of the relationship.

Combinatorial constraints determine the assembly of cells at the boundaries for quasi-manifold conditions such that, for an involution function α_i , the constraint is expressed as $\forall i, j, 0 \leq i \leq i+2 \leq j \leq n$ with $\alpha_i \circ \alpha_j$ being an involution (Kraemer et al. 2014). Constraints are fundamentally important for cellular decompositions of at least two-dimensions and can be easily expressed and checked. In the special case of non-manifolds, an n -chain $C = ((G^i)_{0 \leq i \leq n}, \sigma)$ is constructed such that $G^i = (D^i, \alpha_i^i, \alpha_{i-1}^i, \dots, \alpha_0^i)$ is an i -Gmap with α_i^i being an undefined involution and $\sigma: G^i \rightarrow G^{i-1}$ ($1 \leq i \leq n$) a mapping between a cell on G^i and on G^{i-1} (Alayrangues et al. 2015; Lienhardt et al. 2004).

The cellular decomposition is encoded through sets of darts and involutions creating a one-to-one correspondence with i -dimensional cells. For instance, in figure 4.10, vertex v9 (zero-dimensional cell) contains height darts that are linked together with involutions α_1 and α_2 . Dart d1 is defined by vertex v9, edge e8 and face f1 which is equivalent to a cell-tuple. Using involution α_1 on dart d1, we obtain dart d8 belonging to the same vertex v9 and face f1 but to a different edge e2. Edge e2 (one-dimensional cell) contains four darts that are linked together with involutions α_0 and α_2 . On the other hand, involution α_2 on dart d1 results on dart d21 which belongs to the same vertex v9 and edge e8 but to a different face f3. A face (two-dimensional cell) is in turn composed by eight darts contained by four edges that are linked together with involutions α_0 and α_1 . Hence, involution α_0 on dart d1 results on dart d2 which belongs to the same face f1 and edge e8 but to a different vertex v8. N -g-maps are therefore dimension-independent which means that higher dimensional cellular decompositions can be easily adapted to this scheme by just adding more darts and the corresponding involution functions as shown in figure 4.12. Involution functions and combinatorial constraints provide a consistent and constructive way to build and modify topologic models. In doing so, an efficient way to

store all the cells of the subdivision is created along with their incidence and adjacency relationships that are solved in optimal time without having to maintain any additional information.

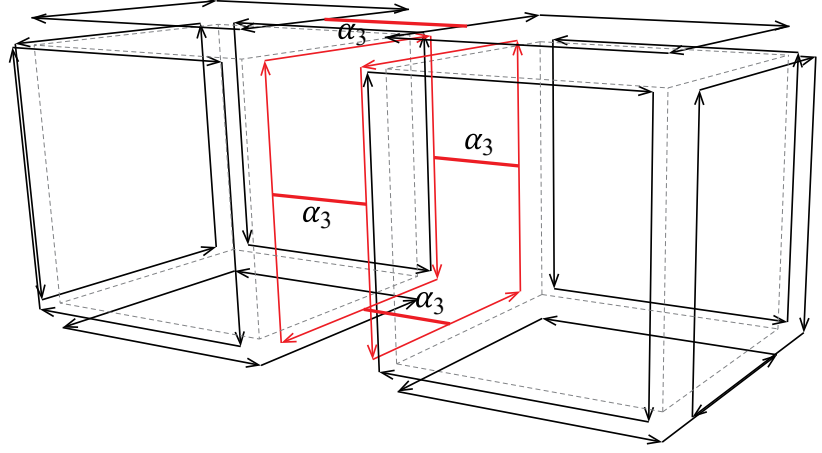


Fig. 4.12

Combinatorial map extension to higher dimensions.

4.2.3 ORBITS

An important characteristic of Gmaps is the idea of orbits which relate to the notion of graph traversals. A Gmap may be in fact considered as a detailed graph in which vertices are darts and edges are the corresponding links between those darts (involutions) (Figure 4.13a). An orbit $\langle S \rangle(b)$ is the set of all darts in \mathbf{B} that can be reached from b by following any combination of involution functions $\langle \alpha_0, \dots, \alpha_n \rangle$. An i -dimensional cell is therefore a special case of an orbit describing a set of darts that are connected with all involution functions except the involution of the current cell α_i (Meseure et al. 2010). Orbits provide an appropriate and formal way to describe all the cells that were explicitly defined in the subdivision. For instance, all the darts belonging to vertex v_9 are obtained through any combination of involutions α_1 and α_2 that return the set of darts $\{1,8,10,9,28,27,22,21\}$ (Figure 4.13b). The orbit of this zero-dimensional cell is denoted as $\langle \alpha_1, \alpha_2 \rangle(1)$. In a similar manner, the orbit representing edge e_8 using the same dart, denoted as $\langle \alpha_0, \alpha_2 \rangle(1)$, returns the set of darts $\{1,2,20,21\}$ by using involutions α_0 and α_2 (Figure 4.13c). Lastly, all the darts belonging to the face f_1 can be obtained by any combination of involutions α_0 and α_1 returning the set of darts $\{1,2,3,4,5,6,7,8\}$ in which case the orbit is denoted as $\langle \alpha_0, \alpha_1 \rangle(1)$ (Figure 4.13d).

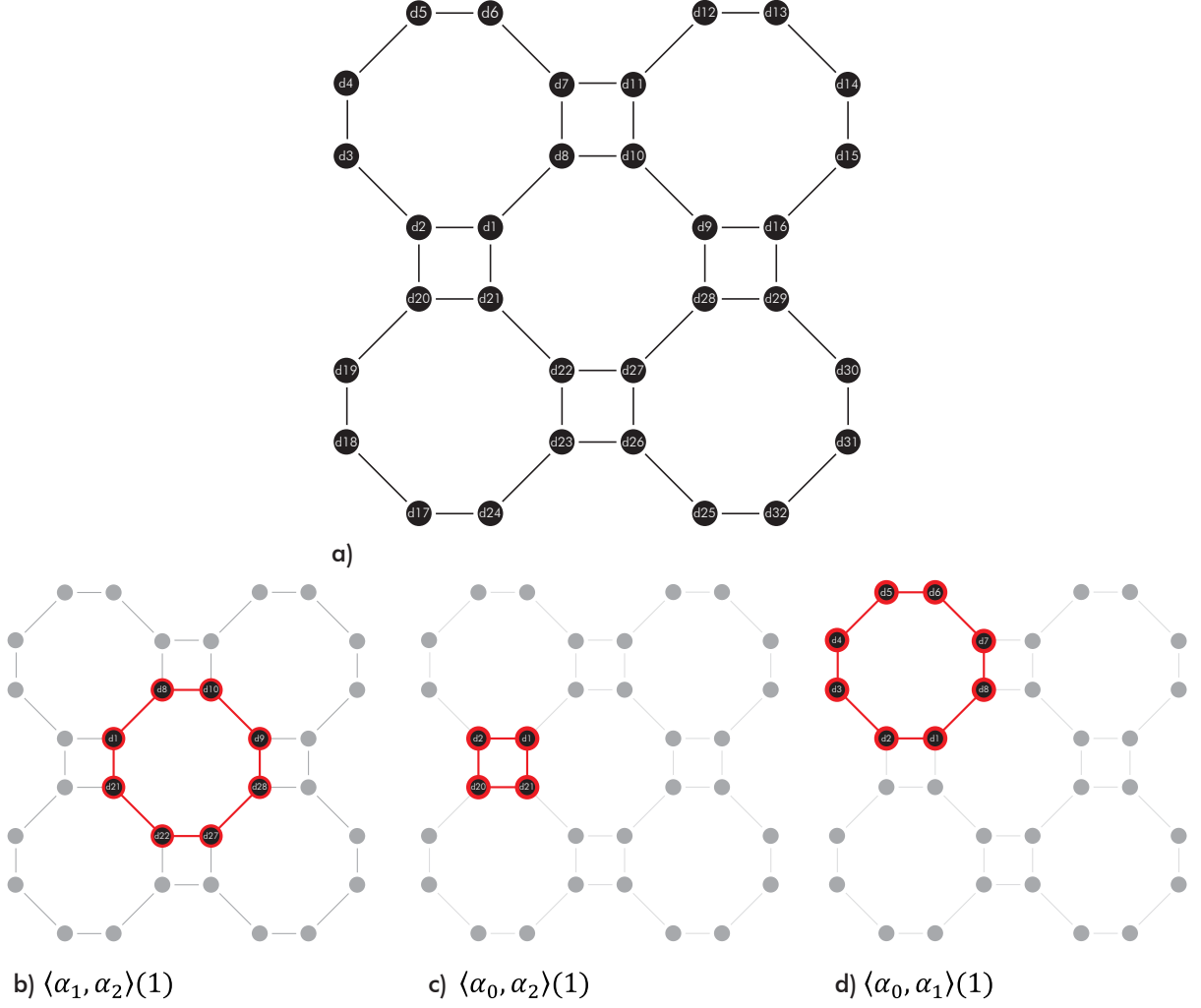


Fig. 4.13

Different types of orbits on a combinatorial map.

From a graph perspective, an orbit is a general type of subgraph that is constructed from a vertex and a combinatorial set of edges. They constitute the fundamental medium for the description of cellular structures, but the notion of orbits may not be limited to the representation of explicitly defined cells. In the specific context of a PS model, a one-to-one correspondence with mechanical elements can be established on the basis of orbits. For instance, the orbit $\langle \alpha_1, \alpha_2 \rangle(1)$ representing edge e8 can be equivalent to an elastic bar element when the orbit contains additional information to define its initial length and material/geometric stiffness. The idea would be to find the most appropriate orbit to embed information of a given mechanical element or set of elements.

Combinatorial maps are implemented through special types of data structures permitting an efficient way to solve topologic queries at constant time and also to modify information. In the case of half-facet or radial-edge data structures supporting non-manifolds, a quasi-one-to-one correspondence between orbits and mechanical elements can, in most of the cases, be established. On the contrary, for half-edge or quad-edge data structures that only support manifolds, it is important to assume that only part of the mechanical elements can be directly mapped onto the topologic model and that these data structure may not be practical for dealing with non-manifolds with a dimensionality large than two. In any case, an orbit can be used to represent explicitly defined cells associated with the initial cellular decomposition, but also to describe cells that are associated with the topologic requirements of mechanical elements. Those cells are constructed from unspoken associations using topologic queries.

4.2.4 EMBEDDINGS

At this point, an important consideration that needs to be taken is that Gmaps can include complementary information, called embeddings, that can be used to determine the geometric and mechanical properties of PS models. From a conceptual viewpoint, embeddings cause different layers of abstraction that can be modeled as a graph (Figure 4.14). Eventually, this entire set of graphs can be used to comprehensive describe the structure of a PS model. A graph $G = (V; E)$ is the most basic mathematical entity for representing and analyzing objects and their relationships. It consists of a finite set of vertices V in combination with a non-empty set of edges E in which each edge $e \in E$ connects a pair of vertices. The lowest level of abstraction is represented by an undirected graph describing the structure of the Gmap in a way that vertices are darts and edges are involutions (Figure 4.14c). This graph represents the mode of storing and retrieving topologic data through maps in which case the equivalences with cells are established on the basis of orbits. The second level of abstraction correspond to embeddings of mechanical properties that are represented via directed graphs (Figure 4.14b). Here, a vertex is equivalent to an orbit defining a zero-dimensional cell of the Gmap with embedded information regarding geometric

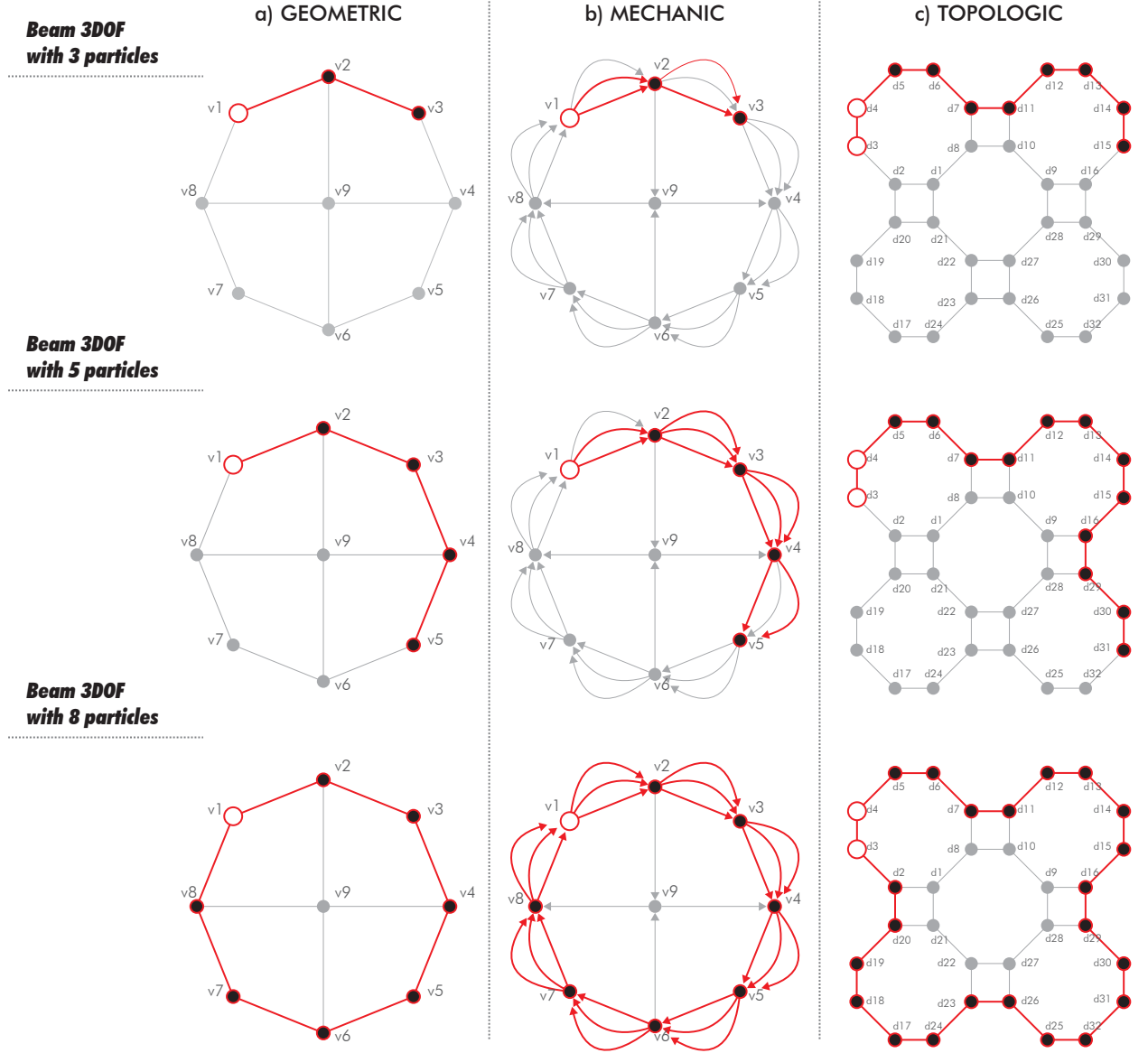
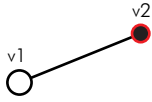


Fig. 4.14

Layers of abstraction caused by embeddings associated with a beam element with 3DOF per particle. a) Geometric, b) Mechanic, c) Topologic.

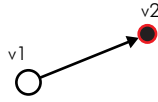
and mechanic properties. To put it in another way, vertices of this graph are equivalent to the notion of particles. Directed edges are in turn used to describe the connectivity among these particles for computing the required forces/potential-energies. Hence, a subgraph can be associated with a particular type of mechanical element or set of elements (Figure 4.15). This means that subgraphs may be equivalent to orbits representing nonzero-dimensional cells on the Gmap. Finally, the highest level of abstraction corresponds to a simple undirected graph describing the initial cellular decomposition of the PS model (Figure 4.14a).

a. ELASTIC BAR



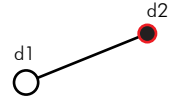
Geometric Graph

Min. Number of Vertices: 2
Min. Number of Edges: 1



Mechanic Graph

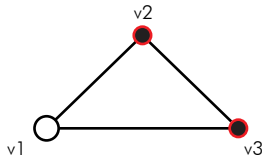
Min. Number of Vertices: 2
Min. Number of Edges: 1



Topologic Graph

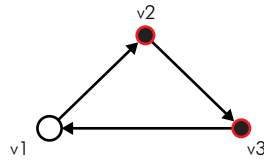
Min. Number of Vertices: 2
Min. Number of Edges: 1

b. CONSTANT STRAIN TRIANGLE



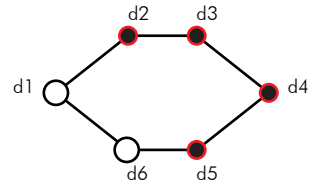
Geometric Graph

Min. Number of Vertices: 3
Min. Number of Edges: 3



Mechanic Graph

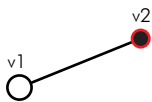
Min. Number of Vertices: 3
Min. Number of Edges: 3



Topologic Graph

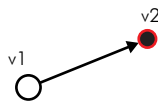
Min. Number of Vertices: 3
Min. Number of Edges: 3

c. BEAM 6DOF (per particle)



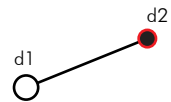
Geometric Graph

Min. Number of Vertices: 2
Min. Number of Edges: 1



Mechanic Graph

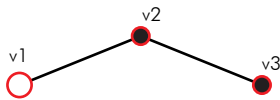
Min. Number of Vertices: 2
Min. Number of Edges: 1



Topologic Graph

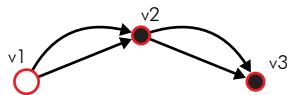
Min. Number of Vertices: 2
Min. Number of Edges: 1

d. BEAM 3DOF (per particle)



Geometric Graph

Min. Number of Vertices: 3
Min. Number of Edges: 2



Mechanic Graph

Min. Number of Vertices: 3
Min. Number of Edges: 4



Topologic Graph

Min. Number of Vertices: 6
Min. Number of Edges: 5

Fig. 4.15

Subgraphs associated with basic mechanical elements.

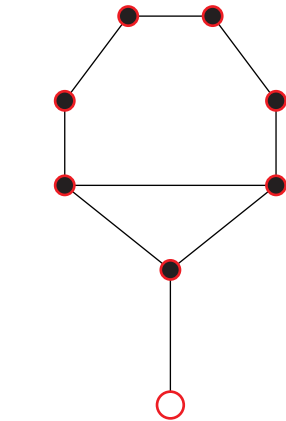
This graph is embedded within a three-dimensional Euclidean space to represent a geometric model which is what designers ultimately see displayed on their screen. Vertices are equivalent to points positioned in the space and edges describing the global connectivity of the cellular decomposition. Subgraphs have a one-to-one correspondence with nonzero-dimensional cells. Graphs describing the mechanical and topologic properties are implicitly rooted within the geometric graph but, for the sake of simplicity, they don't need to be explicitly visualized.

4.2.5 SUBGRAPHS

Gmaps and embeddings permit to efficiently store, retrieve and modify all the properties of a PS model, but they do not provide additional information to specify how to address those transformations over time or through direct user interactions. Since any type of information can be embedded, it is also possible to incorporate simple rules to define valid operations that do not compromise topologic, mechanic or geometric properties. For this purpose, graph grammars need to be introduced to specify all allowed transformations on the corresponding graphs defining the structure of the PS model by using a vocabulary in combination with a set of production rules and axioms. Vertices and edges of each graph are considered to be the vocabulary, whereas axioms are structured combinations of the vocabulary describing basic subgraphs. An axiom describes a small part of the PS model which is why it has an equivalent representation among all the different graphs describing its complete structure. At this point, graph grammars are only applied on subgraphs defined on the basis of the mechanical elements presented in the previous chapter. To avoid confusions with new terminology, subgraphs addressed in the context of the mechanic graph will be referred in the following as building blocks. The proposed topologic model with embeddings is extendible for other types subgraphs and eventually more mechanical elements. For the sake of simplicity, basic subgraphs and their equivalences among all graphs are presented in figures 4.15-17.

- **Cable** (Figure 4.15a): A simple type of subgraph consisting of a pair of vertices connected with a directed edge.

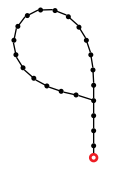
SPLINE-BEAM



Topologic Graph

Min. Number of Vertices: 8
Min. Number of Edges: 9

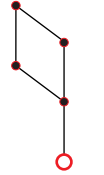
a._



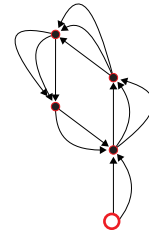
High Res.

Geometric Graph

Min. Number of Vertices: 5
Min. Number of Edges: 5

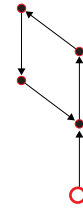


Low Res.



Mechanic Graph

Min. Number of Vertices: 5
Min. Number of Edges (3DOF): 13
Min. Number of Edges (6DOF): 5



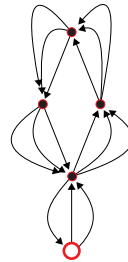
High Res.

Geometric Graph

Min. Number of Vertices: 5
Min. Number of Edges: 5



Low Res.



Mechanic Graph

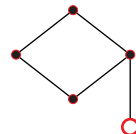
Min. Number of Vertices: 5
Min. Number of Edges (3DOF): 15
Min. Number of Edges (6DOF): 5



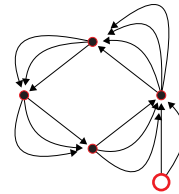
High Res.

Geometric Graph

Min. Number of Vertices: 5
Min. Number of Edges: 5

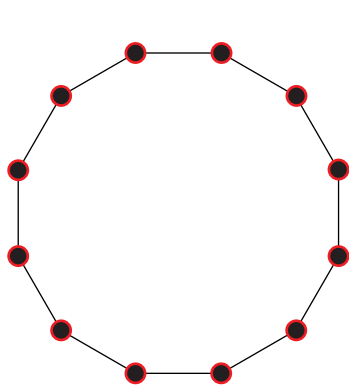
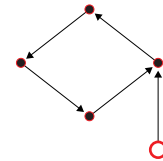


Low Res.



Mechanic Graph

Min. Number of Vertices: 5
Min. Number of Edges (3DOF): 15
Min. Number of Edges (6DOF): 5



Topologic Graph

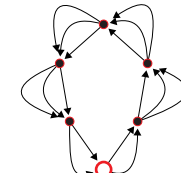
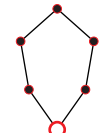
Min. Number of Vertices: 12
Min. Number of Edges: 12

b._



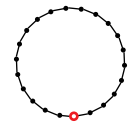
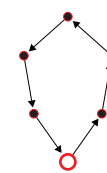
Geometric Graph

Min. Number of Vertices: 6
Min. Number of Edges: 6



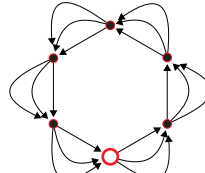
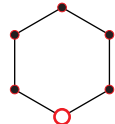
Mechanic Graph

Min. Number of Vertices: 6
Min. Number of Edges (3DOF): 16
Min. Number of Edges (6DOF): 6



Geometric Graph

Min. Number of Vertices: 6
Min. Number of Edges: 5



Mechanic Graph

Min. Number of Vertices: 6
Min. Number of Edges (3DOF): 18
Min. Number of Edges (6DOF): 6

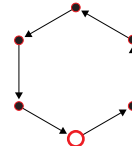


Fig. 4.16

Subgraphs associated with a spline-beam.

• **Spline-beam** (Figure 4.15-4.16): A simple directed subgraph composed by n interior vertices with valence 2 and two boundary vertices with valence 1, in which case the set of edges creates a Hamiltonian path (Figure 4.15c-4.15d). That is, a path that visits each vertex exactly once. This condition is also applied for spline-beams represented by directed cycle subgraphs where all vertices have a valence 2 (Figure 4.16b). A special case that is also considered is a simple directed subgraph composed by n interior vertices with valence 2, one boundary vertex with valence 1 and one interior vertex with valence 3. In this case, the set of edges does not create a Hamiltonian path (Figure 4.16a). This graph is associated with mechanical elements such as elastic bars, beam with 3DOF per particle and beams with 6DOF per particle. The topologic conditions for computing the corresponding force/potential energies are therefore inferred from the connectivity of the subgraph.

TENSILE SURFACE PATCH

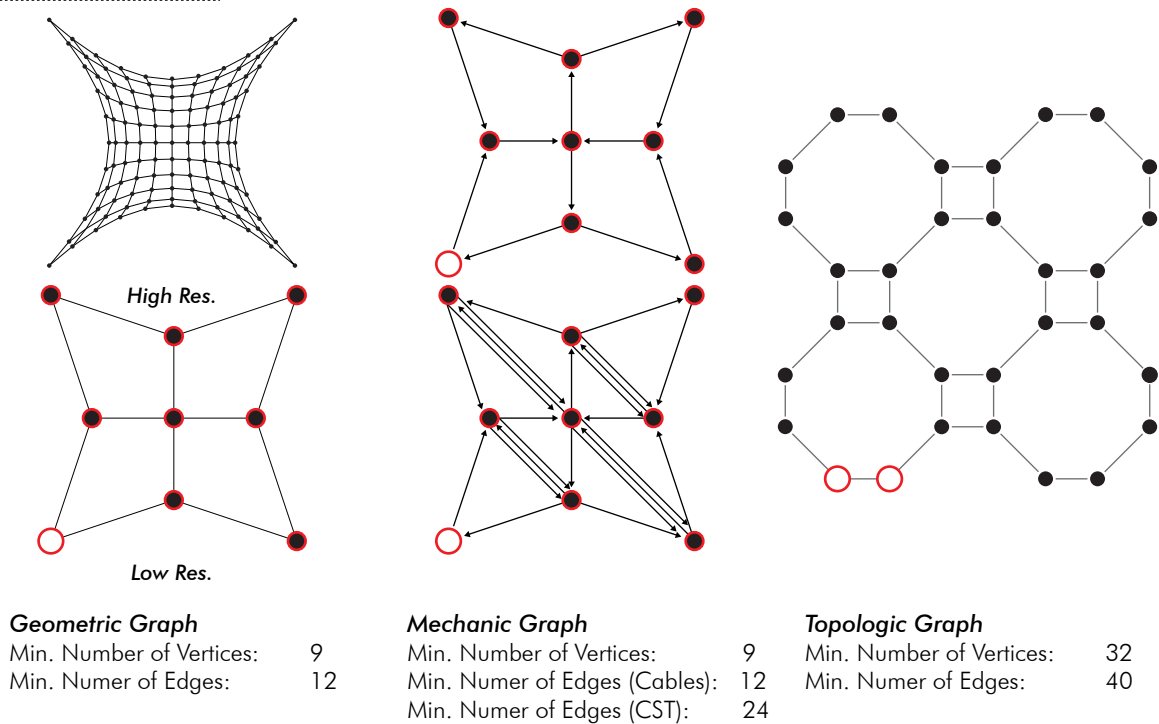


Fig. 4.17

Subgraphs associated with a tensile patch.

• **Tensile surface patch** (Figure 4.17): A square grid

subgraph composed by boundary vertices with valence 2 and 3, interior vertices with valence 4 and directed edges. This graph is associated with mechanical elements such as elastic bars and CST. In the case of elastic bars, a one-to-one correspondence with the edges of the graph can be created. However, the topologic conditions for CST elements need to be inferred from the connectivity of the subgraph.

4.2.6 VERTEX STATE

To enhance the vocabulary of graphs, additional attributes are embedded on vertices to have a specific description of their current state. The state of a vertex is first determined at the level of the mechanic graph and then spread to the rest of graphs (geometric and topologic). This state is used to easily describe the role that a particle has among different mechanical elements. It is therefore determined by the different types of building blocks (i.e. subgraphs) to which it is attached and by the type of motion constraint to which is subject. To facilitate the visualization of such attributes, vertices are colour coded in relation to their state as followed:

- ***Isolated (Black)***: a particle without any type of building block associated to it.
- ***Passive (Grey)***: a particle temporarily excluded from the integration process that has at least one building block associated with it.
- ***Lock (Red)***: a particle whose translational motion is constrained in at least one of its local axes.
- ***Full-Lock (Yellow)***: a particle whose translational and rotational motion are constrained in at least one of its respective local axes.
- ***Simple (Cyan)***: a particle without any type of motion constraint that has only one building block associated with it.

- **Shared (Purple)**: a particle without any type of motion constraint that has two or more building blocks associated with it.

4.2.7 GRAPH GRAMMARS

Graph Grammars are formal and generative techniques established from a vocabulary and a set of production rules that determine how to create, delete or modify vertices and edges (Helms 2012). The implementation of graph grammars permits the extendibility of the proposed topologic model with embeddings since any type of topologic transformation can be applied if a production rule is adequately formulated for it. It is important to note that the given set of vertex states can also be extendable if a given graph grammar requires it. A production rule R consists of a conditional part A that when evaluated to true, produces a resultant part B , such that $R: A \rightarrow B$. The conditional part (left-hand side term) is a subgraph, sometimes called host-graph, that determines when the rule can be applied (axiom). The resultant part (right-hand side term) contains a finite set of sequential operations to automatically modify the data structure so that the old subgraph A can be converted into the new subgraph B .

Graph grammars act across all the corresponding graphs of the PS model and updates the state of their vertices to avoid further inconsistencies. To describe them, let be G the geometric graph, M the mechanical graph and T the topologic graph. Particles are modeled via subgraphs in T but have a one-to-one correspondence with vertices in G and M . In some cases, the three graphs representing all the abstract layers of the PS model can be disconnected for modeling isolated particles. A detailed information of rules is presented below. They are grouped into rulesets and consist of basic operations for adding, deleting and modifying vertices, edges and subgraphs.

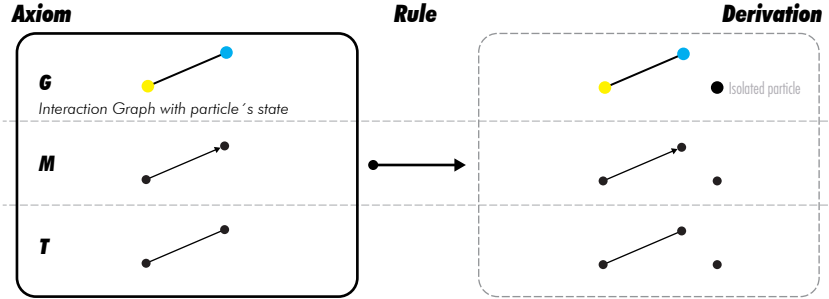
4.2.7.1 Ruleset for Particles

Rule 1 (Figure 4.18). *Adding individual particles.*

(Axiom) – If M , G and T are graphs with empty sets of vertices and edges (null-graphs) or graphs in which each vertex is incident

to at least one edge (connected graphs); (**Production rule**) – **Then**, a particle p_{new} can be created by adding an isolated vertex in each of the corresponding graphs. Due to the possibility of external forces, the state is initially defined as “lock” to avoid p_{new} moving towards infinity.

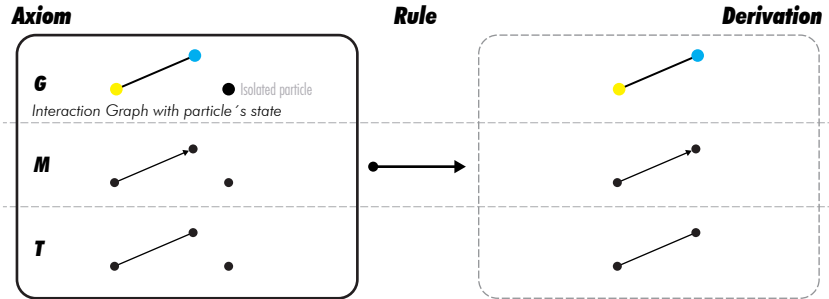
Fig. 4.18
Rule 1.



Rule 2 (Figure 4.19). *Deleting individual particles.*

(**Axiom**) – If p_{old} is a particle is equivalent to an isolated vertex in each graph which means that M, G and T are graphs with empty sets of edges (empty-graphs) or graphs containing isolated vertices (disconnected graphs); (**Production rule**) – **Then**, p_{old} is deleted by erasing the corresponding vertex in each graph.

Fig. 4.19
Rule 2.



Rule 3 (Figure 4.20). *Merging particles.*

(**Axiom**) – If a subset of particles P contains only particles with a simple or shared state, constrained (lock or full-lock) or not; (**Production rule**) – **Then**, the entire subset P is deleted and a new particle p_{new} is created by combining all the information of the subset. Hence, p_{new} is placed at an average position and with an average initial velocity. Furthermore, the motion of the new particle is constrained if at least one particle of the subset was also constrained. This is equivalent to connect multiple building blocks together.

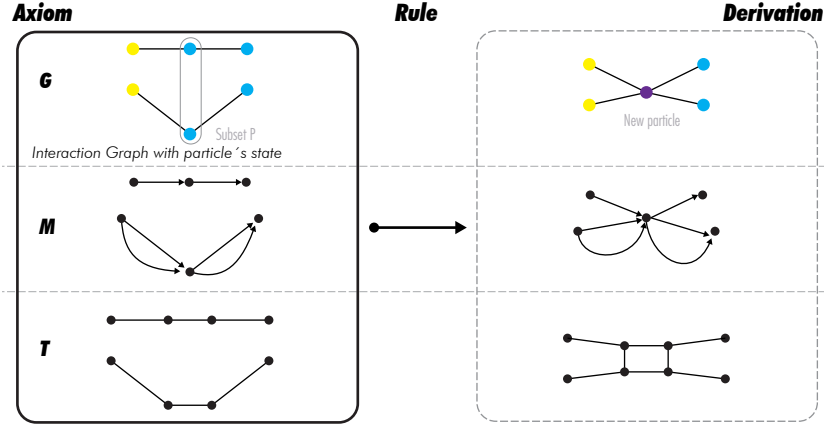


Fig. 4.20

Rule 3.

Rule 4 (Figure 4.21). Exploding particles.

(Axiom) – If a particle p_{old} has a shared state, constrained (lock or full-lock) or not; **(Production rule)** – **Then**, particle p_{old} is deleted and a new subset of particles P is created by adding a new particle p_i to each building block containing p_{old} . Each newly created particle of the subset is initialized with the position and velocity of p_{old} . This is equivalent to disconnect multiple building block.

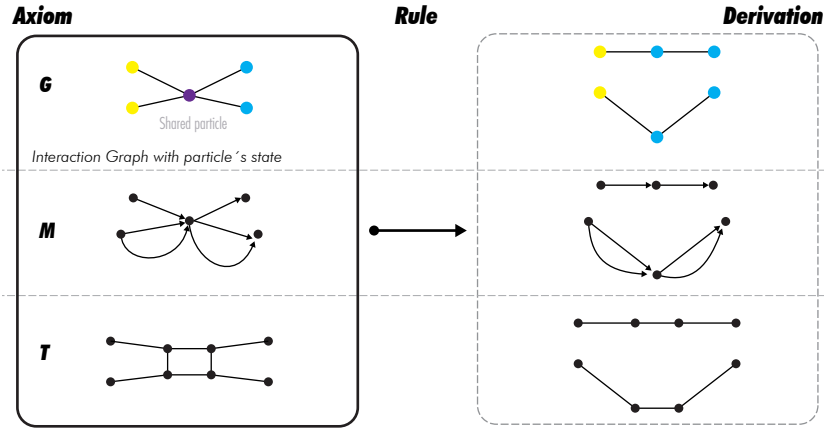


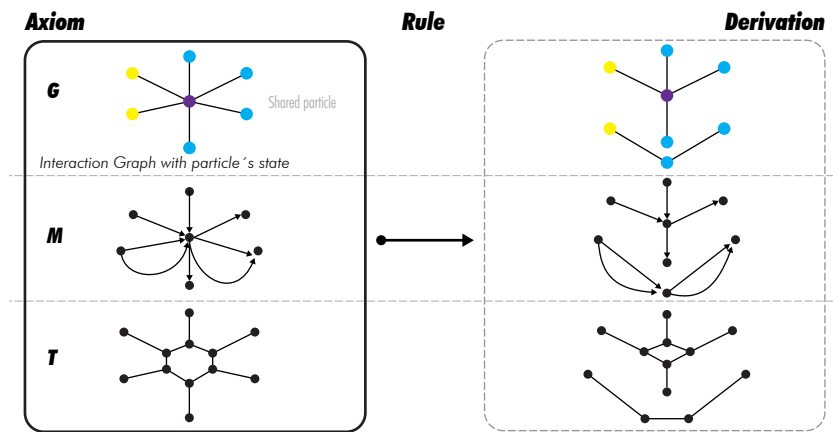
Fig. 4.21

Rule 4.

Rule 5 (Figure 4.22). Exploding specific particles.

(Axiom) – If a particle p_{old} has a shared state, constrained (lock or full-lock) or not; **(Production rule)** – **Then**, a new particle p_{new} associated with a given building block is created from particle p_{old} . This newly created particle is initialized with the position of and velocity of p_{old} . This is equivalent to disconnect specific building blocks.

Fig. 4.22
Rule 5.

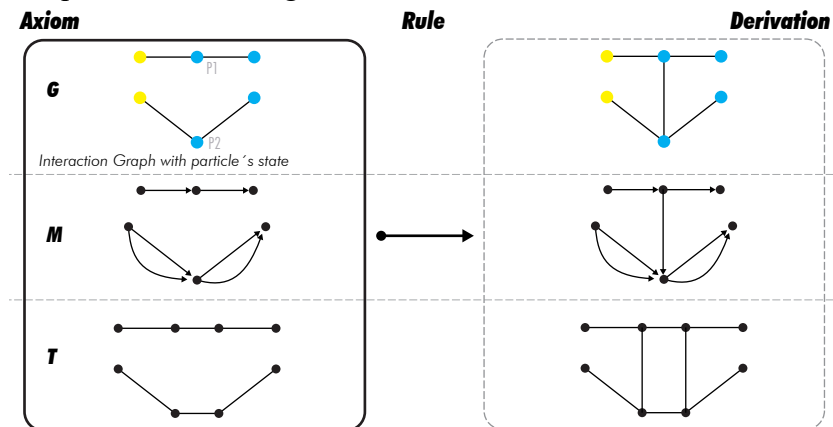


4.2.7.2 Ruleset for Cables

Rule 6 (Figure 4.23). *Creation of a cable.*

(Axiom) – If two particles p_1 and p_2 are selected; **(Production rule)** – **Then**, a link between both particles can be created which is equivalent to adding a cable element.

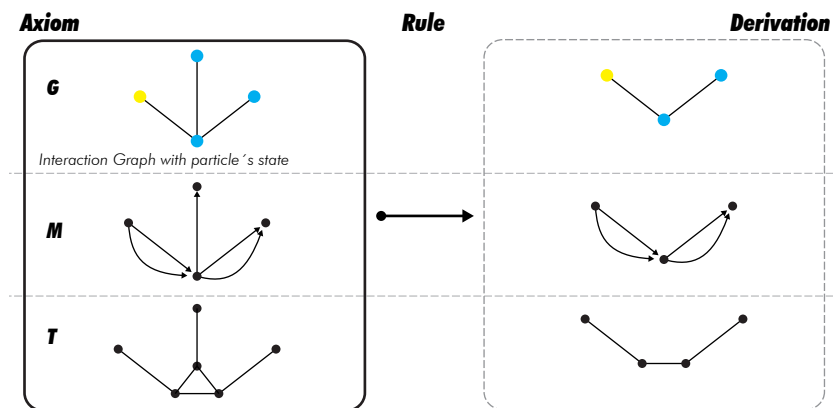
Fig. 4.23
Rule 6.



Rule 7 (Figure 4.24). *Deletion of a cable.*

(Axiom) – If subgraph P represents a pair of connected particles; **(Production rule)** – **Then**, the link between both particles is removed and isolated particles are the deleted by applying rule 2.

Fig. 4.24
Rule 7.



4.2.7.3 Ruleset for Beams

Rule 8 (Figure 4.25). *Creation of a spline beam from pre-existent particles.*

(Axiom) – If a subset of particles P contains at least one particle with any type of state and at least two particles to be added;
(Production rule) – **Then**, the required particles are first added and then the mechanical links on the M graph for computing forces/potential energies of either a beam with 6DOF- or 3DOF- per particle are also created. If the particles are not explicitly connected, then the corresponding links on G and T .

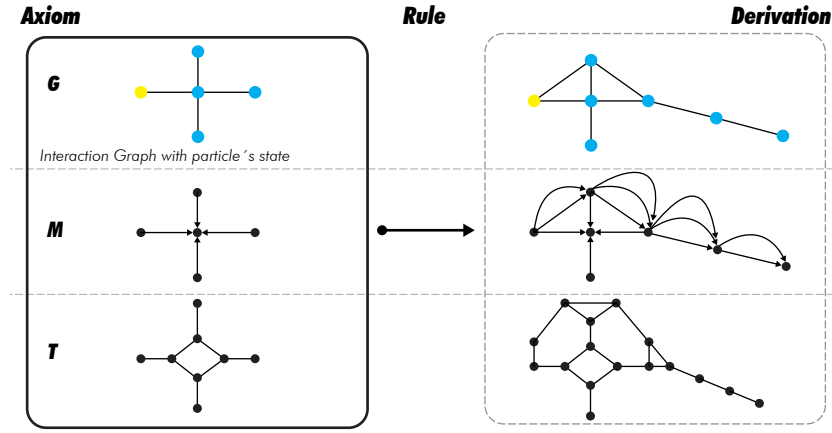


Fig. 4.25
Rule 8.

Rule 9 (Figure 4.26). *Creation of a spline beam from newly added particles.*

(Axiom) – If an initially empty subset of particles P is progressively fill with particles by using rule 1 until attaining a minimum number of three particles; **(Production rule)** – **Then**, the mechanical links on M are dynamically created for computing forces/potential energies of either beams with 6- or 3-DOF- per particle, as well as the corresponding connectivity for G and T .

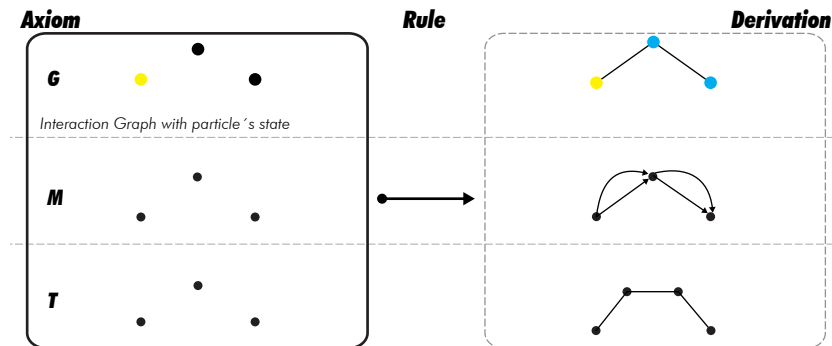


Fig. 4.26
Rule 9.

Rule 10 (Figure 4.27). *Creation of a spline beam from a particle.*
(Axiom) – If a particle p_{old} is selected and a vector direction is specified with a predefined number of subdivisions; **(Production rule)** – **Then**, a new subgraph is created on each of the corresponding graphs for computing forces/potential energies of either a beam with 6DOF- or 3DOF- per particle.

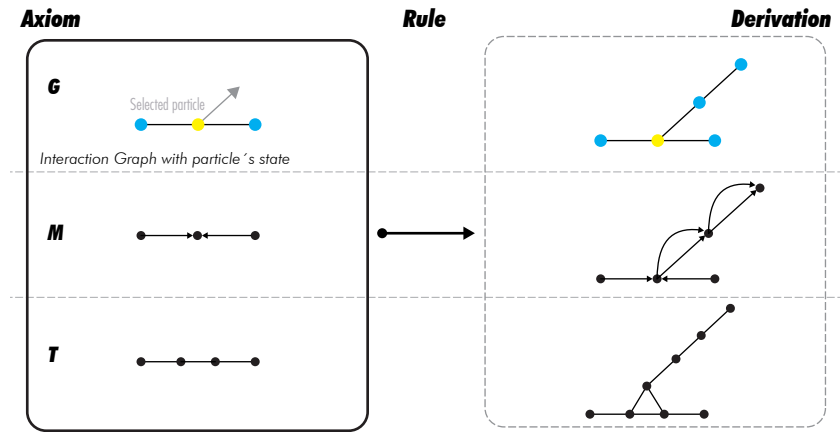


Fig. 4.27
Rule 10.

Rule 11 (Figure 4.28). *Deletion of a simple particle on a spline beam.*

(Axiom) – If a particle p_{old} is part of a spline beam and its state is simple; **(Production rule)** – **Then**, the particle is deleted by first removing all the links on the corresponding graphs and then deleting the isolated vertices by applying rule 2. In case that the particle is not an end-particle, then the two neighbors are bridged with new connections.

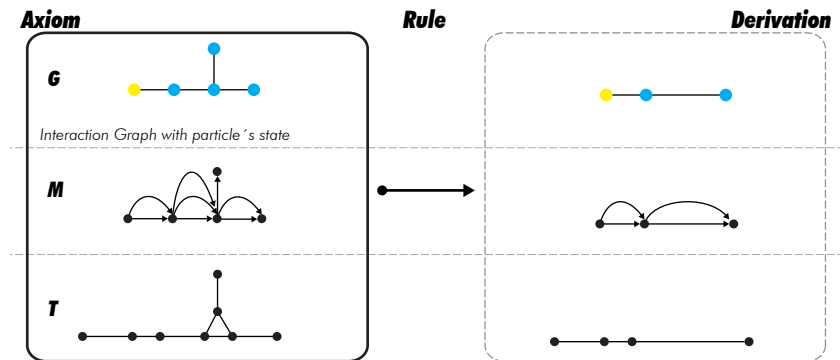
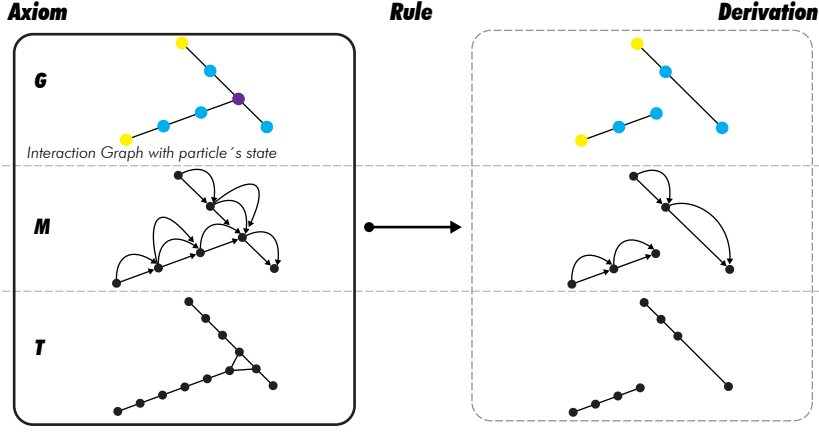


Fig. 4.28
Rule 11.

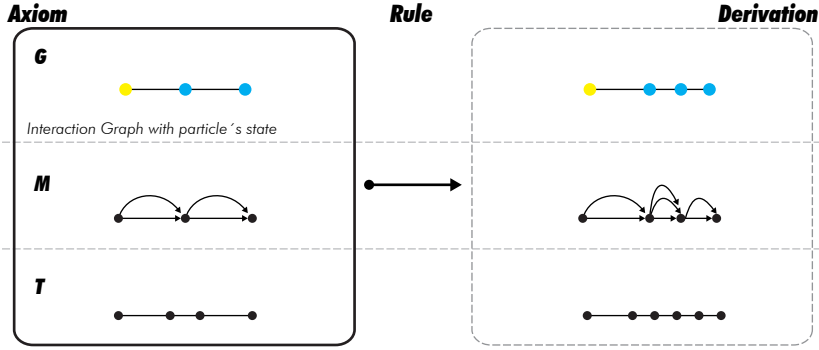
Rule 12 (Figure 4.29). *Deletion of a shared particle on a spline beam.*

(Axiom) – If a particle p_{old} is part of a spline beam and its state is shared; **(Production rule)** – **Then**, the particle is first exploded using rule 4 and then, the corresponding new particles with a simple state are deleted using rule 11.


 Fig. 4.29
Rule 12.

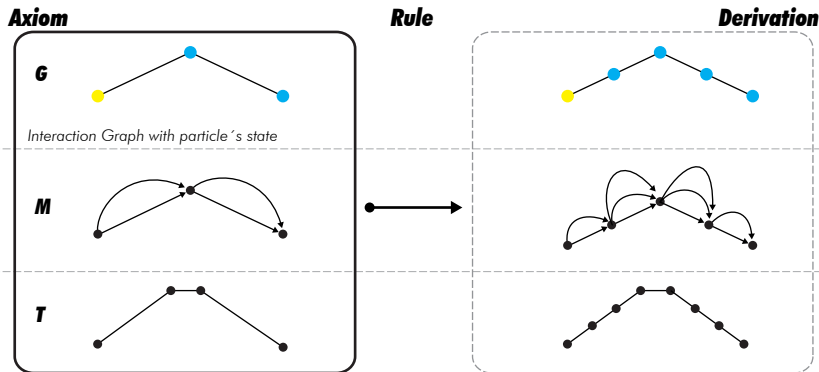
Rule 13 (Figure 4.30). *Adding a particle on a spline beam.*

(Axiom) – If a pair of particles, p_1 and p_2 , is selected and both particles are consecutive; **(Production rule)** – **Then**, a new particle p_{new} is created between the given pair using an average of their position and velocity. All the links on the corresponding graphs are updated accordingly.


 Fig. 4.30
Rule 13.

Rule 14 (Figure 4.31). *Subdivision of a spline beam.*

(Axiom) – If a subset of particles P defines a spline beam element; **(Production rule)** – **Then**, for each pair of two consecutive particles, p_1 and p_2 , a new particle p_i is created by applying rule 11.


 Fig. 4.31
Rule 14.

Rule 15 (Figure 4.32). *Inverse subdivision of a spline beam.*

(Axiom) – If a subset of particles P defines a spline beam element;
(Production rule) – **Then**, for each sequence of three consecutive particles the mid particle is deleted with rule 11.

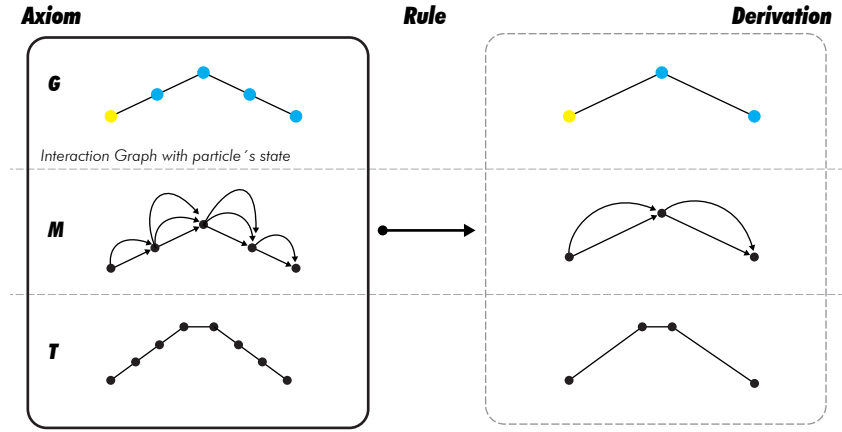


Fig. 4.32
Rule 15.

Rule 16 (Figure 4.33). *Breaking a spline beam on a simple particle.*

(Axiom) – If a particle p_{old} is part of a spline beam and its state is simple;
(Production rule) – **Then**, a particle p_{new} is created with part of the connectivity of the spline beam associated to it and the other part to p_{old} .

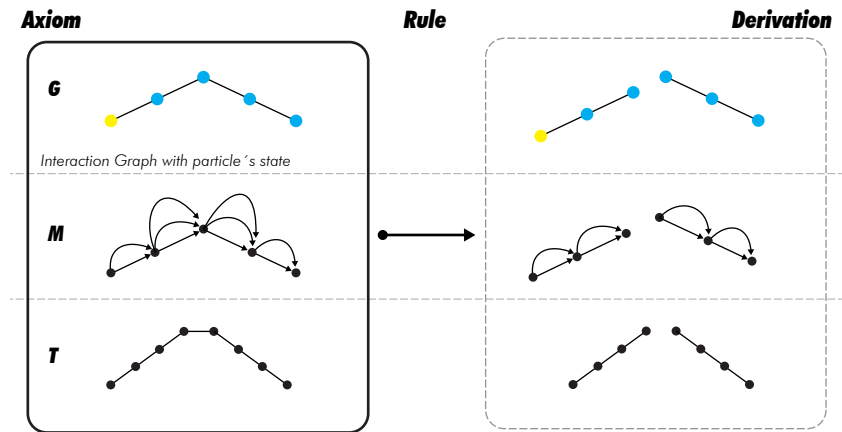


Fig. 4.33
Rule 16.

Rule 17 (Figure 4.34). *Breaking a spline beam on a shared particle.*

(Axiom) – If a particle p_{old} is part of a spline beam and its state is shared;
(Production rule) – **Then**, p_{old} is first selectively exploded with respect to the given building block using rule 5 and then, broken by applying rule 16 on particle p_{new} .

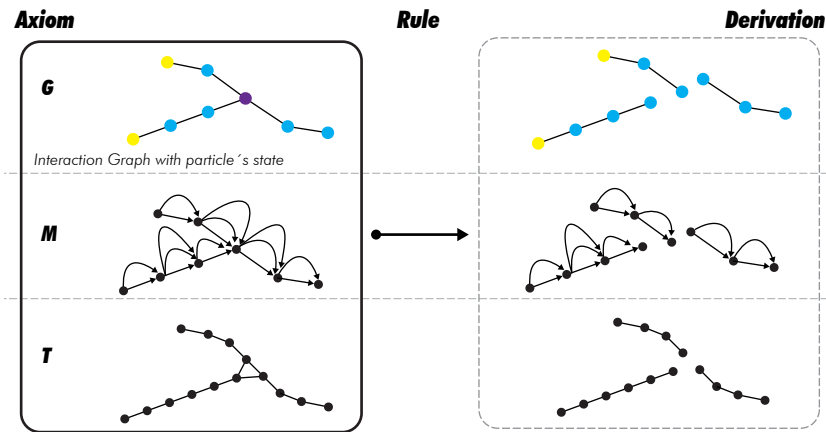


Fig. 4.34

Rule 17.

Rule 18 (Figure 4.35). Deleting a spline beam.

(Axiom) – If a subset of particles P defines a spline beam element;
(Production rule) – **Then**, all the links between these particles are first deleted and all the state updated. In case that the deletion of the spline beam derives into isolated particles, then those particles are deleted by using rule 2.

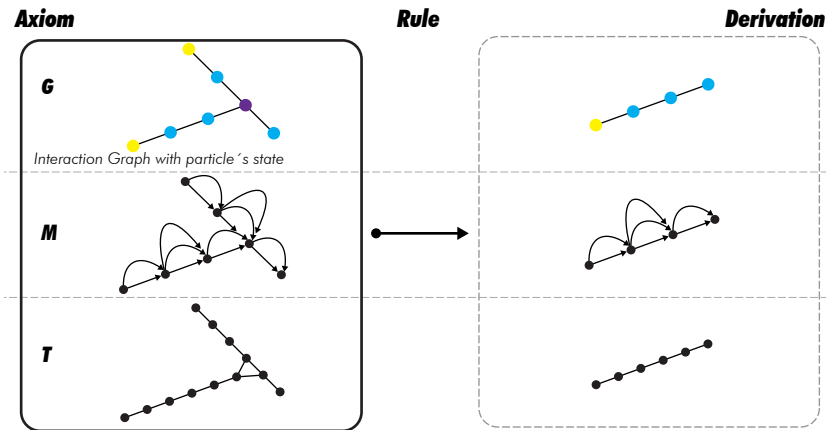


Fig. 4.35

Rule 18.

4.2.7.4 Ruleset for Tensile Surfaces

Rule 19 (Figure 4.36). Creation of a tensile patch from particles using subdivision.

(Axiom) – If a subset of particles P contains at least 3 particles;
(Production rule) – **Then**, a two-dimensional cellular decomposition can be created by using a Catmull-Clark subdivision in which case the state of particles is updated and the newly created particles are defined with a simple state.

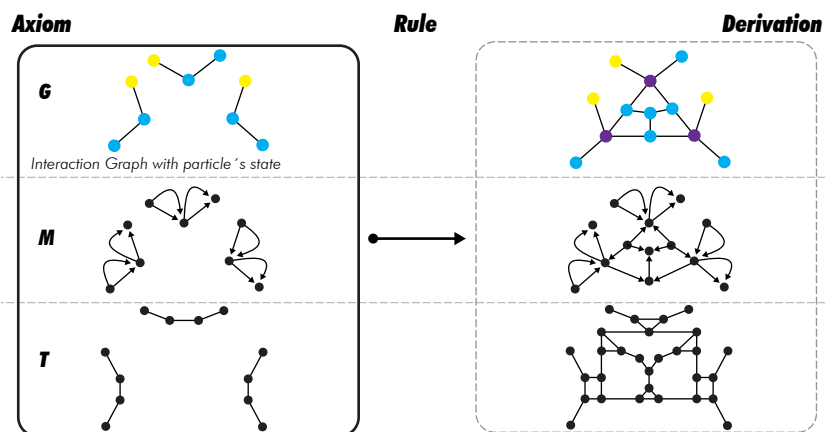


Fig. 4.36

Rule 19.

Rule 20 (Figure 4.37). *Creation of a tensile patch from particles using paving.*

(Axiom) – If a subset of particles P contains at least 8 particles;

(Production rule) – *Then*, a two-dimensional cellular decomposition can be created by using a paving technique in which case the state of particles is updated and the newly created particles are defined with a simple state.

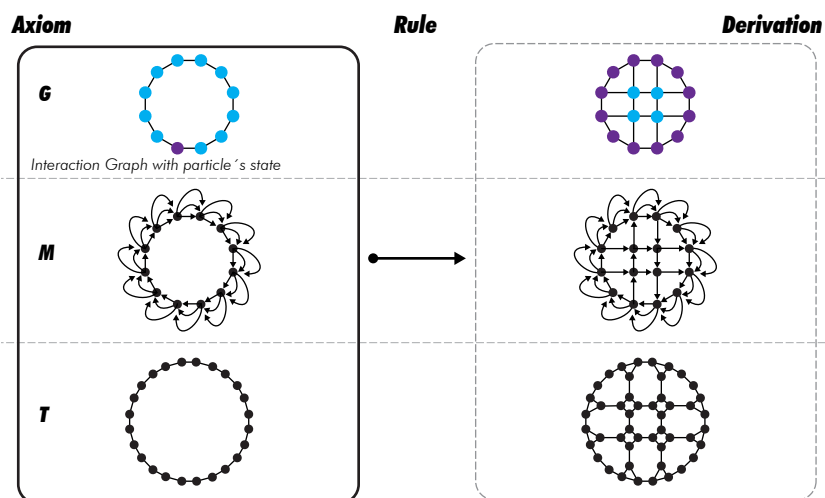


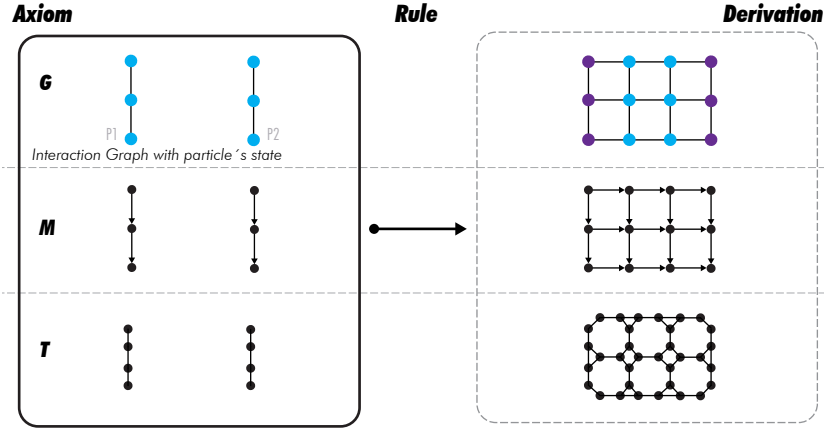
Fig. 4.37

Rule 20.

Rule 21 (Figure 4.38). *Creation of a tensile patch from particles using projections.*

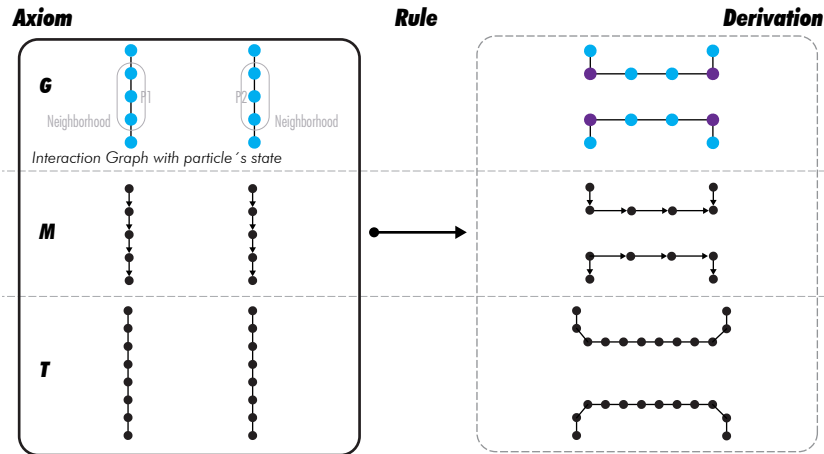
(Axiom) – If two subsets of particles P_1 and P_2 contains an equivalent number of particles; **(Production rule)** – *Then*, a two-dimensional cellular decomposition can be created by mapping

particles from one subset to the other.


 Fig. 4.38
Rule 21.

Rule 22 (Figure 4.39). *Creation of a tensile patch from particles on other patches.*

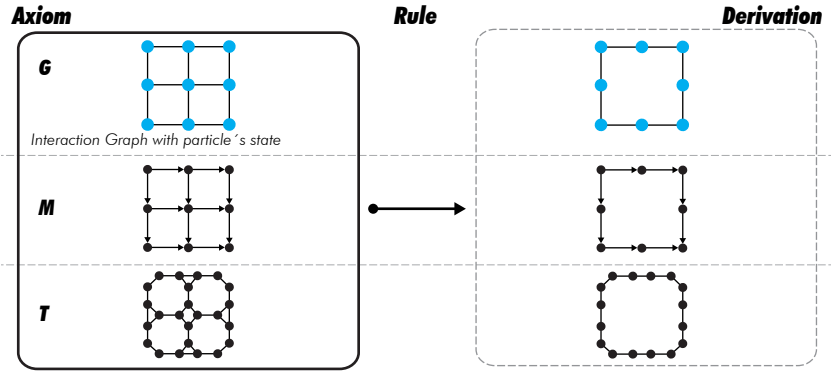
(Axiom) – If two particles p_1 and p_2 with a simple state are placed on the interior part of two patches and contains an equivalent neighborhood; **(Production rule)** – **Then**, all the links of both neighborhoods are deleted as well as isolated particles in order to create a tensile patch on both naked boundaries by using rule 21. On this basis, a kind of volumetric surface is created guaranteeing the 2-manifold of the decomposition.


 Fig. 4.39
Rule 22.

Rule 23 (Figure 4.40). *Deletion of a simple particle on a tensile patch.*

(Axiom) – If a particle p_{old} is part of a tensile patch and its state is simple; **(Production rule)** – **Then**, the particle is deleted as well as all incident cells on the corresponding graphs.

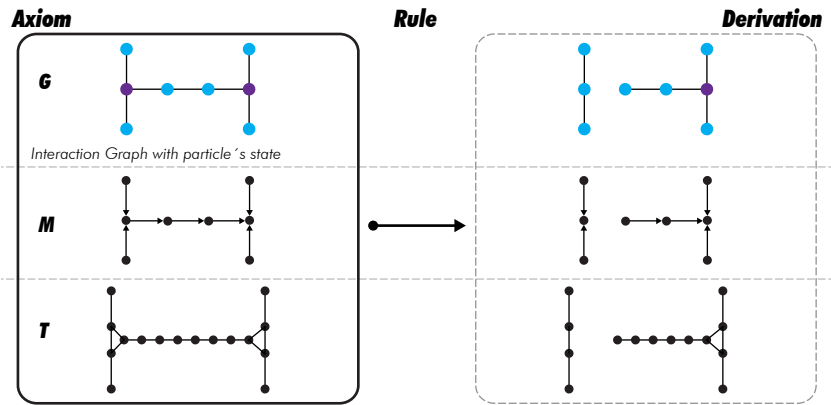
Fig. 4.40
Rule 23.



Rule 24 (Figure 4.41). *Deletion of a shared particle on a tensile patch.*

(Axiom) – If a particle p_{old} is part of a tensile patch and its state is shared; **(Production rule)** – **Then**, the particle is first exploded using rule 4 and then simple particles are selectively deleted with rule 23.

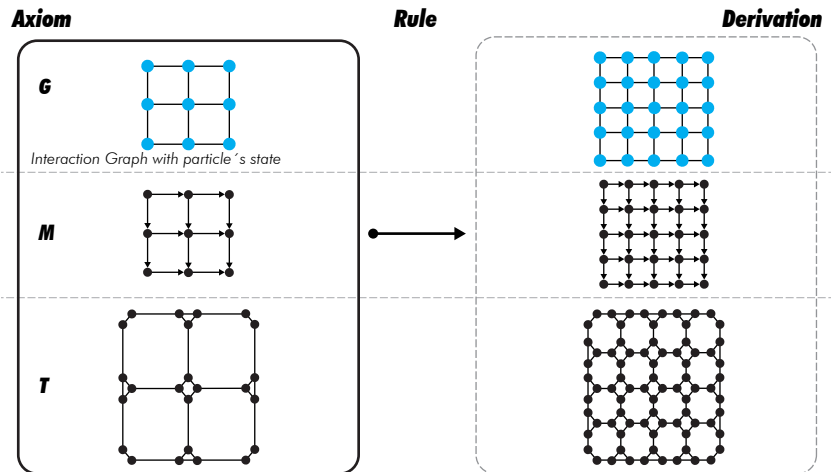
Fig. 4.41
Rule 24.



Rule 25 (Figure 4.42). *Subdivision of a tensile patch.*

(Axiom) – If a subgraph P defines a tensile patch; **(Production rule)** – **Then**, a Catmull-Clark subdivision is applied with new links created on all graphs.

Fig. 4.42
Rule 25.



Rule 26 (Figure 4.43). *Merging of a tensile patch.*

(Axiom) – If two subgraphs, P_1 and P_2 defines two tensile patches with a common subset of particles with shared state; **(Production rule)** – **Then**, both subgraphs can be joined and merged to create one single subgraph.

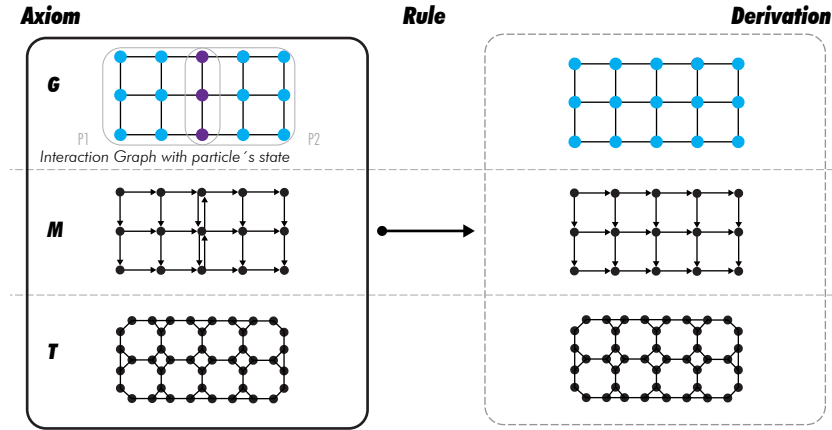


Fig. 4.43

Rule 26.

Rule 27 (Figure 4.44). *Deleting a tensile patch.*

(Axiom) – If a subgraph P_1 defines a tensile patch; **(Production rule)** – **Then**, all the links of the subgraph are first deleted, shared particles selectively exploded with rule 5 and, finally, isolated particles are deleted with rule 2.

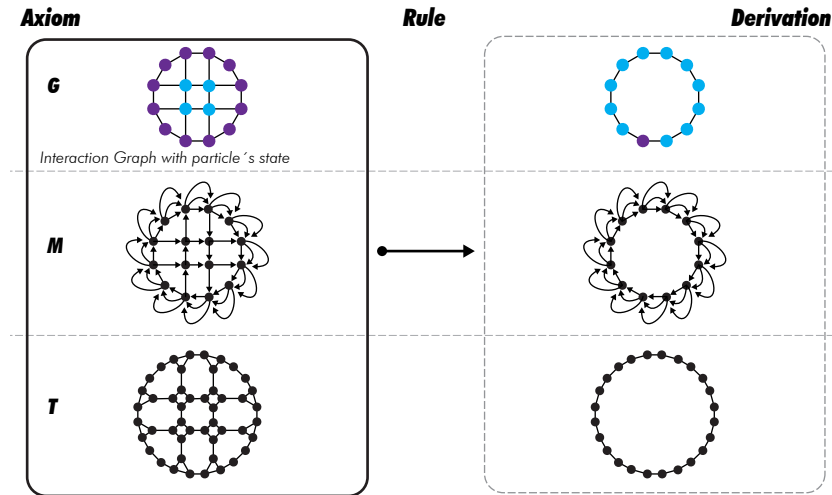


Fig. 4.44

Rule 27.

5

COMPUTATIONAL IMPLEMENTATIONS

5 COMPUTATIONAL IMPLEMENTATIONS

The implementation of numerical simulations in engineering sciences is normally carried by simple and structured routines based on procedural programming paradigm. The purpose is to make an efficient use of computational resources and facilitate the production of highly detailed mechanical models. In most of the cases, the code is so specialized that it cannot be reused neither extended to address a different type of problem. The development of more flexible numerical models can still generate some controversy since this characteristic inversely affects computational performance. However, the basic idea of topology-driven form-finding necessarily implies more flexible implementations permitting the modification of numerical models without breaking the integration process. These types of numerical models are rather difficult to implement without using some of the fundamental concepts of the Object-Oriented Programming (OOP) paradigm such as modularity, encapsulation, inheritance and open recursion. In this context, OOP permits to develop a more robust type of implementation in which code can be easily conceptualize, reuse and extend. Different efforts have already reported that OOP implementations of FEM, through languages like C++ (Zimmermann et al. 1992) or Java (Nikishkov 2010), can reach similar performance levels to those implemented with procedural languages like C or Fortran.

OOP permits to manage a large amount of complexity through the modular concept of “classes” by using abstraction techniques to represent data. A class contains variables and methods to describe the state and behavior of all objects coming from the same type. Note that the instantiation of a class results on the creation of an object. This way, the code of a class can

be reused and extended through the concept of inheritance in order to create a hierarchy of classes. To put it in another way, the use of modularity to handle large pieces of code. Fenves (Fenves 1990) has already described the benefits of developing engineering softwares on the basis of OOP approaches. In 1990, Forde et al. (Forde et al. 1990) described the class structure of basic components in FEM such as nodes, elements, materials, boundary conditions and loads. Later, Archer et al. (Archer et al. 1999) proposed an OOP approach for the development of FEM software in which a flexible and extendable set of classes was used to completely separate the numerical part from the details of the FE-model. Since then, many other efforts have been conducted on the use of OOP for the implementation of numerical models. In this chapter, an OOP implementation of a PS model using our generic topologic model with embeddings is presented along with its use for the development of a software application and physics library.

5.1 OBJECT-ORIENTED PARTICLE SYSTEM

The fundamental requirement to address this implementation is to first separate all the different parts within classes. The top-level classes consist of a *Physical System*, *Topologic Model*, *Numerical Solver*, *Particle Collection* and *Elements Collection*. Figure 5.1 shows the diagram of top-level classes and their associations. A *Physical System* integrates the equations of motions on each *Particle* contained within the *Particle Collection* by calling a method from *Numerical Solver*. This method is associated with a type of integration scheme and damping technique and can be changed at any time. The required forces/potential-energies are obtained from *Elements* contained in *Element Collection* while *Modifiers* contained in *Modifier Collection* are used to change the state of *Elements*. The *Topologic Model* class is a parental class that inherits all the methods and variables to the *Physical System* class for building a comprehensive representation of its connectivity. It refers to a type of data structure build on the concept of combinatorial maps. At this time, a Pointer-based Half-Edge data structure with some modifications and an Array-based Half-Facet data structure have been implemented. A detailed description of both

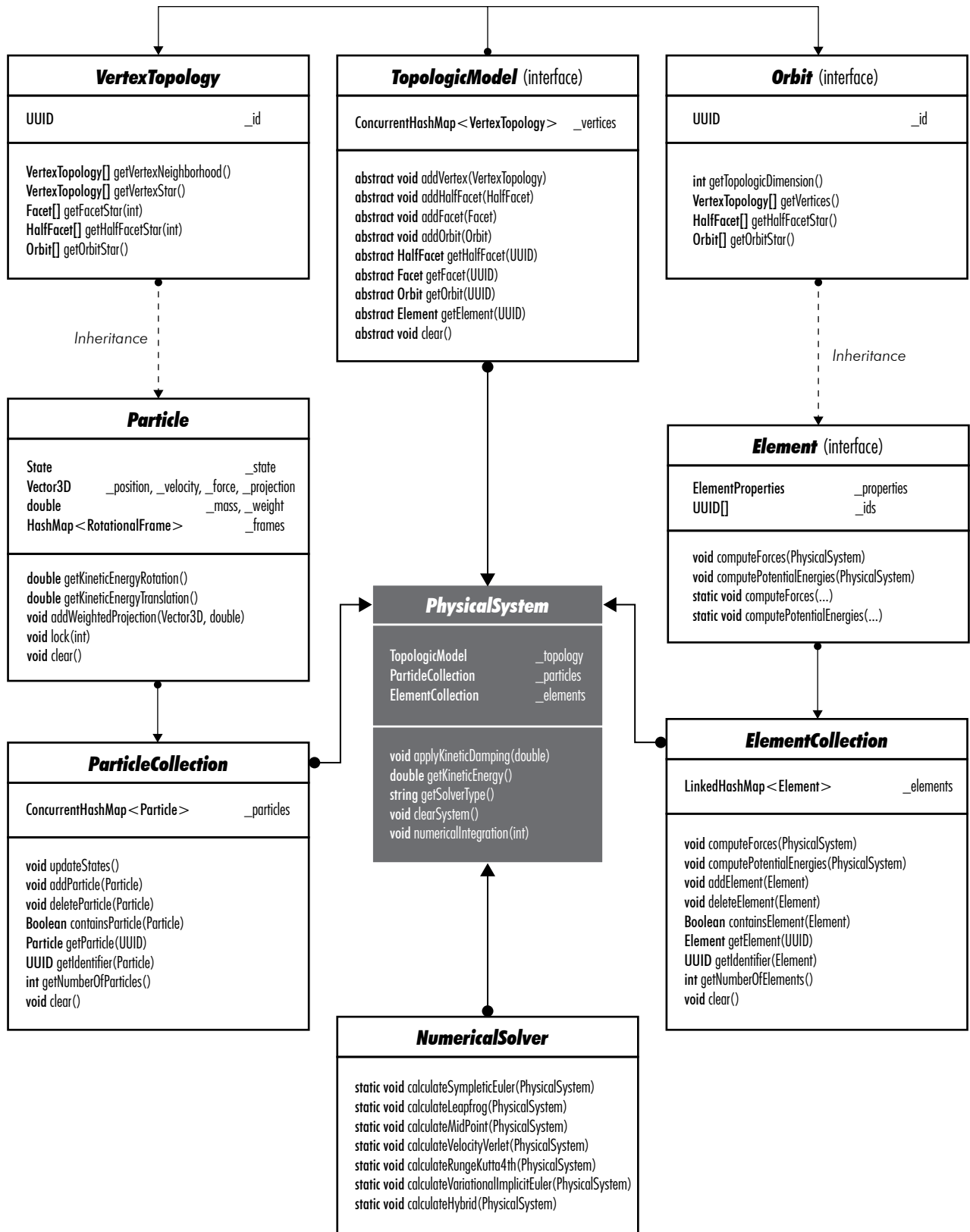


Fig. 5.1

Class taxonomy for implementation.

data structure will come in the following sections. As a result, the *Physical System* class contains different functions that have been incorporated to manage the numerical integration process as well as all types of topologic queries.

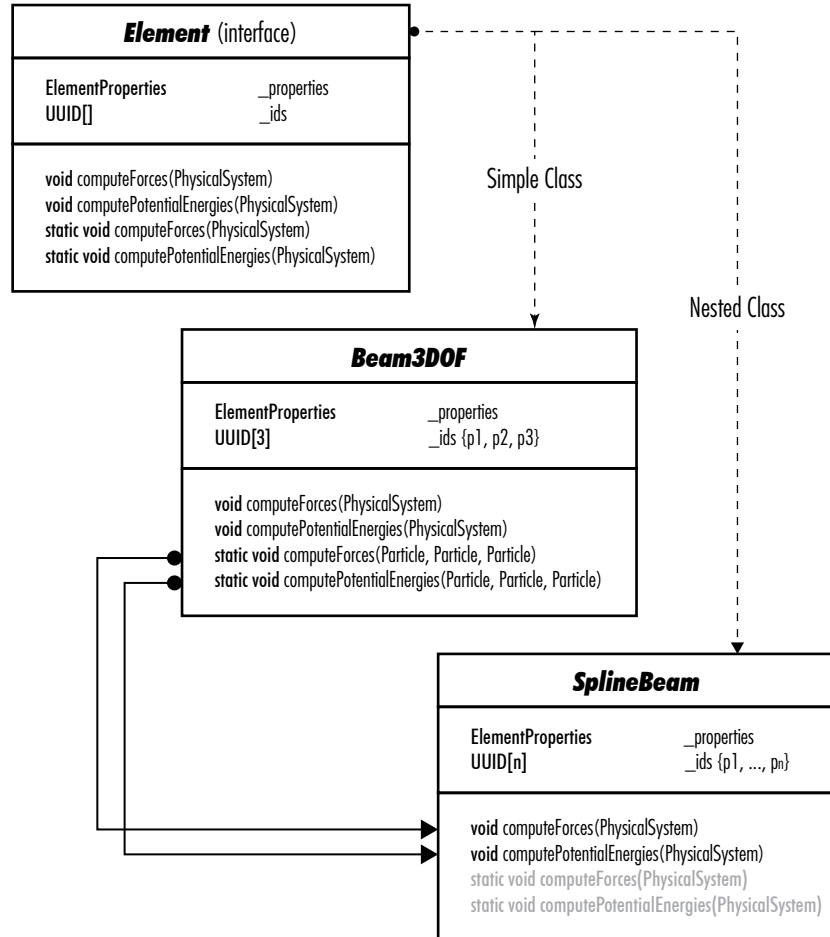


Fig. 5.2

Nested Element class.

A Particle class contains all the mechanic variables presented in chapter 3 as well as methods to calculate forces/potential-energies and kinetic energies. This class inherits a *VertexTopology* class that is entirely associated with the implementation of the data structure associated with a combinatorial map. The state of a particle is in turn defined by a *State* class and store as an object variable in the Particle class. As mentioned before, states constitute an important part of the vocabulary to trigger the application of rules. State and masses of a particle are constantly updated on the basis of *Element* objects associated to it. Particles are stored asynchronously within the *Particle Collection* class by implementing an unsorted associative

array mapping a unique identifier with only one instance of the Particle class. On the other hand, the Element class serves to implement the mechanic formulations described in chapter 3. This class passes four types of methods to compute forces and potential energies from which two are static. This condition permits to derive nested Element classes in a way that formulations to compute forces or potential energies can be dynamically altered by calling static methods of simpler Element classes (Figure 5.2). Such nested element classes are associated with our concept of building blocks. Element classes are then stored within the Element Collection class by implementing a sorted associative array mapping a unique identifier with one instance of the given subclass. This synchronous storage permits to keep track of all the transformations in relation to time. Finally, the Modifier class is a parent class that is used to derive specific subclasses for modifying building blocks by implementing all the topologic modifications described in the previous chapter. In this case, the parent class passes methods for checking axioms as well as for applying production rules (Figure 5.3).

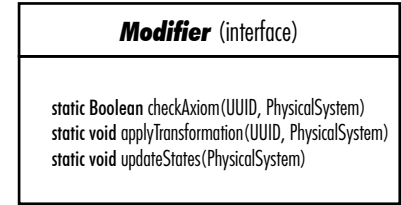


Fig. 5.3

Modifier class.

The selected programming languages for implementation were Java and C#. Java was selected due to its portability between major computer systems and its aptness for an ease development of graphic interfaces. C# was in turn chosen to implement our framework within software applications that are commonly used by designers, architects and engineers. In any case, the great benefit of both programming languages is their built-in garbage collector that permits to simplify the process of preventing memory leaks due to constant modifications of data structures.

5.2 DATA STRUCTURES

5.2.1 POINTER-BASED HALF-EDGE DATA STRUCTURE

The simplest way to implement our topologic model is through a Pointer-based Half-Edge data structure (PHE). A two-dimensional map is equivalent to a PHE data structure with darts being represented by half-edges. The generalized form is a Pointer-based Quad-Edge data structure requiring the double

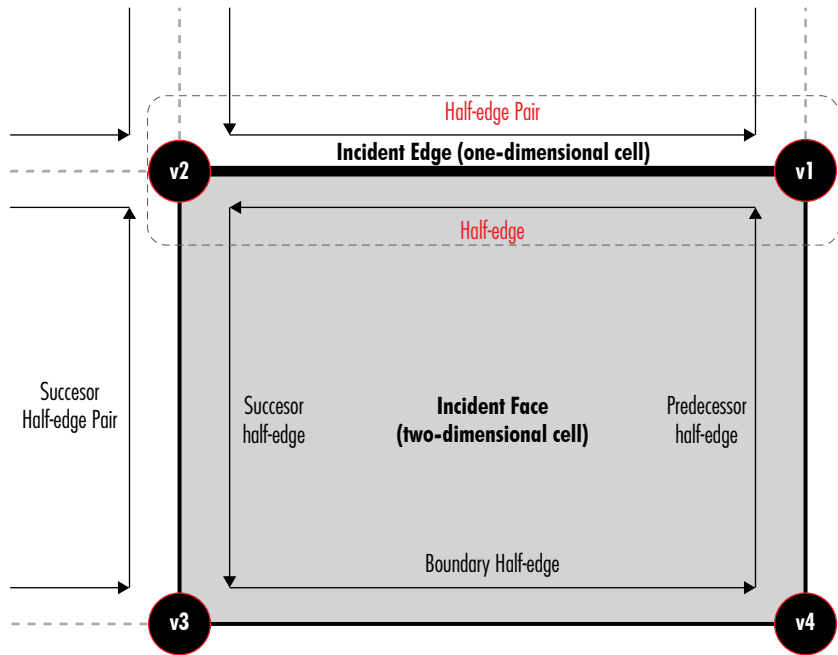


Fig. 5.4

Pointer-based Half-Edge representation.

of computer memory to store the corresponding darts. For the sake of simplicity, the PHE data structure was chosen by limiting the scope of this dissertation to orientable form-finding models. A half-edge is a directed segment associated with a start-vertex and pointing towards an end-vertex (Figure 5.4). A pair of half-edges with opposite directions defines an edge in which, at least, one half-edge is part of a closed polygonal face. A half-edge contained by a face is associated with a predecessor and successor half-edge which eventually forms a closed loop oriented in clockwise or counter-clockwise. The same type of orientation is used throughout the entire cellular decomposition. On the absence of an associated face, a half-edge defines a border which means that an edge can be adjacent to at least one face and, at most, to two faces. This constitutes the manifold constraints of the topologic model. A half-edge stores pointers for retrieving its origin vertex as well as for its predecessor, successor and pair half-edge.

The implementation of this data structure is focused on representing each tensile patch as a two-dimensional manifold with boundaries such that multiple patches can be combined without violating manifold constraints. On the basis of such global manifold representation, spline beams and cables can be created, or, in the worst case, they can be included by permitting some one-dimensional manifold exceptions (Figure 5.5). The

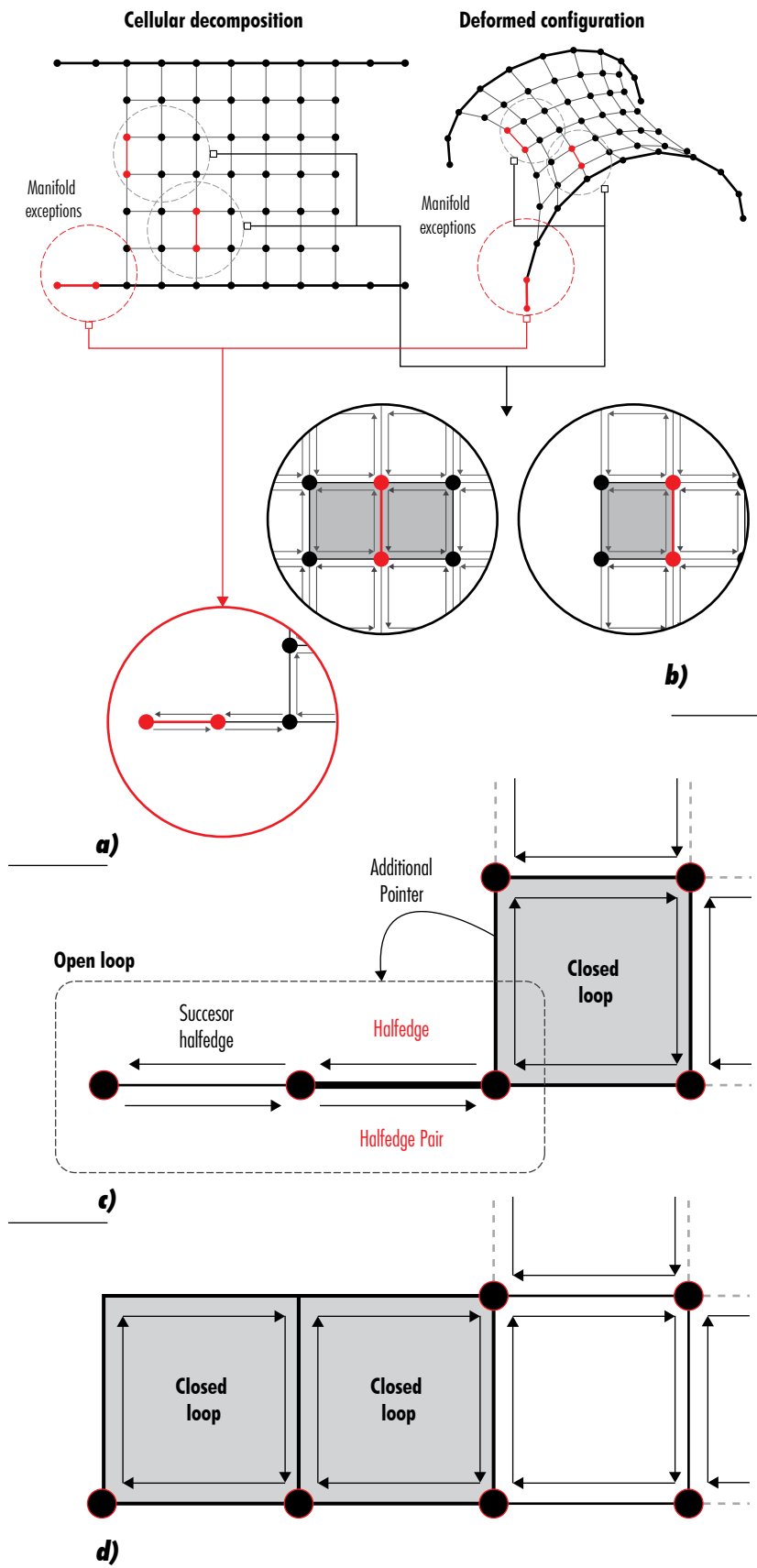


Fig. 5.5

Custom implementation of a Pointer-based Half-Edge data structure supporting edges without adjacent faces for form-finding processes on bending-active and textile hybrid structures. a) Non-manifold conditions, b) Manifold conditions, c) Halfedge relationships on an open loop, d) Addition of faces to develop manifold condition

latter condition produces a highly simplified topologic model supporting mixed dimensionalities (one- and two-dimensional cells). The idea was to allow manifold edges with at most two adjacent faces (Figure 5.5b), but also non-manifold edges defined by pairs of half-edges without associated faces (Figure 5.5a). In the latter case, the half-edge is considered to be part of an open loop so that it can be linked with one half-edge at the ends, either a predecessor or successor, and with both otherwise (Figure 5.5c). To do this, additional pointers need to be included in the PHE data structure to divide the implementation of manifold and non-manifold edges. Since non-manifold edges are also constructed from two halfedges with opposite directions, faces can be dynamically added at any time to define manifold conditions and remove the additional pointers (Figure 5.5d). Even though this PHE implementation is not optimized given the necessity of additional pointers to store certain types of non-manifold edges, it can still traverse all the vertices of the model and solve topologic queries during form-finding.

5.2.2 ARRAY-BASED HALF-FACET DATA STRUCTURE

More flexibility for non-manifold representations is obtained when the topologic model is implemented via an Array-based Half-Facet (AHF) data structure which may be equivalent to the concept of an n -chain. This data structure was initially proposed by Alumbaugh and Jiao (Alumbaugh and Jiao 2005) as a hybrid data structure that combines the comprehensive connectivity of PHE data structures with the compactness of incidence graphs. It aimed to address the problem of implementing more detailed topologic models within numerical simulations. Although the original focus was manifold representations with boundaries, it has been recently extended to non-manifold representations with mixed dimensionalities (Dyedov et al. 2014). The term facet is used to describe an $(n-1)$ -dimensional cell that is associated with an n -dimensional cell. To put it in another way, a face is the facet of a volume, an edge is the facet of a face, and a vertex is the facet of an edge. A facet having a specific orientation in relation to the containing cell is called half-facet which is why it can be perceived as a generalized form of half-edges. All the oriented

half-facets that are part of the same cell are called sibling half-facets and a half-facet without siblings is denoted as a border. Contrary to half-edges, the orientation of sibling half-facets don't need to be reversed. For this reason, a facet can be incident to multiple cells which is an important characteristic for non-manifold representations.

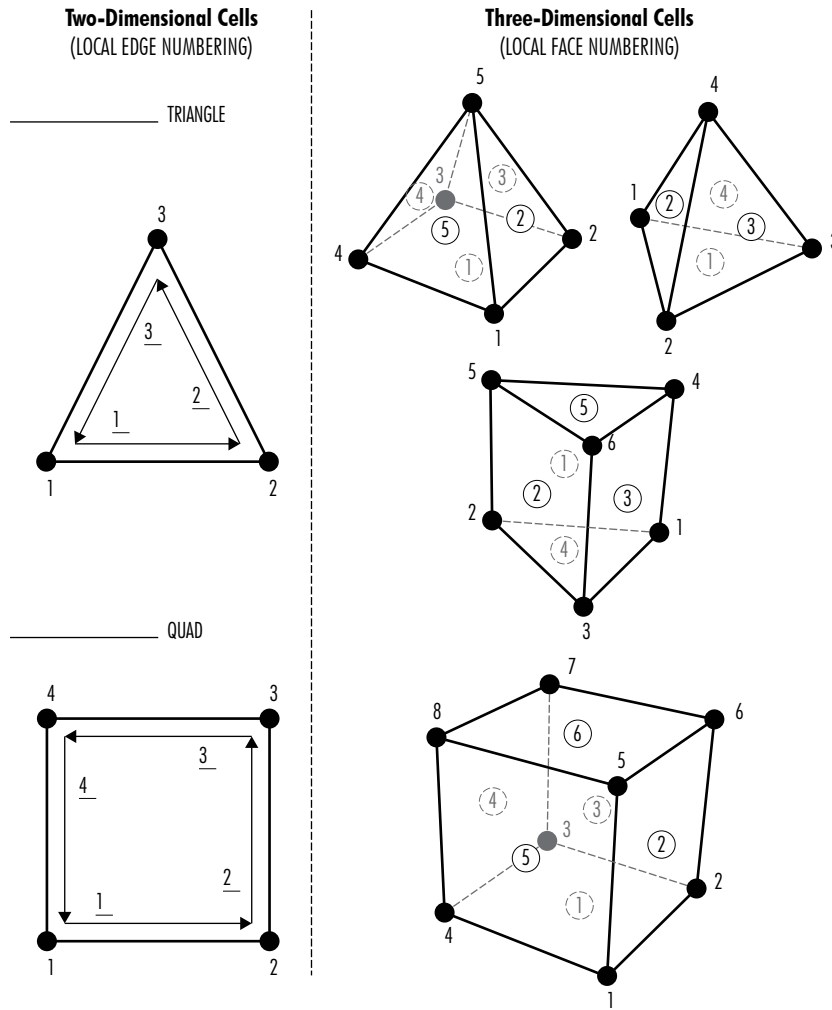


Fig. 5.6

CFD General Notation System for two- and three dimensional cells.

One of the main differences with PHE data structures is that AHF data structures use arrays instead of pointers to store topologic relations. In array-based storage, we already mentioned that the elimination of data is more difficult, but the structure is more compact and efficient in terms of computational resources. The fundamental concept is to treat half-facets as intermediate dimensionalities that can be described from explicitly defined cells (Dyedov et al. 2014). Each cell has a unique identifier (ID) and a local numbering convention that is assigned to sort

its vertices and half-facets (Figure 5.6). The literature suggests a local numbering of half-facet starting from zero (Dyedov et al. 2014). A half-facet can be described by only using the ID of the cell and the local number assigned to it within the cell. For this reason, half-facets are said to be implicitly represented which is not the case of a PHE data structures. The AHF data structure can support topologic models of any dimension, but, in the scope of this dissertation, only two-dimensional models will be used

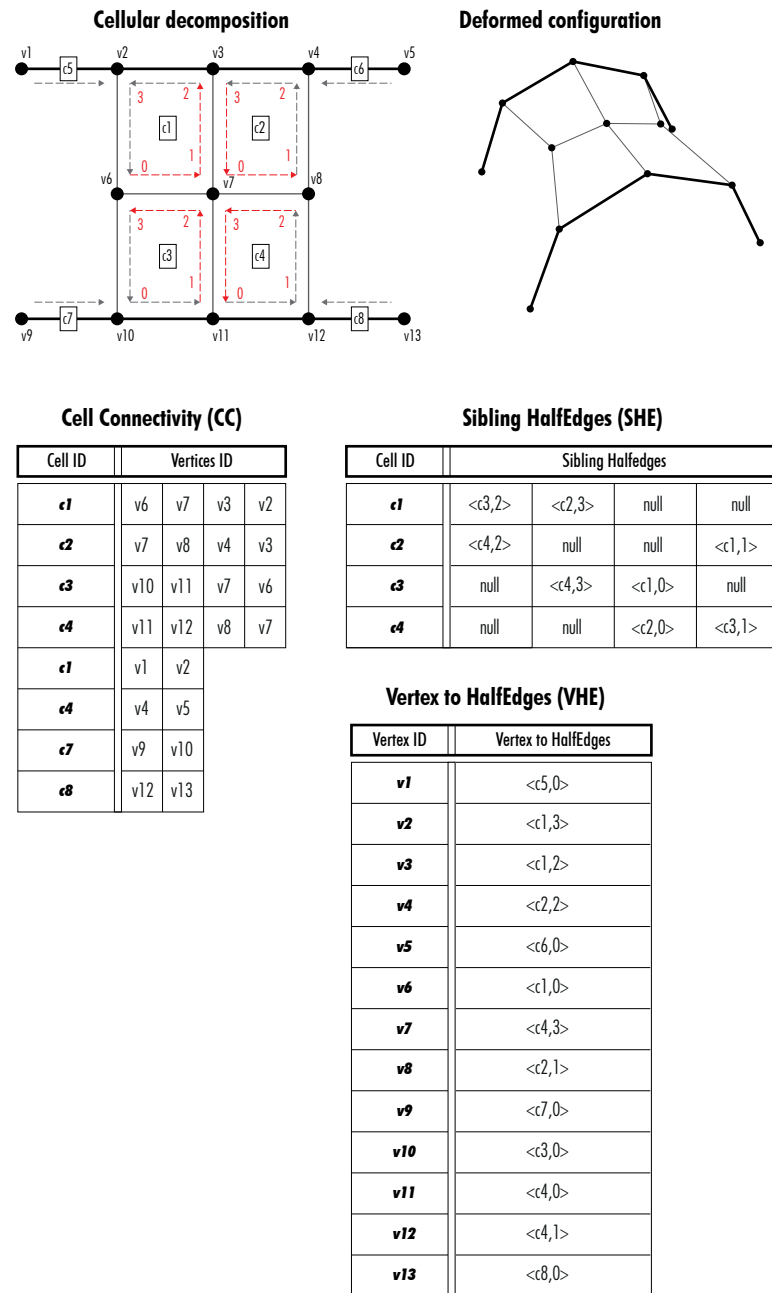


Fig. 5.7

Array-Based Half-Facet data structure for non-manifolds mixing one- and two-dimensional cells.

to conduct exploratory form-finding processes of bending-active and textile hybrid structures. In this context, half-facets are only equivalent to half-edges.

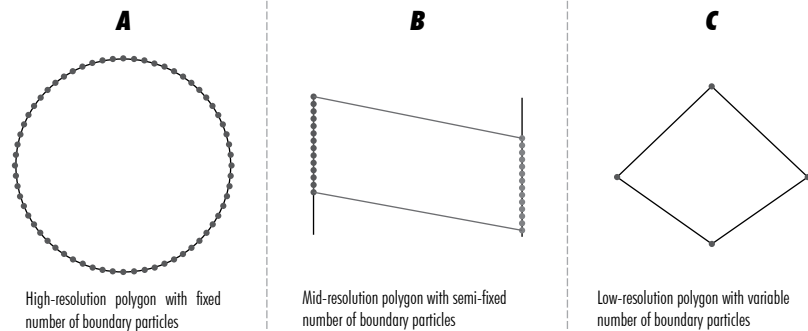
The data structure is then composed by four types of HashMaps called Cell Connectivity (CC), Sibling Half-Edges (SHE) and Vertex to Half-facet (VHE) (Figure 5.7). SHE is used to retrieve the connectivity of two-dimensional cells while VHE for one-dimensional cells. Additional HashMaps will be required for higher dimensional cells. A HashMap store key/values pair asynchronously with constant time performance for basic operations. Having CC, SHE can be constructed by first defining all the half-edges and their correspondent siblings for each cell. It is assumed that sibling half-edges map to each other to form a cycle or a path. A half-edge without a sibling is considered as a border and is stored as a null value in SHE. This HashMap permits to solve most of the adjacency queries of the topologic model. VHE then stores the association between vertices and half-edges with a priority given to border half-edges. As a whole, an AHF data structure permits to build a one-to-to correspondence with the proposed building blocks of our generic topologic model with embeddings. Additionally, form-finding models are not constrained by any manifold condition.

5.3 DISCRETIZATION TECHNIQUES

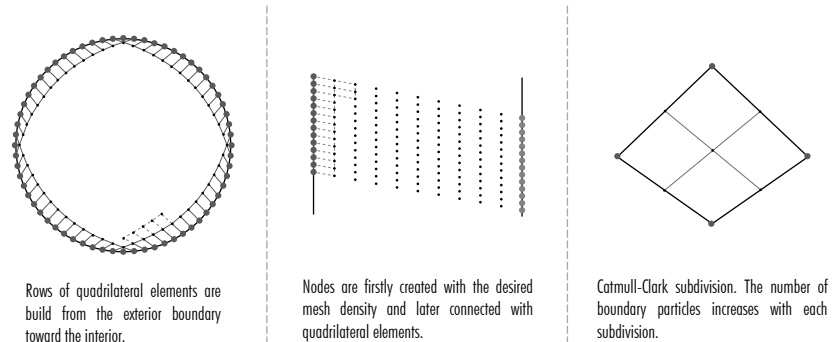
Once defined the types of data structure for implementation, the problem of constructing the cellular decomposition to be described by our topologic model needs to be addressed. In the previous chapter, we introduced the use of several discretization techniques in the definition of graph grammars for tensile patches. The ease of implementation and versatility of these rules relies on the creation of highly regular two-dimensional cellular structures entirely composed by quadrangular faces. In topology-based geometric modeling, cellular structures embedded in a three-dimensional Euclidean space are referred as mesh topologies. Quad-meshes are preferred because its two local directions can be directly associated with weaving directions of textile membranes, and quad-faces can be easily converted into regular triangles with clear directionalities. The problem is, however, how to generate

a regular quadrangular decomposition within different boundary conditions. To simplify the problem, we have identified three types of boundary conditions that can constantly appear during the form-finding of bending active and textile hybrid structures, and where specific meshing algorithms are required (Suzuki and Knippers 2017b).

Boundary conditions



Discretization process



Characteristics

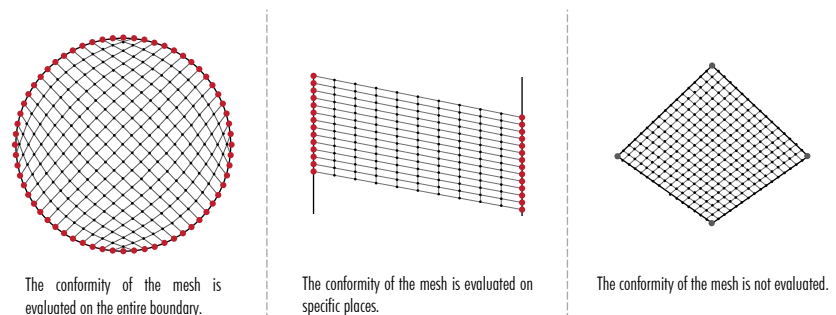


Fig. 5.8

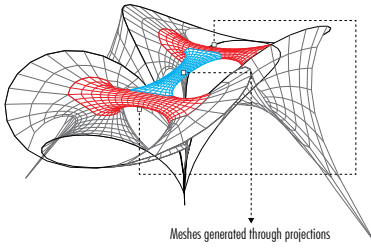
Generic types of boundary conditions appearing during topology-driven form-finding processes with beams and textile patches.

The first type of boundary conditions requires the implementation of an advancing-front method for iteratively “paving” rows of quad elements (Figure 5.8a). This means a

process for directly building quadrilateral elements from the boundary of an object to its interior. The method was introduced by Blacker and Stephenson (Blacker and Stephenson 1991) and has become widely popular in many finite element mesh generators. Important refinements to enhance its properties have been conducted along these years (White and Kinney 1997; Zhou et al. 2016; Staten et al. 2005; CASS et al. 1996). This algorithm permits to generate high quality meshes from any type of arbitrary boundary and includes the possibility of having internal elements like holes or line constraints. For the sake of simplicity, the meshes used for representing tensile patches do not contain any type of interior constraint.

The input of the algorithm is a closed polygon that can be constructed by letting designers select a finite number of vertices in a sequential order. On this basis, the steps of the algorithm are defined as row choice, closure check, row generation, smoothing, seams, row adjustment, intersection and clean-up (Blacker and Stephenson 1991). The algorithm starts by selecting the start- and end-vertices of the boundary to execute the paving process. With this information, it evaluates if the given boundary contains a minimum number of vertices to proceed with either the construction of rows or the closure of the mesh. If the mesh is not closed, the algorithm starts to sequentially create row elements from projections computed on the basis of vertex positions. A Laplacian smoothing technique is applied when the entire row is terminated in order to improve its quality. Interior angles, intersections and element sizes are then checked to avoid inconsistencies on the discretization. Once all rows are generated, the algorithm cleans and completes the mesh by either adding or deleting elements. The paving algorithm constitutes the most flexible solution to adapt the generation of quad-meshes to different types of boundary conditions appearing during the form-finding process.

The two other algorithms implemented to dynamically adapt the construction of quad-meshes are respectively associated with projective (Figure 5.8b) and subdivision processes (Figure 5.8c). The first algorithm takes two equally-sized sets of vertices, placed on opposite sides, to build a set of projections between

**Fig. 5.9**

Assembly of meshes generated from simple construction processes based on projections.

each pair of corresponding vertices from different sets. From here, internal vertices are created and then connected to shape quadrilateral elements. While the number of vertices on the given sides can't change, internal vertices are computed to satisfy the desired resolution of the mesh. Despite the simplicity of this method, it permits the construction of highly complex assemblies that closely approximate volumetric meshes (Figure 5.9). Finally, a Catmull-Clark's subdivision algorithm is also implemented to permit the creation of high-resolution quad-meshes from a minimum number of selected vertices.

5.4 DEVELOPMENTS

The implementation of our generic topologic model with embeddings to build an appropriate PS model for topology-driven form-finding has been realized within the development of a software applications named ElasticSpace and robust physic library named Iguana.

5.4.1 ELASTICSPACE

ElasticSpace is a custom-built software for explorative form-finding of bending-active and textile hybrid structures supporting topologic transformations on the fly (Suzuki and Knippers 2017a). The physics package of the software was written in Java 1.6 while the Graphical User Interface (GUI) in Processing 2.2.1. The modeling workflow is highly interactive and closely approximate shape exploration in analogue form-finding practices. The topologic model is implemented by using a modified PHE data structure. The designer can dynamically control topological transformations in addition to material and geometric properties, and solver's setups. Hence, form finding can start without specifying an input topology since this one can be gradually built through the continuous addition of building block. ElasticSpace includes a wide variety of constructors, associated with graph grammars, for creating and modifying building blocks (Suzuki and Knippers 2017c).

In this implementation, spline-beams only relate to beams with 3DOF per particle. To add a spline-beam, the designer needs

to specify material and geometric properties, number of particles and, in certain occasions, external references to include particles that will be shared by other building blocks. A spline beam that is entirely build with external references based on rule 8, denotes a bundling characteristic. On the absence of external references, the component requires to specify the type of support condition. On the other hand, tensile patches are always constructed from external references following rules 19-21 in which case one or multiple spline-beams may be involved. Another possibility is the catenoid-like constructor associated with rule 22 for the construction of a pseudo-volumetric surface. Components deletion is rather a straightforward and intuitive task based on the rules described in the previous chapter.

5.4.1.1 Implementation

The GUI is divided in six parts excluding the central 3D viewport where the geometry is render and particles are color coded according to their state. The interface is composed by menus for creation, modification, setting and visualization which

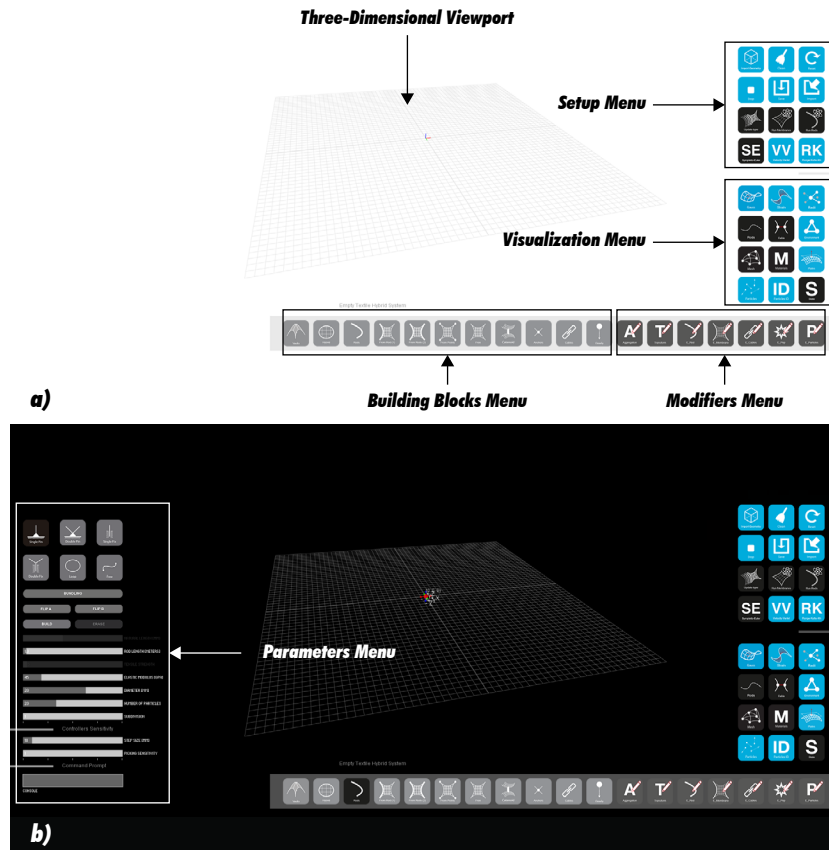


Fig. 5.10
ElasticSpace interface.

can be easily identified by the color of their respective buttons (Figure 5.10). Creation and modification menus contain buttons that are associated with the implementation graph grammars and building blocks. As soon as one of these buttons is pressed, the color of the background is switched to black (Figure 5.10b), and an additional parametric console appears to control the geometric properties of the building block to be created or modified. On the contrary, the color of the background stays white when no button for changing the model has been pressed. In such cases, certain properties like bending-radii, gauss curvature, strain and stress can be clearly visualized. Furthermore, a general purpose console is included to modify specific properties of the PS model or predefined values of the interface.

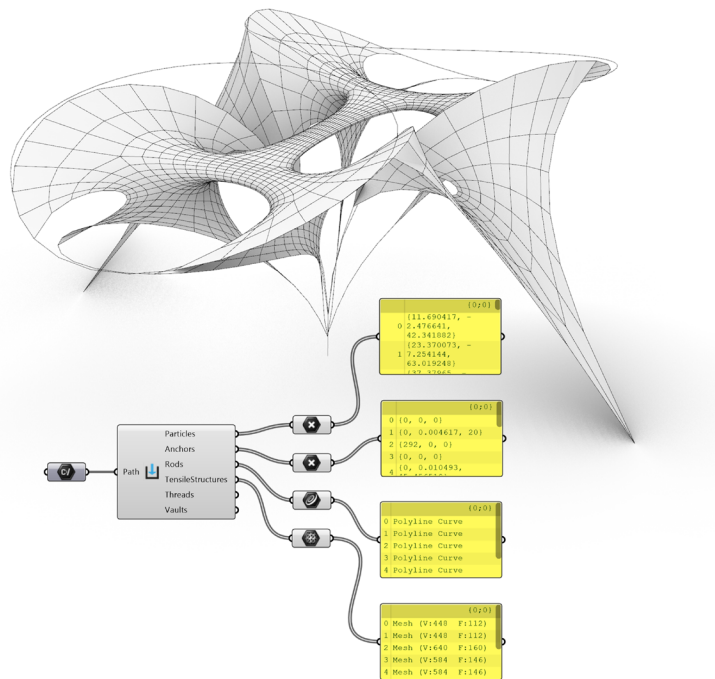


Fig. 5.11

XML communication with Grasshopper.

At any time, the database of the model can be exported via an eXtensible Markup Language (XML) data file (Suzuki and Knippers 2017a). XML is a type of meta-language for describing data in a flexible and adaptable way and is widely available for popular languages like CPP, C#, Python or Java among others (Shea et al. 2005). In fact, XML is an open-standard for communicating data between different programs. Formatted as a text document, an XML file is machine-readable as well as human-readable. As a result, all geometric, topologic and mechanical data is stored in

a custom XML file. The top level document element corresponds to the description of the physical system. This element then encapsulated different child elements that are associated with the individual description of particles, spline beams, tensile patches and cables. Data is organized in such a way that the complete model, or just a part of it, can be easily reconstructed in other environments (Figure 5.11). A Grasshopper plugin was developed to parse this information and reconstruct in the commercial Rhinoceros CAD package. The plugin contains a simple set of components, including a symplectic-Euler solver, for creating PS models in Grasshopper/Rhinoceros and export the information via XML files to ElasticSpace. This means that some initial condition can be defined in Grasshopper to be later modified and/or completed in ElasticSpace. This condition permits to create different modeling pipelines.

5.4.1.2 Numerical Solver

ElasticSpace implements both types of explicit solvers described in chapter 3 and an additional Velocity-Verlet integration scheme. Here, the implemented building blocks are based on the use of beams with 3DOF per particle, elastic bars and CST elements. The rate of convergence of each solver was evaluated on the basis of a spline beam with 71 particles. The Leapfrog solver reached convergence after 2238 milliseconds while Runge-Kutta4th solver after 2760 milliseconds (Figure 5.12). Comparing the deviation in pure bending of both solutions with an elastica curve, the results showed that, at the mid-point, the Leapfrog solver differs in 6 millimeters with the analytical solution while the Runge-Kutta4th in 3 millimeters.

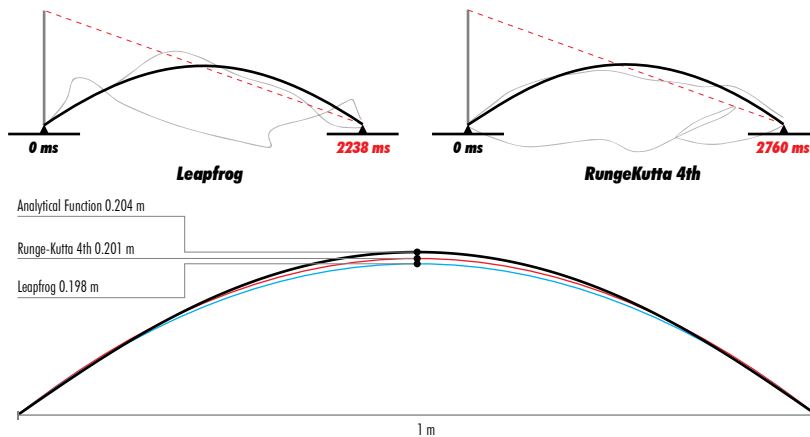


Fig. 5.12

Comparison between the Leapfrog and the Runge-Kutta 4th solvers.

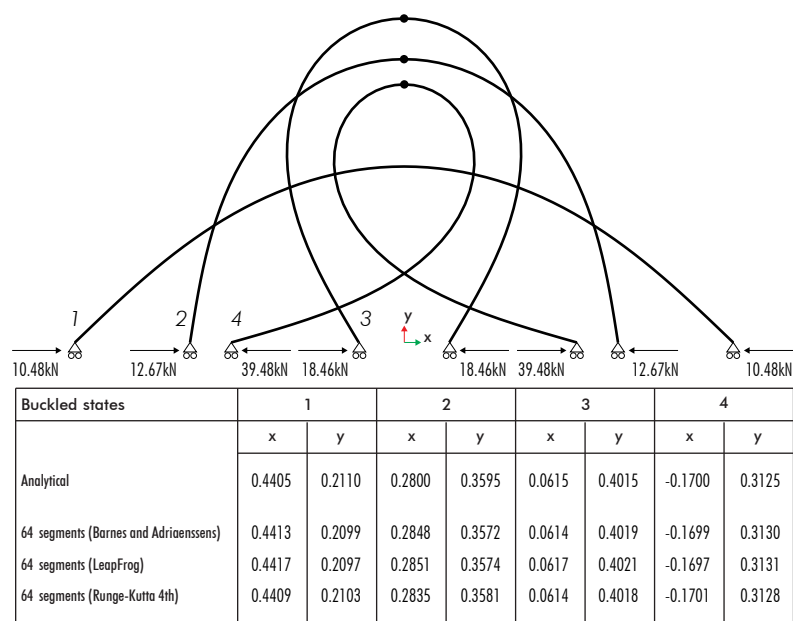
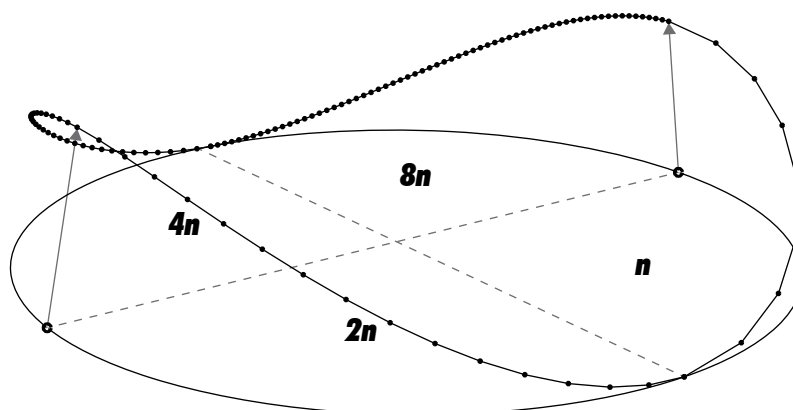


Fig. 5.13

Pure bending comparison of beams with 3DOF per particle.

The implementation of beams with 3DOF per particles was compared with the proposed test case of Adriaenssens and Barnes (Adriaenssens and Barnes 2001; Barnes 1999) based on a pin-ended strut buckling into an elastica curve. The total length of the spline beam is 10 meters divided in 64 segments with $EI = 100 \text{ kNm}^2$ and $EA = 100 \text{ MN}$. The spline beam is supported on both ends with rollers and each end is subject to axial loads. The displacement of the central node was evaluated according to four equilibrium states presented in figure 5.13.



Segments per quadrant	$n = 8$ (Total 120 segments)
Displacement Δ (Barnes and Adriaenssens)	2.42 m
Displacement Δ (LeapFrog)	2.391 m
Displacement Δ (Runge-Kutta 4th)	2.437 m

Fig. 5.14

Out-of-plane loading of a prestressed ring.

Furthermore, our implementation was also compared with the test case of a prestressed ring subject to out-of-plane loading described by Adriaenssens and Barnes (Adriaenssens and Barnes 2001; Barnes 1999). The ring has a diameter of 10m and the same material properties of the previous test case. The ring is divided into 4 quadrants 1-2, 2-3, 3-4, 4-1 and each quadrant is respectively divided into n , $2n$, $4n$, and $8n$, with $n=8$. Loads of 10Kn were applied at nodes 1 and 3 with nodes 2 and 4 constrained in the z direction. The displacements Δ of the out-of-plane loading are presented in figure 5.14.

5.4.2 IGUANA

Iguana is described as a comprehensive physics library developed in Java and C# using concurrent and GPU accelerated computing for numerical integration (Suzuki et al. 2018). In both cases, an easy API is provided to facilitate their implementation in common platforms used by architects, engineers, artists and designers. Iguana aims to provide a flexible framework for pushing further the exploration of non-standard topologic configurations, but mainly for designing adaptable and reconfigurable systems. Custom numerical models can be implemented with highly interactive capabilities for active-geometric and -topologic modelling, and with a selective control of simulation's interactivity and accuracy by means of topologic strategies. The latter is particularly important for providing immediate feedbacks of different equilibrium stages and anticipate a more accurate behaviors before introducing topologic changes.

5.4.2.1 Implementation

Iguana has two modules/packages and is mainly designed for making an extensive use of real-time physic simulations as design tools. Iguana Topology is a module/package that provides discretization algorithms and data structures based on combinatorial maps for facilitating continuous topologic operations. The basic data structure of the module is a PHE data structure for two-dimensional manifolds, similar to the one implemented in ElasticSpace. In addition to this, an AHF data structure is implemented to facilitate non-manifold representations

and multi-dimensionalities. The core module/package is Iguana physics, a comprehensive physics library containing all the mechanic elements described in chapter 3 as well as the entire set of rules described in chapter 4 to modify those elements on the fly. Note that Iguana Physics builds on Iguana Topology which means that the designer needs to specify in advance the type of topologic model to be used for constructing a PS model (PHE or AHF). All data generated by the library can be stored within an XML file to exchange information with other programs. The Java library was initially tested for the development of ElasticSpace 2.0 which was entirely rebuilt with Java 11.0.3 and JavaFX. For this reason, the GUI of the software was completely redesigned to be more robust and standard with a unique viewport using predefined orthographic and perspective cameras, a menu bar containing all the functionalities of the program, a main console for input and output communication and a panel with all the parametric variables controlling the construction of building blocks (Figure 5.15). The color of the background is still used to clearly specify the type of modeling mode.

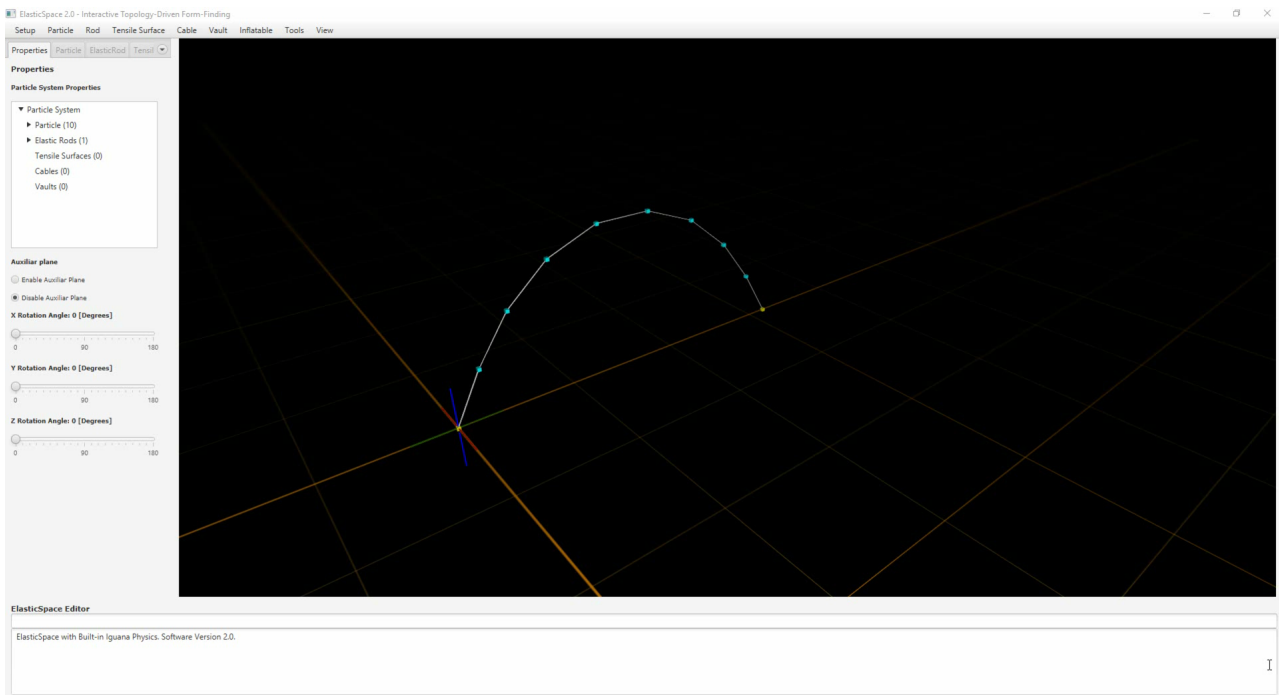


Fig. 5.15

ElasticSpace 2.0 built with Iguana Physics.

The C# library was in turn used to develop a Grasshopper plugin for Rhinoceros 6 permitting to use the topologic module

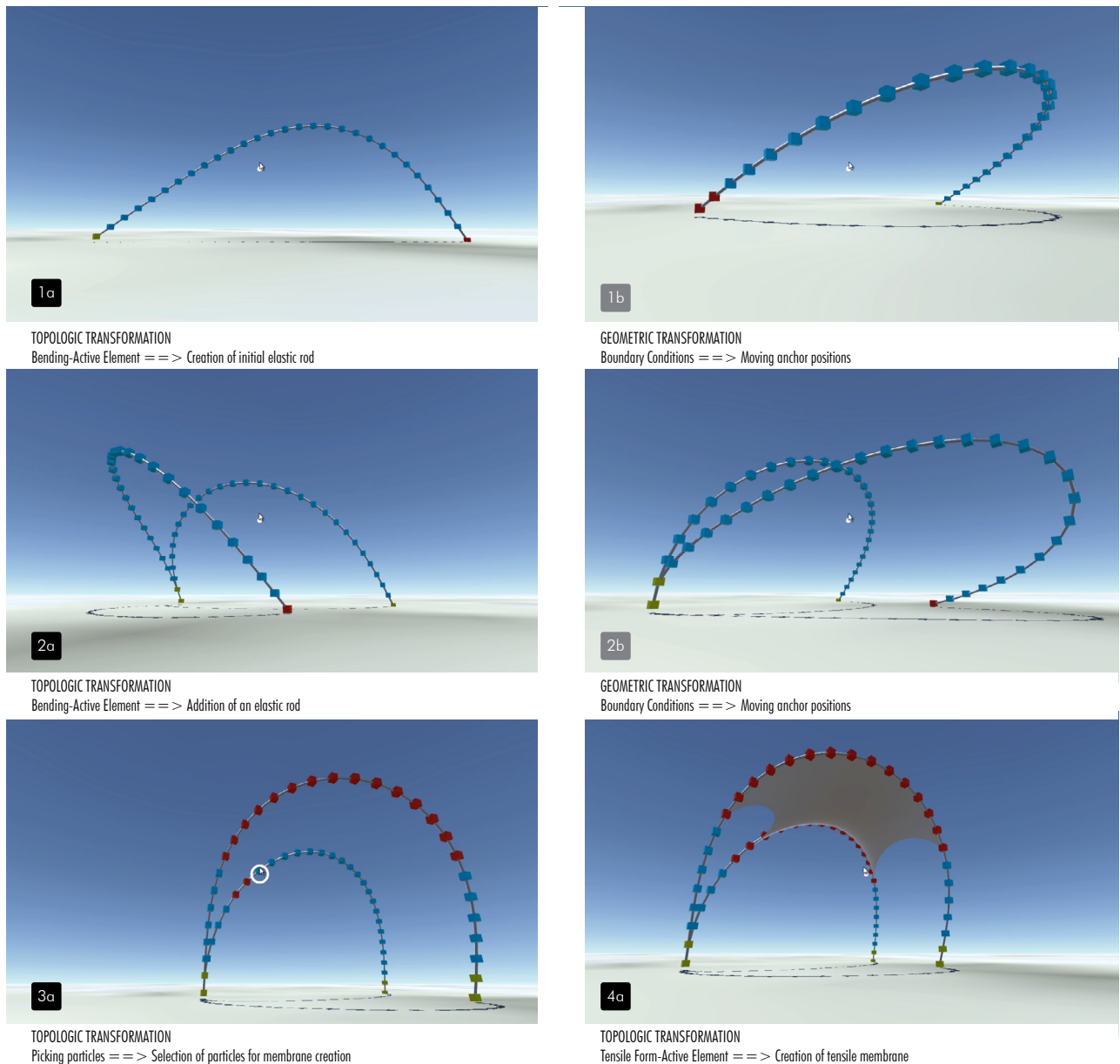


Fig. 5.16

Iguana VR implementation with Unity.

independently from the physic module. Additionally, the C# library was also implemented in Unity3D to start addressing the logical extension of these highly flexible form-finding procedures into immersive spaces for enhancing the design experience (Figure 5.16). The use of virtual reality (VR) technology is here considered as a medium to improve the playfulness of the design process by creating new experiences when modeling with real-time physics. This means a type of gamification of the form-finding process for mastering the design of equilibrium shapes. Hence, a custom android application for Daydream and

Cardboard platforms was developed in Unity3D based on the Iguana API (Suzuki and Knippers 2018). For this first study, this platform was selected because to date it continues to be the most accessible and affordable VR platform. Simplified spline beams and tensile patches were used to conduct the design of a textile hybrid structures within a fully immersive space. The perception of the three-dimensional space radically changes within the VR application which means that form-finding routines were adapted to it. User's head orientation is controlled through a gyroscope sensor in combination with a standard wireless remote controller to set the position in space and manage the dynamic creation of building blocks. The designer can progressively build a system while perceiving deformations and spatial qualities in a comparable process to the design of a vernacular structure using empirical methods. For this reason, immersive form-finding practises can radically change the design experience and drive the designer into a space where analogue and numerical characteristics can be fully merged.

5.4.2.2 Numerical Solver

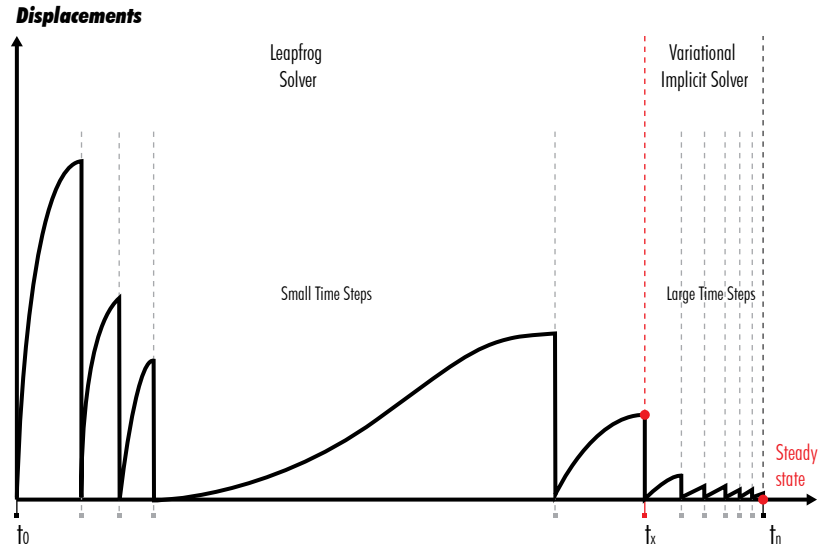


Fig. 5.17
Mixed solver.

Iguana physics implements different types of numerical solvers being in an ascending order these are the Symplectic-Euler, Leapfrog, Mid-Point, Velocity-Verlet, Runge-Kutta4th and Variational Implicit Euler methods. A mixed-solver is also implemented for combining the Leapfrog method with the Variational Implicit Euler method. This is a more sophisticated

solver that automatically switch between both integration schemes to achieve a better computational performance. The Leapfrog method is preferred when the solution is far from the equilibrium condition since small time steps gradually produce large changes on the numerical solution. However, as the system approximate equilibrium, small time steps generates less significant changes. At this point, the Variational Implicit Euler method can be used to increase the size of the time step in order to improve convergence rate (Figure 5.17).

All solvers are decoupled into a translational part which is locally solved using a vector form and a rotational part that, in turn, is locally solved using a matrix form. For stability and convergence, particles are set with optimized lumped masses, inertia-masses, weights and inertia-weights which are directly derived from the maximum stiffnesses occurring during the form-finding process as described in chapter 3. By default, convergence is reached through the application of kinetic damping within a constant timestep. That is, when a peak of kinetic energy is detected, velocities are reset to zero and positions are corrected to the half of the last timestep causing a more rapid convergence rate. Users can still disable the kinetic damping and calibrate a viscous damping to best approximate the dynamic process for arriving to the static solution.

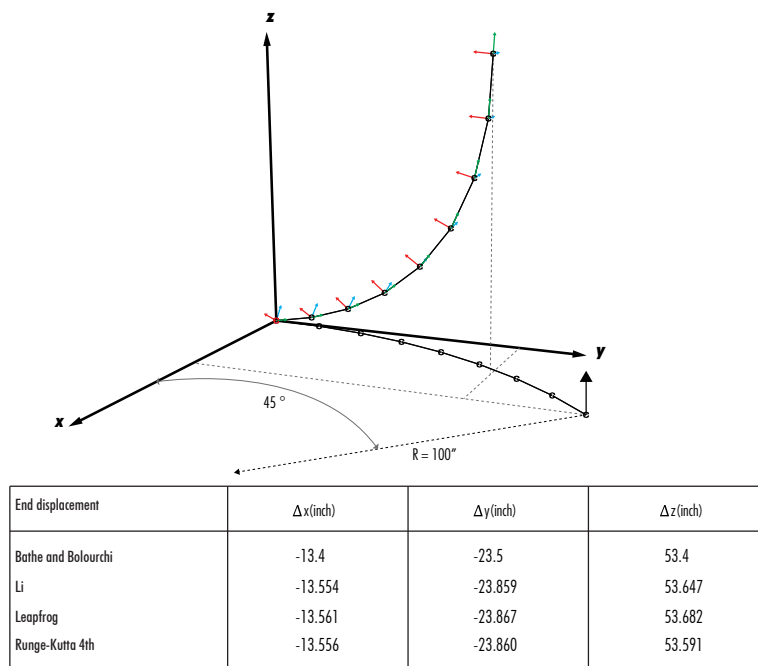


Fig. 5.18

Comparison of end displacements of Bathe's curved cantilever beam.

The implementation of the beam element with 6DOF per particle was compared with the benchmark developed by Li (Li 2017) using Bathe's curved cantilever beam. The beam is a 45 degree bent geometry with a radius of 100 inches divided in eight elements of equal length. Material properties are set to 10^7 psi for the Young's modulus (E) and $E/2$ for the shear modulus (G). The total length of the beam is 78.508 inches with a uniform square cross-section of 1 inch per side. The end of the beam is subject to a transverse load of 600lb in the z-direction. The comparisons of end displacements are presented in Figure 5.18.

5.5 TOPOLOGIC IMPLICATIONS IN NUMERICAL MODELS

As previously stated, in topology-driven form-finding, problems regarding interactivity, data organization, decision-making and discretization need to be solved in parallel (Figure 5.19). In the proposed implementations of our generic topologic model with embeddings, data-structure design and discretization techniques have been already addressed for facilitating the management and generation of recursive form-finding data. The two remaining problematics may be categorized as new within form-finding workflows since they directly derived from the use of dynamic topologies. These two last stages can be activated on the fly when the size of the model increases and are focused on managing the rate of interactivity/accuracy of simulations and the taking of modelling decisions in complex design spaces.

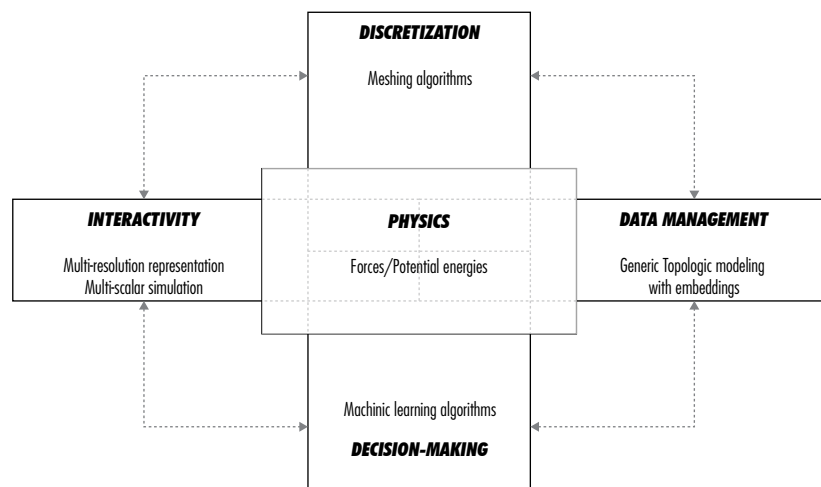


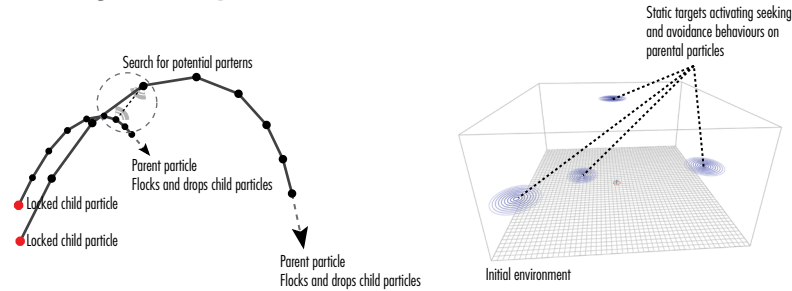
Fig. 5.19

Design implications of form-finding with dynamic topologies.

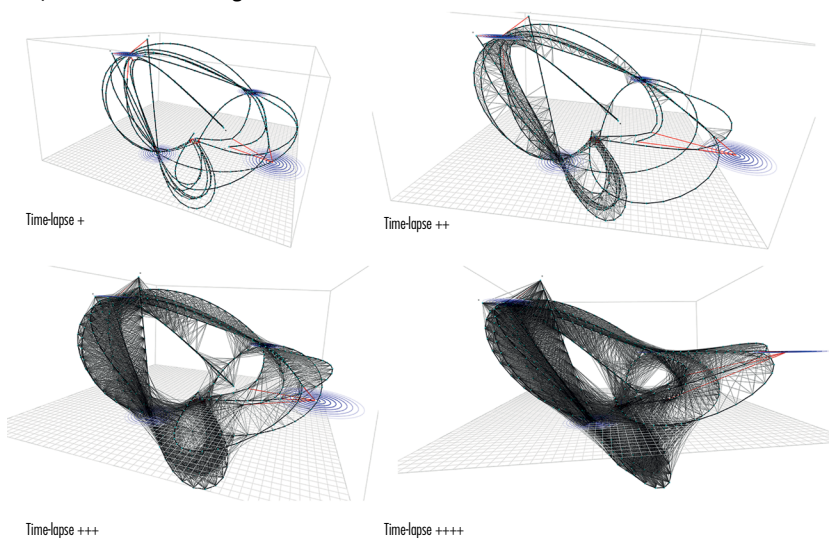
5.5.1 ASSISTED DECISION-MAKING²

Taking decisions regarding the addition or deletion of particles and elements is here a major concern because such decisions need to be carried each time within more complex modeling spaces. So, we noted that the flexibility of our PS models fits well with some machine learning algorithms for assisting designers in such processes. For instance, an additional module in ElasticSpace was implemented to assist the construction of spline-beam elements by either extending behaviours on certain particles, similar to those described by Reynolds (Reynolds 1987), or by using a decision tree algorithm. Extended behaviours were implemented by customizing rule 9 and including two new types of particle's states, denoted as parent and child (Figure 5.20). A particle with a parent state would work analogous to a mouse cursor that when clicked by the designer adds a new particle with a new set of connections into the system. The idea is

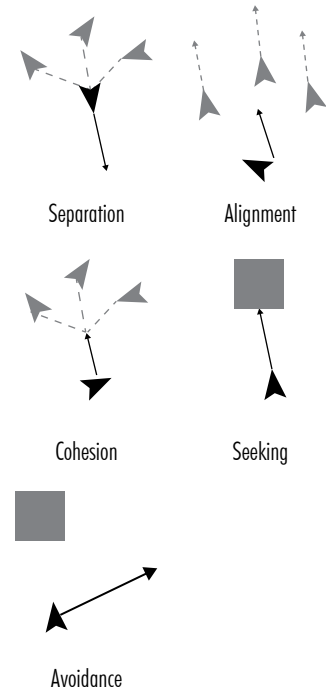
a) Multi-agent based system (families and environment)



b) Creation of bending-active networks with tensile connections



a) Parent particle



b) Child particle

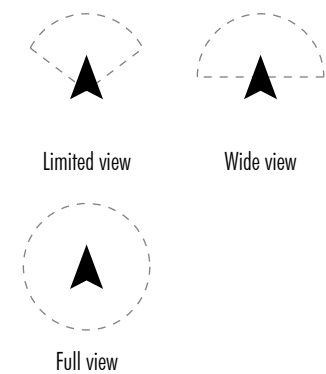


Fig. 5.20

Extended behaviors on particles for assisting the decision making.

Fig. 5.21

Autonomous creation of elastic rods and tensile cables.

²Based on pre-published article (Suzuki and Knippers 2017b): "The Design Implications of Form-Finding with Dynamic Topologies".

that those particles can autonomously move within a digital space while dropping, at the same time, a finite set of child particles and connections within them (Figure 5.21a). This construction method is exactly equivalent to rule 9. In this case, the designer can control the construction of spline-beam by defining the number of child particles to be dropped and by placing static targets around the digital space to activate seeking and avoidance behaviours of parental particles. All the children of a single parent particle describe the topology of a spline-beam. As soon as three child particles are dropped, the spline-beam starts its deformation while the parental particle is still flocking and connected to the last child. Boundary conditions are then established by setting the initial and the last child particles as pinned supports. At the same time, child particles can distinguish other child particles from a different parent based on distance constraints and view ranges to establish a tensile connection (Figure 5.21b). This secondary module was mainly presented to demonstrate the flexibility of the PS model for form-finding with dynamic topologies. Therefore, further efforts going out of the scope of this dissertation are still required on this subject.

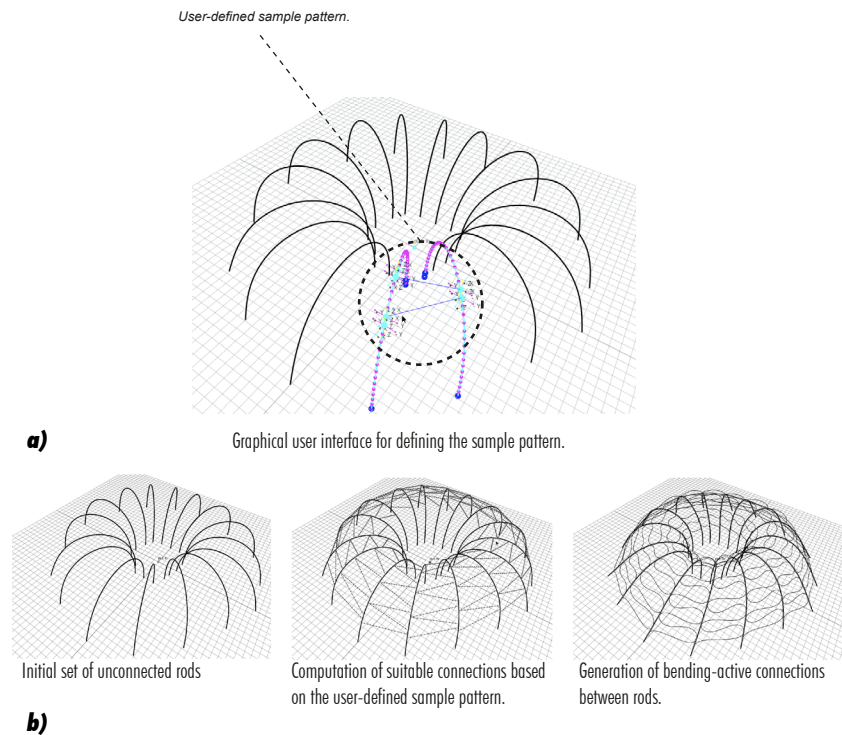


Fig. 5.22

Assisted creation of spline beams using a decision tree algorithm. a) Creation of spline beams, b) Selection of connectivity pattern.

On the other hand, better results were obtained when using a decision tree algorithm. This machine learning algorithm

was implemented for making choices regarding the creation of spline-beam connections between large sets of disconnected spline-beams (Figure 5.22b-5.23).

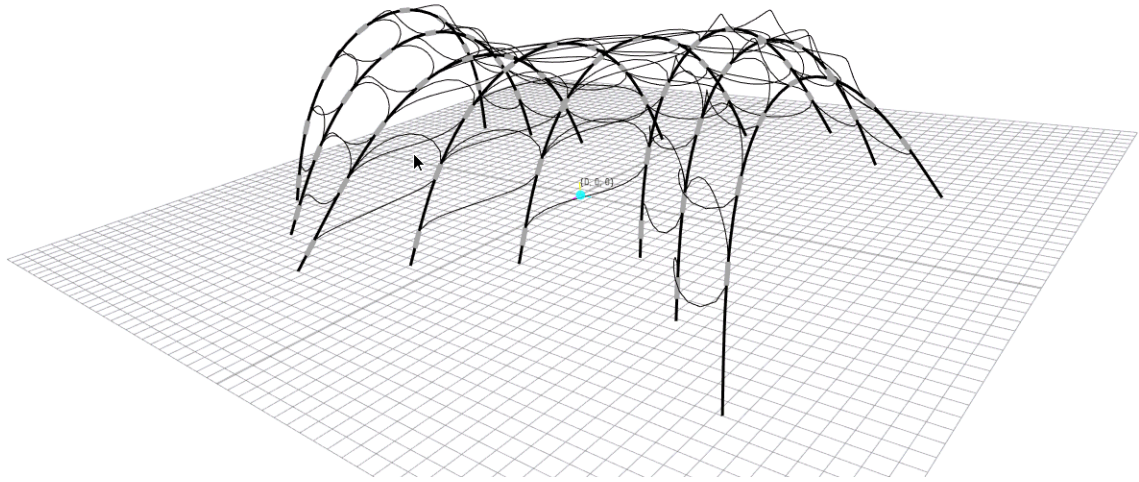


Fig. 5.23

Assisted creation of spline beams using a decision tree algorithm.

Through a Boolean classification, a decision tree returns a positive or negative output by evaluating an input with a predefined value at each tree node (Russell and Norvig 2016). In this case, those predefined values of the decision tree are constructed from a sample pattern that is defined by letting designers select an ordered set of particles representing the desired

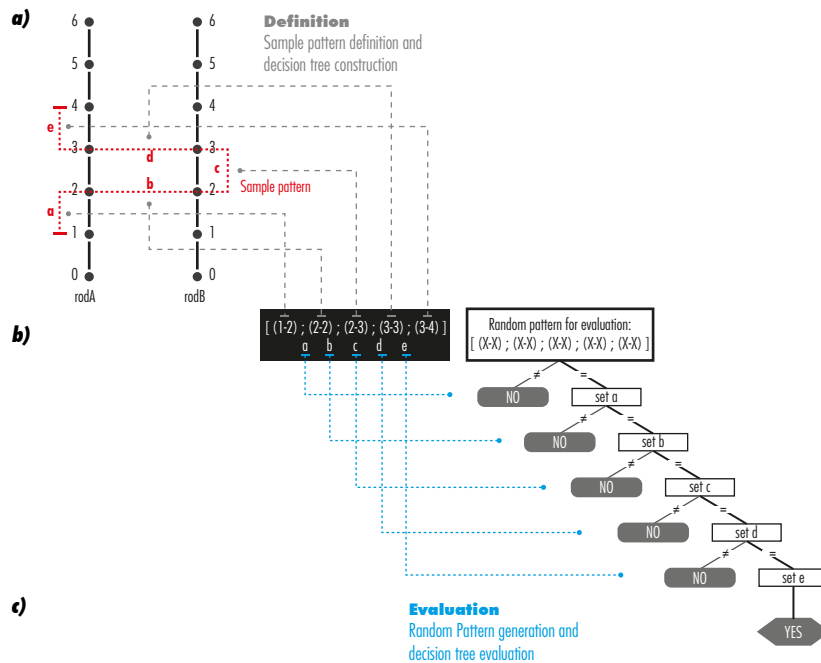


Fig. 5.24

Workflow of the decision tree algorithm. a) Definition of the pattern, b) Structure of the decision tree, c) Evaluation of branches.

connectivity (Figure 5.24). Considering that each spline beam of the disconnected set constitutes a directed subgraph from where a local order of particles indexes can be established, the sample pattern can be mapped on the basis of those indexes (Figure 5.24a). A tree structure is then created for each sample pattern (Figure 5.24b). At each tree-node, the difference of consecutive indexes is computed. If the difference of an input pattern is equal to the difference of the sample pattern, a positive response is triggered. As a result, the algorithm decides whether a computer-generated random pattern satisfies the criteria of the sample pattern generated by the designer (Figure 5.24c). The application of the decision tree algorithm serves to assist designers with the dynamic construction of spline beam elements in complex modeling environments.

5.5.2 SELECTIVE CONTROL OF INTERACTIVITY AND ACCURACY

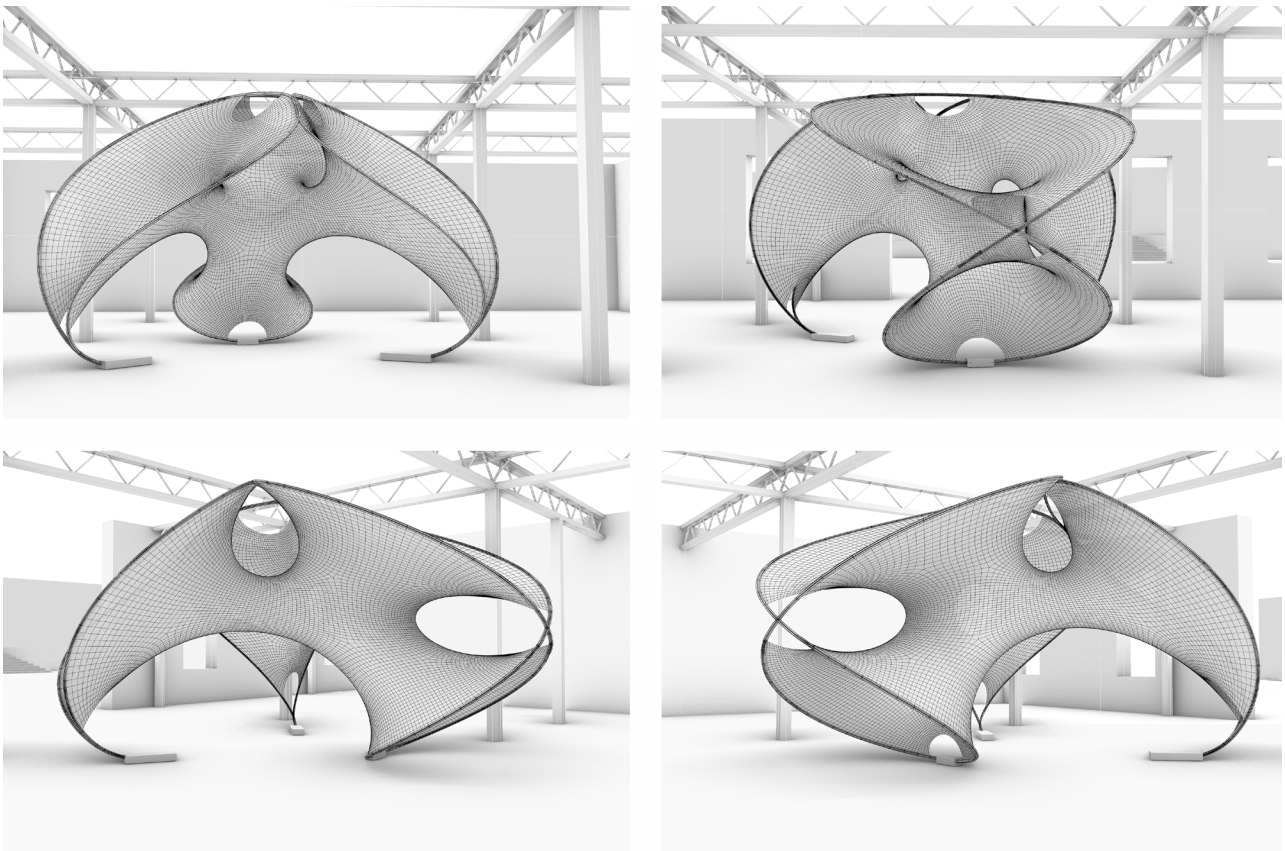


Fig. 5.25
Textile Hybrid structure built with Iguana Physics.

Another problem appearing when dynamically altering the topologic properties of a numerical model can be the progressive diminution of user-model interactivity. This is in part occasioned by the massive number of particles that are gradually introduced, but also by the mechanic elements used to compute forces/potential-energies. For the latter aspect, Iguana may also be used to control the rate of accuracy and interactivity of the numerical model by selectively adapting the use of element's formulations according to modeling requirements (Suzuki et al. 2018). For instance, better levels of accuracy may be desired when reaching a mature design in which case less topologic modifications would be needed. Figure 5.25 shows the model of a textile hybrid structure built with Iguana that will serve in the following to introduce a set of modeling strategies on this subject. The PS model represents a networked arrangement of drop-like modules shaped from elastically bent rods that are coupled with a complex arrangement of tensile surface patches. Material properties on spline-beam are established on the basis of tubular cross-sections with a diameter of ranging from 10 mm to 25 mm, a Young's modulus of 25000 Mpa and a shear modulus equal to $E/2$. Elastic bars and CST elements are only defined with geometric stiffness.

5.5.2.1 Multi-Scalar Spline-Beams³

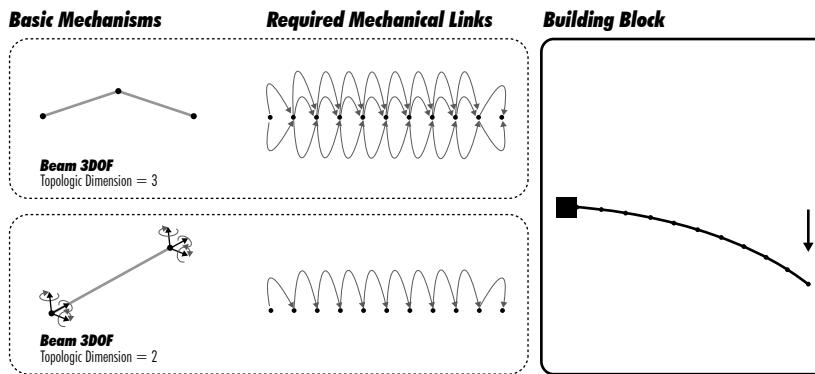


Fig. 5.26

Topologic structure of required mechanical links on a spline beam.

The interactivity of the simulation is directly determined by the amount of computational resources used to solve each integration step. A model entirely built with beams with 3DOF per particle is less expensive to compute than a model composed of beams with 6DOF. To reinforce interactivity, the modeling process can be conducted on the basis of spline-beam elements

³ Based on pre-published article (Suzuki et al. 2018): "Iguana: Advances on the development of a robust computational framework for active-geometric and -topologic modeling of lightweight structures".

using simplified formulation with 3DOF per particle. At any given point, the formulation can be switched to 6DOF per particle without the necessity to break the numerical process. This is only possible because the implementation of our topologic model is divided into different layers with specific types of embeddings. A subset of particles regrouped into what we have defined as a nested class of type spline-beam can be used to compute all internal forces of a sequence of beams with 3- or 6-DOF per particle (i.e. basic element type). Internal forces are computed by creating local links between particles for fulfilling the requirements of a common type of mechanism. By dynamically adapting those local links, the base mechanism can be shifted/changed (Figure 5.26) (see chapter 4). This means that the topologic model doesn't explicitly change but some basic assumptions are taken for computing the new forces/constraints on the fly. The only consideration that needs to be included are rotational constraints on locked particles. In the absence of rotational constraints, those particles are only locked in their translational motion which means that the orientation

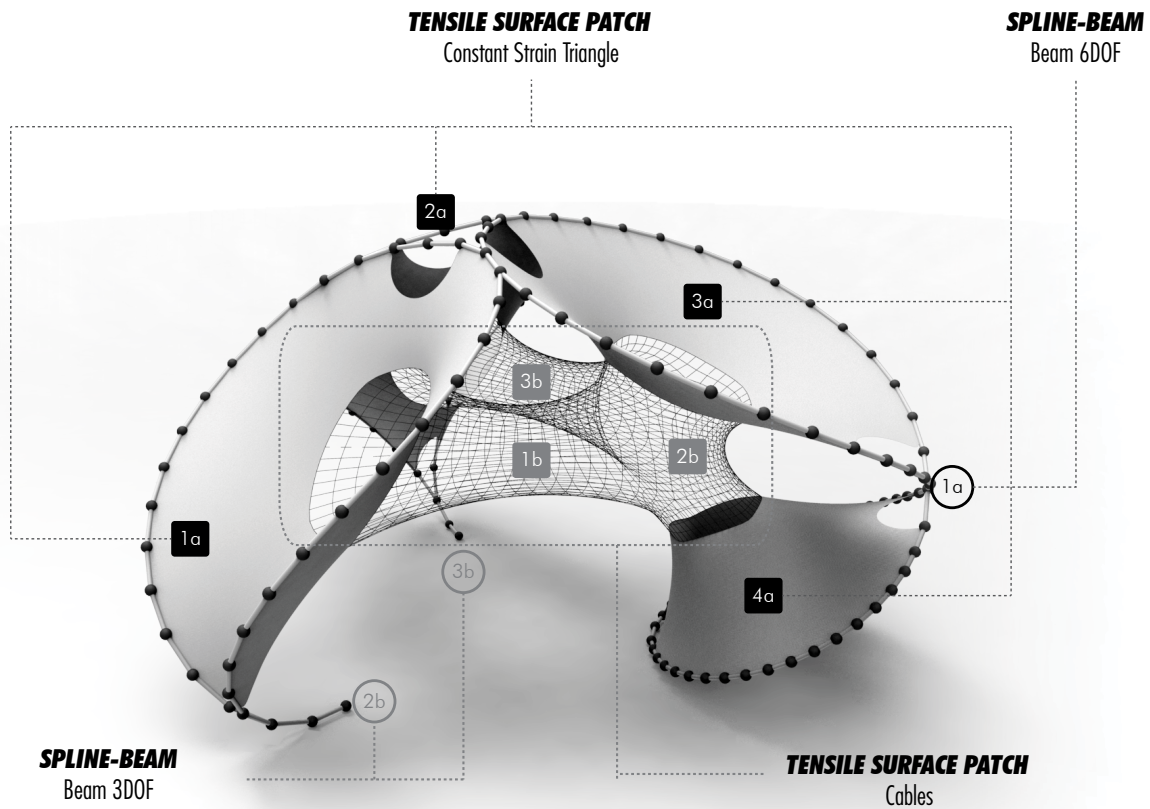


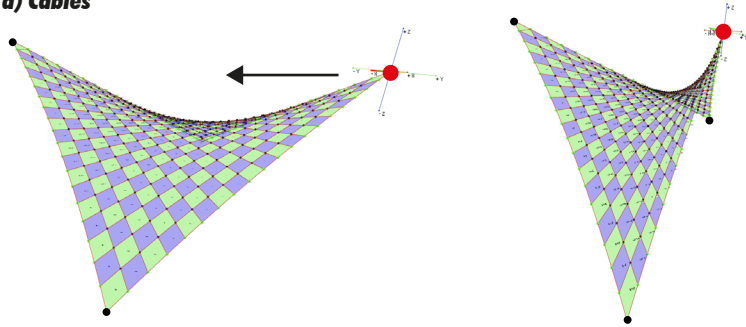
Fig. 5.27

Selective activation of different element formulations for simulating similar physical behaviors.

of free particles is not explicitly affected by the orientations of locked ones. This is not the case when locked particles are also constrained in their rotational motion (i.e. full-lock state). Note that the initial orientations of constrained particle determine the orientations of free particles and ultimately the deformation of the entire beam due to torsional effects. On this basis, all beams with 3DOF per node formulations can be entirely converted in 6DOF per node formulations and vice-versa. An alternative approach is to selectively use more accurate formulations on beams where torsional effects would be more important, and leaving the rest with the simplified formulation (Figure 5.27).

5.5.2.2 Multi-scalar Tensile Patches⁴

a) Cables



b) Constant Strain Triangles

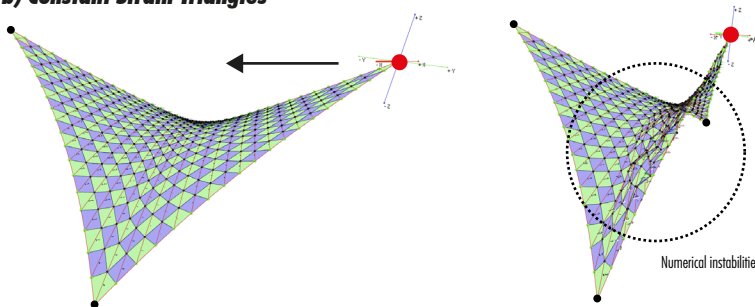


Fig. 5.28

Numerical instabilities of tensile elements when active modeling.

Tensile surfaces are represented by quad-topologies of particles connected via elastic bars. Depending on the orthogonality of the topology, the simulation of membranes can be simplified by modelling a pseudo-cable-net system. This eventually derives into a rough approximations of equilibrium shapes which means that surface elements may be later required to improve the accuracy of results. Therefore, at a given point, the quad-topology based on elastic bars would be automatically

⁴Based on pre-published article (Suzuki and Knippers 2017b): “*The Design Implications of Form-Finding with Dynamic Topologies*”.

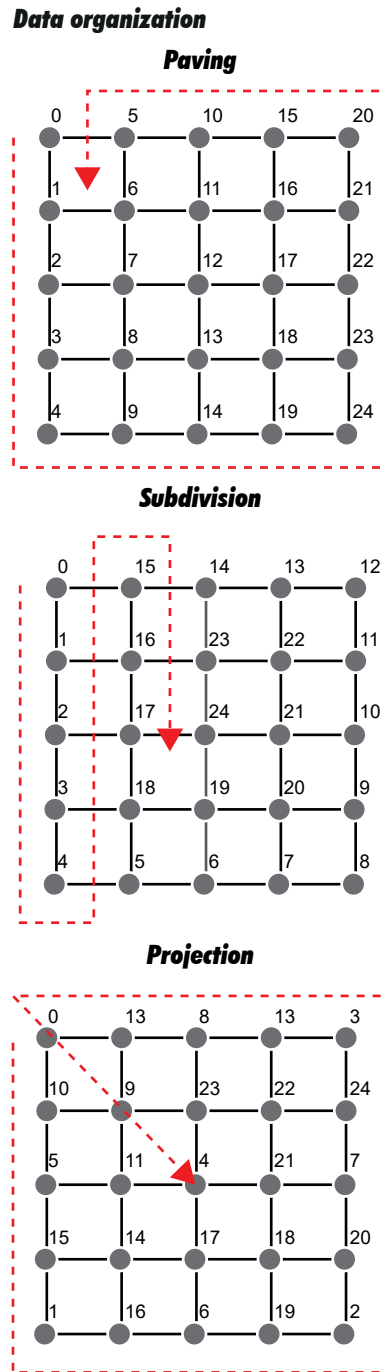
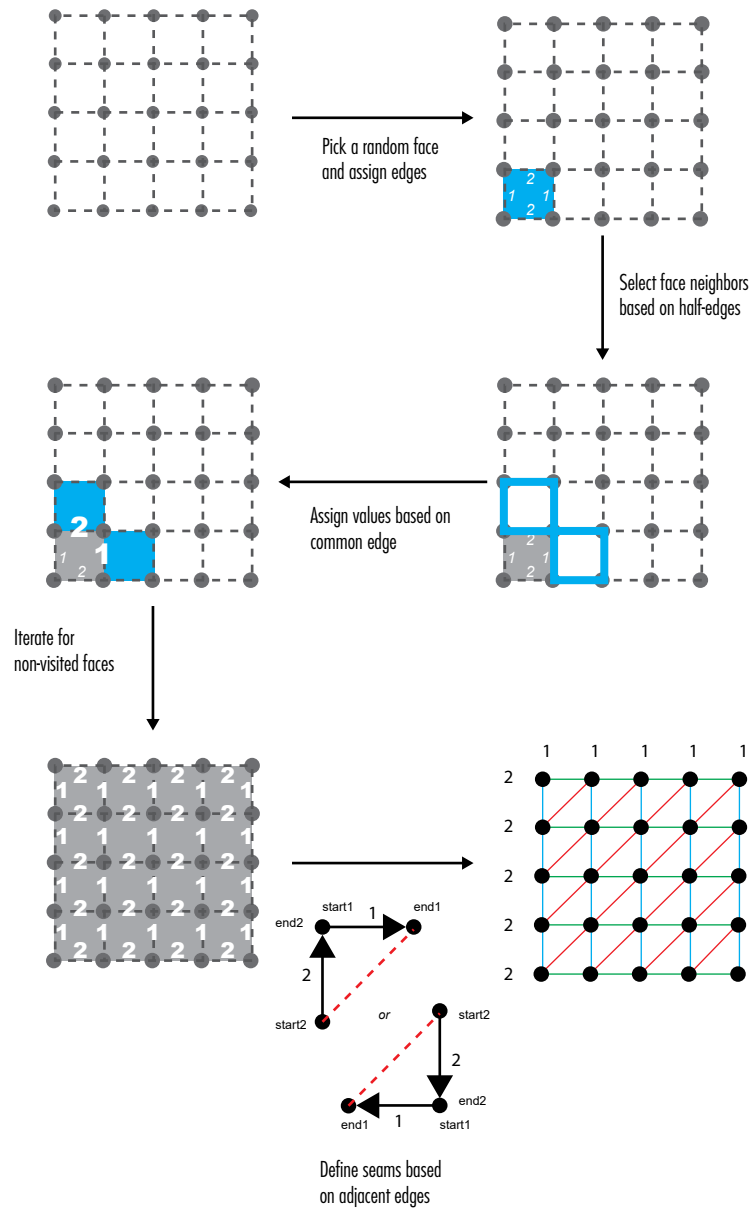


Fig. 5.29
Organization of data according to the corresponding discretization technique.

Fig. 5.30
Generation of suitable triangulations for CST elements.

converted into a triangular-topology of CST elements. To do this, each triangular face needs to have an edge parallel to a geodesic line representing the warp direction of the fabric. CST elements are more expensive to compute than elastic bars and tend to be highly unstable under continuous geometric and topologic modifications (Figure 5.28). It is important to note that this “mechanic” shift is only possible because the modeling process of tensile surfaces is entirely based on discretization techniques that always produce regular quad-topologies.

Edge classification and face triangulation for CST



Considering that each technique sorts data in diverse ways (Figure 5.29), the required triangular topology of CST elements is solved through an edge classification process based on constant adjacency queries (Figure 5.30). The process starts by randomly picking an initial quad-face of a given tensile patch. Since the edges of the face are always cyclically sorted, we alternate the assignation of 1 for edges aligned to the first direction and 2 for those aligned with the second direction. We need to recall that an edge is shaped by at most two half-edges in PHE data structures or multiple half-facets in AHF data structures. Hence, faces containing the corresponding half-edge or half-facet associated with those initial edges are searched. For each face found, if all its edges are not categorized then the assignation process starts from the pre-assigned edge. This process is repeated until all the edges of the tensile patch are assigned. Each face is then triangulated by creating a diagonal link between the end-nodes of two edges assigned with a different direction. This is only possible because of the classification of edges and the cyclic order of face-edges. All diagonals are then aligned with the same direction, creating the required seams for CST. This process only changes the mechanic layer of the topologic model without explicitly changing the rest of the layers. For this reason, a reverse shift from CST to elastic bars is still possible.

5.5.2.3 Multi-Resolution Tensile Patches

The case of tensile patches demands more attention because, compared with spline-beams, they massively increase the number of particles. Ultimately, this condition can compromise the development of real-time responses even with simplified formulations. An alternative topologic strategy is proposed to construct a multiresolution representation of each tensile patch. To do this, an additional rule is created for a tensile patch in which the state of particle can be changed, from passive to any other state and vice versa, in order to include or exclude the particle from the calculation of each integration step. In doing so, all mechanical elements associated with the passive particles are also excluded from the numerical process and some temporary links are created between particles with a non-passive state. This condition creates a low- and a high-resolution representation of the tensile patch.

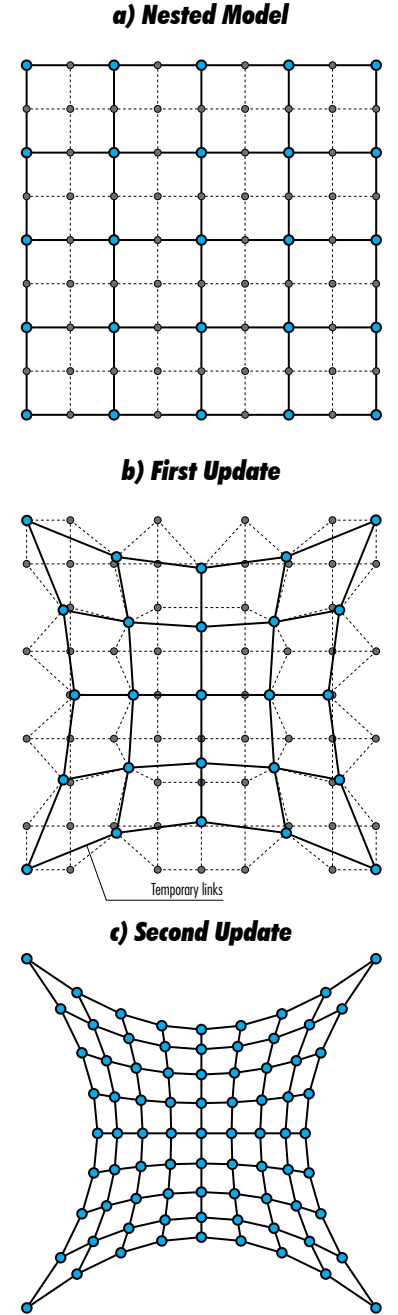
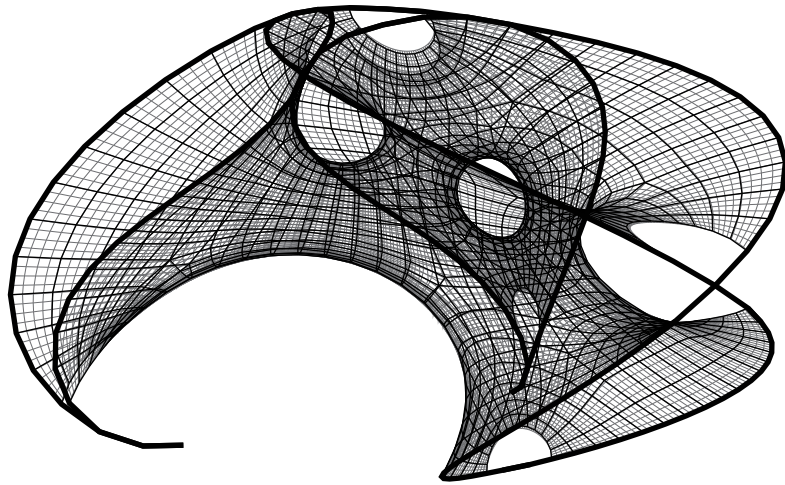
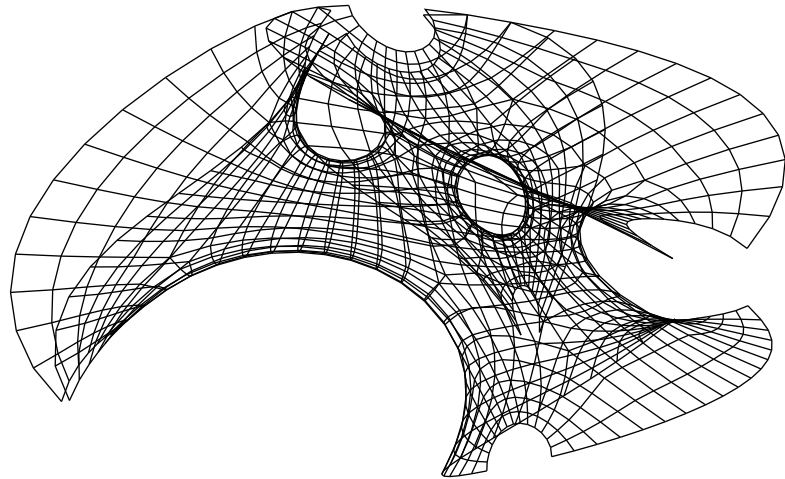


Fig. 5.31

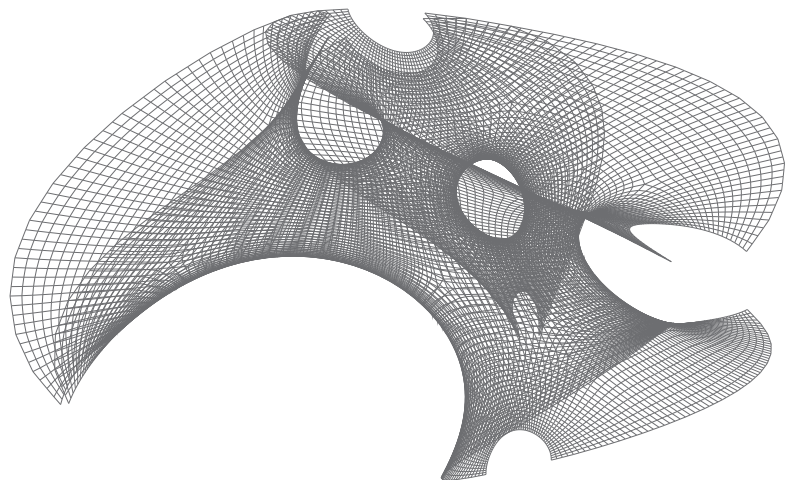
Nested structure of the low- and higher resolution representation. a) Nested model containing passive and non-passive particles, b) First update of particle's positions associated with the coarse topology, c) Second update of all particle's position.



Nested Model



Low-Resolution Model



High-Resolution Model

Fig. 5.32
Multi-resolution representation of tensile patches.

The process is simplified and generalized by the fact that tensile patches are always build from quad-topologies. In a first state, only particles that are associated with the low-resolution representation are included in the integration process. With a given input triggered by an equilibrium condition or a user-interaction, particles with a passive state are now activated to find the equilibrium of the complete model (Figure 5.31). The numerical solver automatically detects the new state of each particle and conducts the numerically procedure accordingly. As a result, the coarse topology is continuously updated whereas the high-resolution topology is only updated at critical points in time. This means that both topologies are not updated simultaneously in order to reduce the consumption of computational resources. The difference between both topologies is presented in figure (Figure 5.32). The adoption of these topologic modeling strategies permitted to dynamically adapt the levels of accuracy and interactivity according to specific modeling requirements without the explicit need to beak the numerical process and reconstruct the PS model.

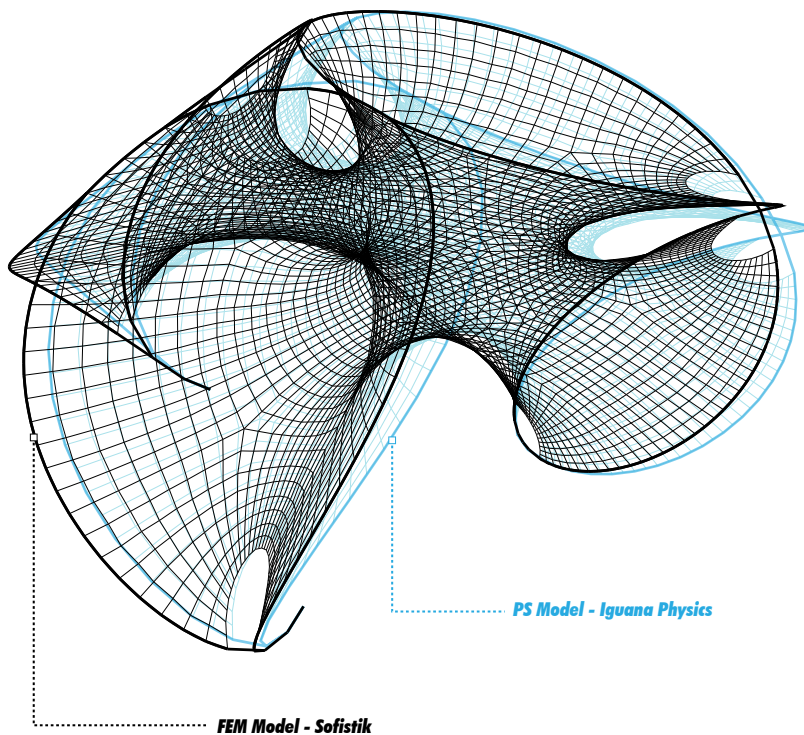


Fig. 5.33

Comparison between the equilibrium geometry resulting from our PS model and an FEM model.

The results obtained with our PS model were compared with an FE-model built in Sofistik with the same type of high-

resolution discretization by using the new bending-active module in which the prestress of an initially straight beam can be directly calculated from its bent shape. Because we are only considering the geometrical accuracy of a conceptual design tool, the results are compared without self-weight. The PS model is composed of beams with 3DOF and 6DOF per particle, cables and CST elements as shown in Figure 5.27. This test case was designed to cover an area of approx. 35 square meters with a maximum height to 3.7 meters and a total span of 6.5 meters. The average deviation calculated between all the nodes of the FEM model and the particles of the PS model is approx. 0.085 meters (Figure 5.33) which can be considered as a good approximation for a conceptual design tool.

6

CASE STUDIES

6 CASE STUDIES

In this chapter, a set of case studies will be introduced to demonstrate the application of topology-driven approaches for exploratory design studies of bending-active and textile hybrid structures. In this context, the first case study showcases the implementation of a routine for reconstructing the database of incidence graphs used in standard PS model in order to support topologic modifications on the fly. In contrast, the rest of examples demonstrates the use of our proposed generic topologic model based on combinatorial maps.

6.1 BENDING-ACTIVE DEMONSTRATOR 2017-18

6.1.1 DESCRIPTION



Fig. 6.1
ICD/ITKE Bending-Active Demonstrator 2017-18.

The Bending-Active Demonstrator 2017-18 (Figure 6.1) is a collaborative project between the Institute for Computational

Design (ICD), the Institute of Building Structures and Structural Design (ITKE) and the Institute for Textile and Fiber Technologies (ITFT), with students of the ITECH master program at the University of Stuttgart. This project demonstrates the use of a topology-driven form-finding approach based on the reconstruction of the topologic model for explorative form-finding studies. The design of the structure is entirely based on the elastic bending behavior of large span Carbon Fiber Reinforced Polymer (CFRP) strips with 10 to 15 mm thickness. The completed structure was shaped from five different strip morphologies with spans ranging from approx. 12 to 20 meters. Each strip was three-dimensional bent, and subsequently connected to define an architectural space with different levels of enclosure. The structure was arranged into two self-equilibrated assemblies of different size that together covered an area of 7.7 by 5.4 meters with a maximum height of 5 meters. The bigger assembly was conceptually designed to integrate a textile hybrid cantilevered module to provide additional levels of enclosure and achieve different spatial experiences. Such textile hybrid module was independently form-found using FE-models. Its equilibrium geometry was then used as an input for conducting the exploratory form-finding process of the strip assembly.

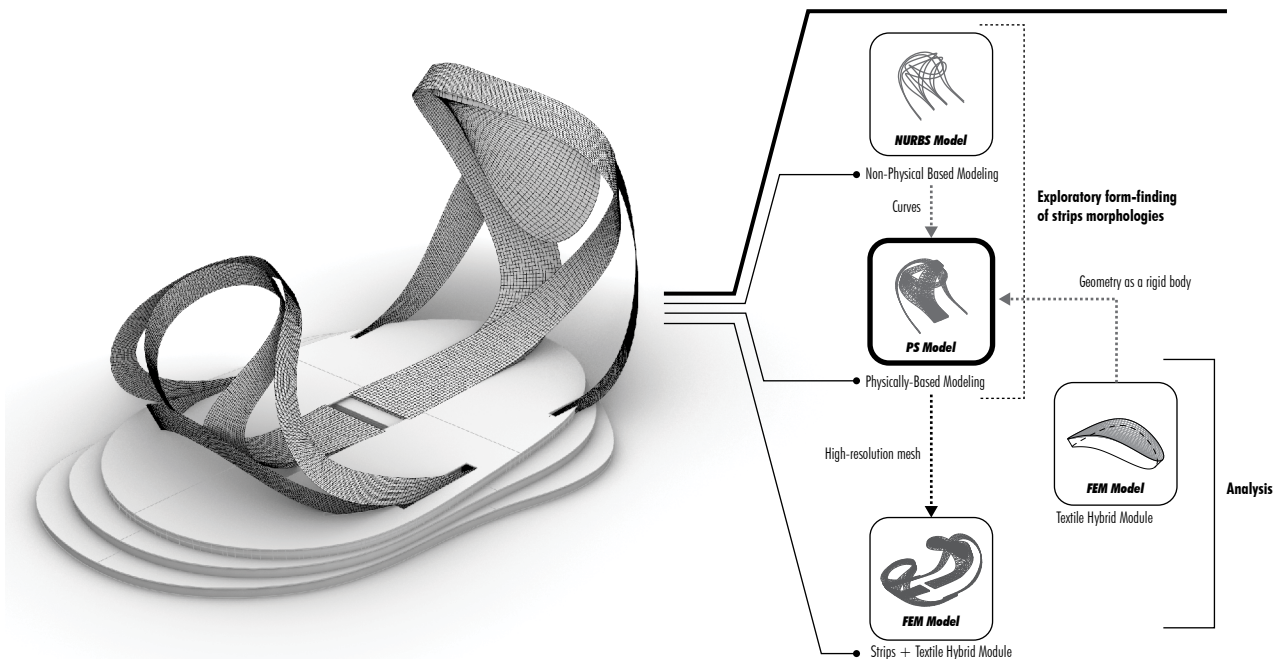


Fig. 6.2

Modeling workflow.

The design development of this structure was handled through a computational modeling workflow consisting of three stages (Figure 6.2). Each stage builds on a specific type of numerical model whose rate of accuracy and interactivity depends on the design problem to which is linked. The conceptual design stage uses exploratory PS models for conducting exhaustive form-finding studies of the global geometry of the structure. The design evaluation stage then takes the results that best satisfy functional and aesthetical criteria to produce simplified FE-models and rapidly check structural capabilities. If the given design iteration doesn't match structural requirements, the exploratory process is started again. Once a mature solution is reached in which all the design criteria are satisfied, the analysis stages produces a detailed FE-model to run a complete analysis of the structure.

The challenge of using PS models for explorative design studies is to numerically approximate the complex deformed geometry of both assemblies of strips, in an ease and interactive manner, in order to facilitate multiple iterations. Simulating the three-dimensional bending process of each strip always requires a pre-define set of complex programmatic displacements on the supports. When major variations on the design requires to continuously redefine those displacements, the explorative design process becomes intractable. To avoid this problem, it was assumed that the explorative form-finding process can start from a non-physical approximation of the deformed shape to eventually produce a refined solution using a physically-based model. In other words, starting from a rough approximation of the target geometry. This is possible because, in a simplified formulation, the prestress of an initially straight surface element can be calculated from dihedral angles of the discretized model. As a result, the flat configuration of each strip and the respective programmatic displacements for the erection process can be reversely solved only for mature solutions.

6.1.2 TOPOLOGY-DRIVEN APPROACH

Ideate from experimental studies on analogue models, the non-physical approximation of the target geometry of each strip is first digital sculpted using mesh-modeling techniques

in MAYA, and then parametrically variegated using NURBS in Rhino/Grasshopper. However, this geometric modeling process does not always guaranty the developability of each strip which was a precondition established for conducting the form-finding process. A two-step numerical procedure was proposed in which the surface geometry of each strip is first discretized into quads to apply a planarization process, and then converted into triangles with a structured logic to include the bending behavior. This exploratory stage demonstrates the implementation of a topology-driven form-finding approach for conceptual design explorations where the local topological properties of the PS model are gradually reconstructed to match the predefined modeling requirements appearing during numerical integration. That is, it builds on enabling topologic modeling capabilities on standard PS models.

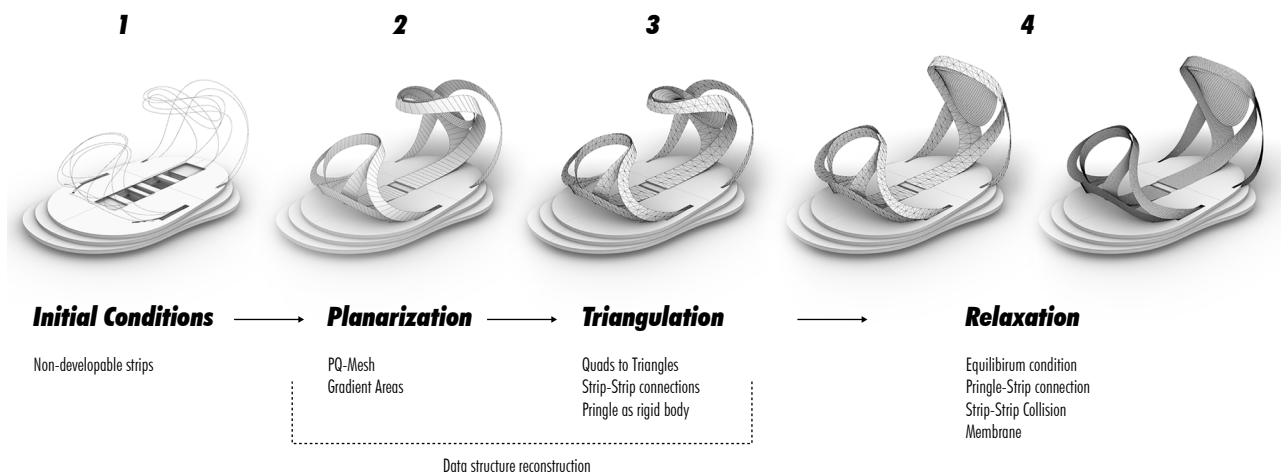


Fig. 6.3

Topologic transformations implemented within the exploratory form-finding tool.

A form-finding tool running in Rhino/Grasshopper CAD environment was developed using C# for customizing the implementation of the Kangaroo constraint solver. This solver implements the Variational Form of the Implicit Euler Method and was originally designed for static or quasi-static topologic properties. Our implementation then permits to automatically reconstruct the entire database of the numerical solver for updating the topological properties and the corresponding constraint functions without necessarily breaking the integration process. The turning point for applying those topologic transformations is

established when the transition from a quad- to a triangular- mesh topology was required (Figure 6.3).

The process starts by discretizing the NURBS geometry of each strip into a structured quad-mesh. Different parameters are included to locally control the density of each mesh according to its curvature. On this basis, the PS model is constructed by using vertices to initialize particles and mesh-topologies to define constraints. Even though, the model stores local topological properties within constraint functions, it does not keep enough information to solve all its topological relationships. Therefore, the complete mesh-topology of each strip, linking vertices with particles, is stored within the PS model for solving upcoming adjacency queries. Numerical integration starts with a planarization process for generating planar quad (PQ) meshes in order to guaranty the developability of each strip. The planarity of those quads is continuously evaluated to trigger the reconstruction of the database of the numerical solver. Once triggered, the reconstruction starts with a subdivision process of those quads. This condition implies updating all auxiliary mesh-topologies, adding particles that are initialized from the new vertices and cleaning the previous set of constraints. The solver now contains a larger number of particles without any constraint functions associated and a set of high resolution mesh-topologies.

The high topologic resolution was a pre-established requisite for facilitating the creation of FE-models. However, adding new constraints with this level of resolution will eventually compromise the generation of fast iterations. To manage the rate of interactivity of the model, a multiresolution topologic representation was created to automatically select the use particles during the continuation of the numerical integration. This implies the creation of a coarse mesh-topology from each fine mesh-topology (Figure 6.4) and then carefully triangulate it to facilitate the definition of the required constraints for simulating bending behaviors. At this point, the geometry of the textile hybrid module, which was form-found in another process, was included into the PS model to be interpreted as a rigid body. Using the textile hybrid module as a rigid body implies having a non-deformable element moving with the deformation of strips (Figure 6.5). With the

database entirely reconstructed, the numerical process continues until a static equilibrium state is finally found.

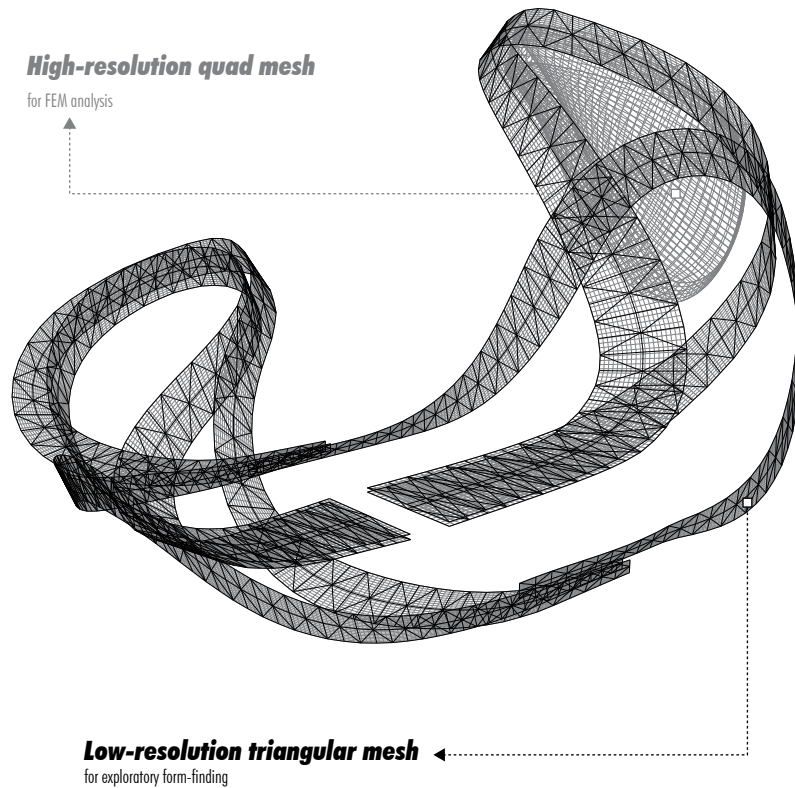


Fig. 6.4

Multi-resolution representation of strips topology.

Once the particles associated with the coarse mesh find an equilibrium position, the rest of the particles associated with the fine mesh are now activated and pulled to an equilibrium position by using a distance constraint function on their edges. The implementation of this exploratory topology-driven form-finding tool aimed to improve modeling interactivity by minimizing the need to manually reconstruct the PS model when addressing different problems appearing throughout the modeling process. Eventually, designers are mainly focused on calibrating the parametric model for producing multiple iterations without putting too much effort on the construction and reconstruction of the numerical model. The main drawback is that the solver stores the topologic model as an incidence graph in which case maintaining its consistency throughout the integration process is a complicated task that requires auxiliary data structures (e.g. mesh-topologies) to be stored and the implementation of multiple operations to avoid topologic conflicts.

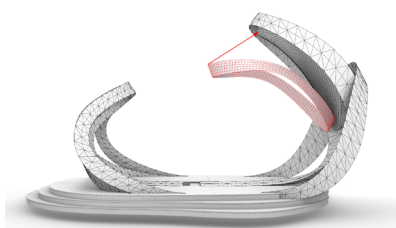


Fig. 6.5

Geometry of the textile hybrid module moving as a rigid body within the simulation.

6.2 AAVS MADRID 2016⁵

6.2.1 DESCRIPTION

The design development of the bending-active part of the Architectural Association Visiting School (AAVS) demonstrator in Madrid 2016 showcases the use of a topology-driven form-finding approach based on evolving topologic properties. This small-scale demonstrator was designed during a two-weeks workshop in Madrid with students that didn't have previous experience on numerical methods or bending-active typologies. The workshop focused on investigating how the intuitive and progressive aggregation of linear active-bending elements can serve as the basis for shaping a morphological complex lightweight system. In doing so, the aim was to introduce these new practitioners to the realm of simulation-based design in a more ludic and intuitive way without the need of exhaustive studies on analogue models. Under these circumstances, it was assumed that a limiting factor for students would be to easily understand the high level of modeling abstraction (e.g. topology, geometry, physical) existing on numerical form-finding processes. This is mainly associated with the concept of solving topologic and geometric properties for deformed physically states by only working at the level of non-deformed ones. The computational modeling experience needed to incorporate additional mechanisms to emulate the ludic characteristics of empirically playing with physical materials for deriving structural forms, without losing its inherent capacity to manage high levels of complexity. This constituted the best scenario for evaluating the use of ElasticSpace in which a more flexible and intuitive modeling process can be reached when introducing topologic transformations on the fly.

The AAVS demonstrator was conceptually designed to work as a temporary registration and information counter at the building of the Colegio Oficial de Arquitectos de Madrid (COAM) (Figure 6.6-6.7). The completed structure covered an area of 6 by 2 meters with a maximum height of 2.5 meters. This structure consisted of a complex connected assembly of robotically-bent aluminum bars shaping the desk of the counter, and a non-standard load-bearing network of actively bent GFRP rods from

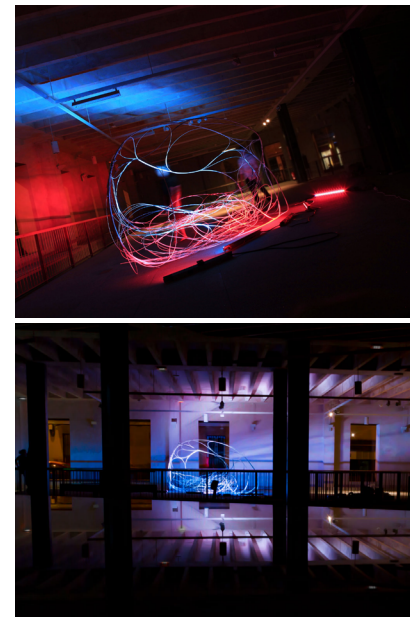


Fig. 6.6

Temporary structure of the AAVS Madrid 2016.

⁵ Based on pre-published articles (Suzuki and Knippers 2017c): "Topology-Driven Form-Finding: Implementation of an Evolving Network Model for Extending Design Spaces in Dynamic Relaxation"; (Suzuki and Knippers 2018): "Digital Vernacular Design: Form-Finding at the Edge of Realities".

where the assembly of aluminum bars hung. The bending-active part was composed of 22 pultruded GFRP rods with cross-section diameters ranging from 5 to 20 mm and span lengths from 1.5 to 6 meters. All forces acting on this part of the structure are solved internally without reaction forces towards the surroundings. The system was designed to be sequentially assembled on-site based on the continuous aggregation and bundling of actively bent GFRP rods. This condition was conceived to gradually increase the global stiffness of the structure by progressively inducing stiffening effects on local areas.



Fig. 6.7

Temporary structure of the AAVS Madrid 2016.

6.2.2 TOPOLOGY-DRIVEN APPROACH

The design development of the bending-active part was entirely conducted with ElasticSpace. The PS model is gradually built during numerical integration by means of explicit modeling operations for adding, deleting or modifying basic building blocks represented as individual polylines with real-time physics. All the information required for the construction of numerical models is computationally solved through real-time user model interactions without any need for analogue models. The simplified spline-beam formulation with 3DOF per node was used to implement those building blocks and the Leapfrog method for integrating the equations of motions. The stiffness of each spline-beam was calibrated with real cross-sectional properties of available GFRP rods and a Young's Modulus of 25000 Mpa which was the value

provided by the manufacturer. In this simplified formulation, the value of the shear modulus is omitted since no rotational degrees of freedom are considered.

The complex global connectivity of the PS model was dynamically explored by intuitively selecting an appropriate location to introduce a new spline beam element, and by defining to which part of the pre-existing PS model this beam connects. An inappropriate operation resulting on a spline beam erroneously placed, or a bad connection, can be easily solved by deleting the element or reverting the connection. For larger systems, important considerations were taken regarding the timing of operations and their potential side effects on existing elements. In fact, it was observed that such operations can easily degenerate into the presence of unnecessary elements that needed to be eventually removed.

Beams were initialized as straight elements and then gradually deformed during the exploration of connections through user's interactions. The geometric values for defining their initial lengths were restrained to a range between the maximum length of available GFRP rods and the minimum length to manually bent them. Connections are modeled by using rule 3 in which a set of particles are merged into a unique particle with cross-reference to the corresponding elements. Under these conditions, no issues were reported from users regarding the instabilities on the numerical solution. PS models are constantly re-shaped through the progressive inclusion of spline beam elements calibrated with real material properties. This computational modeling permits to rapidly developed complex systems of interconnected curved elements. Throughout the numerical process, the PS model continuously informed designers about minimal bending radii, strains, stress and chronological modifications. Note that the latter was only possible because the numerical scheme is entirely designed to keep track of all topologic and geometric changes during time integration. Eventually, this unique chronological record served to inform the construction process by associating those time-based modeling operations with the assembly sequence of the final structure. A quick representation of the construction process can be easily obtained by retrieving the database of past

equilibrium states, in order to anticipate certain actions like the use of scaffolds or temporal elements. This additional information was particularly helpful for the realization of this intricate bending-active structure that progressively passed from arc to cantilever and finally to grid conditions (Figure 6.8).

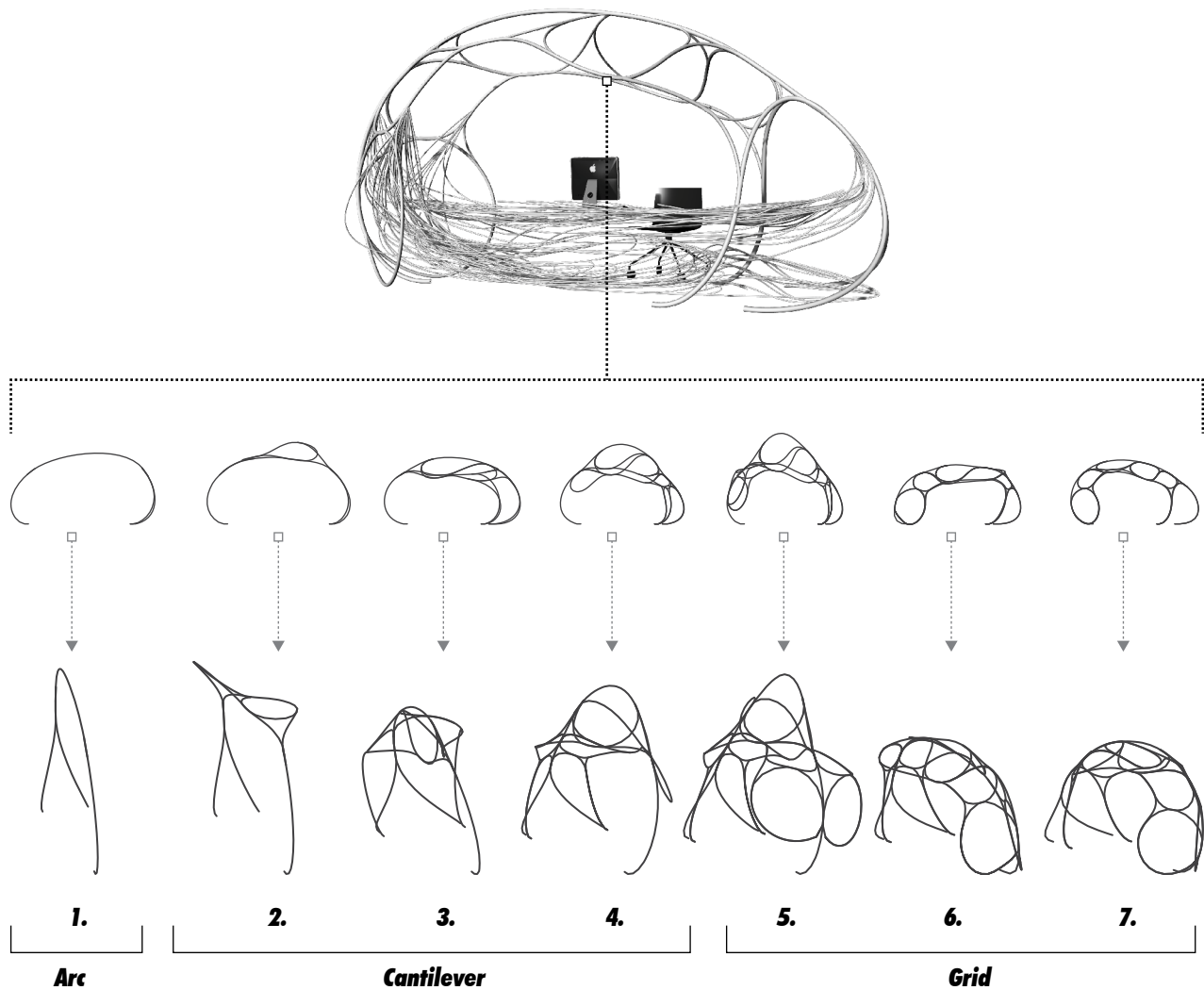


Fig. 6.8

Sequential topologic construction of the bending-active part.

This project has served as the basis for validating, in terms of numerical stability, the reliability and robustness of our topology-driven implementation based on evolving properties. More interesting is that it has proven the increased flexibility that topology-driven approaches have for the exploration of design spaces, but also their educational potential for introducing core concepts of numerical form-finding in a more ludic and intuitive

way. Even though the final design of this structure was entirely derived from direct user interactions on the PS model, we realized that, from a given size, the taking of decisions regarding its modification was not so straightforward. With a large number of deformed elements, design spaces defined by PS models rapidly became more complex and less intuitive to solve. For this reason, design studies were fully oriented towards the exploration of a maximum combinatorial diversity from a minimum usage of beam elements.

6.3 TEXTILE HYBRID MODELS

6.3.1 BAT01⁶



Fig. 6.9

Bat01.

6.3.1.1 Description

The first case study addressed the conceptual design development of a textile hybrid system that resulted on two bending-active parts self-equilibrated by a “patched” doubly-curved tensile surface (Figure 6.9). The design development was conducted on the basis of a bottom-up workflow without any preconceived idea of the target geometry. Therefore, the aim was to evaluate if only with the continuous and intuitive understanding of the interdependent behaviors between tensile and bending-active elements that appear during numerical integration, a hybrid system with certain structural, functional and architectural qualities can be designed. In this case, the topologic properties of the PS model were chronological constructed through explicit

⁶Based on pre-published article (Suzuki and Knippers 2018): “Digital Vernacular Design: Form-Finding at the Edge of Realities”.

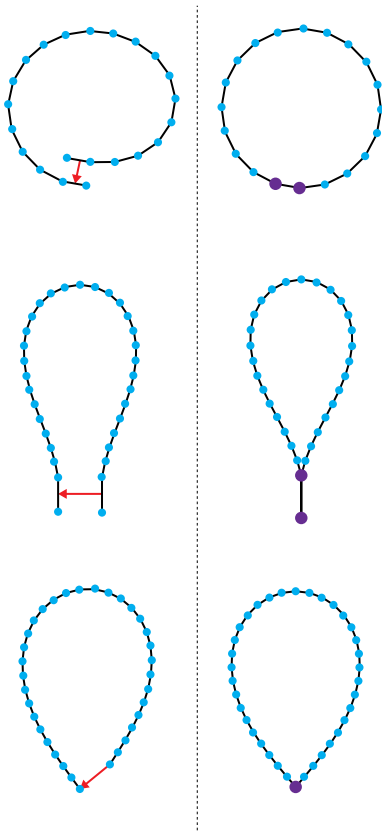


Fig. 6.10
Looped typologies.

modeling operations based on two types of building blocks representing elastic rods and textile membranes. The PS model is simplified as much as possible to benefit the rate of interactivity of the modeling process and, ultimately, facilitate the exploration of new material articulations. Elastic rods are modeled as simplified spline-beam elements with 3DOD per node and tensile surfaces as cable-net models using elastic bar elements. Both types of building blocks may be respectively conceptualized as one and two-dimensional assemblies of particles with real-time physics. The form-finding process of textile membranes only focused on simulating a surface under pure tension in which all the tension forces needed to be equilibrated. The actual material properties of membranes are therefore neglected, and the numerical procedure is only driven by geometrical stiffness. For bending-active elements, material stiffness is not fictitious and was calculated from real properties. It was assumed that the hybrid system would be shaped for pultruded GFRP rods with constant circular cross-section diameter ranging from 10 to 24 mm and an arbitrary maximum span length of 6.5 meters.

6.3.1.2 Topology-Driven Approach

Motion constraint of a PS model is an important consideration to take into account when introducing topological transformations on the fly. This condition does not relate to overshooting problems derived from numerical instabilities, but with the possibility of having multiple equilibrium states from the same set of numerical conditions. Accordingly, minor topological changes can easily produce undesirable equilibrium conditions if the PS model is not adequately constrained. Constraining the motion of a PS model is managed by explicitly locking the movement of particles, in at least one of their local axes, or by associating them with more elements to restrict further displacements. Here, we aimed to explore the primary use of spline-beams that are bent on themselves, to softly constraint the PS model and obtain a better control of the modeling process when introducing further topologic changes.

The exploratory form-finding process started with some topologic studies for connecting a spline-beam with itself. This is

addressed by dynamically altering the local connectivity of each spline-beam through direct user-model interactions resulting on the creation of shared particles. A shared particle is a particle with cross-references to multiple elements, and/or with multiple references to the same element. It was found that a spline-beam could generate at least three different looped typologies when it is connected to itself (Figure 6.10). A closed ring is constructed with two consecutive particles of the first end connected with the corresponding particles of the second end in an ascending order. A drop-like loop with a single joint is the result of connecting both particle's ends together. Finally, a drop-like loop with a double joint is constructed with two consecutive particles of the first end connected with the corresponding particles of the second end in an inverse order. Geometrical variations of these typologies can be obtained by moving the location of connections in only one of their ends.

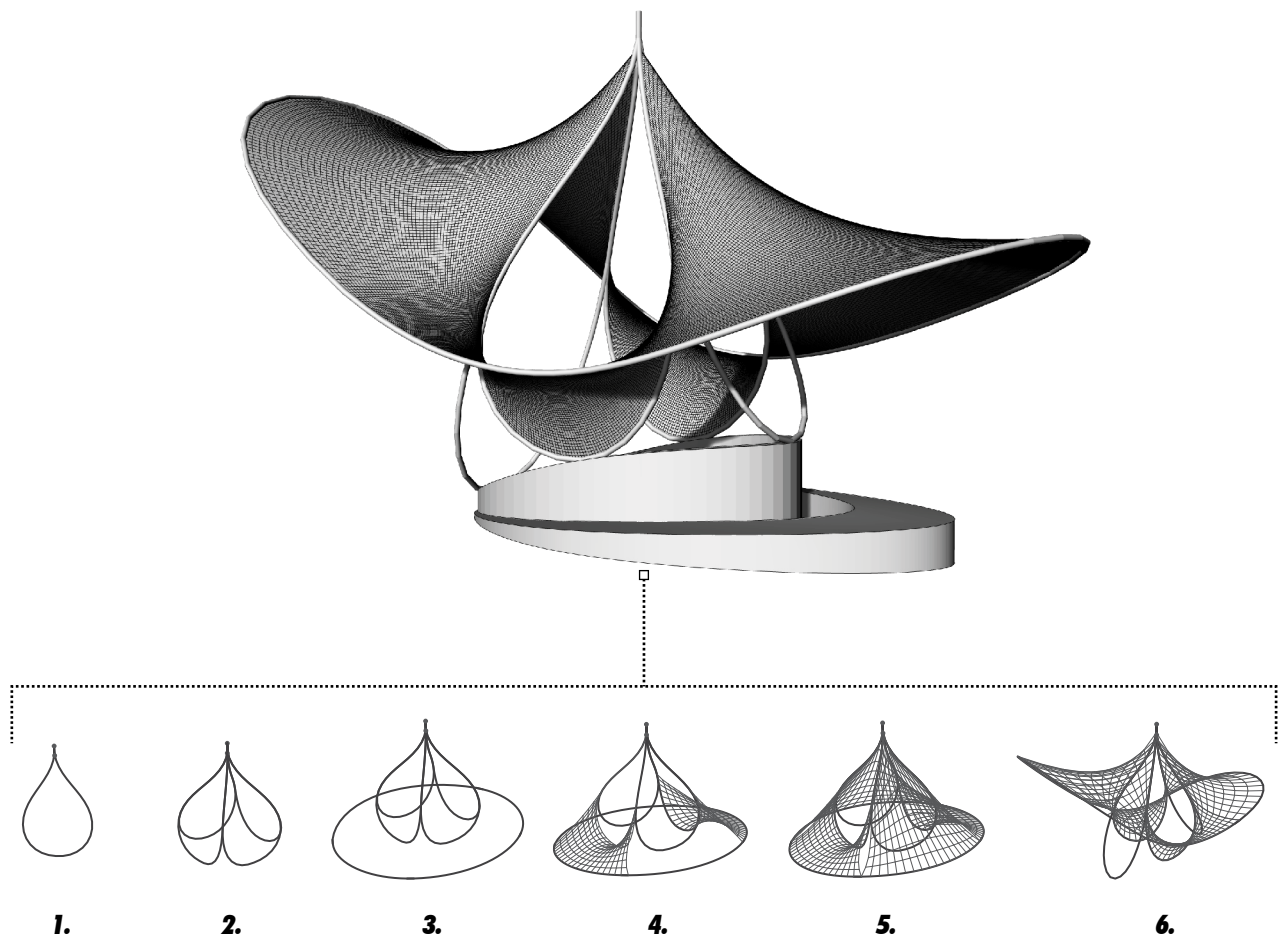


Fig. 6.11

Sequential topologic construction of the textile hybrid system.

The exploratory process (Figure 6.11) ended with an inverted funnel-like assembly of 4 drop-like modules with double pin joints that connected together formed a steady vertical base. This funnel-like assembly was then placed at the center of the closed ring which was in turn shaped from the connection of 4 initially straight beam elements. Both bending-active assemblies have been constructed on the basis of direct user-model interactions through which new spline-beams were gradually added into the PS model and the global connectivity transformed. Both bending-active parts were used to pre-stress the membrane and stabilize the whole system. The continuous tensile surface was progressively modeled by creating simple patches of interconnected particles that were later combined globally (Figure 6.12). Each patch is defined as a structured quad mesh-topology of particles connected via elastic bars with a clear orthogonal directionality that is associated with the weft and warp direction of the textile fabric. The generation of this mesh topology required users to always pick two spline-beam elements within the current PS model and select the corresponding creation areas. The form-finding tool then automatically detects the type of boundary condition and apply the corresponding meshing process. Because the resolution of the mesh is entirely determined by the pre-existent discretization of selected elements, each spline-beam had enough particles to generate a relatively high-resolution mesh.

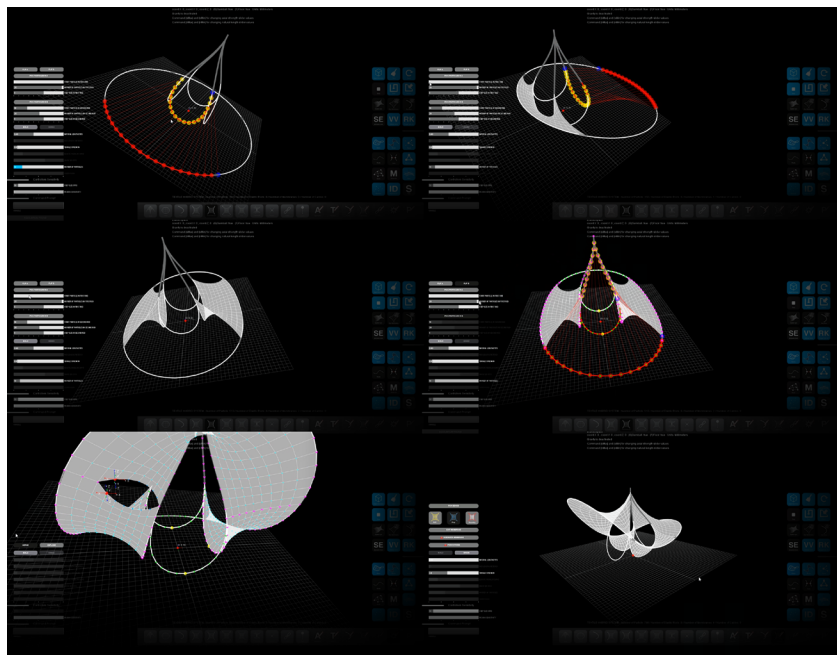


Fig. 6.12
Sequential addition of tensile patches.

The resulting doubly-curved geometry of the completed tensile surface is only held in shape by the delicate equilibrium state of all the parts. This condition permitted to design a self-tensioned hybrid system with null reaction forces projected towards its surroundings. The amount of double curvature of the tensile surface is directly controlled by the magnitude of pre-stress in the weft and warp direction of each patch. The completed tensile surface is composed of a front and back patch with corresponding areas of approx. 5.4 and 4.5 square meters and two lateral patches with equal area of approx. 1.7 square meters. The completed hybrid structure was designed to cover an area of approx. 4.7 meters by 2.3 meters with a highest point locate in the central column of 2.7 meters.

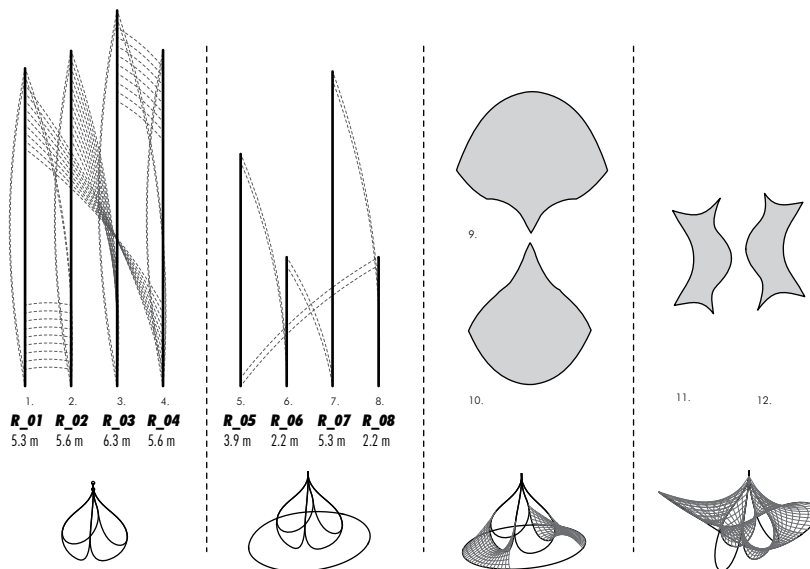


Fig. 6.13

Inventory of transformations to assist the assemblage process.

All topologic operations were tracked and recorded to later retrieve a chronologic inventory of transformations for conceptually guiding the assemblage process of the resultant structure (Figure 6.13) (Suzuki and Knippers 2018). The way the whole tensile surface was modeled, based on the sequential addition of simpler tensile patches, greatly simplified the realization of cutting patterns. For doing this, the final PS model needed to be imported into Grasshopper and reconstructed into polylines and meshes. Each mesh has a specific orthogonal directionality related to the weft and warp directionalities of the tensile patch to which is associated. Because of this, it can be patterned by interpolating seam lines between the warp lines to achieve best use of the available fabric width (Figure 6.14).

This permitted to divide each mesh into smaller pieces that can be easily exported to Sofistik for flattening and compensation. This way, a fast design-to-production workflow was created by processing the information stored during the modification of topologic properties.

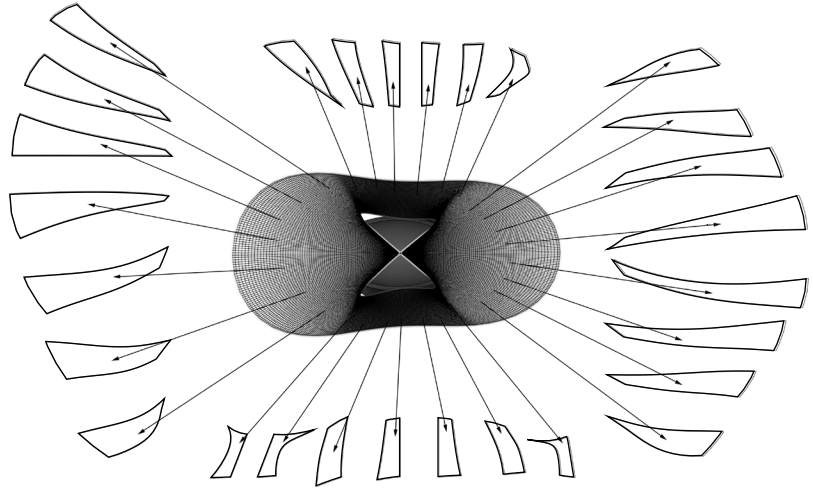


Fig. 6.14

Cutting pattern generation based on the quad topology.

6.3.2 BAT02



Fig. 6.15

Bat02.

6.3.2.1 Description

BAT02 was a temporary textile hybrid demonstrator (Figure 6.15) built in August 2018 at Foster and Partners, and later exhibited at the Danish Architecture Center in Copenhagen during the Innochain conference: “Expanding Information Modelling for a New Material Age”. This demonstrator was designed to work as a personal meeting pod containing an internal space that could adapt to varied levels of enclosure. The

completed structure was designed as a doubly-curved system shaped from the combination of a closed bending-active ring and a hand-made wood-textile membrane produced by Tesler Mendelovitch (Figure 6.16). Even if the mechanical actuator was not integrated into the built demonstrator, such enclosure levels were conceptually associated with a range of equilibrium states that had been numerically explored to define a plausible dynamic behavior of the hybrid system (Figure 6.17). The double curvature was induced by prestressing the membrane with the closed ring and adding two secondary cables. The required stiffness on the bending-active ring to prestress the membrane was attained by bundling four pultruded GFRP tubes with a cross-section diameter of 9.25 millimeters and a thickness wall of 3.25 millimeters. As a result, the structure was solely held in shape by the prestress of tensile elements which avoided the ring deforming back to its flat configuration.

6.3.2.2 Topology-Driven Approach

A series of exploratory studies with different topologic properties were conducted for developing the final design. The main focus was on modeling simple bending-active boundaries from which highly complex configurations of interconnected tensile surfaces can be created. Modeling operations during numerical integration were restricted to two types of building blocks representing elastic rods and textile membranes. Simplified spline-beam elements with 3DOF per node were used to model elastic rods while simple patches of tensile surfaces were modeled with elastic bars. The use of simplified formulations was prioritized for these exploratory studies in order to improve the rate of modeling interactivity. This is because the progressive addition of tensile patches can rapidly increase the size of the PS model, and eventually compromise real-time responses. The Leapfrog implementation in ElasticSpace can produce real-time responses with an approximate number of 16K particles for models entirely composed of spline-beam elements, and 26K particles for those that are composed of elastic bars.

Conceptually, the structure was designed to be transportable and ephemeral without any kind of special

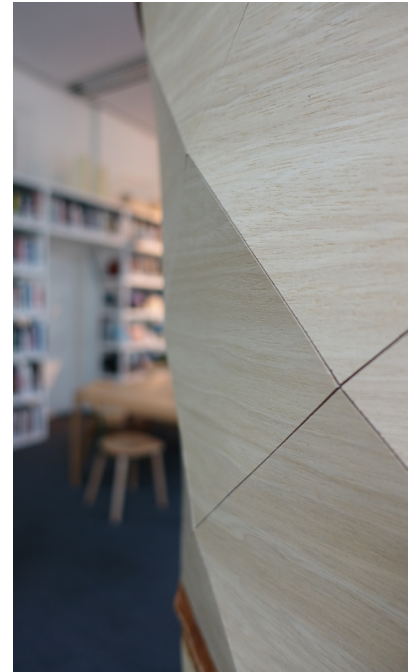


Fig. 6.16

Wood textile membrane.

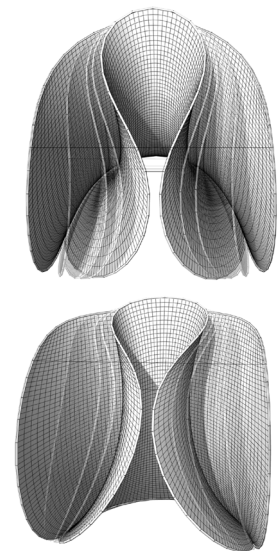


Fig. 6.17

Range of potential equilibrium states between the open and closed conditions.

boundary condition required from the surrounding where it would be placed. To this end, it was assumed that the hybrid system needed to be constructed from closed bending-active rings since they permit to minimize the reaction forces to the surroundings. During the exploratory form-finding process, bending-active rings needed to be created from topologic conditions under continuous transformation. To facilitate such topologic modeling, the software provided designers with multiple creation methods for adapting to such conditions. Bending-active rings can be created from a given selection of at least three non-collinear particles defining a polygonal boundary that is then closed and subdivided to match the target particle resolution. This type of creation was utilized when the bending-active ring needed to be directly connected with other elements of the PS model. Methods not linked with preexistent elements constructed the ring by letting designers manipulate an initially straight spline-beam, as in the preview case study, or by defining a center point, a plane, a radius and a particle resolution. All these ring creation methods were employed for exploratory studies, but the latter was always used for initializing the construction of the PS model. Note that, in such cases, the diameter of the ring is a geometric parameter associated with the span length of the spline-beam.

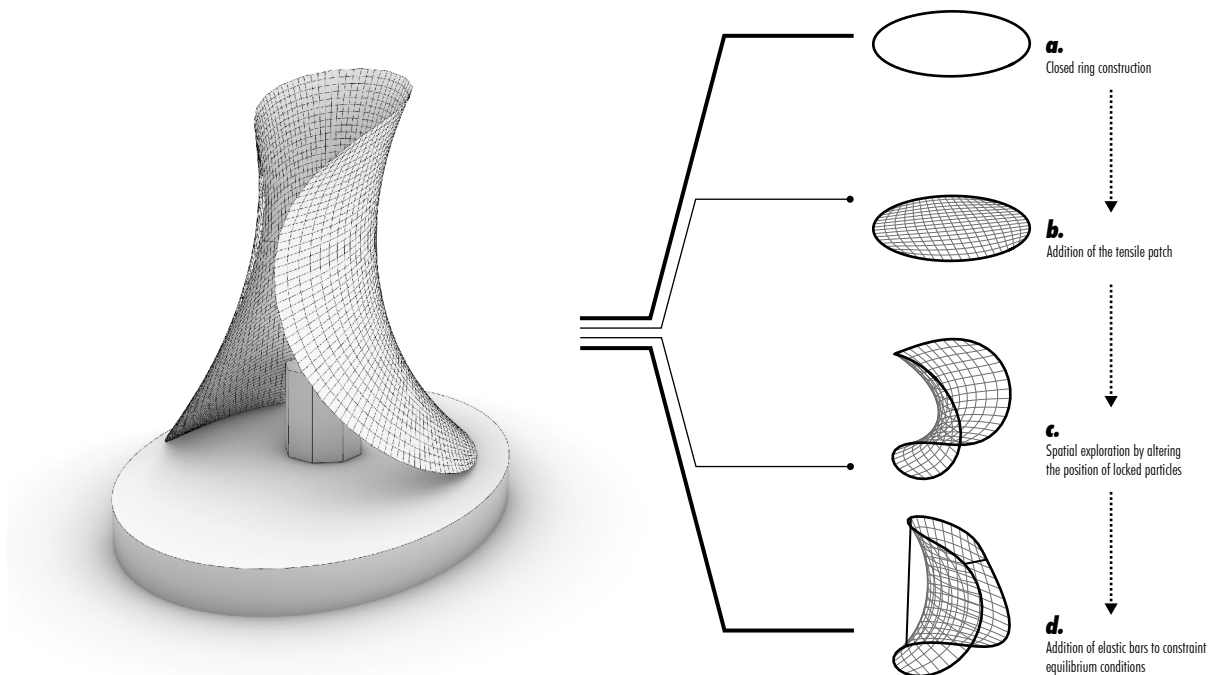


Fig. 6.18
Sequential topologic construction.

A closed spline-beam ring, constructed on the XY plane, served as the basis to start shaping the PS model (Figure 6.18a). This closed ring constituted a polygonal boundary with a given set of particles from which a tensile patch needed to be constructed. The paving algorithm implemented within the functionalities of the software was then used to generate a structured quad mesh-topology of particles with clear orthogonal directionalities (Figure 6.18b). However, the addition of the tensile patch did not necessarily change the initial flat configuration of the ring. Hence, a three-dimensional deformation was induced by limiting the motion of the PS model and explicitly manipulating the position of some particles. Four particles placed in each quadrant of the ring were locked and two of them were moved normal to the direction of the XY plane. The position of those locked particles was gradually adjusted to create different iterations of a cocoon-like geometry (Figure 6.18c). Upper locked particles didn't satisfy initial design criteria since this would require external supports but unlocking them could produce the ring going back to its flat configuration. To prevent this and allow to control the deformation of the entire system through the prestress on the tensile surface, two elastic bars were introduced before unlocking the particles (Figure 6.18d). Under these conditions, the equilibrium state was extremely delicate causing snap-through behaviors when conducting additional topologic transformations and geometric calibrations on the PS model (Figure 6.19). To avoid this, two types of design studies were conducted to evaluate

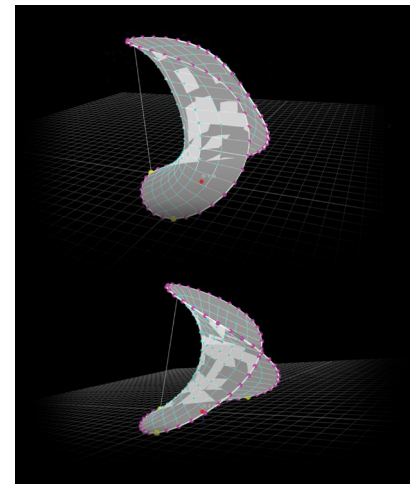


Fig. 6.19
Snap-through behavior.

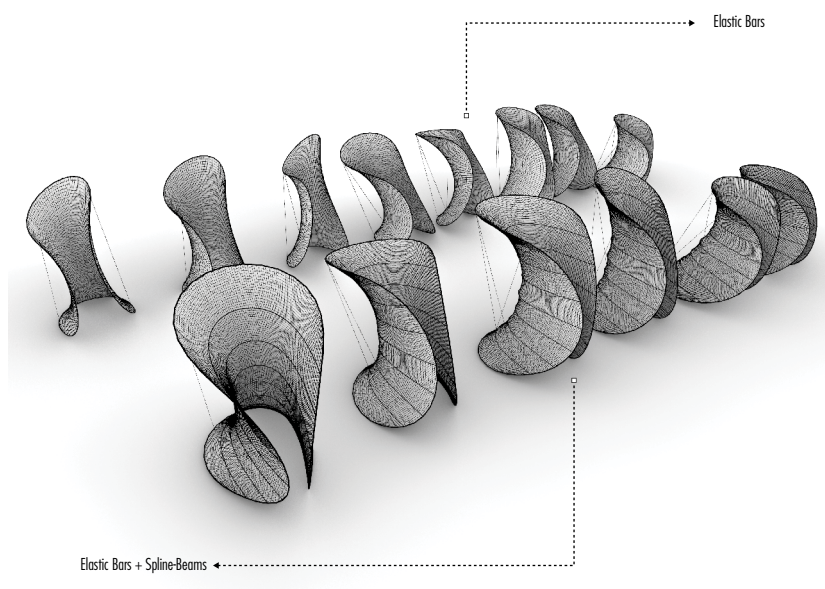


Fig. 6.20
Exploratory studies regarding the addition of elastic bars and spline-beam elements.

the strategic addition of either more elastic bars or spline-beam elements (Figure 6.20). In both types of studies, enclosure levels of the hybrid system were controlled by optimizing the geometric properties of the PS model.

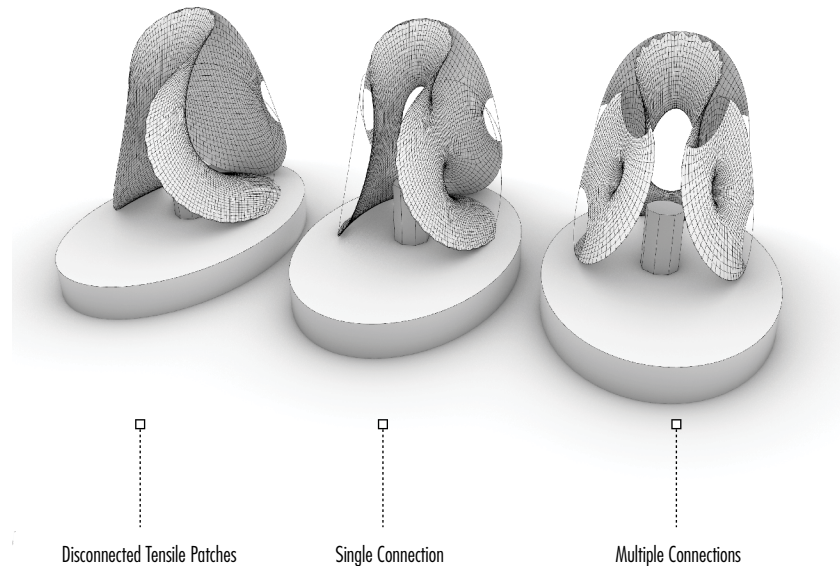


Fig. 6.21

Exploratory studies regarding the addition of tensile patches.

For the first type of exploratory studies, the addition of elastic bars resulted on simpler topological configurations that, in the best case, only required one tensile element centered in the back of the hybrid system. On the other hand, the second type of studies resulted on more intricate network configurations of spline-beam elements. Within these conditions, different spatial and enclosure experiences could be created by breaking the existent surface typology through the progressive creation and connection of additional tensile patches. The result is an interconnected arrangement of tensile patches representing a continuous surface. As discussed in the previous chapter, the topologic properties of the PS model are stored within two different types of data-structure associated with the dimensionalities of the building blocks. In the case of tensile patches, a half-edge data structure is used to support constant queries of the topology. Each tensile patch contains individual geometrical properties and a type of embedded local orthogonality that is associated with the weft and warp directionalities of a textile. Similar to the case of bending-active networks, the great benefit of this incremental construction technique is the generation of highly complex tensile surfaces from the combination of simple elements (Figure 6.21). A detailed

review of the multiple ways to create those patches are found in the topologic modeling rules presented in the previous chapter.

All design explorations were conducted within digital environments without the need of scaled analogue models. At the end, the simplest design iteration containing the back centered tensile element was selected to be materialized since it simplified the fabrication of the wooden-textile membrane. The computer modeling workflow of ElasticSpace proved to be highly intuitive, reaching similar levels of playfulness than analogue techniques. It was this condition that facilitated the exploration of the different design concepts that continuously emerged during the topological modeling with real-time physics of the hybrid system. As a conceptual design tool, ElasticSpace allowed to better explore geometric variations and topologic differentiations of textile hybrid systems shaped from a relatively small number of elastic components with highly intricate relationships. Note that these latter conditions were the ones that made the ICD/ITKE M1 La Tour project so relevant and groundbreaking.

6.4 IAAC-GSS 2018 COMPUTATIONAL BAMBOO⁷

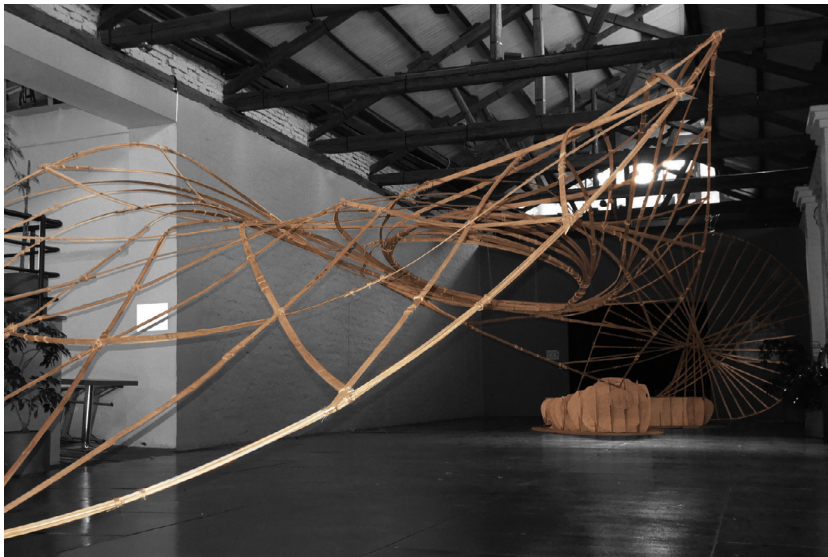


Fig. 6.22

IAAC GSS 2018 bamboo demonstrator.

6.4.1 DESCRIPTION

Computational Bamboo was an experimental demonstrator developed during the Global Summer School (GSS) of the Institute for Advanced Architecture of Catalonia

⁷Based on forthcoming article (Suzuki et al. 2019): “Computational Bamboo: Digital and Vernacular Design Principles for the Construction of a Temporary Bending-Active Structure”.



Fig. 6.23

IAAC GSS 2018 bamboo demonstrator.

(IAAC) in Quito and exhibited at the Museum of Interactive Science in the second semester of 2018 (Figure 6.22-23). All the students taking part of the workshop did not have previous experience on numerical methods or bending-active typologies. The aim was then to investigate the emerging design potentials that bending-active structures have when the relative high-tech technology required for its production is replaced with low-tech technology and locally available natural materials such as bamboo, and when more intuitive parametric and numerical strategies are tailored for these purposes. The structure employed laths and rods of bamboo that were cut with a length of 3 meters from *Phyllostachys canes*. These semi-finished products were produced near the construction site. Laths had a solid rectangular cross-section of 10 by 50 millimeters which was sanded with machinery for homogenization. On the other hand, rods had a circular cross-section of 10 millimeters which was produced by splitting laths and applying a similar sanding process. We suggest to categorize this type of structure in which the geometrical and topological complexity is computationally generated on the basis of natural material constraints and local production logics, as a digital vernacular typology.

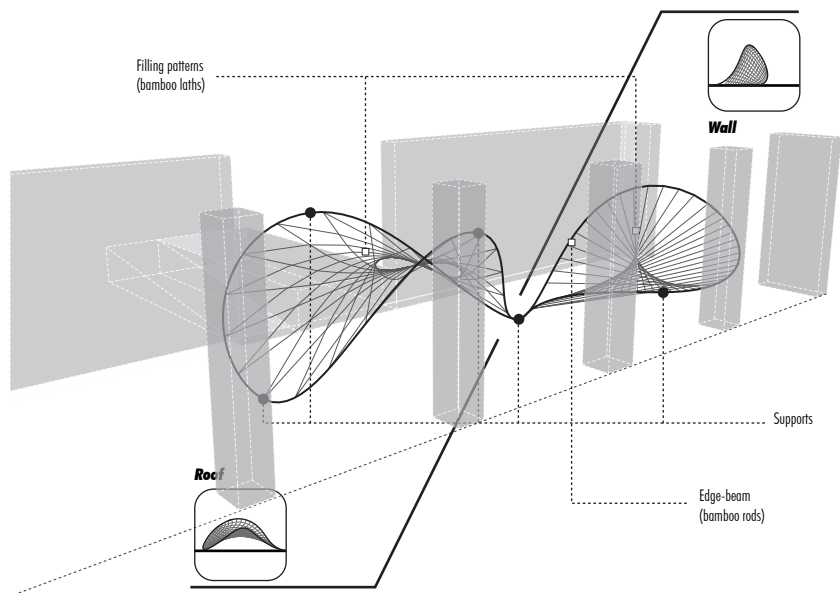


Fig. 6.24

Conceptual scheme of the different parts of the bending-active bamboo structure.

The constraint given by the museum was to develop a structure fitting a medium-sized interior hall of 7.5 by 20 meters without affecting its normal operation. This hall has no specific function and is mostly been used for circulation purposes between

three exhibition areas. The proposed design aimed to channel this circulation by strengthening the entrance to the hall and providing a sitting area for visitors. Thus, the structure was designed from a continuously curved edge-beam made out of bamboo rods that freely extended throughout the entire volume of the hall. The curvature of this element was derived from elastic deformations that were occasioned by the introduction of diverse fixation points (supports) along its entire length. By correctly adjusting the positions of these fixations, deformations were programmed to fulfill with the initial design intentions. The edge-beam was then forced to self-intersect in order to produce a pair of interconnected closed loops. Each loop constituted a different part of the structure and was filled with a unique pattern of actively bent elements. The large part, called the “roof”, was designed as a canopy channeling the main circulations from the galleries, while the small part, defined as the “wall”, integrated a sitting feature (Figure 6.24). The inner-filling pattern for both parts was designed as a non-standard network of interconnected bamboo laths with areas where high curvature was concentrated. The asymmetrical cross-section of the laths enabled a geometric spatial change in torsion for creating structural depth and geometrical stiffness, as well as an easy flat connection to the edge-beam. It also permitted the creation of vertical connections on the center ring of the “roof” so that the weak axis allowed to be bent while following the global curvature of the structure. At the end, the entire structure was 15.5 m long, spanned 7.5 meters across the museum hall and reached a height of 4m at its peak.

6.4.2 TOPOLOGY-DRIVEN APPROACH

The design development of this structure demonstrated a modeling pipeline combining a topology-driven approach for the exploration of the curved edge-beam and a geometric-driven approach for the inner-filling pattern (Figure 6.25). This explorative design study combined both form-finding approaches for providing specialized modeling solutions to different design intentions established for the same structure. In doing so, the output stream of a PS model built with ElasticSpace 2.0 was automatically fed as the input stream of a parametric definition in Grasshopper.

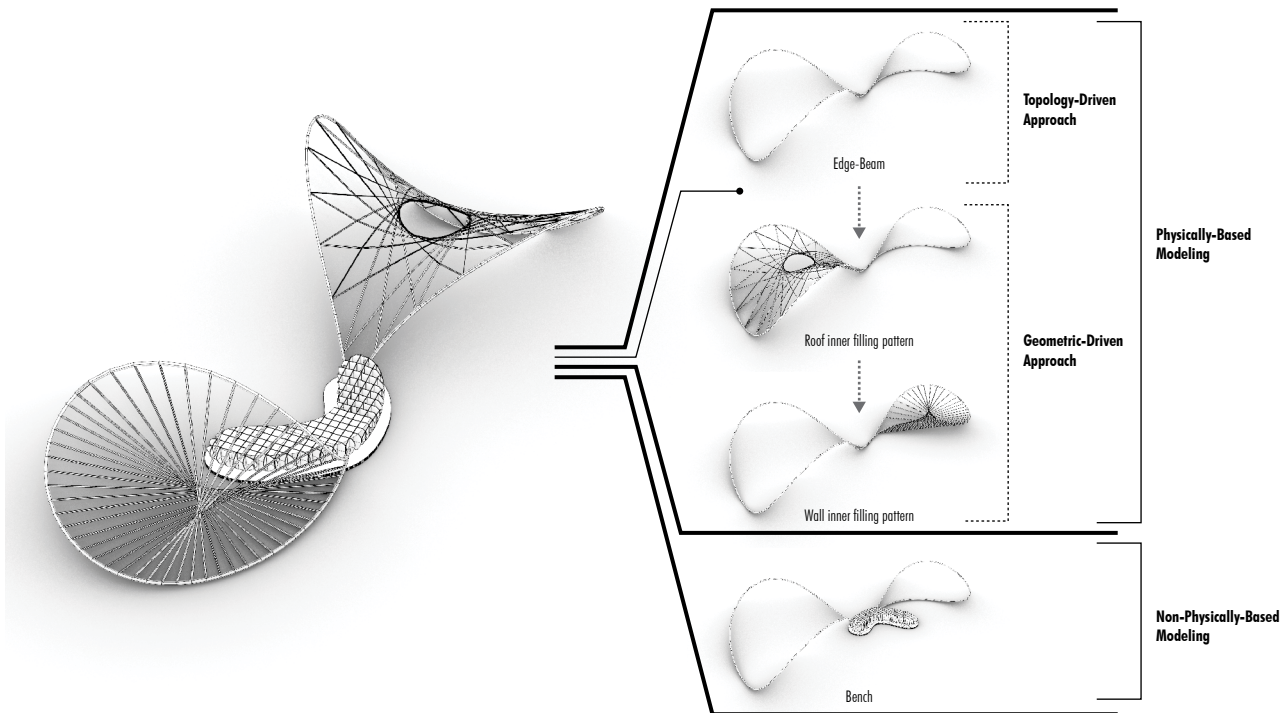


Fig. 6.25
Modeling pipeline.

The material stiffness of each spline-beam element was fictitious and calibrated on the basis of the physical behavior observed through the empirical tests of bamboo materials (Figure 6.26). Through these simple tests, we also concluded that laths were more flexible than rods and that their use within the structure had to be differentiated. In any case, the relative cross-sectional homogeneity of both materials showed that a relatively good approximation of the elastic curve can be reached if we considered that a natural material was used.

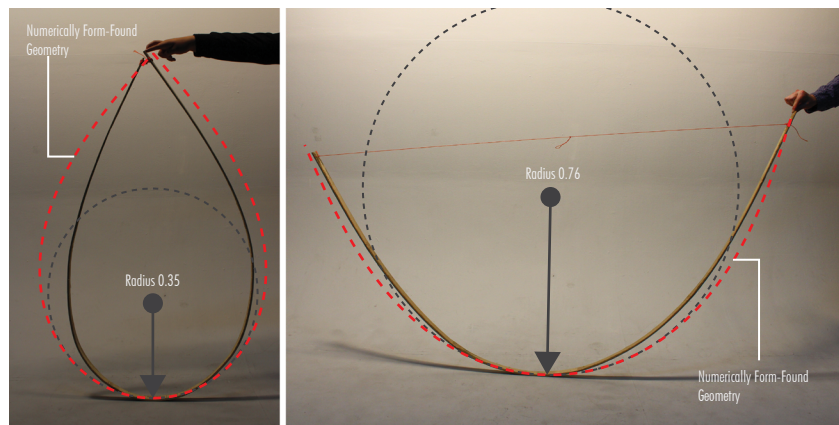


Fig. 6.26
Empirical material tests for calibrating the numerical model.

The topologic and geometric exploration of the edge-beam was entirely conducted with ElasticSpace (Figure 6.27)

on the basis of simplified spline-beam elements with 3DOF per node representing bamboo rods with a span length of 3 meters. Spline-beams were dynamically added, deleted and connected by designers to gradually shape the entire curved edge that, at the end, resulted on a total length of 51 meters. The construction of the PS model in ElasticSpace 2.0 was initialized without any type of topologic properties because those ones gradually evolved throughout time by means of direct designer's interactions. As soon as a mature solution was reached, all the geometric and topological information was sent to Grasshopper using an XML communication protocol via a custom plugin. The plugin then internally reconstructed the PS model to output a non-physically based discrete representation of the edge-beam that served to conduct parametric studies of the inner-filling patterns. Parametric studies corresponded to the pre-processing stage of a geometric-driven form-finding approach where initial topological and geometric conditions need to be defined for building the PS model. The reason for this is because the generation of the inner-filling patterns within the predefined boundaries demanded a stricter control of the modeling process by means of associative logics. Two specific filling patterns were developed for both looped areas with specific syntaxes of connectivity. For the "roof", the filling pattern was generated from curves with end-points lying on the loop and with a path revolved around an attraction point that was positioned inside the loop. The location of this point and the size of the revolved path were parametrically designed according to architectural intents. On the other hand, the pattern of the "wall" had simpler logic with curves connected through shifted sequences. In both cases, the intersections of the curves determined the type of discretization used for the new PS model.

The final PS model was then built with the ElasticSpace plugin for Grasshopper to run the form-finding of the complete structure. That is, the edge-beam was form-found again with the influence of the deformation of filling patterns. It is important to note that at this time some behavioral considerations, like the torsional effects on the laths, were not considered within the numerical model in order to simplify the design process. The complete modeling pipeline proved to be adequate for introducing new practitioners to the realm of simulation-based

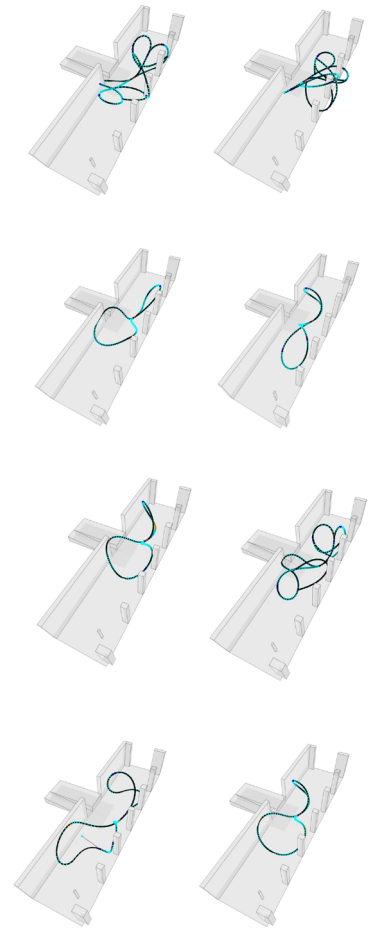


Fig. 6.27

Exploratory design studies of the edge-beam.

design. Moreover, it enabled the full spatial design of a non-standard elastic gridshell structure. Most elastic gridshells are based on orthogonal filling patterns and arch typologies that are supported by two-dimensional bent edges with the exceptions of some openings. One can argue that these gridshells are therefore pseudo three-dimensional (i.e. two-and-a-half-dimensional) mainly because of their erection process. Some built examples like the Asymptotic gridshell (Schling et al. 2018) made a first step into a more spatial approach for making the edge-beams a design parameter.

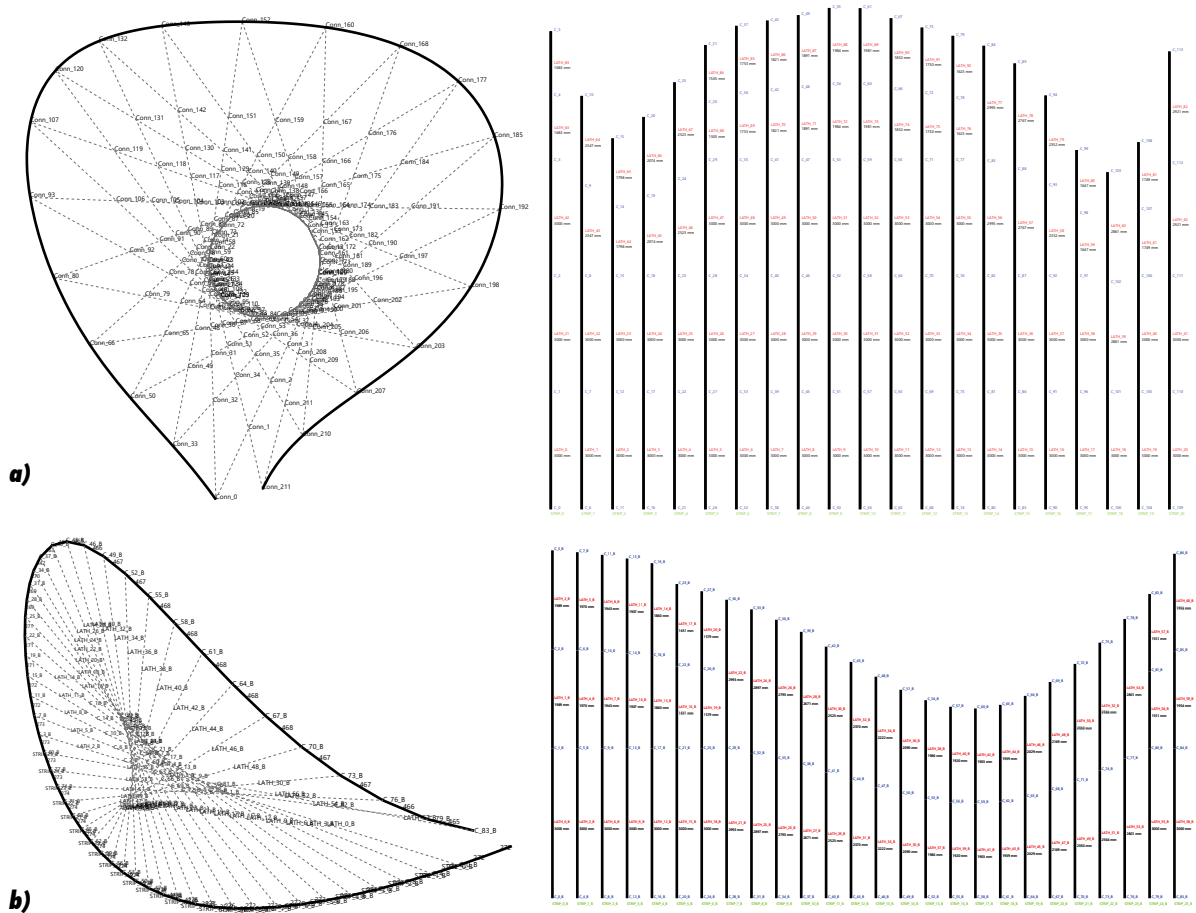


Fig. 6.28

Construction information for the a) “roof” and b) “wall” extracted from the digital model. The diagrams show the labeling of laths and the different connection points.

The structure was finally assembled on-site using information that was entirely extracted from the computational model (Figure 6.28). The edge-beam was shaped from the progressive connection of shorter segments following the connection logic that was stored during the topology-driven form-

finding process. Each segment was made from three bamboo rods that were bundled together to obtain the required stiffness for holding up the structure. Bamboo rods were tied together with a natural jute burlap cord through a high tension knotting system traditionally used in bamboo constructions (Figure 6.29). Hence, the connection had to be firm in tension, compression and rotation. Every joint was unique and had different forces to transfer. At the same time, the construction process of the filling patterns started through the assemblage of longer elements that were shaped from the sequential connection of several laths. Laths were attached with the same burlap cord in a way that the cord prevented lateral movements and rotations at all times. The ends of these longer lath elements were pre-drilled with a hole to facilitate the connection with the edge-beam. At this point, they were inserted one by one into the structure by following a woven logic. Laths were positioned and fixed on the markers of the edge-beam, and then guided in the desired bending and torsional direction. After the exhibition period of six months, the structure was disassembled, and all the components were recycled for further use.



Fig. 6.29

Connection between laths and rods using yute.

6.5 SELF-CHOREOGRAPHIC NETWORKS⁸



Fig. 6.30

Demonstrator.

6.5.1 DESCRIPTION

This case study was developed in collaboration with students Mathias Maierhofer and Valentina Soana as part of their master thesis at the University of Stuttgart. The prototype, built

⁸ Based on pre-published article (Suzuki and Knippers 2018): “Digital Vernacular Design: Form-Finding at the Edge of Realities”.

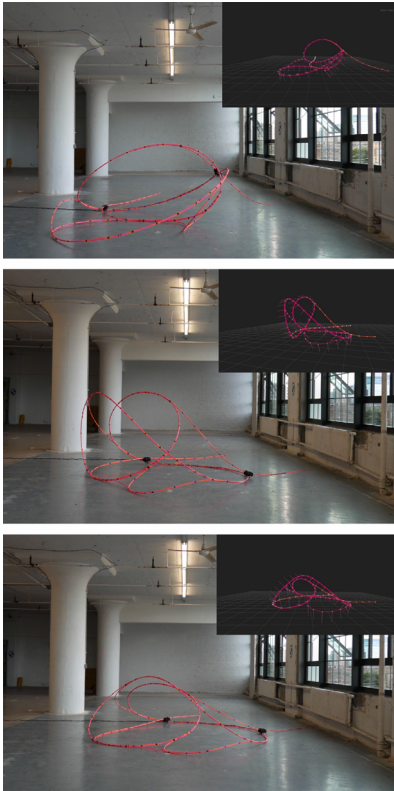


Fig. 6.31
Different equilibrium states reached by the system.

by the students during their residency at the Autodesk Technology Center in Boston, served to demonstrate how the combination of linear elastic elements, robotic actuators and numerical methods can be strategically used to shape an adaptive structure developing different equilibrium states in response to external stimulus (Figure 6.30). A novel feedback communication between PS models and physical data was then developed to model in real-time the current equilibrium state of the structure and predict untested states under quite realistic conditions for assessing further transformations. The structure was shaped as a networked arrangement of actively bent GFRP rods connected via robotic joints that, when activated, produced local deformations propagating globally until reaching a new equilibrium state (Figure 6.31). Each robotic joint was designed to connect two rods together, collect data and activate a given kinematic motion, either a sliding or a rotational motion. The physical data outputted by the joint, at highly interactive rates, was used to directly inform the construction of a digital twin representing the current state of the structure.

6.5.2 TOPOLOGY-DRIVEN APPROACH

The digital twin was numerically represented by a PS model supporting dynamic topological transformations on the fly. In this case, the topology-driven form-finding approach was utilized to continuously modify some pre-existent topologic properties of the PS model. To do this, a custom simulation environment was developed with Processing (Fry and Reas 2014), making use of the Iguana API built with Java 8. Several network configurations had been previously tested with scaled analogue models to determine the initial configuration of the PS model. The aim was to maximize the design space of the dynamic structure (i.e. a maximum number of equilibrium states) by making use of a minimum number of rods and connections. On this basis, the PS model was initialized from a predefined set of geometric and topologic conditions. Simplified spline-beam elements with 3DOF per node were used to model the physical behavior of GFRP rods and facilitate the development of more interactive applications. In contrast, the dynamic behavior of the entire structure caused by the robotic joints was modeled through custom implementations of force elements and topologic modifiers.

As described in the previous chapter, Iguana permits to implement custom classes that can be directly derived from the IModifier class or the IElement class. While every class that has IModifier as a superclass implements rule-based processes for modifying topological properties, classes derived from IElement implement custom methods for computing forces or energies on a given set of particles. The rotational joint was implemented with IElement as a superclass and was only applied on particles that represented this type of robotic joint on the physical structure. Given the predefined topology, those particles had always a shared state with four adjacent particles. Forces contributed by this type of element were only calculated when receiving a signal of the corresponding robotic joint, or when called explicitly by the designer using the simulation environment. Once triggered, the calculation of those forces required to first take the particle associated with the joint and then query for all the particles that were topologically connected to it. Those topologic queries were conducted on the basis of the AHF data structure storing the topologic properties of the PS model. With this information, each element defined a rotational axis aligned with the normal of the shared particle and then asked for a rotational angle to be given by the robotic joint or by the designer. As shown in figure 6.32, two projections were created by rotating the corresponding positions of the adjacent particles clockwise with only half of the angle, and the other two by rotating the remaining adjacent particles counterclockwise. The forces applied on the adjacent particles were finally defined by the corresponding vector created between the current position and the projected position scaled with a fictitious stiffness. This rotational joint permitted to control the internal angles between two crossed spline-beams.

On the other hand, the sliding joint was implemented with IModifier as a superclass which necessarily implied a special treatment of topologic properties. In the physical realm, sliding a rod against another does not represent an explicit transformation on the connectivity of the structure. However, in the discretized digital realm, simulating this sliding behavior implied moving the connection, between two crossed spline-beams, from one known discrete position to another one of the same spline-beam. The problem is that this transition is far from being smooth. This

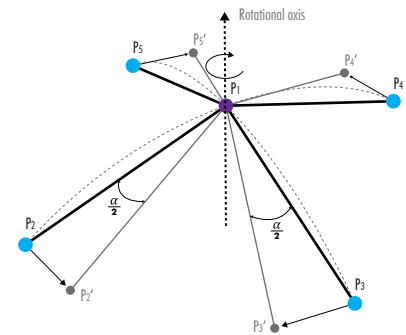


Fig. 6.32

Forces associated with a rotational joint.

lack of smoothness creates instabilities leading to deformations not matching with the current state of the physical structure. A solution to this problem was found by constantly updating the topologic properties of a spline-beam in order to smoothly move the location of the connection along its discrete representation (Figure 6.33). In doing so, a sliding joint can only be applied if the associated particle had a shared or volatile state with at least two spline-beam elements referenced to it. This logical statement constituted the conditional part of the rule which determine the application of further generative rules to transform the topologic properties of a spline-beam elements.

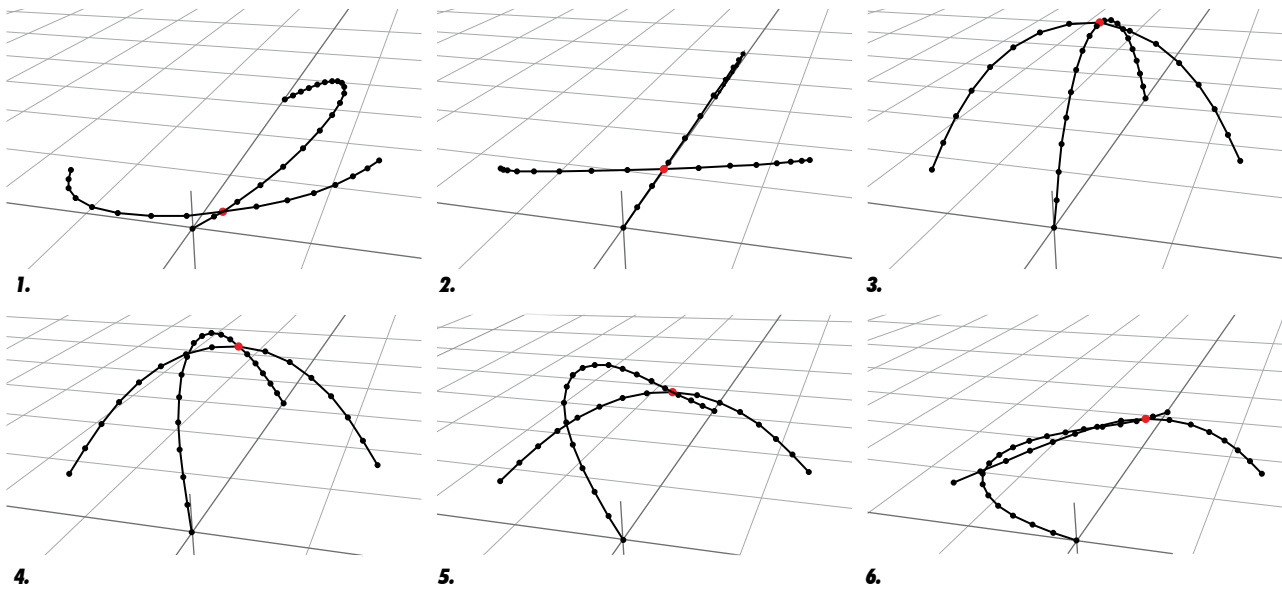


Fig. 6.33
Sliding joint.

Each sliding modifier was associated with a shared particle and with only one of the spline-beams referenced by the particle. The syntax of this rule for modifying the geometric graph of the spline-beam is explained in figure 6.45. The conditional part of the graph rule is presented in the two quadrants Q1 and Q2 with a graph that is embedded in a two-dimensional space. In case such conditional part is true, the generative part of the rule is executed and applies the topologic modifications as described in quadrants Q3 and Q4. Based on the parameterization of the entire spline-beam, quadrant Q3 describes the dynamic creation of a particle with a volatile state that conducts a smooth transition from two consecutive and pre-existent particles. Graphs in Q1 and Q3 are

different in the number of nodes and edges. By contrast, a volatile particle that closely approaches one of the pre-existent particles is automatically deleted, and all its topologic information is transferred to corresponding pre-existent particle. In such cases, graphs in Q2 and Q4 are also different in the number of nodes and edges. All these topologic transformations require the modifier to update the parameterization of the spline beam and recompute the corresponding geometric properties.

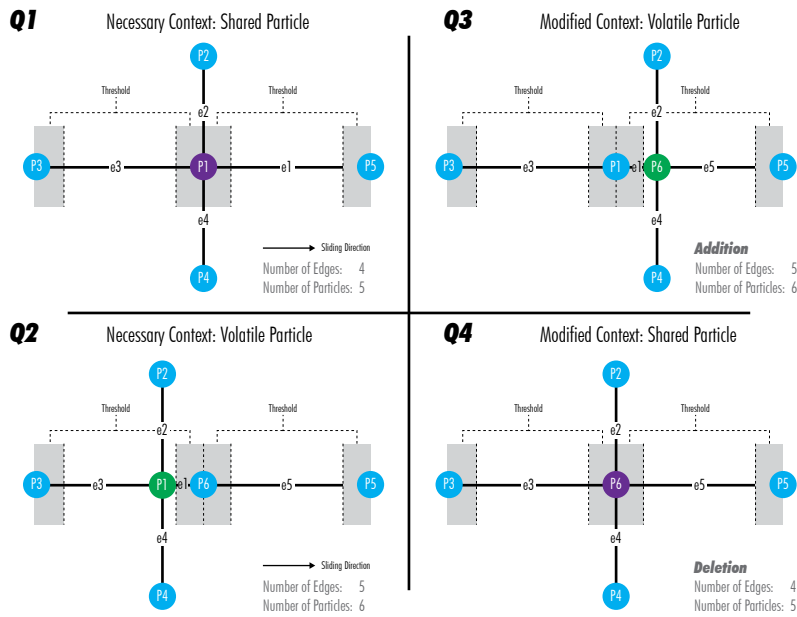


Fig. 6.34

Topologic modifications of the sliding joint represented on the geometric graph.

The instantiation of both class objects in the PS model and their selective activation using sensors data from robotic joints, facilitated the creation of novel ways of interactions among users, designers and structure. The PS model creates an interactive interface of the physical structure to continuously know its current state. Real-time data outputted from it was in turn used to make informed decisions for further transformations. As a result, this hybrid PS model based on evolving topologic models was developed as an alternative for traditional offline procedures in the design of bending-active structures. The final open-ended and reconfigurable prototype created a kind of cyber-physical system in which the gap between digital and analogue realms was bridged for the continuous development of more intricate design solutions.

CONCLUSIONS AND FUTURE DIRECTIONS

The exploration of design spaces in numerical form-finding is only limited by designer's intuition and creativity. This thesis was built under the main idea that combining real-time physics with geometric and topologic modifications on the fly results on a powerful and fully flexible modeling tool for facilitating the design exploration of bending-active and textile hybrid structures during conceptual stages. To our knowledge, this approach has not been entirely treated before in form-finding and structural analysis. Therefore, the main goal was to develop a general insight on the activation of topologic modeling with real-time physics during conceptual stages in order to extend the design space in the form-finding process of bending-active and textile hybrid structures. The principal aspects covered for the development of this work focused on the close relationships among topologic, geometric and numerical conditions build on particle systems for enabling more intuitive design practices promoting new ways for user-model interaction. This should be considered in the context of a novel computational methodology for form-finding that rises new design problematics for shape exploration and computation. This condition does not only imply the development of more intuitive modeling workflows, but most important a deeper understand of rules and conditions that could permit reliable topological transformations on numerical models.

From a broader perspective, the activation of topologic modeling in numerical form-finding can be perceived as the closest way to approximate the playfulness characteristics of analogue methods where material organization and behavior cannot be separated. This means that the inseparable relationship found in scaled physical models regarding topology, geometry

and physics can be reassessed within a digital design context in a way that the gap between analogue and digital form-finding practices can be bridged.

In chapter 2, we presented an overview of bending-active and textile hybrid structures with emphasize to specific research projects exploring the different use of numerical form-finding techniques. The research problematic that this dissertation builds on is derived from a close analysis of the exploratory design studies conducted in these projects and the subsequently identification of open questions. Taking the case of the Umbrella Marrakech, its relative topologic and geometric simplicity constituted the best scenario to conduct a design process entirely based on FEM. Therefore, efforts were devoted for developing an accurate mechanical description of the structure while conveniently restraining shape exploration. On the other hand, the material Equilibria was among the first projects to demonstrate the potential use of PS-models as an interactive design medium for textile hybrid structures. In doing so, it has been demonstrated that these fast numerical simulations can be used to rapidly explore multiple equilibrium shapes emerging from fixed topologic configurations. In both cases, it was implicitly assumed that designers need to provide, either intuitively or through the use of scaled physical models, the required topological information to construct the numerical model.

The full potential of bending-active structures to break formal and structural typologies was demonstrated in projects where the topological problem started to be more intricate like in the case of the ICD/ITKE Research Pavilion 2010, the Textile Hybrid M1 and the Tower. Exploring the topologic properties of these projects in association with their complex geometries required the development of specialized multi-scalar design processes to hierarchically address all design problems through different modeling layers. Rigorous studies on scaled physical models were used to define basic principles of modularity and systemic logics to further explore design potentials via computational models. Consequently, PS models have rapidly become important on the research of bending-active and textile hybrid structures because, to date, they constitute the most

interactive medium for the computational exploration of design concepts requiring the integration of physical behaviors.

The basic principles of Particle System (PS) and Dynamic Relaxation (DR) methods were introduced in chapter 3. Due to close similarities, we have proposed to consider PS as a generalized form of DR supporting more types of time-integration schemes and dynamics to produce the motion of particles. It has been noted that the current use of PS models offers specific benefits for exploratory design stages, but it was also argued that those models exhibit some topologic limitations that cannot be easily overcome. The integration of PS models permitted to tackle the problem of predicting, at highly interactive rates, complex physical behaviors while allowing designers to finely tune the geometric variables of the simulation. Nevertheless, the definition of topologic conditions still needs to be treated outside the numerical process which is in part occasioned by an oversimplification of the topologic model used in the formulation and implementation of PS models. Our assumption was that this problem is entirely derived from relying only on speeding up numerical solutions and not considering specific modeling problems appearing along the design process. In fact, the overview of research projects presented in chapter 2 have shown that many design issues associated with topologic conditions need to be addressed in an analogue form by extensively using scaled physical models. It is therefore in this context that the activation of topology modeling within real time numerical simulations can start to play a significant role for the exploration of design spaces.

To build a theoretical basis, in Chapter 4, we have identified and compared the two main families of generative approaches build on PS models for form-finding bending-active and textile hybrid structures. These approaches have been categorized as geometric-driven and topology-driven form-finding. Geometric-driven approaches have made complex numerical form-finding procedures more accessible to a wider range of user designers. They permit to easily variegate equilibrium shapes by allowing designers to alter geometric parameters while keeping topologic conditions constant over time. The main advantage of this approach is that the initial geometric and topologic models

required for starting the simulation can be easily parameterized and evaluated given fast convergences rates. This design mode is however problematic when highly intricate material relationships are involved. Therefore, geometric-driven approaches are best suited when topologies are to some extent regular and fixed.

Instead, topology-driven approaches are more suitable for exploring the complexity of form and material articulation when designing complex assemblies of bending- and tensile form-active components. The main advantage of these approaches is that simulation can start without any geometric or topologic precondition since these ones can be progressively built during the entire simulation process. On this basis, we presented the formulation of a comprehensive topologic model with geometric and mechanic embeddings for leading the implementation of efficient PS models with complete modeling capabilities (i.e. geometric and topologic) during numerical integration. This topologic model is intended to be reusable to facilitate the development of more intuitive exploratory design processes, extendible for including further problems involving the use of dynamic topologic models with real-time physics, heterogenous in a way that multiple components and behaviors can be included within a single computational model and controllable by expert and non-expert users. Even if the scope of this dissertation studies was only established for specific combinations of bending-active and textile hybrid structures, we believe that the proposed topologic model can certainly embrace modeling more complex deformable objects by adjusting, implementing and/or identifying additional building blocks.

The computational implementation of our topologic model was presented in chapter 5 using an OOP approach. The modularity and level of abstraction of this implementation permits to reuse the code and extend its functionalities. This OOP implementation was primarily used for the development of two types of applications named ElasticSpace and Iguana. ElasticSpace was described as explorative form-finding software for the design of bending-active and textile hybrid structures through the use of dynamic topologic models. Iguana was in turn presented as a robust computational library that can be

implemented via a simple API for the development of custom numerical models supporting dynamic topologic transformations. At this point, we have identified that additional problematics with respect to the rate of interactivity of the simulation and the taking of modelling decisions in complex design spaces can also be tackled from the use of dynamic topologies. In fact, it has been shown that the potential integration of autonomous systems for conducting topology-driven form-finding processes opens a new set of research questions. In this spirit, the wide range of machine learning techniques may offer promising mechanisms for controlling and optimising the addition of new elements within the numerical model.

Based on these developments, chapter 6 introduced a set of case studies showcasing the use of topology-driven approaches for the design development of bending-active and textile hybrid structures. These case studies have shown that the introduction of active-topology modeling during form-finding enables to extend the design space freedom by creating new ways of interactions among designers, digital models and physical structures. From an educational perspective, it has also proven to facilitate the introduction of non-expert-user to the realm of simulation-based design. As a design practice, the activation of topologic modeling during numerical form-finding conceptually implies an important step for bridging the gap between analogue and digital realms.

Addressed from the categorization of bending-active structures proposed by Lienhard et al. (Lienhard et al. 2013c), the activation of topologic modeling would open a door for merging the intuitive and interactive capacities of physically bend, stretch, cut, connect and disconnect elements inherent in the behavior- and geometrical-based approach within the extended benefits of managing geometrical and topological complexity of numerical models inherent in the integral approach. Consequently, our assumption is that such extended modeling flexibilities can give rise to new types of digital vernacular design approaches such as the one developed for the Computational Bamboo demonstrator in which innovation on the use of local natural materials is stimulated by strengthening the creative and intuitive characteristics of the digital design process. Additionally, topology-driven approaches

are also well suited to break standard structural typologies that are based on single equilibrium states. This is because they permit to explore innovative solutions for adaptation and reconfiguration rising from dynamic geometric and topologic conditions such as the open-ended adaptive concept developed in the project of Self-Choreographic Networks.

To finish, further efforts are still required to overcome the different aspects that were briefly covered throughout this work. For this reason, this thesis was conceived as a departing point for future research yet to come on the study of the activation of topologic modeling within real-time physically-based models for exploratory form-finding applications and their potential design applications within architecture and engineering.

Appendix A

A.1 INITIAL BEAM ORIENTATION

In the model proposed by Li (Li 2017), a beam element is always connected to two beam-ends and . For defining the initial orientation of a beam element, the local x-direction is calculated through the positions of both beam-ends by

$$(A.1) \quad \hat{u}_x^0 = \frac{(\vec{x}_b - \vec{x}_a)}{|\vec{x}_b - \vec{x}_a|}$$

The simplest way to define the rest of directions is by using the reference of a global direction. Hence, taking the global z-direction as a reference, the local y-direction of the initial beam orientation is

$$(A.2) \quad \hat{u}_y^0 = \hat{Z} \times \hat{u}_x^0$$

If the cross product of these two reference vectors is the zero vector, then a different global reference axis needs to be taken for this calculation. This is occasioned by having the initial local x-direction of the beam orientation align with the selected global direction. At this point, the local z-direction is calculated by the cross product of the two known directions. Thus,

$$(A.3) \quad \hat{u}_z^0 = \hat{u}_x^0 \times \hat{u}_y^0$$

A.2 INITIAL BEAM-END ORIENTATIONS

For a sequential series of beams with 6 degrees of freedom per node, the type of pre-stress condition is defined by setting specific initial orientations to their beam-ends. Note that the

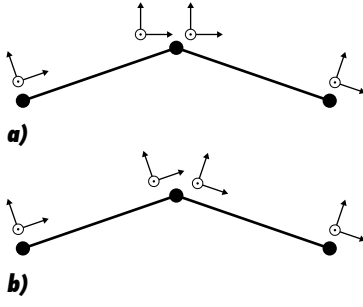


Fig. A.1

Beam-ends orientations, a) Pre-stress of an initially straight joint, b) No pre-stress.

coupling of adjacent beam-ends between a pair of consecutive beams define a joint. The pre-stress of an initially straight joint is calculated by assigning equal orientations to both beam-ends as shown in Figure A.1a (Li 2017). To avoid an excessive pre-stress condition on the joint, this shared beam-end orientation needs to be calculated from the average of the two beam orientations. Thus, the initial local directions of this beam-end orientation are calculated by

$$\begin{aligned}
 \hat{e}_z^0 &= \frac{(\hat{u}_{z,l}^0 + \hat{u}_{z,r}^0)}{|\hat{u}_{z,l}^0 + \hat{u}_{z,r}^0|} \\
 \hat{e}_y^0 &= \frac{\hat{e}_z^0 \times (\hat{u}_{x,l}^0 + \hat{u}_{x,r}^0)}{|\hat{e}_z^0 \times (\hat{u}_{x,l}^0 + \hat{u}_{x,r}^0)|} \\
 \hat{e}_x^0 &= \hat{e}_y^0 \times \hat{e}_z^0
 \end{aligned}
 \tag{A.4}$$

where \hat{e}_z^0 is the initial local z -direction of the beam element orientation on the left side of the joint and $\hat{u}_{x,r}^0$ is the initial local x -direction of the beam element orientation on the right side of the joint. In case that no pre-stress is required, beam-end orientations at the joint are different and need to be equivalent to their corresponding beam orientation (Figure A.1b).

Bibliography

- Adriaenssens, Sigrid (2008): Feasibility Study of Medium Span Spliced Spline Stressed Membranes. In *International Journal of Space Structures* 23 (4), pp. 243–251.
- Adriaenssens, Sigrid; Barnes, Michael (2001): Tensegrity spline beam and grid shell structures. In *Engineering Structures* 23 (1), pp. 29–36.
- Adriaenssens, Sigrid; Barnes, Mike; Harris, Richard; Williams, Chris (2014): Dynamic Relaxation. Design of a strained timber gridshell. In Sigrid Adriaenssens, Philippe Block, Diederik Veenendaal, Chris Williams (Eds.): *Shell Structures for Architecture. Form finding and optimization*. London, New York: Routledge, pp. 88–101.
- Ahlquist, Sean; Askarinejad, Ali; Chaaraoui, Rizkallah; Kalo, Ammar; Liu, Xiang; Shah, Kavan (2014): Post-forming composite morphologies. Materialization and design methods for inducing form through textile material behavior. In *Proceedings of the ACADIA 2014 International Conference: Design Agency*. Los Angeles, USA.
- Ahlquist, Sean; Erb, Dillon; Menges, Achim (2015): Evolutionary structural and spatial adaptation of topologically differentiated tensile systems in architectural design. In *AIEDAM* 29 (4), pp. 393–415. DOI: 10.1017/S0890060415000402.
- Ahlquist, Sean; Fleischmann, Moritz (2008): Elemental Methods for Integrated Architectures: Experimentation with Design Processes for Cable Net Structures. In *International Journal of Architectural Computing* 6 (4), pp. 453–475. DOI: 10.1260/147807708787523259.
- Ahlquist, Sean; Lienhard, Julian; Knippers, Jan; Menges, Achim (2013): Exploring Materials Reciprocities for Textile-Hybrid Systems as Spatial Structures. In *Prototyping Architecture: The Conference Paper*. London, UK., pp. 187–210.
- Ahlquist, Sean; Menges, Achim (2011): Behavior-based Computational Design Methodologies. Integrative Processes for Force Defined Material

- Structures. In *Integration through Computation*, Proceedings of the 31th Conference of the Association For Computer Aided Design In Architecture (ACADIA). Canada., pp. 82–89.
- Ahlquist, Sean; Menges, Achim (2013): Frameworks for computational design of textile micro-architectures and material behavior in forming complex force-active structures. In *Acadia 2013: Adaptive Architecture*. Toronto, Canada.
- Alayrangues, Sylvie; Lienhardt, Pascal; Peltier, Samuel (2015): Conversion between chains of maps and chains of surfaces. Application to the computation of incidence graphs homology. [Research Report].
- Aldinger, Lotte; Margariti, Georgia; Körner, Axel; Suzuki, Seiichi; Knippers, Jan (2018): Tailoring Self-Formation. Fabrication and simulation of membrane-actuated stiffness gradient composites. In *Proceedings of the IASS Symposium 2018: Creativity in Structural Design*. MIT Boston, USA.
- Alumbaugh, Tyler J.; Jiao, Xiangmin (2005): Compact Array-Based Mesh Data Structures. In *Proceedings of the 14th International Meshing Roundtable 31*, pp. 485–503. DOI: 10.1007/3-540-29090-7_29.
- Archer, G. C.; Fenves, G.; Thewalt, C. (1999): A new object-oriented finite element analysis program architecture. In *Computers & Structures* 70 (1), pp. 63–75. DOI: 10.1016/S0045-7949(98)00194-1.
- Baraff, David; Witkin, Andrew (1998): Large steps in cloth simulation. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*. New York, US., pp. 43–54. DOI: 10.1145/280814.280821.
- Barnes, Michael (1999): Form Finding and Analysis of Tension Structures by Dynamic Relaxation. In *Internacional Journal of Space Structures* 14 (2).
- Barnes, Michael R. (1977): Form finding and analysis of tension structures by dynamic relaxation. Doctoral Thesis, London. City University of London.
- Barnes, Michael R. (1988): Form-finding and analysis of prestressed nets and membranes. In *Computer & Structures* 30, pp. 685–695.
- Barnes, Michael R.; Adriaenssens, Sigrid; Krupka, Meghan (2013): A novel torsion/bending element for dynamic relaxation modeling. In *Computers & Structures* 119, pp. 60–67. DOI: 10.1016/j.compstruc.2012.12.027.
- Barzel, Ronen (1992): *Physically-based modeling for computer graphics. A structured Approach*: Academic Press.

- Ben Salah, Fatma; Belhaouari, Hakim; Arnould, Agnès; Meseure, Philippe (2017): A General Physical-topological Framework using Rule-based Language for Physical Simulation. In International Conference on Computer Graphics Theory and Applications. Porto, Portugal., pp. 220–227. DOI: 10.5220/0006119802200227.
- Bender, Jan; Müller, Matthias; Otaduy, Miguel A.; Teschner, Matthias (2012): Position-based Methods for the Simulation of Solid Objects in Computer Graphics. DOI: 10.2312/CONF/EG2013/STARS/001-022.
- Bender, Jan; Müller, Matthias; Otaduy, Miguel A.; Teschner, Matthias; Macklin, Miles (2014): A Survey on Position-Based Simulation Methods in Computer Graphics. In Computer Graphics Forum 33 (6), pp. 228–251. DOI: 10.1111/cgf.12346.
- Blacker, Ted D.; Stephenson, Michael B. (1991): Paving: A new approach to automated quadrilateral mesh generation. In Int. J. Numer. Meth. Engng. 32 (4), pp. 811–847. DOI: 10.1002/nme.1620320410.
- Bouaziz, Sofien; Deuss, Mario; Schwartzburg, Yuliy; Weise, Thibaut; Pauly, Mark (2012): Shape-Up: Shaping Discrete Geometry with Projections. In Computer Graphics Forum 31 (5), pp. 1657–1667. DOI: 10.1111/j.1467-8659.2012.03171.x.
- Bouaziz, Sofien; Martin, Sebastian; Liu, Tiantian; Kavan, Ladislav; Pauly, Mark (2014): Projective dynamics. Fusing constraint projections for fast simulation. In ACM Trans. Graph. 33 (4), pp. 1–11. DOI: 10.1145/2601097.2601116.
- Brandt-Olsen, Cecilie (2016): K2Engineering. Available online at <https://github.com/CecilieBrandt/K2Engineering>, checked on 3/7/2019.
- Breen, David E.; House, Donald H.; Wozny, Michael J. (1994): A Particle-Based Model for Simulating the Draping Behavior of Woven Cloth. In Textile Research Journal 64 (11), pp. 663–685. DOI: 10.1177/004051759406401106.
- Buckley, C. E.; Nugent, E.; Ryan, D.; Neary, P. C. (2012): Virtual Reality – A New Era in Surgical Training. In Virtual Reality in Psychological, Medical and Pedagogical Applications. DOI: 10.5772/46415.
- CASS, R. J.; BENZLEY, S. E.; MEYERS, R. J.; BLACKER, T. D. (1996): Generalized 3D Paving. An Automated Quadrilateral Surface Mesh Generation Algorithm. In Int. J. Numer. Meth. Engng. 39 (9), pp. 1475–1489. DOI: 10.1002/(SICI)1097-0207(19960515)39:9<1475::AID-NME913>3.0.CO;2-W.

- Chak, Dan; Galbraith, Megan; Killian, Axel (2002): CatenaryCAD. An Architectural Design Tool. Final Project Report, Boston, Massachusetts. MIT.
- Chentanez, Nuttapong; Alterovitz, Ron; Ritchie, Daniel; Cho, Lita; Hauser, Kris K.; Goldberg, Ken et al. (2009): Interactive simulation of surgical needle insertion and steering. In *ACM Trans. Graph.* 28 (3), p. 1. DOI: 10.1145/1531326.1531394.
- Chilton, John; Isler, Heinz (2000): Heinz Isler. London: T. Telford (The engineer's contribution to contemporary architecture).
- Chu, Dominique (2011): Complexity. against systems. In *Theory in biosciences = Theorie in den Biowissenschaften* 130 (3), pp. 229–245. DOI: 10.1007/s12064-011-0121-4.
- Cribb, Roger (1991): *Nomads in archaeology*. Cambridge: Cambridge University Press.
- Crolla, Kristof (2017): Building indeterminacy modelling – the ‘ZCB Bamboo Pavilion’ as a case study on nonstandard construction from natural materials. In *Vis. in Eng.* 5 (1), p. 178. DOI: 10.1186/s40327-017-0051-4.
- Cundall, Peter (1976): Explicit Finite Difference Methods in Geomechanics. In *Numerical Methods in Engineering: Proceedings of the EF Conference on Numerical Methods in Geomechanics*, Blacksburg, Virginia. 1, pp. 132–150.
- D’Amico, B.; Zhang, H.; Kermani, A. (2016): A finite-difference formulation of elastic rod for the design of actively bent structures. In *Engineering Structures* 117, pp. 518–527. DOI: 10.1016/j.engstruct.2016.03.034.
- Damiand, Guillaume; Lienhardt, Pascal (2015): *Combinatorial maps. Efficient data structures for computer graphics and image processing*. Boca Raton, FL: CRC Press/Taylor & Francis Group. Available online at <http://proquest.tech.safaribooksonline.de/9781482206524>.
- Day, Alistair (1965): An introduction to Dynamic Relaxation. In *The Engineer* 2019, pp. 218–221.
- Day, Alistair S.; Bunce, J. (1969): The analysis of hanging roofs. In *The Arup Journal* 4 (3), pp. 31–32.
- DeLanda, Manuel (2002): *Philosophies of Design. The Case of Modeling Software*. In *Verb: Architecture Bookazine* 1 (1).
- Deleuran, Anders Holden; Pauly, Mark; Tamke, Martin; Tinning, Ida Friis; Thomsen, Mette Ramsgaard (2016): *Exploratory Topology Modelling of*

- Form-active Hybrid Structures. In *Procedia Engineering* 155, pp. 71–80. DOI: 10.1016/j.proeng.2016.08.008.
- Delingette, H.; Cotin, S.; Ayache, N. (1999): A hybrid elastic model allowing real-time cutting, deformations and force-feedback for surgery training and simulation. In *Proceedings Computer Animation*. Geneva, Switzerland., pp. 70–81. DOI: 10.1109/CA.1999.781200.
- Deuss, Mario; Deleuran, Anders Holden; Bouaziz, Sofien; Deng, Bailin; Piker, Daniel; Pauly, Mark (2015): ShapeOp. A Robust and Extensible Geometric Modelling Paradigm. In *Modelling Behaviour*. London, UK. 31, pp. 505–515. DOI: 10.1007/978-3-319-24208-8_42.
- Douthe, C.; Baverel, O. (2009): Design of nexorades or reciprocal frame systems with the dynamic relaxation method. In *Computers & Structures* 87 (21-22), pp. 1296–1307. DOI: 10.1016/j.compstruc.2009.06.011.
- Du Peloux, Lionel (2017): Modeling of bending-torsion couplings in active-bending structures. Application to the design of elastic gridshells. Milton Keynes, UK: Lightning Source UK Ltd.
- Dvorak, Martin (2008): JBullet. Available online at <http://jbullet.advel.cz/>, checked on 7/3/2019.
- Dyedov, Vladimir; Ray, Navamita; Einstein, Daniel; Jiao, Xiangmin; Tautges, Timothy J. (2014): AHF: Array-Based Half-Facet Data Structure for Mixed-Dimensional and Non-manifold Meshes. In *Proceedings of the 22nd International Meshing Roundtable* 40, pp. 445–464. DOI: 10.1007/978-3-319-02335-9_25.
- Eberhardt, B.; Weber, A.; Strasser, W. (1996): A fast, flexible, particle-system model for cloth draping. In *IEEE Comput. Grap. Appl.* 16 (5), pp. 52–59. DOI: 10.1109/38.536275.
- Edmonds, John R. (1960): A combinatorial representation for polyhedral surfaces. In *Notices of the American Mathematical Society* 7.
- Engel, Heino (2013): *Tragsysteme. Structure systems*. 5. ed. Ostfildern: Hatje Cantz.
- Felbrich, Benjamin (2017): FlexCLI. Available online at <https://github.com/HeinzBenjamin/FlexCLI>.
- Fenves, Gregory L. (1990): Object-oriented programming for engineering software development. In *Engineering with Computers* 6 (1), pp. 1–15. DOI: 10.1007/BF01200200.
- Fenves, Steven J.; Branin, Franklin H. (1963): *Network-Topological*

- Formulation of Structural Analysis. In *Journal of the Structural Division* 89 (4), pp. 483–514.
- Flechon, Elsa; Zara, Florence; Damiand, Guillaume; Jaillet, Fabrice (2013): A generic topological framework for physical simulation. In *21st International Conference on Computer Graphics, Visualization and Computer Vision*. Plzen, Czech Republic., pp. 104–113.
- Fleischmann, Moritz; Ahlquist, Sean (2009): Cylindrical Mesh Morphologies. Studies of computational meshes and form based on parameters of force, material, and space for the design of tension-active structures. In *Computation: The new realm of architectural Design eCAADe Proceedings*. Istanbul, Turkey., 39–46.
- Fleischmann, Moritz; Knippers, Jan; Lienhard, Julian; Menges, Achim; Schleicher, Simon (2012): Material Behaviour: Embedding Physical Properties in Computational Design Processes. In *Archit Design* 82 (2), pp. 44–51. DOI: 10.1002/ad.1378.
- Fleischmann, Moritz; Lienhard, Julian; Menges, Achim (2011): Computational Design Synthesis. Embedding Material Behaviour in Generative Computational Processes. In *Respecting Fragile Places: 29th eCAADe Conference Proceedings*. Ljubljana, Slovenia.
- Forde, Bruce W.R.; Foschi, Ricardo O.; Stiemer, Siegfried F. (1990): Object-oriented finite element analysis. In *Computers & Structures* 34 (3), pp. 355–374. DOI: 10.1016/0045-7949(90)90261-Y.
- Fry, Ben; Reas, Casey (2014): *Processing*. Version 2.2.1: Processing Foundation.
- Ganovelli, Fabio; Cignoni, Paolo; Montani, Claudio; Scopigno, Roberto (2000): A Multiresolution Model for Soft Objects Supporting Interactive Cuts and Lacerations. In *Computer Graphics Forum* 19 (3), pp. 271–281. DOI: 10.1111/1467-8659.00419.
- Gaß, Siegfried (1990): *Experimente*. [physikalische Analogmodelle im architektonischen Entwerfen]. Stuttgart [Germany]: Institut für leichte Flächentragwerke (Form, Kraft, Masse Form, Force, mass, 5).
- Gaß, Siegfried (2016): Physical analog models in architectural design. In *International Journal of Space Structures* 31 (1), pp. 16–24. DOI: 10.1177/0266351116642061.
- Gass, Siegfried; Drüsedau, Heide; Hennicke, Jürgen; Dunkelberg, Klaus (1985): *IL 31 Bambus. Bamboo*. 2. unveränderte Aufl. Stuttgart: Krämer (Mitteilungen des Instituts für Leichte Flächentragwerke, IL, Universität

- Stuttgart, 31).
- Gengnagel, Christoph (2013): Active Bending in Hybrid Structures. In Form-Rule | Rule-Form: Conference Proceedings. Innsbruck, Austria.
- Greenwold, Simon (2012): Upfork-particles. Available online at <https://github.com/juniperoserra/upfork-particles>, checked on 3/4/2019.
- Hansen, Leif; Kim, Sara (2016): 'COCOON' a bamboo building with integration of digital design and low-tech construction. In Structures and Architecture. London, UK., pp. 1159–1165.
- Helms, Bergen (2012): Object-Oriented Graph Grammars for Computational Design Synthesis. Doctoral Thesis. Technische Universität München, Munich, Germany. Fakultät für Maschinenwesen.
- Hernández, E. Lafuente; Gengnagel, C. (2014): A new hybrid: elastic gridshells braced by membranes. In Mobile and Rapidly Assembled Structures IV. Ostend, Belgium., pp. 157–169. DOI: 10.2495/MAR140131.
- Hilleberg Group (2018): Hilleberg. The tent handbook 2018. Edited by Hilleberg Group. Available online at <http://docs.hilleberg.net/Hilleberg2018Handbook-EN.pdf>.
- Holden Deleuran, Anders; Schmeck, Michel; Quinn, Gregory; Gengnagel, Christoph; Tamke, Martin; Ramsgaard Thomsen, Mette (2015): The tower. Modelling, Analysis and Construction of Bending Active Tensile Membrane Hybrid Structures. In Future Visions: Proceedings of the International Association for Shell and Spatial Structures (IASS). Amsterdam, The Netherlands.
- Huerta, Santiago (2006): Structural Design in the Work of Gaudí. In Architectural Science Review 49 (4), pp. 324–339. DOI: 10.3763/asre.2006.4943.
- Joldes, Grand Roman; Wittek, Adam; Miller, Karol (2009): Computation of intra-operative brain shift using dynamic relaxation. In Computer methods in applied mechanics and engineering 198 (41), pp. 3313–3320. DOI: 10.1016/j.cma.2009.06.012.
- Joldes, Grand Roman; Wittek, Adam; Miller, Karol (2011): An adaptive Dynamic Relaxation method for solving nonlinear finite element problems. Application to brain shift estimation. In International journal for numerical methods in biomedical engineering 27 (2), pp. 173–185. DOI: 10.1002/cnm.1407.
- Kao, Gene; Nguyen, Long (2018): PhysX.GH. Available online at <https://>

github.com/TheAsianCoders/PhysX.GH, checked on 3/7/2019.

- Killian, Axel (2004): Linking Digital Hanging Chain Models to Fabrication. In *Fabrication: Examining the Digital Practice of Architecture: Proceedings of the 23rd Annual Conference of ACADIA and the 2004 Conference of the AIA Technology in Architectural Practice Knowledge Community*. Cambridge and Toronto, Ontario, Canada., pp. 110–125.
- Killian, Axel (2006): Design exploration through bidirectional modeling of constraints. Ph.D. Massachusetts Institute of Technology. Dept. of Architecture, Boston, Massachusetts.
- Killian, Axel; Ochsendorf, John (2005): Particle-spring systems for structural form-finding. In *Journal of the International Association for Shell and Spatial Structures* 46 (2), pp. 77–84.
- Knippers, Jan; Cremers, Jan; Gabler, Markus; Lienhard, Julian (2011): *Construction Manual for Polymers + Membranes*. München: DE GRUYTER.
- Kolarevic, Branko (2003): Computing the Performative in Architecture. In *Digital Design: 21th eCAADe Conference Proceedings*. Graz, Austria, pp. 457–464.
- Kotelnikova-Weiler, N.; Douthe, C.; LaFuente Hernandez, E.; Baverel, O.; Gengnagel, C.; Caron, J-F (2013): Materials for Actively-Bent Structures. In *Internacional Journal of Space Structures* 28 (3&4), pp. 229–240.
- Kraemer, Pierre; Untereiner, Lionel; Jund, Thomas; Thery, Sylvain; Cazier, David (2014): CGoGN: n-dimensional Meshes with Combinatorial Maps. In *Proceedings of the 22nd International Meshing Roundtable*. Switzerland. 45, pp. 485–503. DOI: 10.1007/978-3-319-02335-9_27.
- La Magna, Riccardo (2017): Bending-Active Plates. Strategies for the Induction of Curvature through the Means of Elastic Bending of Plate-based Structures. Stuttgart: Universität Stuttgart Inst. f. Tragkonstr (Forschungsberichte aus dem Institut für Tragkonstruktionen und konstruktives Entwerfen der Universität Stuttgart, 43).
- La Magna, Riccardo; Schleicher, Simon; Knippers, Jan (2016): Bending-Active Plates. Form and Structure. In *Advances in Architectural Geometry 2016*, pp. 170–186.
- LaFuente Hernandez, Elisa; Gengnagel, Christoph (2012): Case-studies of arched structures using actively-bent elements. In *From Spatial Structures to Space Structures: IASS-APCS Conference Proceedings*. Seoul, South Korea.

- Lahanas, Vasileios; Georgiou, Evangelos; Loukas, Constantinos (2016): Surgical Simulation Training Systems: Box Trainers, Virtual Reality and Augmented Reality Simulators. In *IJARA* 1 (2), pp. 1–9. DOI: 10.15226/2473-3032/1/2/00109.
- Lázaro, Carlos; Monleón, Salvador; Bessini, Juan; Casanova, José (2016): A review of geometrically exact models for very flexible rods. In *Proceedings of the IASS Annual Symposium: Spatial Structures in the 21st Century*. Tokyo, Japan.
- Lewis, W. J. (2008): Computational form-finding methods for fabric structures. In *Proceedings of the Institution of Civil Engineers - Engineering and Computational Mechanics* 161 (3), pp. 139–149. DOI: 10.1680/eacm.2008.161.3.139.
- Lewis, Wanda J. (2003): *Tension structures. Form and behaviour*. London: Thomas Telford.
- Lewis, Wanda J.; Gosling, Peter D. (1993): Stable Minimal Surfaces in Form-Finding of Lightweight Tension Structures. In *International Journal of Space Structures* 8 (3), pp. 149–166.
- Li, Jian-Min (2017): *Timber Shell Structures. Form-finding and structural analysis of actively bent grid shells*. 1. Auflage. Stuttgart: Universität Stuttgart Inst. f. Tragkonstr (Forschungsberichte aus dem Institut für Tragkonstruktionen und konstruktives Entwerfen der Universität Stuttgart, 42).
- Li, Jian-Min; Knippers, Jan (2012): Rotation Formulations for Dynamic Relaxation. With Application in 3D Framed Structures with Large Displacements and Rotations. In *From Spatial Structures to Space Structures: Proceedings of the International Association for Shell and Spatial Structures (IASS)*. Seoul, Korea.
- Li, Qingpeng; Andrew, Borgart; Yue, Wu (2015): The Vector Form Intrinsic Finite Element method and several other form-finding methods for general networks. In *Proceedings of the IASS Symposium: Future Visions*. Amsterdam, Netherlands.
- Liddell, Ian (2015): Frei Otto and the development of gridshells. In *Case Studies in Structural Engineering* 4, pp. 39–49. DOI: 10.1016/j.csse.2015.08.001.
- Lienhard, Julian (2014): *Bending-active structures. Form-finding strategies using elastic deformation in static and kinetic systems and the structural potentials therein*. Zugl.: Stuttgart, Univ., Diss., 2014. Stuttgart: ITKE (Forschungsberichte aus dem Institut für Tragkonstruktionen und

- Konstruktives Entwerfen, Universität Stuttgart, 36). Available online at <http://nbn-resolving.de/urn:nbn:de:bsz:93-opus-94838>.
- Lienhard, Julian; Ahlquist, Sean; Menges, Achim; Knippers, Jan (2013a): Extending the Functional and Formal vocabulary of tensile membrane structures through the interaction with bending-active elements. In [RE] THINKING Lightweight Structures, Proceedings of Tensinet Symposium. Istanbul, Turkey., pp. 109–118.
- Lienhard, Julian; Ahlquist, Sean; Menges, Achim; Knippers, Jan (2013b): Finite Element Modelling in Integral Design Strategies of Form- and Bending-Active Hybrid Structures. In Structural Membranes: VI International Conference on Textile Composites and Inflatable Structures. Munich, Germany.
- Lienhard, Julian; Alpermann, Holger; Gengnagel, Christoph; Knippers, Jan (2013c): Active Bending. A review on structures where bending is used as a self-formation process. In International Journal of Space Structures 28 (3&4), pp. 187–196.
- Lienhard, Julian; Gengnagel, Julian (2018): Recent developments in bending-active structures. In Proceedings of the IASS Symposium 2018: Creativity in Structural Design. MIT Boston, USA.
- Lienhard, Julian; Knippers, Jan (2015): Bending-active textile hybrids. In Journal of the International Association for Shell and Spatial Structures 56 (1), pp. 37–48.
- Lienhard, Julian; La Magna, Riccardo; Knippers, Jan (2014): Form-finding bending-active structures with temporary ultra-elastic contraction elements. In Mobile and Rapidly Assembled Structures IV. Ostend, Belgium., pp. 107–115. DOI: 10.2495/MAR140091.
- Lienhardt, Pascal; Fuchs, Laurent; Bertrand, Yves (2011): Combinatorial models for topology-based geometric modeling. In Theory and applications of proximity, nearness and uniformity. Quaderni di matematica, dipartimento di matematica, seconda universita di Napoli. Napoli, Italy., pp. 151–198.
- Lienhardt, Pascal; Skapin, Xavier; BERGEY, ANTOINE (2004): Cartesian product of simplicial and cellular structures. In Int. J. Comput. Geom. Appl. 14 (03), pp. 115–159. DOI: 10.1142/S0218195904001408.
- Linkwitz, Klaus; Veenendaal, Diederik (2014): Nonlinear force density method. In Sigrid Adriaenssens, Philippe Block, Diederik Veenendaal, Chris Williams (Eds.): Shell Structures for Architecture. Form finding and optimization. London, New York: Routledge.

- Liu, Tiantian; Bargteil, Adam W.; O'Brien, James; Kavan, Ladislava (2013): Fast Simulation of Mass-Spring Systems. In *Transaction on Graphics* 32 (6).
- Luciani, Annie; Allaoui, Ali; Castagné, Nicolas; Darles, Emmanuelle; Skapin, Xavier; Meseure, Philippe (2014): MORPHO-Map. A New Way to Model Animation of Topological Transformations. In *International Conference on Computer Graphics Theory and Applications (GRAPP)*. Lisbon, Portugal.
- Lynn, Greg (1999): *Animate form*. New York, NY: Princeton Architectural Press.
- Martin, Sebastian; Thomaszewski, Bernhard; Grinspun, Eitan; Gross, Markus (2011): Example-based elastic materials. In *ACM Trans. Graph.* 30 (4), p. 1. DOI: 10.1145/2010324.1964967.
- Martini, Kirk (2001): Non-linear Structural Analysis as Real-Time Animation. Borrowing from the Arcade. In *Proceedings of the Computer-Aided Architectural Design Futures 2001 Conference*. Eindhoven, The Netherlands., pp. 643–656.
- Martini, Kirk (2005): Real-time, Non-linear, Dynamic Simulation in Teaching Structures. Elementary to Advanced. In *Proceedings of the 2005 American Society for Engineering Education Annual Conference*. Portland, Oregon.
- Martini, Kirk (2013): Nonlinear Dynamics: An Intuitive Digital Representation of Structure. In *Journal for Education in the Built Environment* 8 (1), pp. 73–85. DOI: 10.11120/jebe.2013.00002.
- Menges, Achim (2010): Material Information. Integrating Material Characteristics and Behavior in Computational Design for Performative Wood Construction. In *LIFE in:formation, On Responsive Information and Variations in Architecture: Proceedings of the 30th Annual Conference of the Association for Computer Aided Design in Architecture*. New York, USA.
- Meseure, Philippe; Darles, Emmanuelle; Skapin, Xavier (2010): Topology-based Physical Simulation. In *VirtualReality Interaction and Physical Simulation 2010 (VRIPHYS)*. Copenhagen, Denmark., pp. 1–10. DOI: 10.2312/PE/vriphys/vriphys10/001-010.
- Minke, Gernot (2016): *Building with Bamboo. Design and Technology of a Sustainable Architecture* Second and revised edition. 2nd ed. Basel/Berlin/Boston: Birkhäuser. Available online at <http://gbv.eblib.com/patron/FullRecord.aspx?p=4533878>.

- Mitchell, Nathan; Cutting, Court; Sifakis, Eftychios (2015): GRIDiron. In *ACM Trans. Graph.* 34 (4), 43:1-43:12. DOI: 10.1145/2766918.
- Mos, Nicolas; Dolbow, John; Belytschko, Ted (1999): A finite element method for crack growth without remeshing. In *Int. J. Numer. Meth. Engng.* 46 (1), pp. 131–150. DOI: 10.1002/(SICI)1097-0207(19990910)46:1<131::AID-NME726>3.0.CO;2-J.
- Mosconi, M.; Porta, M. (2000): Iteration constructs in data-flow visual programming languages. In *Computer Languages* 26 (2-4), pp. 67–104. DOI: 10.1016/S0096-0551(01)00009-1.
- Müller, Matthias; Heidelberger, Bruno; Hennix, Marcus; Ratcliff, John (2007): Position based dynamics. In *Journal of Visual Communication and Image Representation* 18 (2), pp. 109–118. DOI: 10.1016/j.jvcir.2007.01.005.
- NetworkX Developers (2019): NetworkX. Available online at <https://networkx.github.io/>.
- Nguyen, Long (2017): DynaShape. Available online at <https://forum.dynamobim.com/t/dynashape/11666>.
- Nikishkov, Gennadiy (2010): *Programming finite elements in Java*. London: Springer.
- Otto, Frei; Glaeser, Ludwig (1972): *The work of Frei Otto*. New York, Greenwich, Connecticut: Museum of Modern Art; Distributed by New York Graphic Society.
- Oxman, Rivka; Hammer, Roey; Ben Ari, Shoham (2007): Performative design in Architecture. Employment of Virtual Prototyping as a Simulation Environment in Design Generation. In *Predicting the Future: 25th eCAADe Conference Proceedings*. Frankfurt am Main, Germany., pp. 227–234.
- Pan, Li; Metzger, Don; Niewczas, Marek (2004): Meshless Dynamic Relaxation Technique for Simulating Atomistic Models Subjected to External Forces Under the Periodic Symmetry. In *ASME/JSME Pressure Vessels and Piping Conference*, San Diego, California, pp. 11–17. DOI: 10.1115/PVP2004-2740.
- Pian, T.H.H.; Balmer, H. A.; Bucciarelli, L. L. (1967): Dynamic buckling of a circular ring constrained in a rigid circular surface. In *Dynamic stability of structures. Proceedings of an international conference held at Northwestern University, Evanston, Illinois.*, pp. 285–297. DOI: 10.1016/B978-1-4831-9821-7.50021-1.

- Piker, Daniel (2013): Kangaroo: Form Finding with Computational Physics. In *Archit Design* 83 (2), pp. 136–137. DOI: 10.1002/ad.1569.
- Piker, Daniel (2014): Mesh Machine. Available online at <https://github.com/Dan-Piker/MeshMachine>.
- Quinn, Gregory; Deleuran, Anders Holden; Piker, Daniel; Brandt-Olsen, Cecilie; Tamke, Martin; Thomsen, Mette Ramsgaard; Gengnagel, Christoph (2016): Calibrated and Interactive Modelling of Form-Active Hybrid Structures. In *Spatial Structures in the 21st Century: Proceedings of the IASS Annual Symposium*. Tokyo, Japan.
- Reeves, William T. (1983): Particle Systems. A technique for Modeling a Class of Fuzzy Objects. In *ACM Transactions on Graphics* 2 (2), pp. 91–108.
- Reynolds, Craig W. (1987): Flocks, herds and schools: A distributed behavioral model. In *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*. New York, USA., pp. 25–34. DOI: 10.1145/37401.37406.
- Russell, Stuart J.; Norvig, Peter (2016): Artificial intelligence. A modern approach. With assistance of Ernest Davis, Douglas Edwards. Third edition, Global edition. Boston, Columbus, Indianapolis: Pearson (Always learning).
- Rutzinger, Stefan; Schinegger, Kristina; Knippers, Jan; Scheible, Florian (2012): Kinetic Facade. Theme Pavilion Expo 2012 Yeosu, South Korea. In *Prototyping Architecture: The Conference Papers*, London, pp. 83–91.
- Schek, H.-J. (1974): The force density method for form finding and computation of general networks. In *Computer methods in applied mechanics and engineering* 3 (1), pp. 115–134. DOI: 10.1016/0045-7825(74)90045-0.
- Schleicher, Simon; Rastetter, Andrew; La Magna, Riccardo; Schönbrunner, Andreas; Haberbosch, Nicola; Knippers, Jan (2015): Form-Finding and Design Potentials of Bending-Active Plate Structures. In *Modelling Behaviour*, Cham, pp. 53–63. DOI: 10.1007/978-3-319-24208-8_5.
- Schling, Eike; Killian, Martin; Wang, Hui; Schikore, Jonas; Pottmann, Helmut (2018): Design and Construction of Curved Support Structures with Repetitive Parameters. In *Advances in Architectural Geometry*. Gothenburg, Sweden.
- Schmeck, Michel; Gengnagel, Christoph (2016): Calibrated Modeling of Knitted Fabric as a Means of Simulating Textile Hybrid Structures. In *Procedia Engineering* 155, pp. 297–305. DOI: 10.1016/j.proeng.2016.08.032.

- Schmidt, Karsten (2007): ToxiClibs. Available online at <https://github.com/postspectacular/toxiclibs>, checked on 3/4/2019.
- Senatore, Gennaro; Piker, Daniel (2015): Interactive real-time physics. An intuitive approach to form-finding and structural analysis for design and education. In *Computer-Aided Design* 61, pp. 32–41. DOI: 10.1016/j.cad.2014.02.007.
- Shea, Kristina; Aish, Robert; Gourtovaia, Marina (2005): Towards integrated performance-driven generative design tools. In *Automation in Construction* 14 (2), pp. 253–264. DOI: 10.1016/j.autcon.2004.07.002.
- Sorkin, Olga; Alexa, Marc (2007): As-Rigid-As-Possible Surface Modeling. In *Proceedings of the fifth Eurographics symposium on Geometry Processing*. Barcelona, Spain., pp. 109–116.
- Staten, Matthew L.; Owen, Steven J.; Blacker, Ted D. (2005): Unconstrained Paving & Plastering: A New Idea for All Hexahedral Mesh Generation. In *Proceedings of the 14th International Meshing Roundtable*. Berlin, Germany. 32, pp. 399–416. DOI: 10.1007/3-540-29090-7_24.
- Suzuki, Seiichi; Knippers, Jan (2017a): ElasticSpace: A computational framework for interactive form-finding of textile hybrid structures through evolving topology networks. In *International Journal of Parallel, Emergent and Distributed Systems* 32 (sup1), S4-S14. DOI: 10.1080/17445760.2017.1390101.
- Suzuki, Seiichi; Knippers, Jan (2017b): The Design Implications of Form-Finding with Dynamic Topologies. In *Design Modelling Symposium: “Humanizing Digital Reality”*. Paris, France. 29, pp. 211–223. DOI: 10.1007/978-981-10-6611-5_19.
- Suzuki, Seiichi; Knippers, Jan (2017c): Topology-Driven Form-Finding. Implementation of an Evolving Network Model for Extending Design Spaces in Dynamic Relaxation. In *22nd International Conference of the Association for Computer-Aided Architectural Design Research in Asia (CAADRIA): “Protocols, Flows and Glitches”*. Suzhou, China., pp. 489–498.
- Suzuki, Seiichi; Knippers, Jan (2018): Digital Vernacular Design. Form-Finding At The Edge Of Realities. In *Proceedings of the 38th Annual Conference of the Association for Computer Aided Design in Architecture (ACADIA)*. Mexico City, Mexico.
- Suzuki, Seiichi; Körner, Axel; Knippers, Jan (2018): Iguana. Advances on the development of a robust computational framework for active-geometric

- and -topologic modeling of lightweight structures. In Proceedings of the IASS Symposium 2018: “Creativity in Structural Design”. Boston, USA.
- Suzuki, Seiichi; Slabbinck, Evy; Knippers, Jan (2017): A comparative overview of generative approaches for computational form-finding of bending-active tensile structures. In Proceedings of the IASS Annual Symposium 2017: “Interfaces: Architecture, Engineering and Science”. Hamburg, Germany.
- Suzuki, Seiichi; Slabbinck, Evy; Knippers, Jan (2019): Computational Bamboo. Digital and Vernacular Design Principles for the Construction of a Temporary Bending-Active Structure (Forthcoming). In The Design Modelling Symposium: “Impact: Design With All Senses”. Berlin, Germany.
- Terzopoulos, Demetri; Fleischer, Kurt (1988a): Deformable models. In *The Visual Computer* 4 (6), pp. 306–331. DOI: 10.1007/BF01908877.
- Terzopoulos, Demetri; Fleischer, Kurt (1988b): Modeling inelastic deformation. In *SIGGRAPH Comput. Graph.* 22 (4), pp. 269–278. DOI: 10.1145/378456.378522.
- Terzopoulos, Demetri; Platt, John; Barr, Alan; Fleischer, Kurt (1987): Elastically Deformable Models. In *Computer Graphics* 21 (4).
- Thomsen, Mette Ramsgaard; Tamke, Martin; Deleuran, Anders Holden; Tinning, Ida Katrine Friis; Evers, Henrik Leander; Gengnagel, Christoph; Schmeck, Michel (2015): Hybrid Tower. Designing Soft Structures. In *Design Modelling Symposium*. Copenhagen, Denmark., pp. 87–99. DOI: 10.1007/978-3-319-24208-8_8.
- Tibert, A. G.; Pellegrino, S. (2003): Review of Form-Finding Methods for Tensegrity Structures. In *International Journal of Space Structures* 18 (4), pp. 209–223. DOI: 10.1260/026635103322987940.
- van Mele, Tom; Laet, Lars de; Veenendaal, Diederik; Mollaert, Marijke; Block, Philippe (2013): Shaping Tension Structures with Actively Bent Linear Elements. In *International Journal of Space Structures* 28 (3-4), pp. 127–136. DOI: 10.1260/0266-3511.28.3-4.127.
- Veenendaal, D.; Block, P. (2012): An overview and comparison of structural form finding methods for general networks. In *International Journal of Solids and Structures* 49 (26), pp. 3741–3753. DOI: 10.1016/j.ijsolstr.2012.08.008.
- Veenendaal, Diederik; Block, Philippe (2011): A Framework for Comparing Form-Finding Methods. In Proceedings of the IABSE-IASS Symposium.

London, UK.

- Veenendaal, Diederik; Block, Philippe (2014): Comparison of form-finding methods. In Sigrid Adriaenssens, Philippe Block, Diederik Veenendaal, Chris Williams (Eds.): *Shell Structures for Architecture. Form finding and optimization*, vol. 2014. London, New York: Routledge.
- Vestartas, Petras; Heinrich, Mary Katherine; Zwierzycki, Mateusz; Leon, David Andres; Cheheltan, Ashkan; La Magna, Riccardo; Ayres, Phil (2018): Design Tools and Workflows for Braided Structures. In *Humanizing Digital Reality* 28, pp. 671–681. DOI: 10.1007/978-981-10-6611-5_55.
- Voesenek, C.J. (2008): Implementing a Fourth Order Runge-Kutta Method for Orbit Simulation. Available online at <http://spiff.rit.edu/richmond/nbody/OrbitRungeKutta4.pdf>.
- Volino, P.; Thalmann, N. M.; Jianhua, Shen; Thalmann, D. (1996): An evolving system for simulating clothes on virtual actors. In *IEEE Comput. Graph. Appl.* 16 (5), pp. 42–51. DOI: 10.1109/38.536274.
- Volino, Pascal; Courchesne, Martin; Magnenat Thalmann, Nadia (1995): Versatile and efficient techniques for simulating cloth and other deformable objects. In *SIGGRAPH 95 Conference proceedings*. New York, USA., pp. 137–144. DOI: 10.1145/218380.218432.
- Wakefield, David Simon (1980): *Dynamic Relaxation Analysis of Pretensioned Networks Supported by Compression Arches*. Doctoral Thesis, London. City University of London.
- White, David; Kinney, Paul (1997): Redesign of the Paving Algorithm. Robustness Enhancements through Element by Element Meshing. In *Proceedings, 6th International Meshing Roundtable*, Sandia National Laboratories, pp. 323–335.
- Williams, Chris (2001): The analytic and numerical definition of the geometry of the British Museum Great Court Roof. In *Mathematics & Design 2001*. Geelong, Victoria, Australia., pp. 434–440.
- Wu, Jun; Westermann, Rüdiger; Dick, Christian (2015): A Survey of Physically Based Simulation of Cuts in Deformable Bodies. In *Computer Graphics Forum* 34 (6), pp. 161–187. DOI: 10.1111/cgf.12528.
- Zhou, Xueqian; Sutulo, Serge; Guedes Soares, C. (2016): A paving algorithm for dynamic generation of quadrilateral meshes for online numerical simulations of ship manoeuvring in shallow water. In *Ocean Engineering* 122, pp. 10–21. DOI: 10.1016/j.oceaneng.2016.06.008.

Zimmermann, Thomas; Dubois-Pèlerin, Yves; Bomme, Patricia (1992):
Object-oriented finite element programming: I. Governing principles.
In Computer methods in applied mechanics and engineering 98 (2), pp.
291–303. DOI: 10.1016/0045-7825(92)90180-R.

Image Index

Only images that are not original by the author are listed.

Figure 2.1: (Top) Flickr (2009), Burnham Pavilion [Online Image], Available from: <https://www.flickr.com/photos/picken/3978353952> [Accessed 11/07/19]; (Bottom) The Burnham Plan Centennial (2009), Burnham Pavilion Fabrication Stage [Online Image], Available from: http://burnhamplan100.lib.uchicago.edu/multimedia/image_gallery/detail/2342/ [Accessed 11/07/19].

Figure 2.4: Hubert Berberich (2012), Multihalle Mannheim [Online Image], Available from: <https://commons.wikimedia.org/wiki/File:Multihalle07.jpg> [Accessed 14/07/19].

Figure 2.5/2.11/2.13: Otto, Frei; Glaeser, Ludwig (1972), Images retrieved from: The work of Frei Otto, Museum of Modern Art, p.39.

Figure 2.6: Hiroyuki Hirai (2014), Japan Pavilion in Hannover [online Image], Available from: <https://www.flickr.com/photos/eager/17068464076/in/photostream/> [Accessed 14/07/19].

Figure 2.7: Basher Eyre (2013), Weald and Downland Gridshell [Online Image], <https://www.geograph.org.uk/photo/3762371> [Accessed 18/07/19].

Figure 2.8: Du Peloux (2017), Ephemeral Cathedral of Créteil, Image retrieved from: Modeling of bending-torsion couplings in active-bending structures: Application to the design of elastic. Thesis (PhD), École des Ponts Paris Tech.

Figure 2.9: ICD/ITKE University of Stuttgart (2016), ICD/ITKE Research Pavilion 2015-2016 [Online Image], Available from: <https://icd.uni-stuttgart.de/?p=16220> [Accessed 18/07/19].

Figure 2.10: Rutzinger, Stefan; Schinegger, Kristina; Knippers, Jan; Scheible, Florian (2012): Kinetic Facade. Theme Pavilion Expo 2012 Yeosu, South Korea. In Prototyping Architecture: The Conference Papers, London, pp. 83–91.

Figure 2.12: Yoshito Isono - Structurae (2001) Dance Pavilion [Online Image], Available from: <https://structurae.net/en/structures/dance-pavilion-at-the-federal-garden-exhibition> [Accessed 05/08/19].

Figure 2.14: Josu Orbe (2016), Olympic Stadium of Munich [Online Image], Available from: <https://www.flickr.com/photos/jorbe/26231335796> [Accessed 05/08/19].

Figure 2.15: Hilleberg Group (2018): Hilleberg. The tent handbook 2018. Edited by Hilleberg Group. Available online at <http://docs.hilleberg.net/Hilleberg2018Handbook-EN.pdf>.

Figure 2.17: JSortimo (2011) Pultrusion process [Online Image], Available from: <https://commons.wikimedia.org/wiki/File:Pultrusionsanlage.jpg> [Accessed 05/08/19].

Figure 2.18: Hansen, Leif; Kim, Sara (2016): 'COCOON' a bamboo building with integration of digital design and low-tech construction. In Structures and Architecture. London, UK., pp. 1159–1165.

Figure 2.19: Crolla, Kristof (2017): Building indeterminacy modelling – the 'ZCB Bamboo Pavilion' as a case study on nonstandard construction from natural materials. In Vis. in Eng. 5 (1), p. 178. DOI: 10.1186/s40327-017-0051-4.

Figure 2.20/2.21: Liddell, Ian (2015): Frei Otto and the development of gridshells. In Case Studies in Structural Engineering 4, pp. 39–49. DOI: 10.1016/j.csse.2015.08.001.

Figure 2.22: ICD/ITKE University of Stuttgart (2010), ICD/ITKE Research Pavilion 2010 [Online Image], Available from: <https://icd.uni-stuttgart.de/?p=4458> [Accessed 18/07/19].

Figure 2.23/2.24/2.25: Lienhard, Julian (2014): Bending-active structures. Form-finding strategies using elastic deformation in static and kinetic systems and the structural potentials therein. Zugl.: Stuttgart, Univ., Diss., 2014. Stuttgart: ITKE (Forschungsberichte aus dem Institut für Tragkonstruktionen und Konstruktives Entwerfen, Universität Stuttgart, 36). Available online at <http://nbn-resolving.de/urn:nbn:de:bsz:93-opus-94838>.

Figure 2.26: ICD University of Stuttgart (2012), Material Equilibria [Online Image], Available from: <https://icd.uni-stuttgart.de/?p=7636> [Accessed 18/07/19].

Figure 2.27: Holden Deleuran, Anders; Schmeck, Michel; Quinn, Gregory; Gengnagel, Christoph; Tamke, Martin; Ramsgaard Thomsen, Mette

(2015): The tower. Modelling, Analysis and Construction of Bending Active Tensile Membrane Hybrid Structures. In Future Visions: Proceedings of the International Association for Shell and Spatial Structures (IASS). Amsterdam, The Netherlands.

Figure 6.1: ICD/ITKE University of Stuttgart (2018), ICD/ITKE Bending-Active Demonstrator 2017-18.

Figure 6.6/6.7: Photo: Manuel Jimenez Garcia.

Figure 6.15: Photo: James Solly.

Figure 6.16: Photo: Evy Slabbinck.

Figure 6.30/6.31: Photo: Valentina Soana and Mathias Maierhofer.

Project Acknowledgements

Bending-Active Demonstrator 2017-18.

Responsible for the scientific development: Oliver Bucklin, Marion Lutz, Anja Mader, Evy Slabbinck, Seiichi Suzuki and Lauren Vasey.

With the support of: Leyla Kyjánek Yunis, Behrooz Tahanzadeh, Michael Preisack, Maria Yablonina, Larissa Born, Daniel Reist, Daniel Sonntag, Bas Rongen, Boris Brozovic, Annette Smitt and James Solly.

The student team included: Rasha Alshami, Elaine Bonavia, Brad Elsbury, Samuel Leder, Xun Li, Mathias Maierhofer, Israel Luna Miño, Marie Razzhivina, Jacob Russo, Hosna Shayani, Valentina Soana, Sabīne Vecvagare, Ramon Weber, and Jacob Zindroski, and additional contributions by: Lidia Atanasova, Sven Baur, Ludwig Ebert, and Eric Fuertas.

The author's personal involvement included tutoring, design conceptualisation and exploratory form-finding.

AAVS Madrid 2016.

Responsible for the scientific development: Christina Dahdaleh, Manuel Jimenez Garcia, Seiichi Suzuki.

With the support of: Vicente Soler and Ignacio Viguera.

The student team included: Ana Taboada de Zuniga, Elliot Mayer, Facundo Taborda, Ipek Kuran, Iris Senel, Maria Rodriguez Jurado, Sukriye Robinson, Anna Shishkina, Kevin Moreno and Joey Ibrahim..

The author's personal involvement included design conceptualisation and exploratory form-finding.

Bat01.

Responsible for the scientific development: Evy Slabbinck and Seiichi Suzuki.

The author's personal involvement included design conceptualisation and exploratory form-finding.

Bat02.

Responsible for the scientific development: Evy Slabbinck and Seiichi Suzuki.

The author's personal involvement included design conceptualisation and exploratory form-finding.

IAAC-GSS 2018 Computational Bamboo.

Responsible for the scientific development: Evy Slabbinck and Seiichi Suzuki.

With the support of: Yaner Luis Cangá Chavez, Andres Obregon, and Juan Edison Columba Paucar.

The student team included: Juan Francisco Mayorga Jaramillo, Valeria Latorre, Gordon Leibowitz, Valeria León, Veronica Sofia Molina, Maria Gabriela Chacon Pa-lacios, Gisella Parra, Fausto Andres Davila Pazmiño, and Juan David Barona Trujillo, and additional volunteers: Byron Esteban Cadena Campos, Sophia Natalia Gualán Chacón, Gustavo Francisco Abdo Hernández, Santi-ago Dario Guevara Hidalgo, Karen Elizabeth Landazuri Proaño, Rafael Fernando Suárez Molina, Santiago Gabriel Miño Paz, and Nestor Emmanuel Mendoza Zam-brano.

The author's personal involvement included tutoring, design conceptualisation and exploratory form-finding.

Self-Choreographic Networks.

Responsible for the scientific development: Valentina Soana and Mathias Maierhofer.

With the support of: Axel Körner, Seiichi Suzuki and Maria Yablonina.

The author's personal involvement included tutoring and the development of the physical solver for conducting the exploratory form-finding processes.

