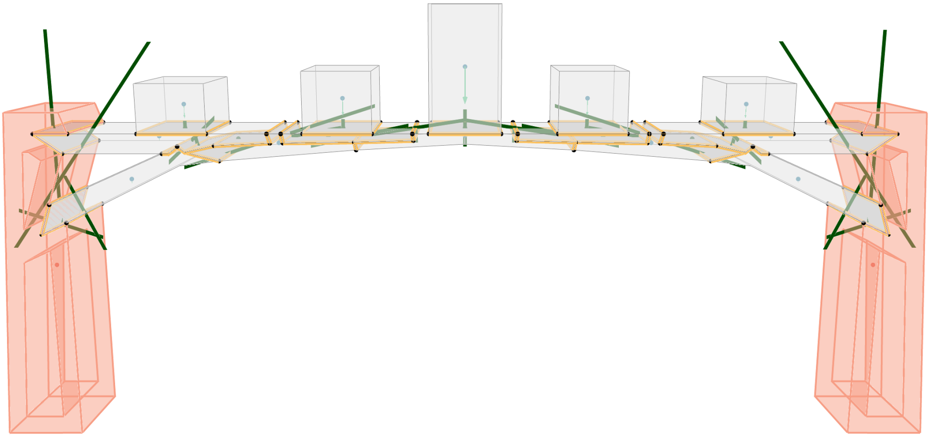
ETH Zurich | PhD Dissertation ETH No. 29151



Stability Assessment of Discrete Shell Structures during Assembly

Gene Ting-Chun Kao

Diss. ETH No. 29151

Stability Assessment of Discrete Shell Structures during Assembly

A thesis submitted to attain the degree of

Doctor of Sciences
(Dr. sc. ETH Zurich)

presented by

Gene Ting-Chun Kao

M.Sc. in Integrated Technologies and Architectural Design Research
Stuttgart University

born on 18.02.1990

accepted on the recommendation of

Prof. Dr. Philippe Block
Prof. Dr. Stelian Coros
Prof. Dr. Jan Knippers
Dr. Tom Van Mele

2023

To Bonnie & Genie

Acknowledgements

Foremost, I want to express the most sincere thankfulness to my advisor, Prof. Dr Philippe Block. This thesis could not be possible without his knowledge and wisdom; I could not be where I am today without his mentorship and guidance. I am particularly grateful to him for giving me such freedom and time to explore unknown knowledge and ideas during my doctoral study. He was extremely patient with me even when my research was stuck. He gave me a lot of encouragement to complete my thesis. Together with Philippe, I am grateful to my thesis reviewer and technical advisor Dr Tom Van Mele, who co-directs the Block Research Group (BRG) at ETH Zurich. Tom was always very supportive and helped me with his profound computational perspective. His energetic characteristic and passion for coding deeply inspired me.

Furthermore, I appreciate all the help from my co-supervisor, Prof. Dr Stelian Coros, from the Computational Robotic Lab (CRL) at ETH Computer Science Department. I am grateful that he became my advisor from the beginning of my doctoral study, spent much time guiding and discussing mathematical and computational details with me, constantly challenged my thinking, and inspired me with different ways of approaching things. I also value Stelian's encouragement during my journey of the CAS ETH study in computer science, which better equipped me for my research topics. Also, I want to acknowledge my thesis reviewer, Prof. Dr Jan Knippers, who also was my supervisor during my master's study at Stuttgart University

back in 2016. His dedication to high-quality education and mentorship well-prepared me for joining BRG.

I am grateful for being part of the incredible research family at BRG, which has a great vibe and made me feel at home in Zurich. I have too many people to thank within BRG. First, I want to give a lot of my heartfelt thanks to Dr Antonino Iannuzzo for his endless support, as my second year of doctoral study in 2020 was particularly challenging. While examining an existing research method, I discovered a significant issue that required attention. I felt trapped in high uncertainty because my research could not go on smoothly as planned. Besides, 2020 was also the start of covid pandemic, and we all had to be at home for lockdown. I felt being in the darkness without seeing any light. The sudden uncertainty of my research, combined with the presence of the pandemic, made everything scary and stressful. Antonino became a source of comfort and unwavering support. He often dedicated numerous hours during the day to discussing and exploring mathematical formulations with me online. I wholeheartedly acknowledge that my research work would not have been possible without him. Second, I also would like to extend my heartfelt gratitude to my colleague and friend, Francesco Ranaudo, for always being considerate and helping me. I learnt so much from him in all kinds of aspects, including in research and daily life. He is one of the kindest and most knowledgeable individuals I have ever met, offering support without hesitation or reservation. Third, I like to thank Dr Matthias Rippmann for supporting and guiding me at the beginning of my time in Zurich. His sudden passing brought all of us sorrow, but his outstanding research work deeply inspired mine. Although he could not guide me entirely, I wish he was still with us to see my final accomplishment. Fourth, I thank Dr Ricardo Maia Avelino and Lotte (Aldinger) Scheder-Bieschin for being my study buddies at ETH. I am grateful particularly Ricardo often helped me with his profound engineering knowledge, and Lotte offered her help with correcting my German abstract writing. Besides, I want to thank Kam-Ming Mark Tam, Chaoyu Du, and Li Chen

for endless funny and humorous discussions in my mother tongue Mandarin, which brought me much joy in the English research environment. Specially and importantly, I want to extend my gratefulness to Dr Noelle Paulson and Kathrin Haake, who gave me endless support in various administrative matters. Last, I would like to thank other researchers, colleagues, and ex-colleagues in BRG, Dr Alesssandro Dell'Endice, Dr Lluis Enrique, Dr Serban Bodea, Dr Juney Lee, Dr Mariana Popescu, Dr Cristián Calvo Barentin, Dr Tomás Méndez Echenagucia, Dr Andrew Liew, Dr Robin Oval, Dr Shajay Bhooshan, Dr David López López, Dr Rebecca Fugger, Dr Anjali Mehrotra, Selina Bitting, Keerthana Udaykumar, Sam Bouten, Sandie Kate Fenton, Rafael Pastrana, Hao Wu, Andrea Manardo, Kerstin Spiekermann, Beril Önalan, Artemis Maneka, Stefan Linger, Nora Ravanidou, Wenqian Yang, Aurèle Gheyselinck, Julie Assunção, Minxiang Gao, Stephen Lamb, Jef Rombouts, Una Wang, and everyone else, who directly or indirectly helped me.

I am also thankful for the many brilliant collaborators I worked with, Prof. Dr Benjamin Dillenburger, Dr Bernhard Thomaszewski, Dr Ziqi Wang, Dr Andrea Carron, Dr Andrea Jipa, Ioanna Mitropoulou, Angela Yoo, and Georgia Chousou. Also, some bright minds inspired me during my stay at ETH, Dr Zhao Ma, Dr Yijiang Huang, Dr Patrick Ole Ohlbrock, Dr Giulia Boller, Davide Tanadini, Pok Yin Victor Leung, and many others. Moreover, I would like to express my gratitude to my outstanding students, Jingwen Wang, Wenjun Liu, and Victor Stolbovoy, for their dedication and inspiration.

I would like to also thank institutions and people, ETH Zurich, Department of Architecture, ITA Institute of Technology in Architecture and NCCR Digital Fabrication, for facilitating my interdisciplinary research environments.

Additionally, I want to thank Autodesk Research — Simulation, Optimization and System Group (SOS) for hosting me twice as a research intern in my last year of doctoral study. Specifically, I want to thank Dr Nigel Morris, Dr Pradeep Kumar Jayaraman for close mentorship, Dr Mehran Ebrahimi,

Dr Farhad Javid, and Dr Hyumin Cheong for collaboration.

I thank my current manager Dr Jan Frykestig for seeing my potential and welcoming me to join Ansys. Moving to Stockholm with my wife and knowing my next step gave me extra comfort in the final stage of wrapping up my thesis writing.

I want to thank my close friends for all their endless support. Dr Lucas Peng and Lenny Fan gave me a lot of inspiration in the field of physics and applied mathematics perspectives. Hojoong Chung, Kenryo Takahashi, Michael Chen, and Addison Ow Jia Wei, for all chats and discussions that brought me great joy.

I want to express my tremendous appreciation to my mentors and families in Taiwan. Especially, Prof. Dr Chen-Cheng Chen inspired me in many ways in the academy and daily life. Uncle Joshua Jih Pan and Aunt Bonnie Sun for encouragement in architecture and engineering study. My parents, Chien-Yao Kao and Hsien-Min Wu, for raising me and providing me with a great environment. I cannot be where I am today without their inspiration, care, and support.

Finally, I want to express my deepest thanks to my beloved wife, Bonnie Pei-Yu Chung and my forever dog Genie. They hold the most special place in my heart and have been invaluable sources of love and support. For my dear wife, Bonnie, words cannot adequately express my gratitude for her unconditional support, understanding, and love. I truly appreciate that she accompanied me from Taiwan to Germany and Switzerland. She has been continuously helping me become a better me and giving me a lot of courage. For our first dog and faithful companion, Genie, who has sadly passed away, it was one of my best decisions in life to adopt him. He had the best and funniest character, bringing so much joy and comfort into our home. He became my emotional support before I even realised it. I want to thank him for staying with us and not leaving us before my final doctoral defence. He was a true blessing and must have been sent from God to accompany us during my last year of doctoral study. I could not finish my doctoral journey without both of them.

Contents

Abstract

This dissertation presents a new numerical solver with an assembly-aware design workflow to help users better design and build discrete shell structures. Discrete shell structures are aesthetically ravishing but challenging to design and build efficiently. Although discrete structures can be prefabricated using efficient material and increasing productivity on-site, building discrete shell structures typically requires extensive scaffolding during their assembly process, which is associated with massive material and energy waste. Moreover, discrete shell structures usually require additional design steps normally not necessary with standard continuous structures, such as shape discretisation and stability analyses during and after construction.

The state-of-the-art stability solver rigid-block equilibrium (RBE) method uses quadratic programming with penalty formulation to measure structural infeasibility. For unstable structures, RBE provides a solution with a tensile force at the unstable regions, which gives a quantified measure of instability. Moreover, this dissertation extends the RBE method with some prominent features to thoroughly understand its optimisation results and mechanical meanings. Several experiments are performed and studied, including comparing the original quadratic with a linear function to illustrate nodal forces and interface resultants' roles and reducing the structure's contact interface into its kern area to explore different admissible internal stress states.

However, although RBE has various benefits that can be applied to the design problem, it incorrectly assesses unstable structures as stable when complex interface geometries are involved. Therefore, this dissertation provides insight into the RBE flaws, builds upon its strengths, and presents a new robust solver that overcomes the problem. Our newly proposed coupled rigid-block analysis (CRA) solver combines equilibrium and kinematics in a nonlinear programming problem. Furthermore, similar to RBE, CRA with penalty formulation makes it possible to measure infeasibility and gives valuable information to users to change their design. Comparing a wide range of benchmarks with commonly used commercial software shows CRA's robustness and accurate mechanical description of the complex three-dimensional discrete-element assemblies formed by rigid blocks.

Utilising CRA, this dissertation proposes a stability-aware design process to iteratively help the user and guide their design towards structurally-sound assemblies. Embedding stability-aware design process, an assembly-aware design (AAD) is further proposed to allow the user to consider the assembly sequence while designing discrete shell structures. Several computational and physical models were designed and assembled with limited scaffolding using our proposed workflow to verify our research findings in real-world scenarios. Finally, our CRA solver is publically available as an open-source Python package — COMPAS CRA — to help researchers and designers around the globe to build better discrete shell structures upon our work.

Keywords: 3D assembly, rigid blocks, equilibrium analysis, discrete element, concave shapes, friction, contact mechanics, computational physics, nonlinear optimisation, stability-aware design, assembly-aware design, discrete shell structures, computational fabrication.

Zusammenfassung

In dieser Dissertation wird ein neuer numerischer Löser eines montagegerechten Entwurfsablaufs vorgestellt, der BenutzerInnen unterstützt, diskrete Schalentragwerke besser zu entwerfen und zu erbauen. Diskrete Schalentragwerke sind ästhetisch reizvoll, jedoch schwierig zu entwerfen und effizient zu bauen. Diskrete Tragwerke können unter Verwendung effizienter Materialien vorgefertigt werden und erhöhen so die Produktivität auf der Baustelle. Jedoch erfordert der Montageprozesses diskreter Schalentragwerke in der Regel einen umfangreichen Gerüstbau, der mit einer massiven Material- und Energieverschwendung einher geht. Darüber hinaus erfordern diskrete Schalentragwerke in der Regel zusätzliche Entwurfsschritte, die bei herkömmlichen kontinuierlichen Tragwerken nicht erforderlich sind, wie beispielsweise die Diskretisierung der Form und Stabilitätsanalysen während und nach Abschluss des Bauprozesses.

Das hochmoderne Stabilitätslösungsverfahren Rigid-Block-Equilibrium (RBE) verwendet eine quadratische Programmierung mit Straffunktionen, um die strukturelle Undurchführbarkeit zu messen. Für instabile Tragstrukturen liefert die RBE-Methode eine Lösung mit Zugkraft in den instabilen Regionen, die ein quantifiziertes Maß für die Instabilität darstellt. Darüber hinaus wird in dieser Dissertation die RBE-Methode um einige bedeutende Merkmale erweitert, um ihre Optimierungsergebnisse und ihre mechanische Bedeutung gründlich zu verstehen. Experimente untersuchen den Vergleich der ursprünglichen quadratischen Funktion mit einer linearen Funktion um die Rolle der Knotenkräfte und der Schnittstellenresultierenden zu veranschaulichen. Des Weiteren wird die Reduzierung der Kontaktflächen des Tragwerks auf ihren Kernbereich untersucht, um verschiedene zulässige Eigenspannungszustände zu ergründen.

Obwohl RBE diverse Vorteile hat, die auf das Bemessungsproblem angewendet werden können, werden instabile Tragwerke bei komplexen Geometrien der Kontaktflächen fälschlicherweise als stabil eingestuft. Daher gibt diese Dissertation einen Einblick in die Schwachstellen der RBE-Methode, baut auf ihren Stärken auf und stellt einen neuen robusten Löser vor, der das Problem überwindet. Unser neu vorgeschlagener gekoppelter Coupled-Rigid-Block-Analysis (CRA)-Löser kombiniert Gleichgewicht und Kinematik in einem nichtlinearen Programmierproblem. Ähnlich wie bei RBE ermöglicht CRA mit einer Straffunktion die Messung der Nichtdurchführbarkeit und liefert BenutzerInnen wertvolle Informationen, um den Entwurf anzupassen. Der Vergleich einer breiten Palette von Benchmarks mit gängiger kommerzieller Software zeigt die Robustheit von CRA und die genaue mechanische Beschreibung eines komplex dreidimensionalen, aus diskreten starren Blöcken bestehenden Tragwerks.

Unter Verwendung von CRA wird in dieser Dissertation ein stabilitätsbewusster Entwurfsprozess vorgeschlagen, der BenutzerInnen iterativ unterstützt und zu strukturell soliden diskreten Tragwerken führt. Zur Einbettung des stabilitätsbewussten Entwurfsprozesses wird außerdem ein Assembly-Aware-Design (AAD) vorgeschlagen, der es BenutzerInnen ermöglicht, die Montagereihenfolge beim Entwurf diskreter Schalenstrukturen zu berücksichtigen. Mehrere rechnerische und physische Modelle wurden mit limitierten Gerüsten unter Verwendung unseres vorgeschlagenen Arbeitsablaufs entworfen und montiert, um unsere Forschungsergebnisse mit realen Szenarien zu verifizieren. Schlussendlich ist unser CRA-Löser als Open-Source-Python-Paket COMPAS CRA öffentlich verfügbar, um ForscherInnen und KonstrukteurInnen weltweit zu helfen, bessere diskrete Schalenstrukturen auf der Grundlage unserer Arbeit zu entwickeln.

Part I

Introduction

Chapter 1

Background

This dissertation develops a new structural solver and workflow for designing discrete shell structures. Section 1.1 introduces the thesis topic, explaining discrete-element assemblies and shell structures and how discrete shell structures can be helpful in architectural contexts. In Section 1.2, we point out the current inefficiencies of building shell structures and list two significant obstacles that cause the unproductive designing and building process of such structures. Section 1.3 explains the arguments for developing tools to build discrete shell structures efficiently. Lastly, Section 1.4 outlines the structure of this thesis to give the reader a better overview.

1.1 Introduction

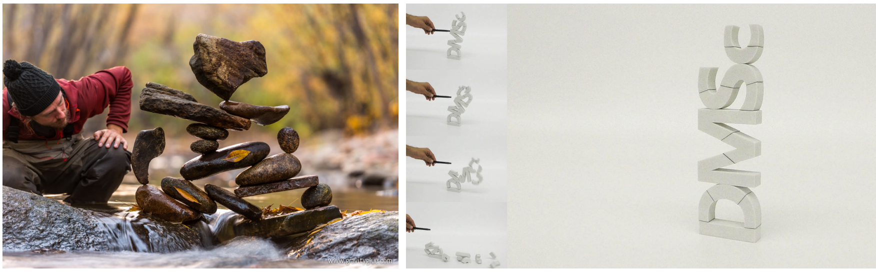
Discrete-element assemblies are structures formed by individual (rigid) units put together without glue or other joinery. The dimension of these assemblies can range from small products to buildings. Designing discrete-element assemblies that can stand under their self-weights has many applications in architectural contexts. For example, the unit size can be relatively small in some masonry structures but large in prefabricated housing. Besides architecture, discrete-element assemblies have various applications, such as designing furniture, 3D puzzles, toys, or even robotic assembly planning.

Sometimes, those assemblies can stand in a surprising configuration that is intuitively hard to believe stable. Figure 1.1 shows two surprising discrete-element assemblies that stand under their self-weight and rely on only contact and friction. A stone-balancing artist describes that achieving such stone balancing requires patience and the "feel" of the stone:

> Balance requires a minimum of THREE contact points. Luckily, every rock is covered in a variety of indentations that can act as a NATURAL TRIPOD for the rock to stand upright. When the rocks touch, you can feel vibrations or "clicks", as the surfaces move against one another... Achieving a challenging balance requires patience, and becoming fully in touch with NOW (where the vibrations live). You must get to know the rocks you are working with. Some rock characters will coordinate better with others, vice versa, back, forth, right, left, up, or down. The trick I've found is to PLAY and experiment. Start simple. Step by step, add complications if you feel.
>
> (Michael Grab, Gravity Glue)

Figure 1.1a shows the artist working on a stone equilibrium. He carefully puts one or two pieces of stone at a time, ensuring that all stones are stable. Instead of finding suitable stone shapes in nature, in Figure 1.1b, researchers use computational methods to precise design and 3D-print customised unique blocks to achieve a stable configuration. In order to push the limit and achieve some *unbelievable* or *extreme* stable configurations, whether designed physically or digitally, designers need to be able to "feel" the equilibrium of the assemblies from design to construction.

A shell is a type of structure with a small thickness compared to its other dimensions. Shell structures are structurally efficient and capable of carrying additional loads. Shell structures usually have long spans relative to their structural thicknesses, enabling extensive spatial design freedom. They provide various design flexibility in architectural and engineering practice and

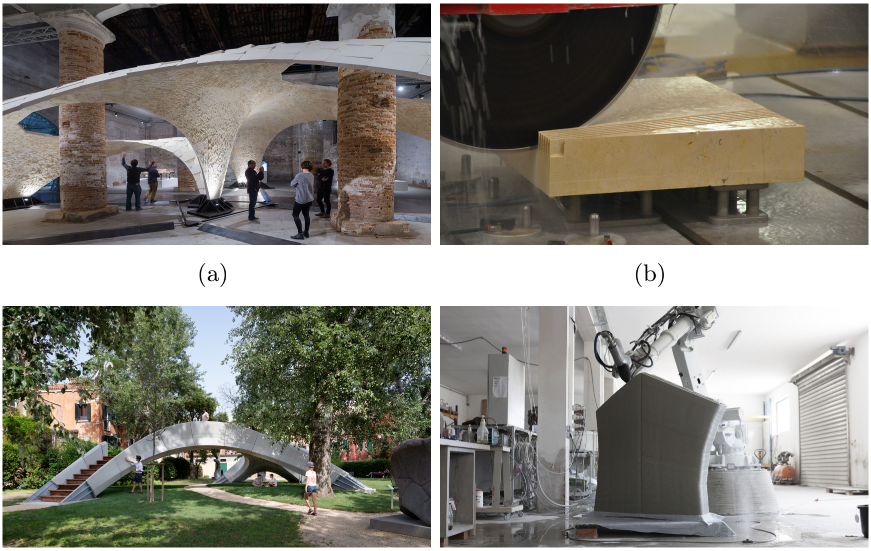


(a) (b)

Figure 1.1: Surprising discrete-element assemblies are stable in static equilibrium under self-weight and rely only on contact and friction: (a) Leaf Catcher: Stone Balance Art created and photographed by Michael Grab — 27 October 2016 — Boulder, Colorado, USA [Michael Grab, 2008]; (b) computationally designed and 3D-printed assembly by Frick et al. [2015].

have caught much research attention [Block et al., 2017b; Van Mele et al., 2022]. Funicular shell structures are a particular type of shell structures in which all stresses are either compressive or tensile under a defined loading condition [Rippmann, 2016].

In this research, we focus on discrete shell structures, which combine the terms of discrete-element assemblies and shell structures. In addition to their several structural and architectural design benefits, they are also convenient to fabricate, assemble, and disassemble. Historically, ancient masons have designed and constructed masonry structures for a long time. Those masonry structures are composed of discrete stones and many have been standing for thousands of years, showing their outstanding structural potential and architectural durability. Learning from the past, Figures 1.2a and b show an astonishing discrete shell design that comprised 399 customised cut limestones, standing without any mortar, reinforcements, or mechanical joints. In modern construction, building structures with prefab components vastly decreases the construction time and improves construction logistics [Li et al., 2014]. Figures 1.2c and d show the use of the latest robotic 3D-printing technology to produce prefab unreinforced-concrete components for discrete shell structures. However, although recent advances in fabrication and construction technologies, such as large-scale additive manufacturing and robotic fabrication, have created new opportunities to design, fabricate, and assemble customised architectural units with complex geometries, efficiently designing and building such discretised shell structures remains challenging.



(a)

(b)

(c)

(d)

Figure 1.2: (a) Armadillo: a discrete shell structure constructed in stone without any mechanical joints or reinforcements [Rippmann et al., 2016]; (b) one of the stones of the Armadillo vault being CNC cut; (c) Striatus: 3D concrete printed masonry bridge [Bhooshan et al., 2022]; (d) robotic 3D-printing of a component of the Striatus bridge [Bhooshan et al., 2022].

1.2 Motivation and challenges

This section describes this dissertation's main motivations and critical challenges. It tries to address reasons and missing parts that cause shell construction to be inefficient.

1.2.1 Construction inefficiency of shell structures

In architectural practice, shell structures, such as large-span roofs, are challenging to design and construct [Block et al., 2017c; Van Mele et al., 2022]. These structures require a lot of (temporary) foundations, scaffolding, formwork, and falsework during construction. Typically, the formwork for a concrete structure is estimated to be 35–60% of the construction cost [Lloret et al., 2015], and this number can go up to 75% for complex concrete structures [de Soto et al., 2018]. Besides, building a dense falsework after leads to a lot of wasted material, consumes a lot of time and complicates project logistics.

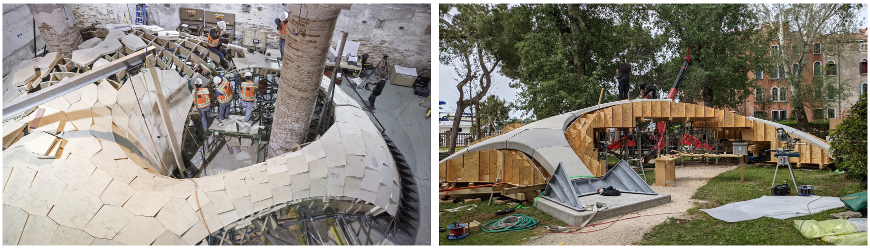

(a) (b)

Figure 1.3: The construction and assembly process of (a) Armadillo Vault [Block et al., 2017c] and (b) Striatus [Bhooshan et al., 2022] require dense falsework.

The efficient load-bearing behaviour of a shell structure results from its doubly-curved geometry, but this also makes it very difficult to construct them as stability during the assembly is not given (and typically not part of the design). The Armadillo vault and the Striatus bridge, both built in Venice (Figure 1.2), show efficient discrete shell structures with complex freeform surfaces, but they both required dense falsework during the assembly process, as can be seen in Figure 1.3. Rippmann et al. [2016] present the workflow of the Armadillo vault from design to assembly. Figure 1.4 shows that assembly is usually the last step in the workflow for such structures and is indeed not considered in the design cycle. As a result, the structure requires much falsework as it is only structurally sound when the whole structure is built, not during all intermediate stages. Moreover, typical structural analysis tools require designers to have structural domain knowledge with extensive experience, and it does not provide information for better design improvements.

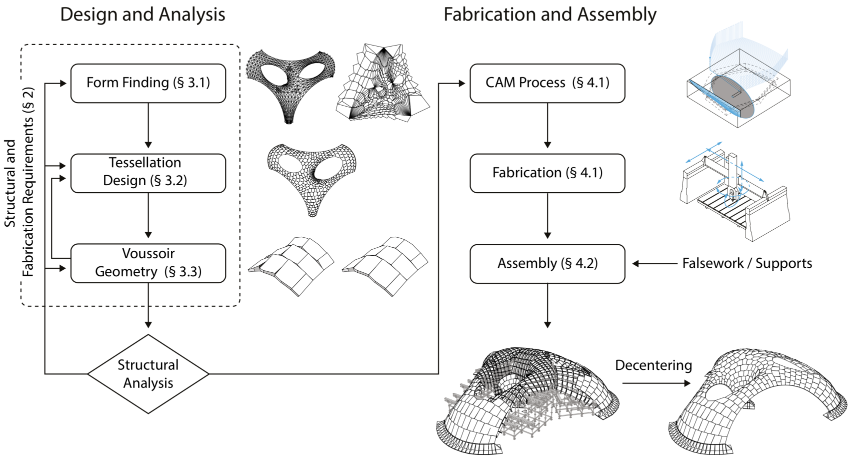


Figure 1.4: Armadillo workflow from design to assemblage (Rippmann et al. [2016])

1.2.2 Lack of tailored structural solvers

Although there are many software tools for structural design, a proper tool to design discretised equilibrium structures is lacking and assessing stability during the assembly process remains difficult. The main challenge in designing such structurally sound or scaffolding-free discrete-element assemblies is to interactively assess their stability at each design iteration and the modifications needed to redirect the initial design. Unfortunately, quick interactive tools are rarely used in engineering practice due to insufficient accuracy, while more accurate engineering software suffers from heavy computational effort and complicated parameter tuning. The generation and analysis of discrete assemblies can be cumbersome and time-consuming. On the geometry generation side, usually, discrete element solvers do not offer extensive parametric modelling capabilities and standard Computer-Aided Design (CAD) software do not offer a straightforward analysis toolbox. From an analysis point of view, engineering solvers typically require detailed mechanical parameter settings, often not known in the early design exploration. Therefore, there is a need for an integrated design and analysis tool for efficiently designing and accurately analysing discrete-element assemblies.

1.2.3 Lack of assembly in the design workflow

Design for Assembly (DFA) or Design for Manufacture and Assembly (DFMA) is a product design process that considers assembly and manufacturing constraints. DFA/DFMA is a common idea in mechanical engineering and is often used to reduce assembly time and costs. Figure 1.5 shows the typical steps of the DFMA workflow. By applying the method, the user can iteratively redesign the product to reduce assembly costs. Boothroyd [1994] shows that productivity is greatly increased by considering assembly at the design stage, even more effectively than by improving automation. However, despite the current development of digital fabrication and computational design, designers are only beginning to consider the manufacturing constraints in the design process and attempting to automate manufacturing processes. However, DFA or DFMA is still rarely used in architecture, engineering, fabrication and construction (AEFC). Recently, though, the concept has gotten some traction and is being introduced by the industry (e.g., Laing O'Rourke in the UK and Implenia in Switzerland) as an essential strategy to increase labour productivity and resulting profit margins.

Inspired by the DFA and DFMA processes, this thesis is motivated to propose an *assembly-aware design* process for *discrete shell structures*. We aim to help designers "feel" the equilibrium of the discrete-element assemblies for shell structures during the early design phase, improving how we design, build and reduce the time and cost of assembling discrete shell structures.

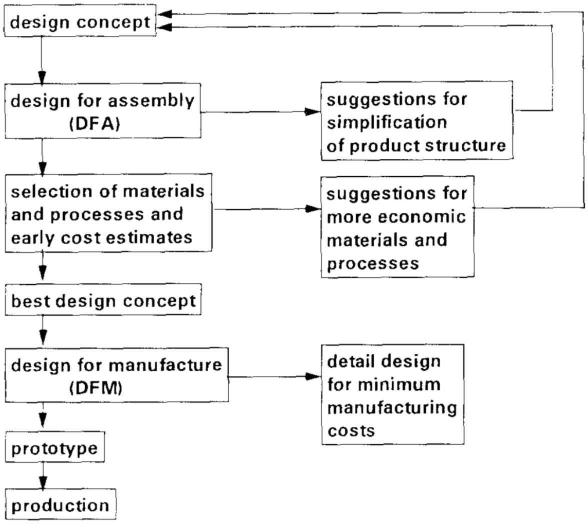
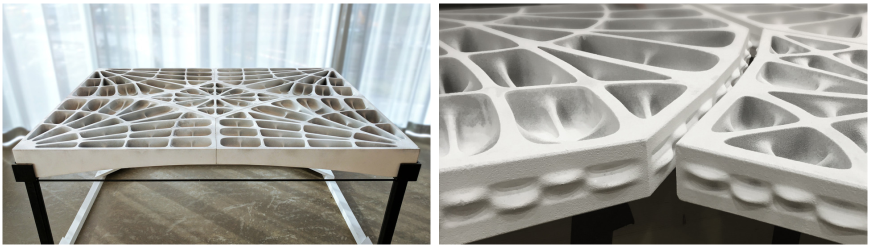

design concept
design for assembly (DFA)
suggestions for simplification of product structure
selection of materials and processes and early cost estimates
suggestions for more economic materials and processes
best design concept
design for manufacture (DFM)
detail design for minimum manufacturing costs
prototype
production

Figure 1.5: Typical workflow for Design for Manufacture and Assembly (DFMA) Boothroyd [1994]

1.3 Potential applications

Assembly-aware design of discrete shell structures can apply to many different contexts. Firstly, it can help to design and construct free-form shell structures more efficiently. Secondly, it can enable a new way to build self-supporting formwork for concrete casting. For example, the concrete shell structures designed by Félix Candela require much support and formwork for concrete casting [Barrallo and Sánchez-Beitia, 2011]. An easy-to-assemble and light-weighted (possibly stay-in-place) formwork can increase the productivity of building such structures. Thirdly, it can help design and assess for restoring traditional masonry structures (such as Gothic cathedrals or traditional vaults). Fourthly, it can help to build shell structures in developing countries with limited technology. For example, labour costs are low in such contexts, but materials and expensive technology are not affordable; learning from traditional formwork-less masonry vaulting techniques can help designers better understand design constraints in various contexts. Fifthly, it can be used to design and efficiently assemble discrete structures, such as the funicular concrete floor system in Figure 1.6, to reduce environmental impact and embodied carbon emissions. Last but not least, the assembly-aware design of discrete shell structures can be applied to constructing large-span structures that can be self-supporting during assembly. Potentially, it can be integrated with other robotic feedback systems to have autonomous construction, see Figure 1.7.



(a) (b)

Figure 1.6: 3D-printed funicular floor system from Block Research Group [Rippmann et al., 2018; Ranaudo et al., 2021].

1.4 Thesis structure and outline

Some of the contents and results in this thesis are based on the following first-author journal publications by the author during his PhD studies:

- Understanding the rigid-block equilibrium method by way of mathematical programming [Kao et al., 2021]
- Coupled Rigid-Block Analysis: Stability-Aware Design of Complex Discrete-Element Assemblies [Kao et al., 2022]

This dissertation consists of five parts. The first part, "Introduction", lays out the research topics, motivations, challenges, research objectives, and reviews relevant research. The second part, "Equilibrium solvers", presents

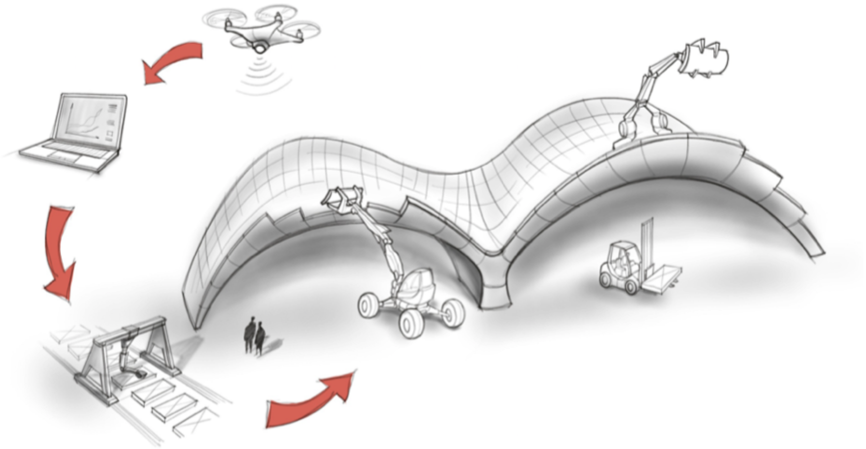


Figure 1.7: Building large-span, self-supporting structure, using autonomous robots (vision sketch from NCCR DFAB research stream 1B 'Large-span self-supporting assemblies').

the theoretical background and mathematical foundation of a robust method for solving static equilibrium. The third part, "Applications", demonstrates the use of our method for different kinds of applications. In the fourth part, "Implementation", our open-source code implementation is presented. Finally, the fifth part, "Conclusion", summarises our work and identifies some further research directions.

Part I: Introduction

Chapter 1: Background

This chapter introduces the research topic, shows the motivations, discusses the current research challenges and shows potential applications. It intends to give the readers a thorough explanation of the research topics: assembly-aware design and discrete shell structures.

Chapter 2: Literature review

This chapter reviews the relevant works to the topic. We begin with the reviews of discrete shell structures. We look at ancient masonry, modern prefabrication constructions, and discretisations for shell structures. After that, we look deep into different structural analysis methods and all kinds of scaffolding-free methods. Lastly, we review possible assembly strategies for assembly shell structures.

Chapter 3: Scope of work

Based on the literature review, this chapter defines the thesis scope. We list the research statements and highlight the research objectives.

Part II: Equilibrium solvers

Chapter 4: Rigid-Block Equilibrium (RBE) method

This chapter aims to review, develop, and extend the state-of-the-art Rigid-Block Equilibrium (RBE) method. We first go through the mathematical details of the problem in depth with a consistent notation. Then we introduce the numerical procedure to calculate the kern formulation in a general polygon. After that, we explain the calculation of the resultant to prepare the reader with tools to understand better the optimisation results. In the end, we utilise a graph-based data structure to store our assembly information.

Chapter 5: Coupled Rigid-Block Analysis (CRA) method

This chapter formulates a new nonlinear optimisation Coupled Rigid-Block Analysis (CRA) method. We first show two simple 2D examples to highlight RBE in specific scenarios that wrongly predict the equilibrium. Therefore, we build CRA upon the RBE method's strengths, appending additional constraints to enhance the problem formulations. We also extend the existing graph-based data structure to blocks with freeform interfaces.

Part III: Applications

Chapter 6: Stability assessment

In this chapter, we perform several experiments and benchmarks RBE and our newly-developed CRA methods. After that, we showcase several assessment examples and propose a stability-aware design workflow.

Chapter 7: Assembly-Aware Design (AAD)

This chapter proposes the assembly-aware design workflow. Firstly, we use a simple example to demonstrate the process of designing scaffolding-free assembly and formalise the workflow by demonstrating some examples with step-by-step analysis. Moreover, we describe the necessary steps to create discrete shell structures, propose a possible assembling strategy, and analyse stability during assembly. Finally, we perform several examples to demonstrate the workflow.

Part IV: Implementation

Chapter 8: COMPAS CRA

This chapter demonstrates our open-source software implementation COMPAS CRA. We introduce the software package's code and online documentation structure. In our implementation, we include several solvers and geometry calculations as we showed in the Part II: Equilibrium solvers. Additionally, we also include many examples as we demonstrate in Part III: Applications.

Part V: Conclusion

Chapter 9: Conclusion

In the last chapter, we summarise all findings and list the main contributions of this thesis. In the end, we also discuss some limitations and possible future work.

Chapter 2

Literature review

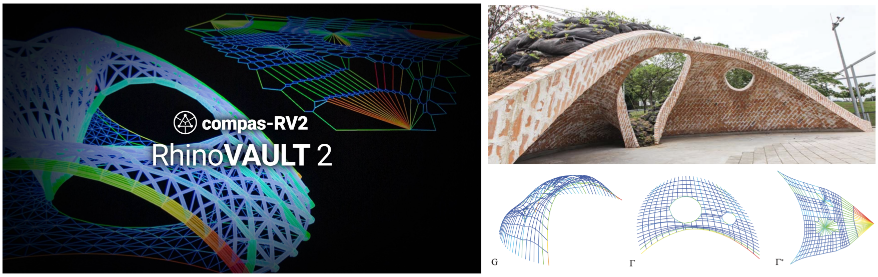
This dissertation focuses on the discrete shell structures and the assembly-aware design process for such structures using as less scaffolding as possible. This chapter provides an in-depth overview of relevant literature to our research topics. Section 2.1 reviews the discrete shell structures from their geometrical and structural aspects. Masonry structures and some prefabricated projects using computational methods are listed. Afterwards, Section 2.2 gives an in-depth review of various structural analysis methods for assessing complex assembles. Several methods are covered, including the finite element, physics-based simulation, discrete element, and rigid-block equilibrium methods. Section 2.3 studies methods to reduce construction scaffolding from traditional and modern techniques. Finally, Section 2.4 discusses some designs for assembly works related to this dissertation.

2.1 Discretised shell structures

Discrete shell structures are particular types of shells whose surfaces are discretised with smaller units. In what follows, we look into discrete masonry and self-supporting structures. We also review some discretisation methods and prefabrication works that particularly relate to the shell structures.

2.1.1 Masonry and self-supporting structures

Self-supporting structures are discrete-element assemblies that can stably stand under their self-weight. In particular, much effort has been devoted to exploring freeform self-supporting shapes such as masonry structures [Vouga et al., 2012; Panozzo et al., 2013; Deuss et al., 2014]. Much research is based on Thrust Network Analysis (TNA), an equilibrium approach developed by Block and Ochsendorf [2007]. In the architectural design and fabrication community, RhinoVault, an interactive design tool, has been popular and broadly used for designing funicular structures [Rippmann et al., 2012; BlockResearchGroup, 2020], see Figure 2.1. Rippmann [2016] thoroughly studied funicular form-finding of self-supporting shapes. However, TNA applies specifically to shell structures. It is not the appropriate tool for designing general discrete-element assemblies. Along with the TNA method, Iannuzzo et al. [2021c] provided a Python-based computational framework with several solvers, which will be reviewed in Section 2.2, for assessing unreinforced masonry structures.



(a)

(b)

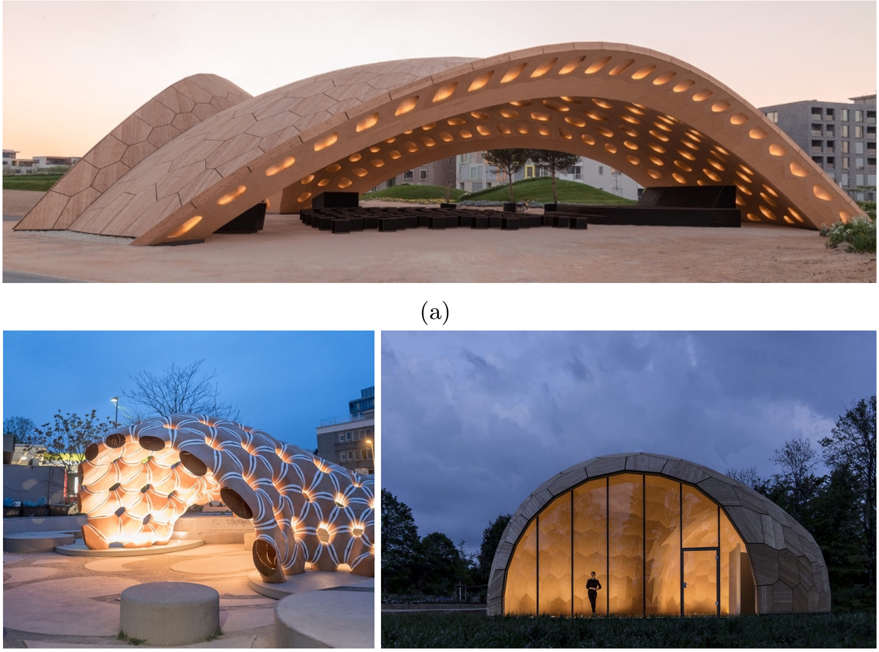
Figure 2.1: (a) RhinoVAULT 2 is an open-source tool that can help users perform funicular form-finding and create self-supporting structures [Rippmann et al., 2012; BlockResearchGroup, 2020]. (b) ColomBrick Thintile Vault by López et al. [2014] was designed using RhinoVAULT (image from Rippmann [2016] and Sergio González).

2.1.2 Tessellation and prefabrication of shell structures

Tessellation is essential in mathematics, geometry, art, and architecture. It covers the surface with smaller units with one or more geometrical shapes and without gaps between them. The tessellation of free-form architecture has been an important research topic for aesthetic and economic reasons. Proper panelisation of architectural elements, i.e., tessellating the roof of a building with proper panels, such as designing planar panels with similar sizes or repeated dimensions, can significantly reduce fabrication costs [Pottmann and Wallner, 2008; Pottmann et al., 2008; Eigensatz et al., 2010]. Moreover, much research in graphics has been devoted to studying and optimising constraint meshes that can be applied more intuitively in the design process and consider fabrication constraints [Yang et al., 2011; Bouaziz et al., 2012, 2014; Deuss et al., 2015].

In recent years, digital fabrication has assisted the development of robotic-assisted additive manufacturing in building construction. 3D printing vastly increases custom component design possibility and the complexity of prefabricated structures [Mitropoulou et al., 2020]. Bhooshan et al. [2022] studied the non-conventional 3D-printed method for the component of the Striatus bridge, see Figures 1.2c and d.

In engineering practice, much research investigates plate shell structures. It takes both tessellation and prefabrication constraints into account. e.g., Tangent Plane Intersection (TPI), the biomimetic principles of natural plate structures for finding the planar plate. The Agent-Based Modelling (ABM) approach to finding tessellation with plate geometry fulfilling fabrication constraints. [Li and Knippers, 2015; Krieg et al., 2015; Sonntag et al., 2017; Bechert et al., 2021], see Figure 2.2.



(a)

(b)

(c)

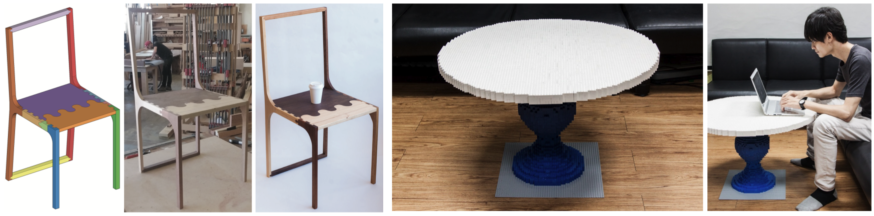
Figure 2.2: (a) BUGA Wood Pavilion 2019, photograph by ICD/ITKE University Stuttgart [Bechert et al., 2021]. (b) ICD/ITKE Research Pavilion 2015–16, photographs by ICD/ITKE University Stuttgart [Sonntag et al., 2017]. (c) Landesgartenschau Exhibition Hall, photographs by ICD/ITKE/IIGS University Stuttgart [Li and Knippers, 2015; Krieg et al., 2015].

2.2 Structural equilibrium of complex assemblies

The study of structural equilibrium and complex assemblies has been a core topic in many research fields, especially in computer graphics, robotics, architecture, and structural mechanics. In what follows, we mainly focus on some research topics relevant to our goal.

2.2.1 Complex assemblies

The study of complex assemblies is also a popular research topic and much effort has been paid to designing and creating different types of objects using smaller units, such as toys, furniture, mechanisms, and architectures. Such complex shapes are in some cases designed from a pure geometrical perspective [Fu et al., 2015; Tai, 2012; Wang et al., 2018; Schwartzburg and Pauly, 2013; Testuz et al., 2013; Song et al., 2016], while, in other cases, the mechanical stability is also taken into account. Frick et al. [2016] introduced a graph-based data structure to handle complex assemblies for stability analysis considering imperfections, but it requires convex decomposition for concave shapes, see Figure 2.4a. Yao et al. [2017] provided an interactive process to design joinery for furniture and consider structural stability at the same time, see Figure 2.3a. However, their analysis result is over-conservative and does not align well with well-accepted analytic or other numerical solutions. Looking at structurally-informed LEGO design, Luo et al. [2015] presented a force-based approach that requires physical experiments to tune the optimisation, see Figure 2.3b.



(a)

(b)

Figure 2.3: Embedding structural equilibrium analysis of complex assembly in the design tool enables various applications, including furniture design: (a) analysed and fabricated chair joinery design from Yao et al. [2017]; and, (b) the LEGO table was structurally designed and fabricated by Luo et al. [2015].

2.2.2 Finite element method

The classic Finite Element Method (FEM) is the most commonly used method for analysing structures in many engineering fields. However, much research shows that the standard use of FEM approaches does not provide accurate results when assessing unilateral structures [Block et al., 2006a; DeJong, 2009; Whiting, 2012; Shin et al., 2016]. In recent works, FEM methods have been used to assess unilateral assembly modelling each block as a distinct element [Papadopoulos, 2021]. Nonetheless, they can only provide a *yes-no* answer and, thus, cannot be embedded into other optimisation processes such as gradient-based optimisation formulations.

2.2.3 Physics-based simulation

Physics-based simulations and robot grasp planning have been studied intensively [Baraff, 1991, 1993, 1994; Mattikalli et al., 1995, 1996; Guendelman et al., 2003; Erleben, 2007; Stewart and Trinkle, 1996]. However, most research is devoted to giving a visually convincing rather than a physically accurate result. Contact problems with friction have been recognised as challenging problems [Mason and Wang, 1988]. Much research points out that finding an equilibrium solution of unknown contact forces is a necessary but not sufficient condition [Palmer, 1989; Pang and Trinkle, 2000; Mosemann et al., 1997], and determining the stability is co-NP complete [Mattikalli et al., 1996]. Kaufman et al. [2008] proposed a staggered algorithm to simulate rigid-body dynamics to realistically model the sliding behaviour using the Maximum Dissipation Principle (MDP) [Goyal et al., 1991]. We, however, are interested in the static analysis as it is easier to steer the structural design. Haas-Heger and Ciocarlie [2020] gave good insights into coupling kinematics and static equilibrium problems in the optimisation constraints using MDP. However, their approach is based on Mixed-Integer Programming (MIP) to distinguish between sliding and resting frictions and requires a long computational time and does not apply to general complex assembly problems with many elements.

2.2.4 Discrete element method

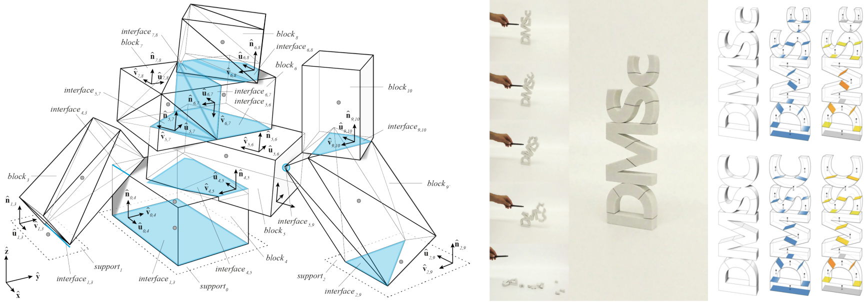
Masonry structures have been standing for thousands of years, and they were designed and constructed by ancient master builders without the aid of modern engineering methods and technologies. Much research has developed computational methods to understand its structural mechanics. Cundall [1971] developed Discrete Element Modelling (DEM) to analyse structures composed of discrete particles for granular materials, which was later used in the software code for 3DEC [Itasca, 2013]. Starting from the work of Lemos [1995], it became a popular tool to assess masonry structures or to benchmark new methodologies [Lemos, 1995; DeJong, 2009; Iannuzzo et al., 2021b; Dell'Endice et al., 2021; Bui et al., 2017; Sarhosis et al., 2016, 2014]. However, DEM solvers require detailed mechanical parameter tunings and a long computational time. Besides, simulating complex concave shapes requires an additional convex decomposition step. In general, historical masonry structures are not modelled using complex block shapes.

2.2.5 Rigid-block equilibrium method

The use of the Safe Theorem of Limit Analysis, as proved by Heyman [1966], is a widespread approach to assessing masonry structures. It states that a structure is safe if an admissible stress state can be found in equilibrium with the external loads and lying within the structural domain [Heyman, 1966; Ochsendorf, 2002; Huerta, 2006a,b; Como, 2013]. Moreover, Limit Analysis is also a powerful method when the aim is to assess statically indeterminate structures and, thus, to explore the infinite set of admissible, internal stress states [Como, 2013]. Many strategies apply such theorem computationally to explore different equilibrium solutions. In particular, Fraternali et al. [2002], Angelillo and Fortunato [2004], Block et al. [2006a], Block and Ochsendorf [2007], Fraternali [2010], Block and Lachauer [2014], De Chiara et al. [2019], Gesualdo et al. [2019], D'Ayala and Casapulla [2001], and Mousavian and Casapulla [2020a] modelled the structures as a continuum, while in Livesley [1978], Livesley [1992], Gilbert and Melbourne [1994], Orduña and Lourenço [2005], Gilbert et al. [2006], Portioli et al. [2014], and Portioli et al. [2015] modelled the structure as an assembly of rigid blocks having a finite friction capacity.

The Rigid-block Equilibrium (RBE) method was developed by Whiting et al. [2009, 2012], based on the work of Livesley [1978, 1992], who proposed a formulation based on linear programming to analyse the limit load of masonry structures. Whiting et al. [2009, 2012] added a penalty term to Livesley's formulation that measures structural instability such as to enable forward design. RBE has since been successfully applied to a wide range of problems, including forward-design tools and shape optimisation [Deuss et al., 2014; Wang et al., 2019, 2021a; Shao et al., 2014; Hsu and Keyser, 2012; Beyeler et al., 2015; Frick et al., 2015; Mousavian and Casapulla, 2020b; Bhooshan et al., 2018]. In particular, Frick et al. [2015] proposed its use for designing discrete-element assemblies in an interactive CAD environment, see Figure 2.4b. Shin et al. [2016] gave a thorough comparison between RBE and FEM approaches. Wang et al. [2019, 2021a] formulated a shape optimisation of topological interlocking for convex assemblies based on an RBE stability analysis without using friction. Wang et al. [2021b] stated that RBE represents one of the best tools in the design process.

In general, the RBE approach is preferable over FEM, DEM, or other dynamic simulations because it can more intuitively guide the design from unstable toward stable configurations. RBE's strength is its penalty formulation and the possibility of looking at the internal stress state to understanding the structural response. The penalty formulation provides the users with the localisation of unstable regions instead of only collapse mechanisms that cannot give meaningful structural hints for complex structures. Therefore, it enables different structural design options, such as changing discretisation or connection design (i.e., geometry, shape or using rebar). Although RBE is widely used to design stable assemblies, it suffers from its limitations when friction is involved. We will review RBE formulation in detail in Chapter 4 and address its limitation in Chapte 5.


interface $_{7,8}$
block $_7$
block $_8$
interface $_{6,8}$
block $_6$
interface $_{5,7}$
interface $_{6,7}$
interface $_{5,6}$
interface $_{4,3}$
block $_{10}$
interface $_{9,10}$
block $_3$
block $_9$
interface $_{5,9}$
block $_5$
block $_4$
support $_1$
interface $_{1,3}$
interface $_{1,3}$
interface $_{4,5}$
support $_0$
support $_2$
interface $_{2,9}$
$\hat{n}_{1,3}$ $\hat{u}_{1,3}$ $\hat{v}_{1,3}$
$\hat{n}_{0,4}$ $\hat{u}_{0,4}$ $\hat{v}_{0,4}$
$\hat{n}_{4,5}$ $\hat{u}_{4,5}$ $\hat{v}_{4,5}$
$\hat{n}_{5,6}$ $\hat{u}_{5,6}$ $\hat{v}_{5,6}$
$\hat{n}_{5,7}$ $\hat{u}_{5,7}$ $\hat{v}_{5,7}$
$\hat{n}_{6,7}$ $\hat{u}_{6,7}$ $\hat{v}_{6,7}$
$\hat{n}_{6,8}$ $\hat{u}_{6,8}$ $\hat{v}_{6,8}$
$\hat{n}_{7,8}$ $\hat{u}_{7,8}$ $\hat{v}_{7,8}$
$\hat{n}_{9,10}$ $\hat{u}_{9,10}$ $\hat{v}_{9,10}$
$\hat{n}_{2,9}$ $\hat{u}_{2,9}$ $\hat{v}_{2,9}$
$\hat{z}$ $\hat{y}$ $\hat{x}$
DMSc

(a)

(b)

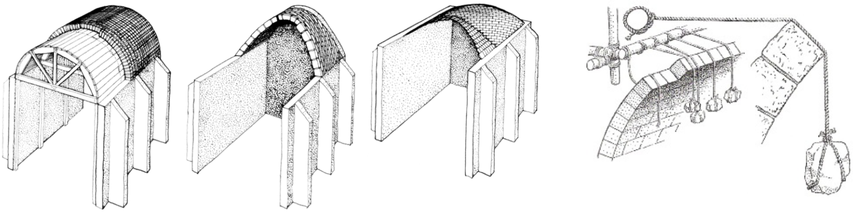
Figure 2.4: (a) Assembly data structure proposed in Frick et al. [2016] to handle complex assembly geometrical information; (b) Surprising design of discrete-element assembly using the Rigid-block Equilibrium (RBE) method, image from Frick et al. [2015].

2.3 Scaffolding-free constructions

Shell or vault structures typically require dense scaffolding during construction. Intensive falsework or formwork leads to a large amount of material waste. This section reviews traditional techniques from ancient masons and modern construction techniques to reduce scaffolding.

2.3.1 Traditional techniques to reduce falsework

Although typical traditional vaulting requires intensive wooden frame structure as falsework, in Figure 2.5a left, traditional masons have developed several methods to build shell and vault structures with limited support. Pitched-brick or Nubian vaulting (Figure 2.5a middle, [Allen and Zalewski, 2009]) is an ancient technique that allows structure erection with regular brick and mortar without centring. It utilises incline masonry courses as stable sections to support blocks during construction. Mexican vaulting technique extended the range of vaulted geometries. However, their design space is highly constrained [Ramírez Ponce and Ramírez Melendez, 2015].



(a) (b)

Figure 2.5: (a) from left to right: traditional vaulting, pitched-brick/Nubian vaulting, tile/Catalan vaulting (Allen and Zalewski [2009]). (b) Temporary supporting using strings (Fitchen [1981]).

On the other hand, tile or Catalan vaulting offers more formal flexibility. However, it requires fast-setting mortars to provide the local bending capacity to temporarily hold the lightweight brick until the next stable section/arch during the construction (Figure 2.5a, right, [Ochsendorf, 2010]). Traditional vault construction can be constructed without formwork. Wendland [2005, 2007] studied the most common half-stone vault, discussing how the geometry of the masonry courses and traditional building techniques could help with scaffolding-free vault construction. Furthermore, Fitchen [1981] reviewed an ancient Gothic technique using stone-weighted rope to hold voussoirs in place temporarily, see Figure 2.5b. Although those traditional techniques allow spanning space with a limited amount of support from below, they are limited to specific geometry, certain types of brick units and mortars, additional stone-weighted ropes, or expert masonry skills to be able to construct the structure stably.

2.3.2 Modern techniques to reduce falsework

Rippmann [2016] thoroughly reviewed some modern techniques for reducing temporary scaffolding and formwork. This subsection summarises some of his findings.

Prefabricated shells constructed in the 1950s–1960s typically require tem-

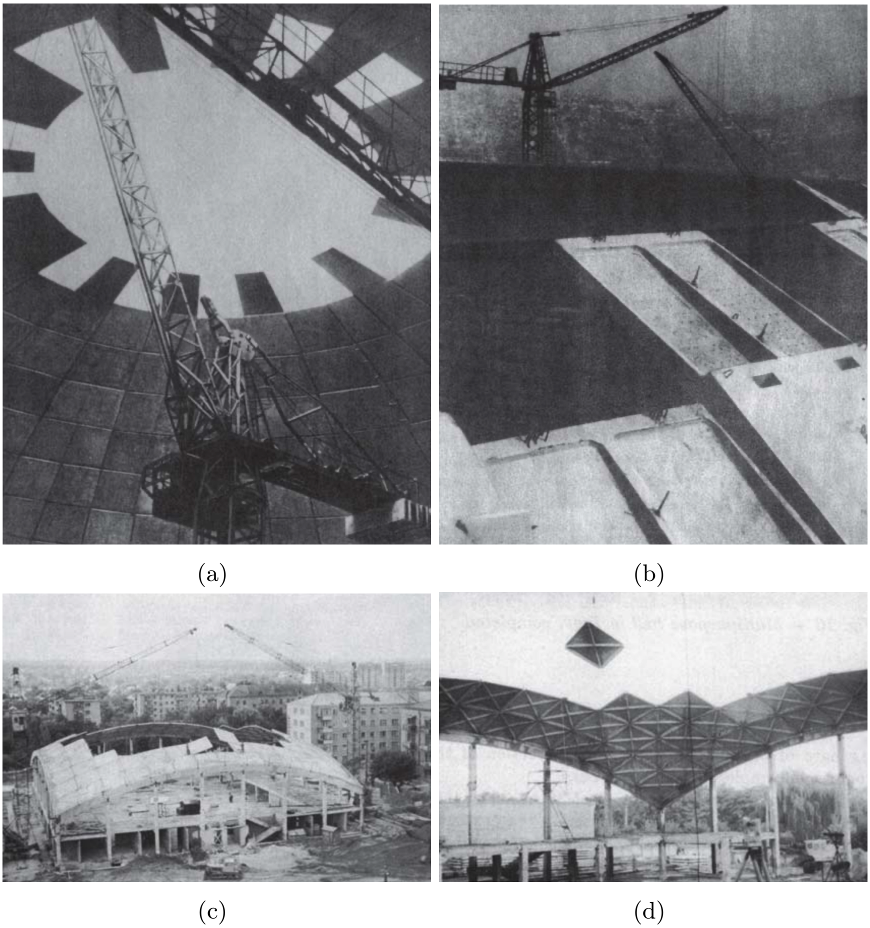


Figure 2.6: Formwork-free and scaffolding-free prefabrication construction, images from Rippmann [2016]: (a,b) constructing the Sport Palace in Tbilisi, Georgia (1961) using an overhang method and its prefabricated element in detail [Kadzhaya, 1966]; and, (c,d) the construction process of a market roof in Belaya Zerkov, Ukraine (1978) using a cantilevering method [Kaplunovich and Meyer, 1982].

porary scaffolding. They use post-tensioned precast elements [Matthews, 1955] or as permanent formwork and later form a structural bond by filling it with cast-in-place concrete [Nervi, 1953]. On the contrary, prefabricated shells constructed in the former Soviet Union in the 1950s–1970s use an overhang and cantilever method to reduce intensive scaffolding to support the cast-in-place element during construction [Sanchez-Arcas, 1961]. Figure 2.6 shows that the two projects' prefabricated units were installed without any scaffolding. The previously installed units temporarily support the current cantilever units with interlocking mechanisms and boltings. Strategically, two projects were built up from the outer support towards the centre, which can be seen as a shell structure with a massive opening in the centre.

2.4 Design for assembly

Design for assembly (DFA) is commonly used in the manufacturing industries to consider assembly in the design process. This section reviews some assembly sequence planning methods and assembly-aware designs related to shell structures.

2.4.1 Assembly sequence planning

Assembly and disassembly sequences are closely related. Much research uses disassembly to predict assembly sequences [Laperriere and ElMaraghy, 1992; Boothroyd and Alting, 1992; Boothroyd, 1994; Yokota and Brough, 1992; Goda and Davis, 2003; Ghandi and Masehian, 2015]. Although it is a research area that has been studied extensively, instead of considering the stability of dis/assembly, most research in architecture focuses on the correct assembly direction or sequence planning [Tai, 2012; Wang et al., 2018]. Beyeler et al. [2015] proposed a heuristics-based approach to find a sequence to deconstruct a pile of object step by step without compromising the overall structure's stability. However, the heuristics algorithm proposed by Beyeler et al. [2015] requires a long computational time that only applies to small-scale problems.

2.4.2 Assembly-aware design

Assembly-aware design or stability during assembly is a new research area that has received much attention recently. Kao et al. [2017] proposed a design process with a heuristic strategy to assemble discrete shell structures stably without using falsework, see Figure 2.7. They proposed an efficient way to compute a feasible assembly sequence based on a stable disassembly sequence. They utilised an existing game engine as a stability analysis tool, which is not guaranteed to be reliable.

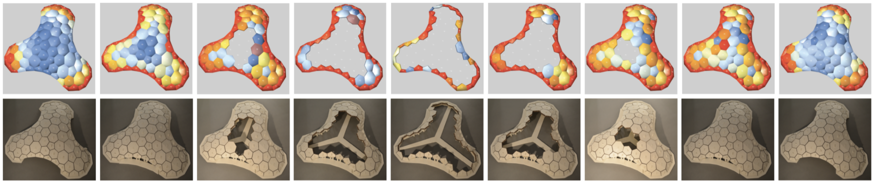


Figure 2.7: Assembly sequence prediction through disassembly: (top) simulated models using a physics-based game engine; and, (bottom) 3D-printed physical model Kao et al. [2017].

Learning from traditional masonry techniques described in Figure 2.5b, inspired by DREW [2013], and utilising RBE formulation, Deuss et al. [2014] proposed using a sparse set of tensile chains in replacing dense supports. A more recent study showed the potentiality of using three cooperative robots to achieve scaffold-free construction [Bruun et al., 2021].

Chapter 3

Scope of work

This dissertation aims to provide designers with a robust tool to assess and guide the structural design of discrete shells.

Based on the background and literature review in the previous Chapters 1 and 2 respectively, we clarify the research statements in Section 3.1 and list our research objectives in Section 3.2.

3.1 Problem statements

Essentially, this dissertation aims to answer the following question:

- **How to improve how we design and build discrete shell structures?**

As shown in the background review and literature studies, this question has many aspects. Thus, the previous question can be decomposed into the following:

- **How to improve the RBE method and assess discrete-element assemblies correctly and efficiently?**

  The Rigid-Block Equilibrium (RBE) method is a state-of-the-art method that can guide designers to improve their design of discrete-element assemblies. However, Yao et al. [2017] pointed out that RBE fails to give a reasonable solution when sliding occurs. To improve it and make it more robust, we need to understand better its optimisation result related to mechanical behaviour.

- **How to better assess the stability of assemblies with complex geometry?**

  The existing graph-based data structure for storing equilibrium calculations requires convex decomposition of concave shapes, which requires tedious manual steps or an additional tetrahedralisation algorithm step. Therefore, we aim to find a better and easier strategy for analysing assemblies with complex shapes.

- **Can the shapes of elements improve the stability during assembly?**

  Different shapes and configurations can affect the assembly's stability, so we aim to investigate block shapes that have better stability during assemblage and aid/allow scaffold-free assembly constructions.

- **What assembly strategies are adequate for the discrete shell structures?**

  Since both assembly shape and sequencing affect how discrete shell structures can be assembled, we aim to find one of the feasible assembly sequences to construct the structure efficiently.

3.2 Research objectives

From the research statement, we list the following research objectives for this dissertation:

- **Better understanding of the mechanical behaviour of RBE results**

  The RBE method uses quadratic programming to obtain one of the equilibrium solutions. However, that particular solution is incorrect in some scenarios when sliding occurs. Therefore, this dissertation investigates some methods for better understanding RBE results' mechanical behaviour.

- **New robust structural solver**

  A new robust solver fixes the existing RBE solver's flaws and provides users with quick, accurate, and helpful design feedback.

- **Benchmark solver results**

  This research plans to compare the theoretical solver results with commercial solvers and validate them with physical models.

- **Improve assembly data structure**

  Improve assembly data structure that enables handing assemblies with complex block shapes.

- **Study of geometry and stability**

  Block shapes play an essential role for the stability during assembly, so we implement studies of historical stereotomy principles to examine the relation between block geometry and stability.

- **Efficient construction sequencing strategy**

  We aim to propose assembly strategies that can reduce scaffolding during assembly. It is worth noting that our tool may only offer a particular solution that follows what we learn from masonry logic or historical contexts and that the solution may not be the only or the optimal solution.

- **Workflow for efficient design and construction of discrete shell structures**

  A new assembly-aware design workflow should be proposed specifically for designing and assembling discrete shell structures.

- **Open-source software implementation**

  Implementation of our research contributions and our new structural solver will be available to the public. Additionally, thorough documentation will be published online for practical referencing and usage.

Part II

Equilibrium solvers

Chapter 4

Rigid-Block Equilibrium (RBE) method

This chapter aims to review, further develop, and extend some of the main features of the rigid-block equilibrium (RBE) method, and interpret its mathematical optimisation results. RBE is a numerical approach that frames the equilibrium problem of rigid-block assemblies as an optimisation problem to compute possible internal and equilibrated singular stress states. The contact between blocks is considered to have a finite friction capacity and the unilateral behaviour is modelled through a penalty formulation. In particular, the penalty formulation widens the standard admissible solution space of compressive-only forces by allowing for tensile forces to appear in potentially unstable regions. The RBE objective function minimises the interface forces while the constraints are linear functions enforcing the static equilibrium of the whole assembly.

In Section 4.1, we first introduce what RBE is and its history of development. In Section 4.2, we go through the mathematical details of RBE. We define the problem using a consistent notation and go through the constraints and objective functions step by step to build a foundation for understanding mechanical behaviours using mathematical programming. In Section 4.3, we summarise the numerical approach to define the kern of generic and planar polygonal contact interfaces, which can be used to explore *safer* internal stress states. Additionally, in Section 4.4, we provide a procedure to post-process the results of the virtual nodal forces and reduce them to their resultants on the contact interfaces, which can be used to have a more intuitive structural understanding of the RBE results. After that, we review the assembly data structure in Section 4.5 and explain how to include all block and contact information in a graph-based data structure computationally. Finally, we summarise RBE's pros and cons in Section 4.6.

4.1 Introduction

The Rigid-Block Equilibrium (RBE) method frames the Equilibrium Problem (EP) [Angelillo et al., 2018] as an optimisation problem that minimises the total amount of contact forces, having equilibrium relations and friction conditions as constraints. It is based on a formulation that was first proposed by Livesley [1978, 1992] and later extended by Whiting et al. [2009, 2012] and Frick et al. [2015, 2016].

Livesley [1978] proposes a mathematical model that adapts the analysis of rigid-plastic structural frames for finding the limit load of masonry structures formed by rigid blocks. In particular, the solution is obtained as the result of an optimisation problem, in which the load factor is maximised subject to the linear equilibrium constraints. Later on, Livesley [1992] extends his previous work to three-dimensional masonry structures and uses the lower-bound approach to handle the collapse mechanisms that involve sliding, hinging, and twisting. In his work, masonry structures are modelled as discrete elements that connect with planar quadrilateral interfaces and have a finite friction capacity, Furthermore, he assumes that two surfaces in contact are slightly concave, and thus only interact through forces at the corners.

Whiting et al. [2009, 2012] extend the method by including penalty forces to enlarge the solution space by allowing infeasible solutions to virtually provide the tension capacity of an interface as imaginary glue.

Frick et al. [2015] review the work from Whiting et al. [2009, 2012] and propose some visualisation methods to design discrete-element assemblies more intuitively. Finally, Frick et al. [2016] propose a computational method that enables the calculation of arbitrarily-placed assemblies with polygonal planar interfaces using a combined graph and mesh data structure. Their method allows "general" contacts, e.g., partial overlaps of faces between blocks, which were not possible previously in Livesley [1978] and Whiting et al. [2009, 2012].

4.2 Theoretical background

To provide the reader with a sufficient amount of information to go through our method in Chapter 5, we provide the state-of-the-art Rigid-Block Equilibrium (RBE) method in detail and define the problem with consistent notations.

4.2.1 Assembly, blocks, and contact interfaces

In Figure 4.1, an assembly $\mathcal{A}$ is composed of a finite set of blocks $\mathcal{B}$ and contact interfaces $\mathcal{C}$. Each block is a compact closed subset of $\mathbb{R}^3$ whose boundary is defined as the union of finite planar polygons. The interface $\mathcal{C}_{jk}$ between the block j and k is defined as the intersection of $\mathcal{B}_j$ and $\mathcal{B}_k$, which is a compact polygonal subset of $\mathbb{R}^2$. Therefore, for an assembly $\mathcal{A}$ with l blocks $\mathcal{B} = \{\mathcal{B}_j \mid j \in \{1, \ldots, l\}\}$, the set of all interfaces is: $\mathcal{C} = \{\mathcal{C}_{jk} = (\mathcal{B}_j \cap \mathcal{B}_k) \mid \exists j, k \in \{1, \ldots, l\}, \text{with } j \neq k \wedge \mathring{\mathcal{C}}_{jk} \neq \emptyset\}$. Let m_{jk} be the total number of vertices $\mathbf{c}^i_{jk}$ of the interface $\mathcal{C}_{jk}$ with $i \in \{1, \ldots, m_{jk}\}$. $(\hat{\mathbf{u}}_{jk}, \hat{\mathbf{v}}_{jk}, \hat{\mathbf{n}}_{jk})$ denotes the unit basis vector representing the local reference system of the planar interface $\mathcal{C}_{jk}$ where $\hat{\mathbf{n}}_{jk}$ is the unit normal vector pointing towards the block $\mathcal{B}_k$. Note that although $\mathcal{C}_{jk}$ and $\mathcal{C}_{kj}$ denote the same interface, the corresponding reference systems are not the same.

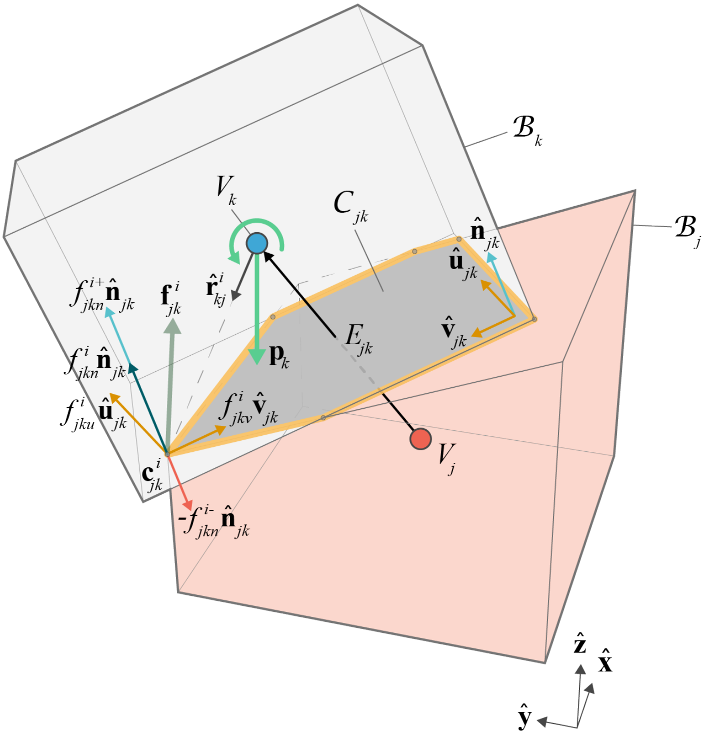


Figure 4.1: Notations adopted to describe the equilibrium equations of the assembly. The pink colour is used for supports, i.e., fixed blocks, and grey is used to denote free blocks. The assembly data structure is a directed graph $G(V, E)$, where vertex V_j stores information of block $\mathcal{B}_j$ while edge E_{jk} stores all information of interface $\mathcal{C}_{jk}$. $\mathbf{f}^i_{jk}$ represents the unknown nodal force acting at the vertex i of interface jk; it can be decomposed into three mutually orthogonal vectors using the local reference system of the interface $(\hat{\mathbf{u}}_{jk}, \hat{\mathbf{v}}_{jk}, \hat{\mathbf{n}}_{jk})$.

4.2.2 Unilateral contact

The unilateral contact between blocks is modelled through compressive forces occurring at the interface vertices. In Figure 4.1, the unknown reaction force $\mathbf{f}^i_{jk} \in \mathbb{R}^3$ acting at the vertex $\mathbf{c}^i_{jk}$ can be decomposed into three components $f^i_{jkn}\,\hat{\mathbf{n}}_{jk}$, $f^i_{jku}\,\hat{\mathbf{u}}_{jk}$, and $f^i_{jkv}\,\hat{\mathbf{v}}_{jk}$. The requirement that only compressive forces are admitted is modelled as a non-negativity constraint on the normal force components:

$$f^i_{jkn} \geq 0\,, \quad \forall i, j, k\;. \tag{4.1}$$

For simplicity, we use bold symbol $\mathbf{f}^i_{jkn} \in \mathbb{R}^3$ to represent $f^i_{jkn}\,\hat{\mathbf{n}}_{jk}$. Similarly, we consider the nodal friction force $\mathbf{f}^i_{jkt} \in \mathbb{R}^3$ combining the two arbitrary, mutually-orthogonal, in-plane shear components $f^i_{jku}\,\hat{\mathbf{u}}_{jk} + f^i_{jkv}\,\hat{\mathbf{v}}_{jk}$, where $f^i_{jku}, f^i_{jkv} \in \mathbb{R}$.

4.2.3 Equilibrium equations

Referring to Figure 4.1, for the assembly $\mathcal{A}$ with l blocks and h interfaces to be in static equilibrium requires that every block has to be in static equilibrium. Unknown internal net forces and torques must balance all known external and unknown reaction forces and torques. In particular, for each block six equilibrium equations can be written, three for the net force and three for the torque. Looking at the block $\mathcal{B}_k$, the equilibrium equations due to all nodal reaction forces $\mathbf{f}_{jk}$ acting on the m_{jk}-sided polygonal interface $\mathcal{C}_{jk}$ can be compactly expressed as:

$$\mathbf{A}_{k,jk}\,\mathbf{f}_{jk} = \begin{bmatrix} \mathbf{a}_{jk,x} & \mathbf{a}_{jk,x} & \cdots & \mathbf{a}_{jk,x} \\ \mathbf{a}_{jk,y} & \mathbf{a}_{jk,y} & \cdots & \mathbf{a}_{jk,y} \\ \mathbf{a}_{jk,z} & \mathbf{a}_{jk,z} & \cdots & \mathbf{a}_{jk,z} \\ \mathbf{b}^1_{jk,x} & \mathbf{b}^2_{jk,x} & \cdots & \mathbf{b}^m_{jk,x} \\ \mathbf{b}^1_{jk,y} & \mathbf{b}^2_{jk,y} & \cdots & \mathbf{b}^m_{jk,y} \\ \mathbf{b}^1_{jk,z} & \mathbf{b}^2_{jk,z} & \cdots & \mathbf{b}^m_{jk,z} \end{bmatrix} \begin{bmatrix} \mathbf{f}^1_{jk} \\ \mathbf{f}^2_{jk} \\ \vdots \\ \mathbf{f}^m_{jk} \end{bmatrix}\,,$$

where $\mathbf{a}_{jk,\imath} = [\hat{n}_{jk,\imath} \quad \hat{u}_{jk,\imath} \quad \hat{v}_{jk,\imath}]$, $\mathbf{b}^i_{jk,\imath} = [(\mathbf{r}^i_{kj} \times \hat{\mathbf{n}}_{jk})_\imath \quad (\mathbf{r}^i_{kj} \times \hat{\mathbf{u}}_{jk})_\imath \quad (\mathbf{r}^i_{kj} \times \hat{\mathbf{v}}_{jk})_\imath]$, $\imath \in \{x, y, z\}$, and $\mathbf{f}^i_{jk} = [f^i_{jkn} \quad f^i_{jku} \quad f^i_{jkv}]^\intercal$. Particularly, $\hat{n}_{jk,x}$, $\hat{u}_{jk,x}$, and $\hat{v}_{jk,x}$ are the x coordinate of $\mathcal{C}_{jk}$'s basis expressed in the global reference system. $\mathbf{r}^i_{kj}$ is the torque arm vector pointing from the centre of mass of $\mathcal{B}_k$ to $\mathbf{c}^i_{jk}$. The first three rows of $\mathbf{A}_{k,jk}$ collect the translational coefficients, while the last three rows collect the torque coefficients. These six equilibrium equations have to be written for all interfaces $\mathcal{C}_{jk}$ between $\mathcal{B}_k$ and each of its neighbouring blocks $\mathcal{B}_j$. Once all equilibrium equations for all blocks are collected, the global equilibrium of the entire assembly $\mathcal{A}$ reads:

$$\begin{array}{cccc} \begin{bmatrix} \mathbf{A}_{1,1} & \cdots & \mathbf{A}_{1,h} \\ \vdots & \ddots & \vdots \\ \mathbf{A}_{l,1} & \cdots & \mathbf{A}_{l,h} \end{bmatrix} & \begin{bmatrix} \mathbf{f}_1 \\ \vdots \\ \mathbf{f}_h \end{bmatrix} & = - & \begin{bmatrix} \mathbf{p}_1 \\ \vdots \\ \mathbf{p}_l \end{bmatrix} \\ & & & \\ \mathbf{A} & \mathbf{f} & = - & \mathbf{p} \quad , \end{array}$$

where row k and column $\jmath$ of sub matrix element $\mathbf{A}_{k,\jmath}$ represent the equilibrium equations of $\mathcal{B}_k$ as coming from interface $\mathcal{C}_\jmath$. Note that all interfaces $\mathcal{C}_{jk}$ are collected in a given order through the map $g : \mathcal{C}_{jk} \in \mathcal{C} \mapsto \mathcal{C}_\jmath \in \mathcal{C}$ and $\jmath \in \{1, \ldots, h\}$. Typically, $\mathbf{A}$ matrix is sparse. Indeed, each column $\jmath$ only has at most two non-zeros sub-matrices $\mathbf{A}_{k,\jmath}$ as each interface is shared by two adjacent blocks. Besides, if the block $\mathcal{B}_k$ is a support, the corresponding sub-matrix is zero. Specifically, the row k in the equation, $\begin{bmatrix} \mathbf{A}_{k,1} & \cdots & \mathbf{A}_{k,h} \end{bmatrix} \begin{bmatrix} \mathbf{f}_1 & \cdots & \mathbf{f}_h \end{bmatrix}^\intercal = -\mathbf{p}_k$, represents the equilibrium condition for $\mathcal{B}_k$ respective to all its neighbouring interfaces, where $\mathbf{p}_k$ is the vector collecting the external loads acting on the $\mathcal{B}_k$.

To summarise, we can compactly write all equilibrium equations in matrix form:

$$\mathbf{A}_{eq}\, \mathbf{f} = -\mathbf{p} \ , \tag{4.2}$$

where the coefficients of the equilibrium equations are collected in the matrix $\mathbf{A}_{eq} \in \mathbb{R}^{6 \cdot l \times 3 \cdot s}$, with $s = \sum_{k=1}^{l} \sum_{j=1}^{l} m_{jk}$ the total number of interface corners; $\mathbf{f} \in \mathbb{R}^{3 \cdot s}$ collects all interface unknown forces, and $\mathbf{p} \in \mathbb{R}^{6 \cdot l}$ the external forces lumped at each block's centre of mass.

To have stable assemblies, Eq. 4.2 is the necessary condition that needs to be satisfied. If we cannot find any solution $\mathbf{f}$, we can state that the structure is not stable. Typically, $\mathbf{A}_{eq}$ is not a square matrix because of the structure's indeterminacy, having fewer equations than unknowns; in general, $6 \cdot l < 3 \cdot s$. Therefore, the solution $\mathbf{f}$ is not unique, and it does not necessarily represent the actual force distribution over the interface. For these reasons, optimisation approaches represent a valid strategy to select one of the possible internal stress states.

4.2.4 Analysing infeasible structures

For unstable assemblies, equation (4.2) does not have a solution. To extend the solution space and to measure the corresponding infeasibility, Whiting et al. [2009, 2012] allowed for tensile capacity through the introduction of a penalty formulation. Specifically, they decoupled the normal force f^i_{jkn} into its positive f^{i+}_{jkn} and negative f^{i-}_{jkn} parts, such that $\forall i, j, k,\ f^i_{jkn} = f^{i+}_{jkn} - f^{i-}_{jkn}$, with $f^{i+}_{jkn},\ f^{i-}_{jkn} \geq 0$ [Bertsimas and Tsitsiklis, 1997]. Therefore, an additional unknown for each vertex was introduced. All the unknowns can be collected in the vector $\tilde{\mathbf{f}} \in \mathbb{R}^{4 \cdot s}$ and the equilibrium of the entire assembly can be written using the same equilibrium matrix in Eq. 4.2 as:

$$\mathbf{A}_{eq}\,\mathbf{B}\,\tilde{\mathbf{f}} = -\mathbf{p}\ , \tag{4.3}$$

where the different dimensional $\tilde{\mathbf{f}}$ is represented element-wise $\tilde{\mathbf{f}}^i_{jk} = [f^{i+}_{jkn} \quad f^{i-}_{jkn} \quad f^i_{jku} \quad f^i_{jkv}]^\intercal$ and $\mathbf{B}$ is a matrix that maps $\tilde{\mathbf{f}}$ to $\mathbf{f}$. We define this relation as $\mathbf{B}^i_{jk}\,\tilde{\mathbf{f}}^i_{jk} = \mathbf{f}^i_{jk}$ with:

$$\mathbf{B}^i_{jk} = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \forall i, j, k\ .$$

4.2.5 Linearised friction constraint

To model the Mohr-Coulomb friction cone constraint, the friction is bounded by the normal forces. Since the normal force is decomposed into two parts, two strategies can be adopted. The first relates the tangential with the net normal forces [Whiting et al., 2009, 2012], namely:

$$\left| \, f^{i}_{jkt} \, \right| \leq (\mu \, f^{i+}_{jkn} - \mu \, f^{i-}_{jkn}) \, , \quad \forall i, j, k \; . \tag{4.4}$$

The second indicates that the friction is bounded by the positive part of the normal forces [Frick et al., 2015]:

$$\left| \, f^{i}_{jkt} \, \right| \leq \mu \, f^{i+}_{jkn} \, , \quad \forall i, j, k \; . \tag{4.5}$$

where μ is the static friction coefficient. We call the first strategy (Eq. 4.4) the *friction-net* approach; while the second (Eq. 4.5) the *friction+* approach. With the *friction-net* approach, the optimisation problem can get infeasible if the friction capacity is exceeded. Indeed, if the net force of Eq. 4.4 is zero, the corresponding tangential force is constrained to be zero. With the *friction+* approach, the problem is always feasible since the solution in terms of tangential forces affects the nodal normal forces because of the penalty formulation. In Section 6.1, we illustrate this aspect through a simple benchmark. Throughout this research, if we do not specifically mention which strategy is applied, we use *friction+* by default because it provides better feasibility to the problem.

Theoretically, the Mohr-Coulomb criterion is represented by a cone. Nonetheless, it is common to approximate the original Mohr-Coulomb friction cone with a pyramid, linearised with a given number of planes, to speed up the solving process. To have a good proximation of the cone, we detail the eight-sided linearised friction constraint used in this research. Figure 4.2 shows an eight-sided cone approximation and its vertex coordinates concerning two tangential axes $\mathbf{f}^{i}_{jku}$ and $\mathbf{f}^{i}_{jkv}$.

Looking at an interface vertex, Eq. 4.5 can be rewritten using the following two inequalities:

$$\begin{aligned} -\mu\, f^{i+}_{jkn} + f^{i}_{jkt} &\leq 0 \\ -\mu\, f^{i+}_{jkn} - f^{i}_{jkt} &\leq 0\,, \quad \forall i, j, k\ . \end{aligned}$$

If we replace f^i_{jkt} with f^i_{jku} and f^i_{jkv}, we can define the eight-sided approximation of the Mohr-Coulomb criterion as:

$$\mathbf{G}\,\mathbf{B}^i_{jk}\,\tilde{\mathbf{f}}^i_{jk} \leq \mathbf{0}\,, \quad \forall i, j, k\ ,$$

where matrix $\mathbf{G}$ is:

$$\begin{bmatrix} -\mu & 1 & 0 \\ -\mu & 0 & 1 \\ -\mu & -1 & 0 \\ -\mu & 0 & -1 \\ -\mu & 1/\sqrt{2} & 1/\sqrt{2} \\ -\mu & -1/\sqrt{2} & 1/\sqrt{2} \\ -\mu & -1/\sqrt{2} & -1/\sqrt{2} \\ -\mu & 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix},$$

matrix $\mathbf{B}^i_{jk}$ is:

$$\mathbf{B}^i_{jk} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \forall i, j, k\ ,$$

which can be stacked into a big diagonal matrix considering all vertices of all interfaces:

$$\begin{bmatrix} \mathbf{G} & & \\ & \ddots & \\ & & \mathbf{G} \end{bmatrix} \begin{bmatrix} \mathbf{B}^i_{jk} & & \\ & \ddots & \\ & & \mathbf{B}^i_{jk} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{f}}_1 \\ \vdots \\ \tilde{\mathbf{f}}_s \end{bmatrix} \leq \mathbf{0}\,,$$

where $\begin{bmatrix} \tilde{\mathbf{f}}_1 & \cdots & \tilde{\mathbf{f}}_s \end{bmatrix}^\mathsf{T}$ are all the nodal forces.

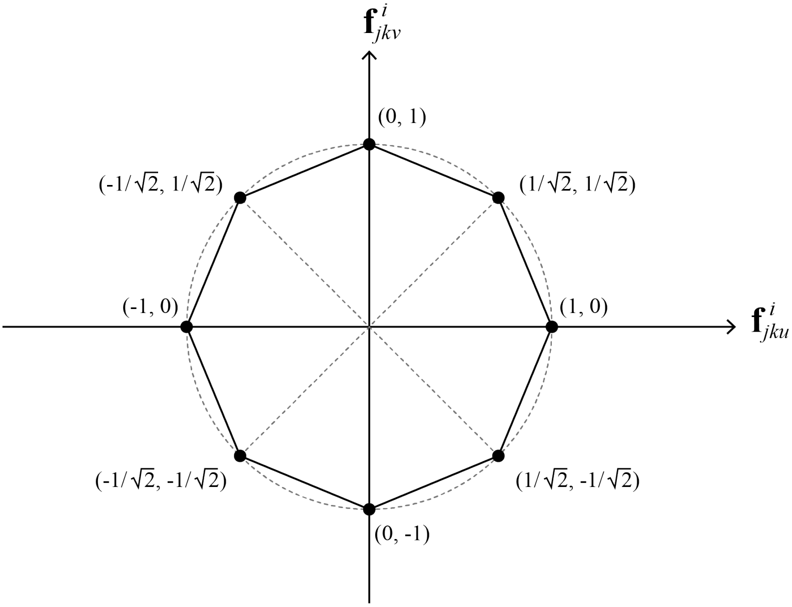


Figure 4.2: Linearised eight-sided friction cone

To summarise, the linearised equations have to be written for all interface vertices and can be collected in the following matrix form:

$$\mathbf{A}_{fr}\, \mathbf{B}\, \tilde{\mathbf{f}} \leq \mathbf{0}\ , \tag{4.6}$$

where $\mathbf{A}_{fr}$ collects all coefficients in a large sparse matrix.

4.2.6 Optimisation problem

Combining the equilibrium condition (Eq. 4.3) with the linearised friction constraint (Eq. 4.6), the entire optimisation problem can be expressed as a quadratic programming (QP) problem:

$$\begin{aligned} \min_{\tilde{\mathbf{f}}} \quad & \frac{1}{2}\,\tilde{\mathbf{f}}^{\intercal}\,\mathbf{H}\,\tilde{\mathbf{f}} \\ \text{s.t.} \quad & \mathbf{A}_{eq}\,\mathbf{B}\,\tilde{\mathbf{f}} = -\mathbf{p} \\ & \mathbf{A}_{fr}\,\mathbf{B}\,\tilde{\mathbf{f}} \leq \mathbf{0} \\ & f^{i+}_{jkn}\,, f^{i-}_{jkn} \geq 0\,, \quad \forall i, j, k\,, \end{aligned} \tag{4.7}$$

where $\mathbf{H}$ is the square diagonal matrix that collects all weights attributed to different force components [Whiting et al., 2012], with the tensile forces highly penalised. Compared to a linear objective function in linear programming (LP) problem:

$$\min_{\tilde{\mathbf{f}}} \quad \mathbf{c}^{\intercal}\,\tilde{\mathbf{f}}\,, \tag{4.8}$$

the quadratic one returns a linear-elastic force distribution on the interface. In Section 6.1, we discuss this aspect in more detail. Optimisation problem (4.7) represents the original RBE formulation as in Whiting et al. [2009, 2012]. It is worth noting that RBE is a strictly force-based approach, without coupling internal stress states with corresponding displacements. As in Whiting et al. [2009, 2012], an assembly is assumed stable if problem (4.7) returns a solution without tensile forces ($f^{i+}_{jkn} - f^{i-}_{jkn} \geq 0\,,\ \forall i, j, k$).

4.3 Kern of a generic polygonal interface

In this section, we present the numerical procedure to define the kern of a generic, planar polygonal interface, which can be either convex or concave (in the latter, the convex hull of the interface is used to define the kern). For more details, the reader is referred to [Hally, 1987; Romano, 2002]. For unilateral materials, the kern of an interface is the area in which a compressive point load may be applied without producing any tensile stress.

Looking at the vertex local reference system of the m_{jk}-sided polygonal interface $\mathcal{C}_{jk}$ (Figure 4.3a), defining a planar contact between two adjacent blocks, let $\boldsymbol{\varsigma}^i_{jk}$ be the position vector of the i-th vertex in the local coordinate system.

4.3.1 The moments of polygons

In this subsection, we briefly summarise the equations of moments of polygons derived from Hally [1987]:

$$\boldsymbol{I}^{(n)} \equiv \int_{\partial\Omega} {\boldsymbol{\varsigma}^i_{jk}}^n \, du \, dv \; , \tag{4.9}$$

where the superscript n is the order of the moment, $\partial\Omega$ denotes integration over the interface domain, and u, v are the local coordinate axes.

The following three linear equations express the zero-, first- and second-order area moments, namely:

$$\boldsymbol{I}^{(0)} = \frac{1}{2} \sum_{i=1}^{m_{jk}} \mathcal{D}^i_{jk} \; , \tag{4.10}$$

$$\boldsymbol{I}^{(1)} = \frac{1}{6} \sum_{i=1}^{m_{jk}} \mathcal{D}^i_{jk} (\boldsymbol{\varsigma}^i_{jk} + \boldsymbol{\varsigma}^{i+1}_{jk}) = \begin{bmatrix} I_u \\ I_v \end{bmatrix} \; , \tag{4.11}$$

$$\begin{aligned} \boldsymbol{I}^{(2)} =& \frac{1}{12} \sum_{i=1}^{m_{jk}} \mathcal{D}^i_{jk} ({\boldsymbol{\varsigma}^i_{jk}}^2 + \frac{1}{2} (\boldsymbol{\varsigma}^i_{jk} \otimes \boldsymbol{\varsigma}^{i+1}_{jk} + \boldsymbol{\varsigma}^{i+1}_{jk} \otimes \boldsymbol{\varsigma}^i_{jk}) + {\boldsymbol{\varsigma}^{i+1}_{jk}}^2) \\ =& \begin{bmatrix} I_{uu} & I_{uv} \\ I_{vu} & I_{vv} \end{bmatrix} \; , \end{aligned} \tag{4.12}$$

where,

$$\mathcal{D}^i_{jk} = \left| \boldsymbol{\varsigma}^i_{jk} \times \boldsymbol{\varsigma}^{i+1}_{jk} \right| = \varsigma^i_{jk,u} \, \varsigma^{i+1}_{jk,v} - \varsigma^i_{jk,v} \, \varsigma^{i+1}_{jk,u} \; ,$$

$${\boldsymbol{\varsigma}^i_{jk}}^2 = \boldsymbol{\varsigma}^i_{jk} \otimes \boldsymbol{\varsigma}^i_{jk} \; ,$$

and,

$$I_u = \frac{1}{6}\sum_{i=1}^{m_{jk}} \mathcal{D}^i_{jk}(\varsigma^i_{jk,u} + \varsigma^{i+1}_{jk,u}) \ ,$$

$$I_v = \frac{1}{6}\sum_{i=1}^{m_{jk}} \mathcal{D}^i_{jk}(\varsigma^i_{jk,v} + \varsigma^{i+1}_{jk,v}) \ ,$$

$$I_{uu} = \frac{1}{12}\sum_{i=1}^{m_{jk}} \mathcal{D}^i_{jk}({\varsigma^i_{jk,u}}^2 + \varsigma^i_{jk,u}\varsigma^{i+1}_{jk,u} + {\varsigma^{i+1}_{jk,u}}^2) \ ,$$

$$I_{vv} = \frac{1}{12}\sum_{i=1}^{m_{jk}} \mathcal{D}^i_{jk}({\varsigma^i_{jk,v}}^2 + \varsigma^i_{jk,v}\varsigma^{i+1}_{jk,v} + {\varsigma^{i+1}_{jk,v}}^2) \ ,$$

$$I_{uv} = I_{vu} = \frac{1}{12}\sum_{i=1}^{m_{jk}} \mathcal{D}^i_{jk}(\varsigma^i_{jk,u}\varsigma^i_{jk,v} + \frac{1}{2}(\varsigma^i_{jk,u}\varsigma^{i+1}_{jk,v} + \varsigma^i_{jk,v}\varsigma^{i+1}_{jk,u}) + \varsigma^{i+1}_{jk,u}\varsigma^{i+1}_{jk,v}) \ ,$$

4.3.2 Position of kern

Let $\mathbf{R}$ be the second-order, a skew-symmetric tensor that rotates in the positive direction (right-hand rule) any vector by $\pi/2$:

$$\mathbf{R} = \begin{bmatrix} \cos(\pi/2) & -\sin(\pi/2) \\ \sin(\pi/2) & \cos(\pi/2) \end{bmatrix} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \ . \tag{4.13}$$

In Figure 4.3b, the unit normal $\hat{\boldsymbol{n}}_i$ to the edge $(\boldsymbol{\varsigma}^i_{jk}, \boldsymbol{\varsigma}^{i+1}_{jk})$ can be expressed as:

$$\hat{\boldsymbol{n}}_i = \mathbf{R}\,(\boldsymbol{\varsigma}^{i+1}_{jk} - \boldsymbol{\varsigma}^i_{jk})/\|\boldsymbol{\varsigma}^{i+1}_{jk} - \boldsymbol{\varsigma}^i_{jk}\|_2 \ . \tag{4.14}$$

Particularly, Eqs. (4.10 - 4.12) can be thought of as a scalar, a 2D vector and a matrix, respectively. The position of the centroid of the interface $\mathbf{r}_G$ (Figure 4.3a) can be evaluated through Eqs. (4.10) and (4.11) as:

$$\boldsymbol{r}_G = \boldsymbol{I}^{(1)} \,/\, \boldsymbol{I}^{(0)} \ , \tag{4.15}$$

while the vertex i' of the kern corresponding to line $\boldsymbol{L}_i$ (Figure 4.3b), as:

$$\boldsymbol{r}_{i'} = -\boldsymbol{I}_G^{(2)} \hat{\boldsymbol{n}}_i \,/\, (\boldsymbol{I}^{(0)} (\boldsymbol{r}_i \cdot \hat{\boldsymbol{n}}_i)) \,, \tag{4.16}$$

in which:

$$\begin{aligned} \boldsymbol{I}_G^{(2)} =& \boldsymbol{I}^{(2)} - \boldsymbol{I}^{(0)} (\boldsymbol{r}_G \otimes \boldsymbol{r}_G) \\ =& \boldsymbol{I}^{(2)} - \boldsymbol{I}^{(0)} \begin{bmatrix} \varsigma^G_{jk,u}\, \varsigma^G_{jk,u} & \varsigma^G_{jk,u}\, \varsigma^G_{jk,v} \\ \varsigma^G_{jk,v}\, \varsigma^G_{jk,u} & \varsigma^G_{jk,v}\, \varsigma^G_{jk,v} \end{bmatrix} . \end{aligned}$$

By applying Eq. (4.16) for each edge of the convex hull, it is possible to explicitly define the kern of a polygonal interface [Romano, 2002]. In Section 6.1, we will show how the kern of an interface can be introduced directly in RBE. Furthermore, we will illustrate how, for statically indeterminate structures, we can use it to seek one of the infinite, admissible, internal stress fields that fully activates a set of interfaces.

Remark 1. Instead of adopting a geometric safety factor using the concept of "kern" by shrinking the boundaries of the contact polygon [Whiting et al., 2009], we implement the explicit kern formulation. The reason for introducing the "kern" in Whiting et al. [2009] was to prevent potential high concentrations of compressive stresses/forces just taking into account reduced interfaces. For a real material having a finite capacity in compression, high-level, localised compressive stresses can lead to material crushing. The main consequence of local material crushing is the redistribution of the compressive stresses over an augmented area, and the corresponding kinematic effects may be modelled in terms of relative displacements as a penetration between the original rigid blocks. Nonetheless, in common unreinforced, historic masonry structures, the crushing effects on the stability are usually two orders of magnitude smaller than the common effects due to the typical crack pattern, e.g., the ones caused by settlements. Therefore, it is very conservative to consider an assembly with reduced interfaces. It is more reasonable to take into account these secondary effects in the

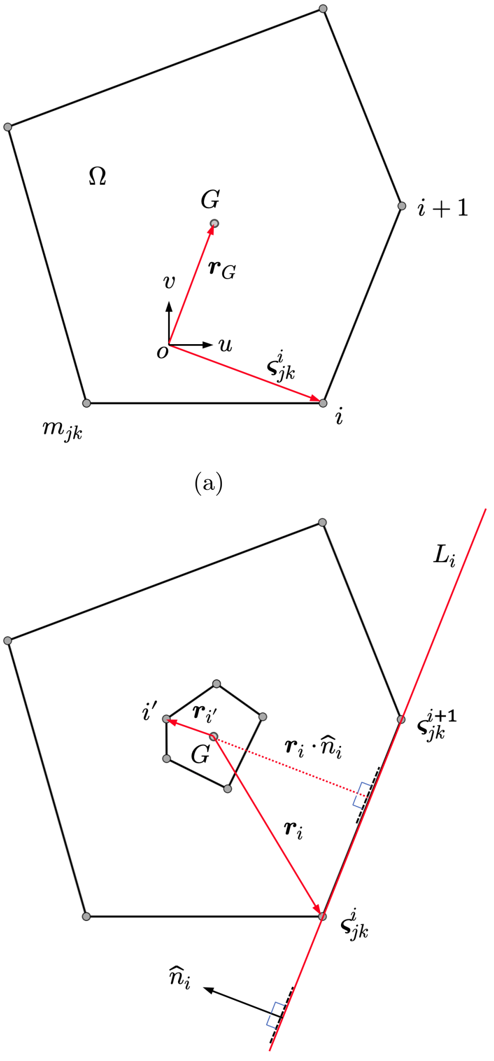

Ω
G
$i+1$
$\boldsymbol{r}_G$
v
u
o
$\boldsymbol{\varsigma}^i_{jk}$
i
m_{jk}
(a)
L_i
i'
$\boldsymbol{r}_{i'}$
$\boldsymbol{\varsigma}^{i+1}_{jk}$
$\boldsymbol{r}_i \cdot \widehat{n}_i$
G
$\boldsymbol{r}_i$
$\boldsymbol{\varsigma}^i_{jk}$
$\widehat{n}_i$
(b)

Figure 4.3: In (a), a convex, polygonal region Ω on the interface $\mathcal{C}_{jk}$, whose generic vertex i is denoted through the vector $\boldsymbol{\varsigma}^i_{jk}$ with respect to a generic reference system (o, u, v). In (b), the kern of the polygonal interface, and the correspondence among the tangent line $\boldsymbol{L}_i$ and its pole i'.

post-processing phase unless the adopted model allows accounting for plastic compressive deformations directly. For a rigid-block model considering a finite compressive capacity, the reader is referred to Portioli et al. [2015].

4.4 Resultants

In physics and engineering, the resultant force (sometimes called net force) combines a system of forces to help simplify the problem. As mentioned in Section 4.2, the optimisation problem selects one of the many solutions for the indeterminacy structures. By computing and visualising the resultant, we can more intuitively interpret the RBE results and understand the structural behaviour of the assemblies.

4.4.1 Resultant forces

The normal resultant force $\boldsymbol{F}_{jkn}$ on the interface $\mathcal{C}_{jk}$ is the sum of all nodal normal forces:

$$\boldsymbol{F}_{jkn} = \sum_{i=1}^{m_{jk}} \mathbf{f}_{jkn}^{i} \; . \tag{4.17}$$

Similarly, we can sum up all nodal friction forces as an in-plane frictional resultant $\boldsymbol{F}_{jkt}$:

$$\boldsymbol{F}_{jkt} = \sum_{i=1}^{m_{jk}} \mathbf{f}_{jkt}^{i} \; . \tag{4.18}$$

4.4.2 Position of the resultant

We locate the resultant at the centre of the pressure of the interface $\mathcal{C}_{jk}$. Its position $\boldsymbol{P}_{jk}$ in the local coordinate system can be calculated with the following relation:

$$(\boldsymbol{P}_{jk} - \boldsymbol{o}_{jk}) \times \boldsymbol{F}_{jk} = \sum_{i=1}^{m_{jk}} (\mathbf{c}_{jk}^{i} - \boldsymbol{o}_{jk}) \times \mathbf{f}_{jkn}^{i} \; ,$$

where $\boldsymbol{o}_{jk}$ is the local reference origin. $\boldsymbol{P}_{jk}$ is a vector that can be expressed as $\begin{bmatrix} P_{jk,u} & P_{jk,v} & 0 \end{bmatrix}^{\mathsf{T}}$, with:

$$P_{jk,\imath} = o_{jk,\imath} + \frac{1}{\|\boldsymbol{F}_{jkn}\|} \sum_{i=1}^{m_{jk}} (c^i_{jk,\imath} - o_{jk,\imath})\, f^i_{jkn,\imath} \ , \ \imath \in \{u, v\} \ . \tag{4.19}$$

4.4.3 Torque

Since both normal and frictional resultant forces are placed at the centre of pressure $\boldsymbol{P}_{jk}$, additional torque caused by the friction forces in the local reference system can be expressed as:

$$\begin{aligned} \mathbf{T}_{jk} &= \sum_{i=1}^{m_{jk}} (\mathbf{c}^i_{jk} - \boldsymbol{P}_{jk}) \times \mathbf{f}^i_{jkt} \\ &= \begin{bmatrix} 0 & 0 & \sum_{i=1}^{m_{jk}} ((c^i_{jku} - P_{jku}) f^i_v - (c^i_{jkv} - P_{jkv}) f^i_u) \end{bmatrix}^{\mathsf{T}} \ . \end{aligned} \tag{4.20}$$

4.5 Assembly data structure

To efficiently handle RBE information, we utilise the assembly data structure proposed in Frick et al. [2016]. The assembly data structure is a graph-based data structure where a directed graph $G(V, E)$ is used to represent the entire assembly $\mathcal{A}$. The vertices V store block geometries information $\mathcal{B}$, and the directed edges E store all contact information $\mathcal{C}$. The graph edge is directed. Indeed, although interfaces $\mathcal{C}_{jk}$ and $\mathcal{C}_{kj}$ collect the same vertices, the unknown forces corresponding to the same corner are reversed for Newton's third law. This shows the reason why the axes of the local reference systems of $\mathcal{C}_{jk}$ and $\mathcal{C}_{kj}$ are pointing in opposite directions. Several interface typologies are available to model non-perfect contacts such as *face-face*, *face-edge*, and *face-vertex* contacts. For a more detailed discussion and implementation of the assembly data structure, we refer the reader to Frick et al. [2016] and COMPAS Assembly [BlockResearchGroup, 2018].

4.6 Summary

In this chapter, we have reviewed, explored, and further developed the RBE method. RBE is a force-based method that solves the equilibrium problem through an optimisation where the objective function minimises the total amount of interface forces and the constraints are represented by linear relations enforcing the static equilibrium and the friction failure conditions. It is worth pointing out that tensile forces are allowed even though highly penalised: in this sense, RBE enlarges the space of admissible stress states. After that, we presented the kern of a general polygonal interface and summarised the resultant calculation procedure. We utilised a graph-based assembly data structure to handle all forces and geometrical properties for the complex assembly.

Compared to other commonly used tools, such as discrete element modelling (DEM), e.g., using the 3DEC code [Cundall, 1971], RBE is open-source, fast, explicit, and straightforward. It only needs one mechanical parameter (i.e., the friction angle, see also Iannuzzo et al. [2020]). On the other hand, compared to real-time interactive environments such as physics-based game engine PhysX [Nvidia, 2013] and Bullet [Coumans, 2013], RBE is a *static* problem and does not require simulating the movement of the objects, such that there is no additional time parameter to tune. In general, RBE combines the rigour of both DEM and the interactivity speed of a game engine in order for it to be used for design purposes. Furthermore, RBE provides more transparent and accurate results beyond a *Yes/No* answer, i.e., stable or collapsing. In this sense, RBE can be used to better guide users during the design process [Wang et al., 2021b]. Additionally, it can also be further applied to other optimisation problems as in Whiting et al. [2012].

Although RBE is used to design stable assemblies, it suffers from limitations in some cases. Yao et al. [2017] hinted that the RBE model has a problem and is not reliable when sliding is involved, thus developing a different alternative method. However, instead of completely giving up on the RBE approach and reinventing or looking for an alternative solver from scratch, we recognise RBE's merits, benefits, and advantages for redirecting users' unstable designs. In the next Chapter 5, we build upon the RBE approach, clarify its issues, and solve them to propose our newly developed method.

Chapter 5

Coupled Rigid-Block Analysis (CRA) method

The rigid-block equilibrium (RBE) method uses a penalty formulation to measure structural infeasibility or to guide the design of stable discrete-element assemblies from an unstable geometry. However, RBE is a purely force-based formulation, and it incorrectly describes stability when complex interface geometries are involved. To overcome this issue, this chapter introduces the coupled rigid-block analysis (CRA) method, a more robust approach building upon RBE's strengths. The CRA method combines equilibrium and kinematics as a penalty formulation in a nonlinear programming problem. In addition, we show how CRA enables accurate modelling of complex three-dimensional discrete-element assemblies formed by rigid blocks.

In Section 5.1, we first describe what RBE is missing conceptually and give some insights to the problem. To introduce the CRA method, in Section 5.2, we first formulate a pure force-based optimisation problem and then we couple it with kinematics through specific additional constraints. After that, we introduce the penalty formulation to measure the level of instability of the unstable assemblies (Section 5.3). The penalty formulation represents the extended version of the original RBE method discussed in Chapter 4. In

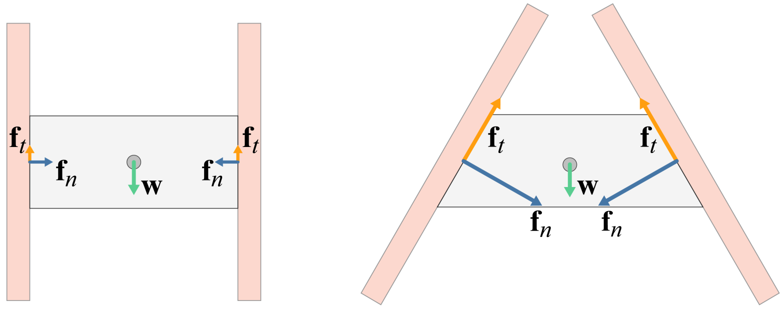
Section 5.4, we extend the data structure in Section 4.5 to avoid the tedious convex decomposition step for analysing complex assemblies. Lastly, we summarise our proposed CRA method in Section 5.5.

5.1 Introduction

To show when RBE fails to predict stability when friction is involved, we illustrate in Subsection 5.1.1 two paradigmatic examples, called model H (shown in Yao et al. [2017]) and A (Figure 5.1). After that, in Subsection 5.1.2, we explain how we propose to overcome these issues by looking at a simple 2D wedged model V (Figure 5.2).

5.1.1 Issues of the RBE approach

For model H in Figure 5.1, we can imagine one free block placed between two parallel fixed walls, which can also represent robot-hand grippers. Regardless of the size of the central block and the friction coefficient, RBE always finds a solution such that the free block is stable. Indeed, to be in static equilibrium, all forces have to cancel out. In the vertical direction, the free block's self-weight $\mathbf{w}$ is balanced by two tangential contact reactions $\mathbf{f}_t$. In the horizontal direction, RBE finds two normal forces $\mathbf{f}_n$ that cancel each other out. Thus, for any tangential force distribution with any friction coefficient, RBE always finds two large enough normal forces that simultaneously satisfy the Mohr-Coulomb criterion and cancel each other out. In other words, those valid static equilibrium solutions always exist in the constraint solution space defined by Eq. 4.2. However, if the middle block is not *prestressed*, e.g., due to robotic grippers, the normal forces should not exist in a real-world scenario. In fact, such a bizarre equilibrium solution is not wrong; it is because of the original rigid-body assumption that a normal force can occur without any deformation. It is a *physically unrealisable* solution when the free block is not prestressed, and generated by a pure force-based optimisation that does not take those considerations into account. In the case of associative behaviour, the solution of the force-based



(a) model H

(b) model A

Figure 5.1: Two peculiar examples show some of RBE's issues. The free block (grey) is always detected as in equilibrium.

problem is linked to the solution of a dual displacement-based problem, as shown in Portioli [2020], Portioli et al. [2021] for a finite friction capacity, or through an energy-based dualisation of an Heymanian material model in Iannuzzo et al. [2020, 2021a]. RBE returns a force-based solution that assumes its dual displacement-based solution exists, regardless if it needs prestress or not. A simple way to avoid the H model's problem is to couple equilibrium with kinematics and allow for a negligible numerical overlapping; since RBE assumes all blocks to be rigid and non-deformable, having two simultaneous overlaps between supports is impossible.

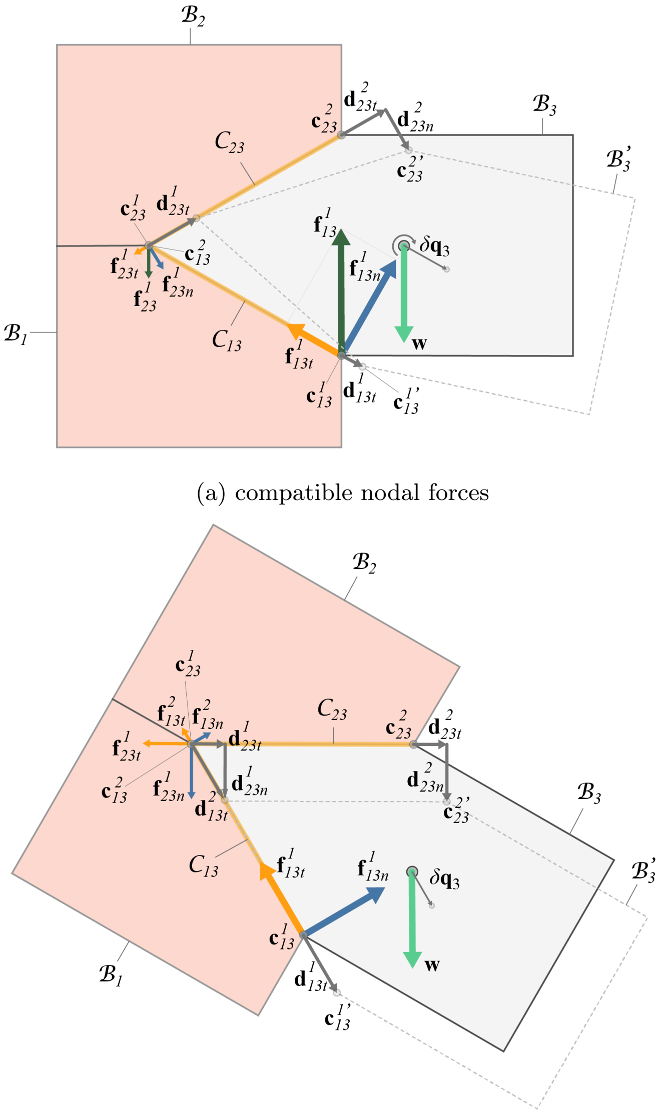
Comparable to model H, the reader can intuitively imagine that in model A the free block would fall down if it is rigid and not prestressed. However, if we try to solve this model in the same way used for the H model, RBE would still provide the solution shown in Figure 5.1b. Therefore, something is still missing; besides the deformation aspect, all reaction forces, particularly tangential forces, have to be compatible with possible rigid-body motions [Weißenfels and Wriggers, 2015], and, all tangential forces are possible only if the corresponding normal forces are activated. With these assumptions, model A without prestressing is no longer feasible. In other words, all possible rigid-body motions have to be considered. In model A, the only admissible movement is a downward vertical displacement of the central block. Once the contact of one of the two surfaces is lost, admissible force solutions are no longer possible.

As the RBE method is a pure force-based approach, it fails to capture some scenarios as discussed in the H and A model. The optimiser greedily finds *physically unrealisable* solutions (without prestressing) as long as those solutions satisfy equilibrium and friction cone constraints. In some cases, wrong solutions included in the RBE's solution space can be taken out by looking at the admissible dual movements, but in many other cases it is not possible to simply exclude them from the solution space.

5.1.2 Insight problems

Inspired by Omata and Nagata [2000], we observe that not all forces are *physically realisable* concerning rigid-body movements. To be more specific, the reaction forces are *realisable* only if we can find a compatible rigid-body motion. The friction force occurs in the opposite direction of a potential relative movement to prevent the object from sliding such that the acceleration remains zero. Additionally, the normal reaction force is activated only if the two adjacent points are in contact. When there is a detachment, the normal force has to be zero and, consequently, the friction force must be zero as well (bounded by the Mohr-Coulomb criterion).

Figure 5.2 shows two possible equilibrated solutions of the wedged model V obtained with RBE. Both models share the same geometry and boundary conditions but are inclined at a different angle. In Figure 5.2a, we depict a possible rigid-body motion defined by an in-plane rotation and translation, collected in the vector $\delta\mathbf{q}_3$ describing the motion of block $\mathcal{B}_3$. The nodal friction forces $\mathbf{f}^1_{13t}$ and $\mathbf{f}^1_{23t}$ found by RBE are compatible with nodal displacements $\mathbf{d}^1_{13t}$ and $\mathbf{d}^1_{23t}$. Similarly, if a contact point tends to detach, the nodal normal force cannot be activated. Indeed, no reaction force can occur at vertex $\mathbf{c}^2_{23}$ because of the relative detachment $\mathbf{d}^2_{23n}$.



(a) compatible nodal forces

(b) incompatible nodal forces

Figure 5.2: Model V: nodal forces have to be compatible with virtual rigid-body motion; otherwise, those nodal forces are not physically *realisable*.

Conversely, Figure 5.2b illustrates how an admissible equilibrated solution is not compatible with a potential rigid-body motion. The contact forces obtained using RBE are in equilibrium as they balance the block's self-weight. However, these forces are not compatible with a pure translation of the block $\mathcal{B}_3$. Indeed, contact forces $\mathbf{f}^1_{23n}$ and $\mathbf{f}^1_{23t}$ are not compatible because of the normal detachment $\mathbf{d}^1_{23n}$. Additionally, this equilibrated solution is not compatible with the rigid-body motion depicted in Figure 5.2a, as $\mathbf{f}^2_{13n}$ and $\mathbf{f}^2_{13t}$ are not admissible because of the dual detachment.

Importantly, in this work, all friction forces are *static*, while *infinitesimal* rigid-body motions represent *virtual* displacements. The aim is to couple internal forces with *infinitesimal* rigid-body motion in order to discard equilibrated solutions that are not compatible with any rigid-body motion, and in this sense, not *realisable*.

The physical limitations highlighted on the H, A, and V models can frequently appear in 3D assemblies, and they are intrinsic in geometries with multiple planar interfaces (we will call it multi-planar for short). Indeed, in Figure 6.13d, the interface between free and fixed blocks is multi-planar, and, upon closer inspection the reader can see a parallelism with model A or V. Even though the RBE provides proper solutions in some cases, e.g., an arch subjected to in-plane loads (see Figure 6.11), we cannot determine in advance if treating a specific problem with a pure force-based approach is sufficient. In the worst case, when RBE fails, it falsely claims that a non-prestressed and unstable structure is safe, leading to wrong design choices or assessment strategies. Thus, a procedure that correctly accounts for these aspects is needed. This research aims to tackle these aspects through a non-linear constrained optimisation problem to discard *unrealisable* solutions and provide an accurate approach for discrete-element assembly analysis. We call this approach the Coupled Rigid-Block Analysis (CRA) method as it combines the original equilibrium formulation of RBE with kinematics. In Section 5.2, this new numerical formulation will be introduced and discussed.

5.2 Problem formulation

Similar to the RBE formulation in Eq. 4.7, but excluding the penalty tension parts, the force-based formulation that we start from is:

$$\begin{aligned} \min_{\mathbf{f}} \quad & \|\mathbf{f}_n\|_2^2 && \text{(5.1a)} \\ \text{s.t.} \quad & \mathbf{A}_{eq}\,\mathbf{f} = -\mathbf{p} && \text{(5.1b)} \\ & \mathbf{A}_{fr}\,\mathbf{f} \leq \mathbf{0} && \text{(5.1c)} \\ & f^i_{jkn} \geq 0\ , \quad \forall i, j, k\ , && \text{(5.1d)} \end{aligned}$$

where Eq. 5.1b enforces equilibrium only using compressive forces (Eq. 5.1d) and Eq. 5.1c represents the Mohr-Coulomb material failure criterion. The objective function represents the squared norm of all contact forces based on Gauss's principle of least constraint [Moreau, 1966]. The optimisation problem returns infeasible if no equilibrium solution can be found.

As mentioned in Section 5.1, in some scenarios such as models H and A, Eq. 5.1 gives an equilibrated solution but it is a *physically unrealisable* solution without the existence of prestressing or external forces.

5.2.1 Virtual displacement

As proved in Mattikalli et al. [1996] and Pang and Trinkle [2000], equilibrium alone does not represent a sufficient condition for stability. To impose the compatibility between forces and rigid-body motions, we consider *virtual* displacements, see Figure 5.3. In particular, the first aspect is to define the space of all possible *infinitesimal* admissible rigid-body displacements. We denote with $\delta\mathbf{q}_k \in \mathbb{R}^6$ the virtual displacement vector of $\mathcal{B}_k$, where the first three components of $\delta\mathbf{q}_k$ are rigid translations $\begin{bmatrix} \delta q_{k,x} & \delta q_{k,y} & \delta q_{k,z} \end{bmatrix}^\mathsf{T}$ and the last three the rigid rotations $\begin{bmatrix} \delta q_{k,rx} & \delta q_{k,ry} & \delta q_{k,rz} \end{bmatrix}^\mathsf{T}$. The piecewise rigid displacement field describing the motion of the entire assembly is a function of the Lagrangian parameters collected in the vector $\delta\mathbf{q} \in \mathbb{R}^{6\cdot l}$.

Based on the well-known duality relation for rigid bodies, the kinematic matrix can be expressed as the transpose of the equilibrium matrix $\mathbf{A}_{eq}^{\intercal} \in \mathbb{R}^{3 \cdot s \times 6 \cdot l}$ [Pellegrino, 1993]. Therefore, the relative displacement $\delta \mathbf{d} \in \mathbb{R}^{3 \cdot s}$ of the vertices can be expressed through the following relation:

$$\mathbf{A}_{eq}^{\intercal} \, \delta \mathbf{q} = \delta \mathbf{d} \; . \tag{5.2}$$

Similarly to the nodal force $\mathbf{f}_{jk}^{i}$, the relative *virtual* displacement $\delta \mathbf{d}_{jk}^{i} \in \mathbb{R}^{3}$ of the vertex $\mathbf{c}_{jk}^{i}$ can be decomposed into three components $\delta \mathbf{d}_{jkn}^{i}$, $\delta \mathbf{d}_{jku}^{i}$, and $\delta \mathbf{d}_{jkv}^{i}$, with $\delta \mathbf{d}_{jkn}^{i}$ the relative normal displacement, $\delta \mathbf{d}_{jkt}^{i}$ the relative tangential sliding displacement. Next, we introduce two afterwards nonlinear constraints to exclude *physically unrealisable* forces.

5.2.2 Friction-sliding alignment constraint

Firstly, we need to enforce the compatibility between forces and rigid-body motions in two directions, tangential sliding and normal detachment. For a potential sliding motion, we enforce the alignment between the friction forces and corresponding virtual sliding movements using the following nonlinear constraints for each contact $\mathbf{c}_{jk}^{i}$:

$$\mathbf{f}_{jkt}^{i} = -\alpha_{jk}^{i} \, \delta \mathbf{d}_{jkt}^{i} \; , \quad \alpha_{jk}^{i} \geq 0 \; , \quad \forall i, j, k \; , \tag{5.3}$$

where α_{jk}^{i} is a non-negative scalar used to relate the friction force $\mathbf{f}_{jkt}^{i}$ direction with the relative sliding displacement $\delta \mathbf{d}_{jkt}^{i}$. Note that friction and virtual sliding movements are aligned but pointing in opposite directions. Also, when the virtual sliding movement is zero, the friction force is zero as well. From a physical perspective, constraints (5.3) ensure only that the direction of the friction forces have to be compatible with the corresponding relative virtual tangential displacement. The need for such a requirement was also pointed out in Omata and Nagata [2000], which stated that enforcing a static friction force in the opposite sliding direction restricts the force solution space properly, as also shown in Figure 5.2.

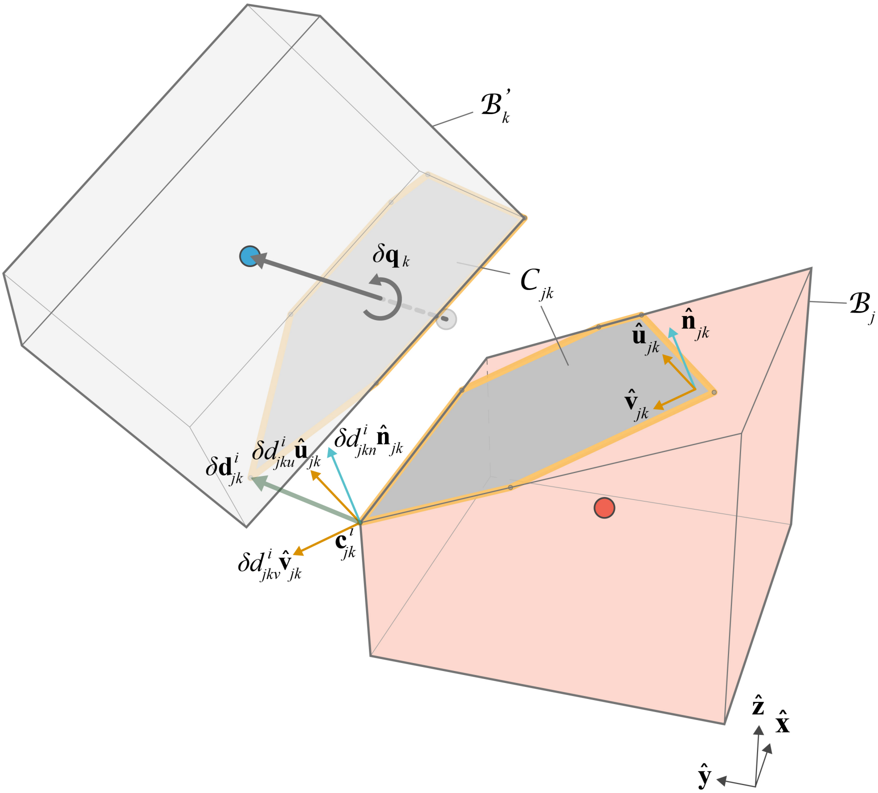


Figure 5.3: Notations adopted to describe the kinematic equations of the assembly. The pink colour is used for supports, i.e., fixed blocks, and grey is used to denote free blocks. Similar to Figure 4.1, the assembly data structure is a directed graph $G(V, E)$, where vertex V_j stores information of block $\mathcal{B}_j$ while edge E_{jk} stores all information of interface $\mathcal{C}_{jk}$. $\delta\mathbf{q}_k \in \mathbb{R}^6$ represents the virtual displacement of $\mathcal{B}_k$, while the relative virtual displacement $\delta\mathbf{d}^i_{jk} \in \mathbb{R}^3$ of the vertex $\mathbf{c}^i_{jk}$ can be decomposed into three components $\delta\mathbf{d}^i_{jku}$, $\delta\mathbf{d}^i_{jkv}$, and $\delta\mathbf{d}^i_{jkn}$ using the local reference system of the interface $(\hat{\mathbf{u}}_{jk}, \hat{\mathbf{v}}_{jk}, \hat{\mathbf{n}}_{jk})$.

5.2.3 Contact complementarity constraint

Secondly, the compatibility between normal forces and corresponding relative displacements has to be enforced. In other words, we need to ensure normal forces appear only if two blocks are in contact; otherwise, when detachment happens, they have to be zero. We model this behaviour as a complementarity constraint:

$$\begin{aligned} f^i_{jkn}\,(\delta d^i_{jkn} + \varepsilon) = 0\,, \\ f^i_{jkn}\,, (\delta d^i_{jkn} + \varepsilon)\,, \varepsilon \geq 0\,, \quad \forall i, j, k\,, \end{aligned} \tag{5.4}$$

where ε is a numerically very small overlapping parameter, i.e., from 10^{-5} to 10^{-4} of the block dimension. The inclusion of ε assures that the normal forces occur only when two blocks slightly overlap.

5.2.4 Positiveness of the external work

Not all rigid motions are physically possible, meaning that the space of kinematically admissible displacement s has to be defined considering specific constraints. Beyond the local constraints expressing the non-overlapping of interfaces, a global requirement is that the work done by the external forces and the corresponding displacements has to be non-negative:

$$\mathbf{p}^\intercal\,\delta\mathbf{q} \geq 0\,. \tag{5.5}$$

However, this requirement is implicitly taken into account by Eq. 4.2 and Eqs. 5.2–5.4. Indeed, applying (4.2) and (5.2) relations, Eq. 5.5 results in:

$$\mathbf{p}^\intercal\,\delta\mathbf{q} = -(\mathbf{A}_{eq}\,\mathbf{f})^\intercal\,\delta\mathbf{q} = -\mathbf{f}^\intercal\,\mathbf{A}_{eq}{}^\intercal\,\delta\mathbf{q} = -\mathbf{f}^\intercal\,\delta\mathbf{d}\,,$$

with the last scalar product decomposable into the sum of the work done by the normal and tangential contact forces and the dual contact displacements, as:

$$-\mathbf{f}_t^\intercal\,\delta\mathbf{d}_t - \mathbf{f}_n^\intercal\,\delta\mathbf{d}_n\,.$$

Note that the first term is always positive because of Eq. 5.3, while because of the normal coupling (Eq. 5.4) the second term is close to zero. Thus, Eq 5.5 can be rewritten as:

$$\mathbf{p}^{\mathsf{T}}\,\delta\mathbf{q} = \sum_{i,j,k} \alpha^i_{jk}\,(\delta\mathbf{d}^i_{jkt})^2\ ,$$

which requires that Eq. 5.5 holds as all α^i_{jk} to be non-negative.

5.2.5 CRA optimisation problem

Finally, combining the original equilibrium formulation (5.1) with the additional constraints expressed in Eqs. 5.2–5.4, the new QP formulation with nonlinear constraints is:

$$\begin{aligned}
\min_{\mathbf{f},\,\delta\mathbf{q},\,\boldsymbol{\alpha}} \quad & \|\mathbf{f}_n\|_2^2 + \|\boldsymbol{\alpha}\|_2^2 \\
\text{s.t.} \quad & \mathbf{A}_{eq}\,\mathbf{f} = -\mathbf{p} \\
& \mathbf{A}_{fr}\,\mathbf{f} \leq \mathbf{0} \\
& \mathbf{A}_{eq}^{\mathsf{T}}\,\delta\mathbf{q} = \delta\mathbf{d} \\
& f^i_{jkn}\,(\delta d^i_{jkn} + \varepsilon) = 0 \\
& \mathbf{f}^i_{jkt} = -\alpha^i_{jk}\,\delta\mathbf{d}^i_{jkt} \\
& \left|\,\delta\mathbf{d}^i_{jk\cdot}\,\right| \leq \eta \\
& f^i_{jkn}\,,\alpha^i_{jk}\,,(\delta d^i_{jkn} + \varepsilon)\,,\varepsilon\,,\eta \geq 0\ , \quad \forall i,j,k\ .
\end{aligned} \tag{5.6}$$

The objective function here includes the additional term $\boldsymbol{\alpha}$ that collects all α^i_{jk}, and we solve the problem in a least square formulation. For simplicity, we follow the RBE-style formulation to model the linearised friction constraints. Alternatively, those constraints can also be modelled with second-order cone constraints to reduce the computational cost [Portioli et al., 2014; Cascini et al., 2018]. It is worth noting that problem (5.6) can be thought of as an optimisation problem with equilibrium constraints. Ferris and Tin-Loi [2001] proposed a nonlinear programming formulation to define limit states of masonry structure composed of rigid blocks. Several differences between our formulation and theirs can be pointed out: i) the objective function in (5.6) minimises the amount of the normal contact forces and does not consider the minimum of the "load factor solutions" as in Ferris and Tin-Loi [2001]; ii) differently from Ferris and Tin-Loi [2001], problem (5.6) also models the alignment between tangential displacements and friction forces, which is key when modelling the mechanics of sharp wedge connections; and, iii) as will be shown in Section 5.3, formulation (5.6) represents an ideal base to implement a penalty formulation in order to measure the structural instability.

Additionally, although small displacement fields are used, optimisation problem (5.6) is nonlinear, so each displacement value $\delta \mathbf{d}^i_{jk\cdot}$ is bounded to a small number η (i.e., $10^{-3} - 10^{-2}$ of the block dimension) to avoid local minimum solutions that are far from the initial reference configuration. Note that problem (5.6) is infeasible when a structure is unstable. The additional constraints combining equilibrium and kinematics reduce the solution space excluding *unrealisable* equilibrated solutions. As we will show, this is reflected in two aspects: i) it allows for better capturing of the limit states; and ii) when the structure is in a stable configuration, it provides more realistic results. As a consequence, it represents a good mathematical base to be used in a penalty formulation.

5.3 Penalty formulation

In comparison with other approaches, one of the true potentials of the RBE method is that it can analyse unstable structures. As the idea behind our formulation is similar to the RBE approach, we include the penalty formulation in the same qualitative way (Subsection 4.2.4). We also allow for tensile forces and decouple the normal force $\mathbf{f}^i_{jkn}$ into its positive $\mathbf{f}^{i+}_{jkn}$ and negative $\mathbf{f}^{i-}_{jkn}$ parts. Thus, the f_{jkn} in Eq. 5.4 can be replaced with the positive part of normal force f^{i+}_{jkn}.

5.3.1 Compression-tension complementarity constraint

To avoid the simultaneous presence of both negative and positive parts on the same vertex, we introduce the following complementarity constraint:

$$f_{jkn}^{i+}\, f_{jkn}^{i-} = 0\,. \tag{5.7}$$

As f_{jkn}^{i+} and f_{jkn}^{i-} cannot coexist, Eq. 4.5 with constraint (5.7) implicitly avoids friction when tension force occurs.

5.3.2 Optimisation to analyse infeasible structures

By adding all penalty forces and additional constraints, optimisation problem (5.6) can be modified to measure unstable structures:

$$\begin{aligned}
\min_{\tilde{\mathbf{f}},\,\delta\mathbf{q},\,\boldsymbol{\alpha}} \quad & \left\|\mathbf{f}_n^{+}\right\|_2^2 + \gamma \left\|\mathbf{f}_n^{-}\right\|_2^2 + \left\|\boldsymbol{\alpha}\right\|_2^2 \\
\text{s.t.} \quad & \mathbf{A}_{eq}\, \mathbf{B}\, \tilde{\mathbf{f}} = -\mathbf{p} \\
& \mathbf{A}_{fr}\, \mathbf{B}\, \tilde{\mathbf{f}} \leq \mathbf{0} \\
& \mathbf{A}_{eq}^{\intercal}\, \delta\mathbf{q} = \delta\mathbf{d} \\
& f_{jkn}^{i+}\, (\delta d_{jkn}^{i} + \varepsilon) = 0 \\
& f_{jkn}^{i+}\, f_{jkn}^{i-} = 0 \\
& \mathbf{f}_{jkt}^{i} = -\alpha_{jk}^{i}\, \delta\mathbf{d}_{jkt}^{i} \\
& \left|\, \delta\mathbf{d}_{jk\cdot}^{i} \,\right| \leq \eta \\
& \gamma\,, \varepsilon\,, \eta \geq 0 \\
& f_{jkn}^{i+}\,, f_{jkn}^{i-}\,, \alpha_{jk}^{i}\,, (\delta d_{jkn}^{i} + \varepsilon) \geq 0\,, \quad \forall i, j, k\,.
\end{aligned} \tag{5.8}$$

The tensile forces are introduced in the objective function using a highly penalised weighting factor γ. As for the RBE formulation, Eq. 5.8 allows for the description of infeasible stress states still favouring compressive solutions.

5.4 Extended assembly data structure for complex assemblies

As discussed in Chapter 4.5, we utilise the assembly data structure described in Frick et al. [2016] and its computational implementation available through BlockResearchGroup [2018] to handle all information for complex assemblies efficiently. For assemblies with non-convex blocks, Frick et al. [2016] proposed the use of multiple convex blocks to discretise a non-convex compound block marking the additional new interfaces as *internal*. This convex decomposition step of complex assemblies, e.g., freeform shapes, is not straightforward, as it requires additional and specific algorithms.

Therefore, we propose an alternative procedure that does not require any the additional convex decomposition step. Specifically, the interface $\mathcal{C}_{jk}$ is approximated through w_{jk} planar compact subinterfaces $\mathcal{C}_{jko}$ with $o \in \{1, \dots, w_{jk}\}$ (see Figure 5.4). All subinterfaces along with their corresponding local reference systems can be stored in the edge E_{jk} of the original directed graph $G(V, E)$. Therefore, constructing the equilibrium matrix $\mathbf{A}_{eq}$ becomes straightforward, since all interfaces are indexed in a compatible way. Our current implementation still requires the interfaces to be discretised either manually or by using an auto-triangulation algorithm. Compared to the existing assembly data structure that requires the convex decomposition of the blocks, our extended version only requires discretising interfaces, making it more straightforward to use.

5.5 Summary

This chapter introduces a new method, named CRA, to assess the stability of complex assemblies. The CRA method provides accurate results, which can push structural stability to the limit during assembly, consequently saving material and allowing for thinner complex structures.

Different from previous methods, CRA provides an accurate evaluation of

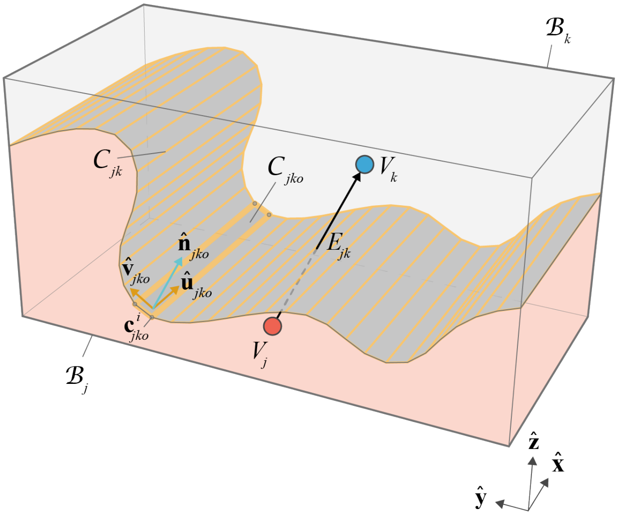


Figure 5.4: The freeform interface $\mathcal{C}_{jk}$ can be approximated through many sub-interfaces $\mathcal{C}_{jko}$.

internal stress states and limit scenarios thanks to a nonlinear programming formulation that couples equilibrium and kinematics. Beyond classical constraints representing the equilibrium and the friction capacity of the assembly, two additional and crucial constraints are considered. The first one is a complementarity condition that relates detachment and normal forces: normal forces are not allowed when blocks are no longer in contact. The second constraint aligns friction forces with *virtual* sliding movements. Its use is particularly relevant to capture the stability of multiple interfaces and sharp wedge connections.

Furthermore, this new nonlinear optimisation problem is rewritten in a penalty formulation, allowing us to evaluate non-equilibrated assemblies and thus explore a larger configuration space. It allows for the detection of non-stable regions, providing additional information to stabilise the assembly. Moreover, the existing graph-based assembly data structure has been extended to enable flexible analysis of complex shapes with curved interfaces.

In Part III, we demonstrate the use of both improved RBE and CRA for the structural assessment and assembly-aware design of discrete assemblies.

Part III

Applications

Chapter 6

Stability assessment

In Part II, we laid out the mathematical foundation and implemented tools for the improved RBE and CRA methods. This chapter demonstrates their use for structural assessment. To understand the correct usage of the RBE method, we perform several studies in Section 6.1. After that, in Section 6.2, we demonstrate the CRA method's accuracy by comparing its numerical results with RBE and various commercial software. Section 6.3 proposes a stability-aware design workflow to help designers iteratively toward a structurally sound design. Section 6.4 demonstrates CRA's usage for complex structural analysis and stability assurance during construction with robots. Lastly, Section 6.5 summarises all structural assessment findings present in this chapter.

6.1 RBE case studies

In this section, to illustrate, compare, interpret, and give meaning to the RBE results, we will reduce all nodal forces acting on an interface to their resultants (forces and torques, if present) applied at the centre of pressure of the interface as described in Section 4.4. Compressive nodal forces are plotted in blue, tensile forces in red, interface resultants in dark green, and interface torques in black. For all examples, we only consider the self-weight as applied loads, and we look at three examples with increasing complexity, and static indeterminacy: an assembly of two stacked blocks, a buttressed arch, and a dome. In all cases, we adopt both optimisation problems to clarify the main differences between the two approaches using quadratic (QP — Eq. 4.7) or linear (LP — Eq. 4.8) programming. Additionally, we will show how to explore/select different internal stress states on the reference configuration and how to define the limit state for increasing horizontal loads.

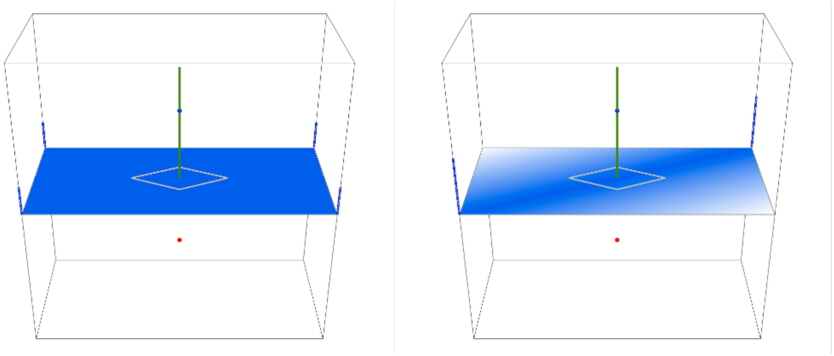
6.1.1 Two stacked blocks

In this subsection, we look at simple assemblies composed of two stacked blocks to illustrate some features of RBE. The first analysis illustrates the role of the normal contact forces by considering two blocks having a horizontal interface in different scenarios obtained by horizontally translating the upper block. With the second analysis, we show the role of the tangential contact forces illustrating how the friction capacity can be taken into account using both *friction-net* and *friction+* strategies.

Horizontal interface

In Figure 6.1, we look at an assembly composed of two vertically stacked blocks with a horizontal contact interface. The upper block, whose centroid is denoted with a blue dot, is a free block, while the bottom block, with a red dot, is assumed to be a support. Figure 6.1 shows a first RBE analysis considering a fully connected interface. The solutions of both QP (Figure 6.1a) and LP (Figure 6.1b) problems return vertical nodal forces only. As one can notice, the distribution of the nodal forces obtained with the QP is different from the one with the LP problem. In this case, we also highlight the different nodal force distributions over an interface by using a colour gradient.

Nonetheless, if we reduce the nodal forces to their resultants (in green), one can see that the same solutions are obtained. These solutions are trivially



(a) QP

(b) LP

Figure 6.1: Two blocks stacked vertically with a horizontal contact interface: nodal forces and resultants using the QP (a) and LP (b) formulations. The interfaces are depicted using a blue colour gradient to differentiate nodal-force distributions. Resultants (in green) from both QP (a) and LP (b) formulations are the same

in equilibrium, with the resultants going through the centre of mass of the upper blocks.

If we horizontally translate the upper block, as shown in Figure 6.2, we always obtain a different distribution of the normal contact forces from the QP and LP problems. However, their resultants are the same as they indeed depend on global equilibrium only which remains the same.

This observation also can be noted in the last analyses (Figures 6.2e and f) where the vertical projection of the centre of mass of the upper block is lying outside the interface; the tensile nodal forces appear because of the penalty formulation expressed by Eq. 4.3. This last result represents a key feature of the RBE approach. Despite other equilibrium approaches adopted in the limit analysis framework, RBE goes beyond *Yes/No* answers coming from the *feasibility/infeasibility* of an optimisation problem. Indeed, RBE allows us to describe unstable solutions and, thus, to identify interfaces subjected to tensile forces. In this sense, the penalty formulation enlarges the space of admissible stress fields.

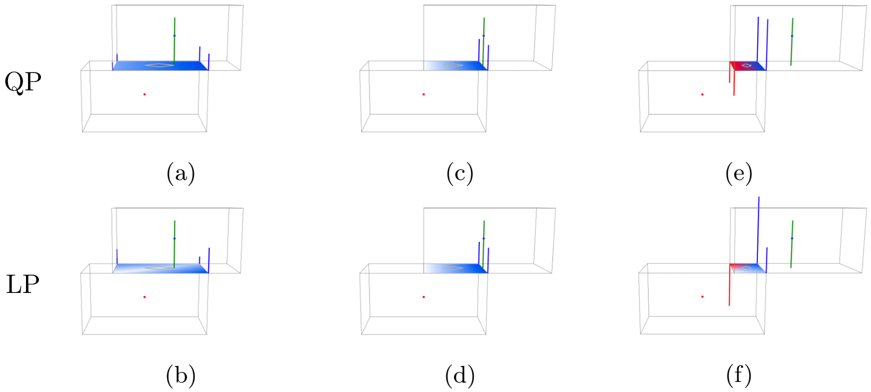


Figure 6.2: Equilibrium results from the RBE when the upper free block is horizontally translated, both QP (a, c, e) and LP (b, d, e). The problem is still feasible even when the upper block is in a unstable condition (e, f) thanks to the tension contact forces (in red).

Inclined interface

In this section, to clarify the main features of both *friction-net* (Eq. 4.4) and *friction+* (Eq. 4.5), we look at an assembly composed of two vertically stacked blocks having an inclined interface. The inclination of the interface is 23.6°, so the minimum friction value needed to guarantee the static equilibrium of the assembly is 0.44. The results are visualised in terms of nodal contact forces, normal and tangential resultants forces, and global resultants (in green).

It is especially worth pointing out that the friction constraint Eq. 4.6 is defined on the interface nodes. In this sense, the friction value used is a *local friction coefficient*. Therefore, we post-process the friction capacity of an interface as a *global friction capacity*, that is, in terms of normal and tangential interface resultants, to better understand the physical behaviour. This does not affect the results; if the friction constraint is fulfilled locally (i.e., on interface corners), namely:

$$\left|\, f^{i}_{jkt} \,\right| \leq \mu\, f^{i+}_{jkn}\,, \quad \forall i \in \{1 \ldots m_{jk}\},$$

then, it is also satisfied globally (i.e., over the whole interface):

$$\sum_{i=1}^{m_{jk}} \left|\, f^{i}_{jkt} \,\right| \leq \mu \sum_{i=1}^{m_{jk}} f^{i+}_{jkn}\,.$$

To also overcome potential infeasibility states due to the *friction-net* approach, one can adopt a *local friction* value to be used in the optimisation and define a threshold to detect if the interface's *global friction capacity* is overcome. In our opinion, the *friction+* approach is consistent with the RBE objective function since penalising the friction forces provides an equilibrium solution with the smallest tangential forces.

Figure 6.3a reports the RBE results using the *friction-net* approach. We assume a local friction value of 0.8 and 0.6 as a global threshold for evaluating the friction capacity. In this case, the interfaces are depicted in yellow since the global friction capacity is not overcome. Supposing a smaller global threshold is chosen (say 0.2) such that the friction capacity is overcome, the solution (in terms of both nodal forces and resultants) is trivially the same, but the interface would appear in red. The use of the *friction+* approach does not change the results if properly coupled with a threshold value. Conversely, if the local friction value is set as 0.2, with the *friction-net* approach, the problem becomes infeasible as the friction constraints (Eq. 4.4) are not satisfied anymore. On the other hand, when using the *friction+* approach, the problem is still feasible since, as shown in Figure 6.3b, the tangential forces affect the normal force solution because of the penalty formulation.

Indeed, Figure 6.3b shows the RBE results using the *friction+* approach and a local friction coefficient equal to 0.2. As one can see, but more evident in the second case, with both QP and LP approaches RBE finds a tensile tangential force in a corner (orange circle in Figure 6.3b). In particular, referring to Eq. 4.5, at that corner, RBE finds both a non-zero positive f_{kn}^{i+} and negative f_{kn}^{i-} part, which results in a negative normal component f_{kn}^{i} (tensile force) with $f_{kn}^{i+} < f_{kn}^{i-}$. In this sense, the friction constraint, written as a function of the positive part, is still satisfied.

Remark 1. It is worth mentioning that when no tensile force appears (i.e., f_{kn}^{i-} is zero), Eqs. 4.4 and 4.5 become identical; hence, *friction+* and *friction-net* provide the same solution.

Discussion

By looking at the nodal-forces distributions, in all cases, the LP solution is different from the QP. In particular, the QP approach tends to activate all nodes selecting solutions whose nodal forces have to be distributed as smooth as possible. Moreover, while every solution obtained with the QP approach can be a solution to the LP problem, the contrary may not happen. Indeed, the value of the QP objective function in the case of Figure 6.1b is greater than the one assumed for the solution depicted in Figure 6.1a, while the LP objective function provides the same value for both solutions. For this reason, it is easy to see that the LP's solution space is larger compared to the QP.

The QP approach selects a nodal force distribution that comes from an elastic interpretation of the equilibrium problem [Angelillo et al., 2010]. The LP formulation, instead, considers the interface as rigid, which then means that infinite admissible stress states are possible [Iannuzzo et al., 2020]. From a limit analysis point of view, there is no reason to prefer one over another, since both represent admissible solutions. Furthermore, in terms of interface resultants, both are the same since the problem is statically determined.

In the second benchmark, we have analysed two blocks with an inclined interface to illustrate the main differences between the *friction-net* (Eq. 4.4)

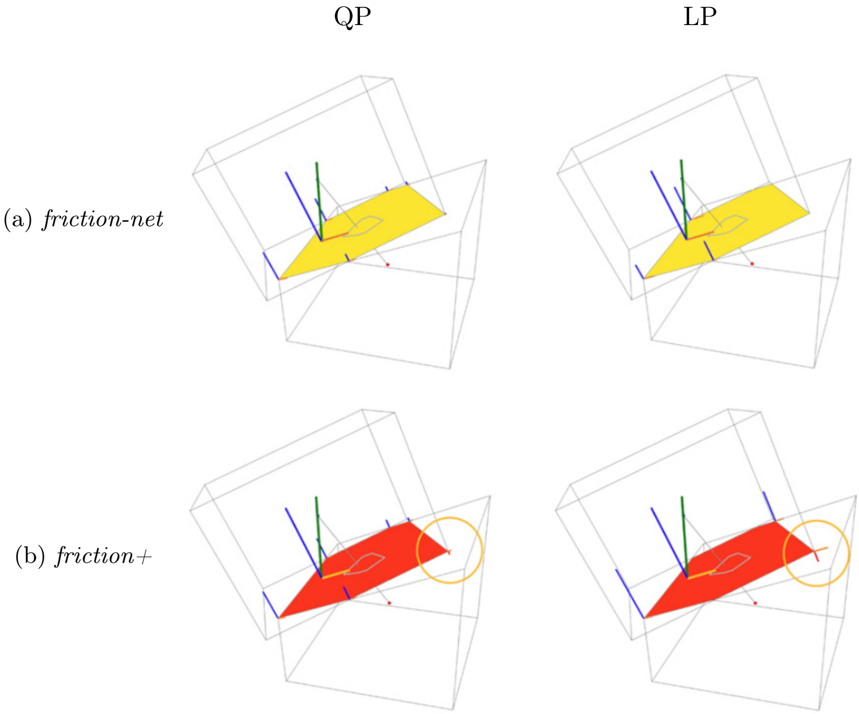


Figure 6.3: Solution of the QP and LP optimisation problems with the *friction-net* (a) and *friction+* (b) approaches. Two blocks stacked vertically but with an inclined interface (23.6°); (a) the friction coefficient is equal to 0.8 and the threshold is set to 0.6: the interfaces are depicted in yellow, meaning that the friction capacity is not overcome; and, (b) the friction coefficient here is assumed equal to 0.2: The interfaces are depicted in red, meaning that the friction capacity is exceeded everywhere.

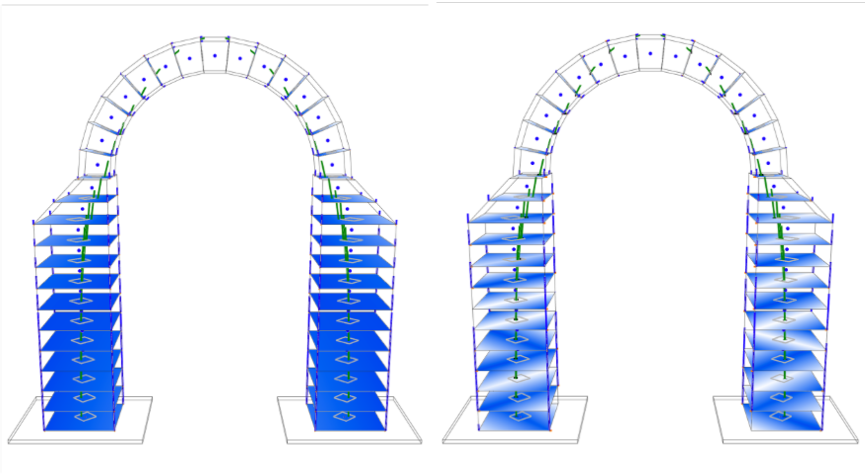
and *friction+* (Eq. 4.5) approaches, solved using QP and LP (Figure 7). Both are valid strategies, particularly when coupled with a threshold to evaluate global friction capacity. The main difference is that when the local friction value is overcome, the problem is infeasible with the *friction-net* approach, while with the *friction+* approach it is still feasible but the normal force distribution is affected by the tangential behaviour because of the penalty formulation. In this sense, the *friction+* approach seems more inline with RBE's aim. Physically, we can imagine that the *friction+* approach activates a tensile capacity (as a glue), which also provides a shear strength. Nonetheless, for many masonry problems, the *friction-net* approach is more useful when the aim is to explore the feasibility of the problem to catch ultimate equilibrium states. For this reason, in what follows, we adopt the *friction-net* strategy assuming 0.8 for the local friction coefficient and 0.6 as the global friction threshold.

6.1.2 Semi-circular arch on buttresses

In this section, we look at a semi-circular arch on buttresses to illustrate how RBE can be applied to assess unreinforced masonry structures. The geometry of the buttressed arch is the same as the one used in Iannuzzo et al. [2020] where it was analysed to show the peculiarities of the piecewise rigid displacement (PRD) method, an energy-based limit analysis approach [Iannuzzo et al., 2018; Iannuzzo, 2019]. The semi-circular arch has an internal radius of 1.0 m, a thickness of 0.3 m, an orthogonal depth of 0.5 m and is discretised into 15 voussoirs. The two buttresses have a height of 2.5 m, a base of 0.7 m, a depth of 1.0 m and are partitioned into 12 elements. The two bottom blocks of the buttresses are considered supports. The structure has a uniformly distributed mass density $\rho = 1800\,\mathrm{kg/m^3}$, and we assume 0.6 as friction threshold. In this Subsection, we perform three studies. First, the initial geometry is analysed. Second, we show how to explore different equilibrium solutions. Third, we perform a tilting test of the structure.

Initial configuration: LP vs QP solutions

The first study looks at the buttressed arch in its initial configuration. Figure 6.4 shows the solutions obtained solving the QP and LP problems, respectively. As in the previous examples, the nodal force distributions, highlighted by the blue colour gradient, are different. Nevertheless, while the two stacked blocks of Section 6.1.1 can have only one equilibrated solution in terms of interface resultants, as the buttressed arch is statically indeterminate, the interface resultants are different. This corresponds to different thrust lines possible for the same arch [Heyman, 1969]. Both results are almost identical though. Indeed, both QP and LP problems return solutions close to the minimum thrust. Moreover, as one can observe, thrusts in the buttresses are everywhere within the kern, meaning that all buttress interfaces are activated fully in compression.



(a) QP

(b) LP

Figure 6.4: A buttressed arch in its initial reference configuration: both QP (a) and LP (b) solutions return a thrust line which is linked to a minimum thrust condition of the semi-circular arch. The interface resultants in the buttresses are everywhere within the corresponding interface kern

Explore different admissible equilibrated stress states

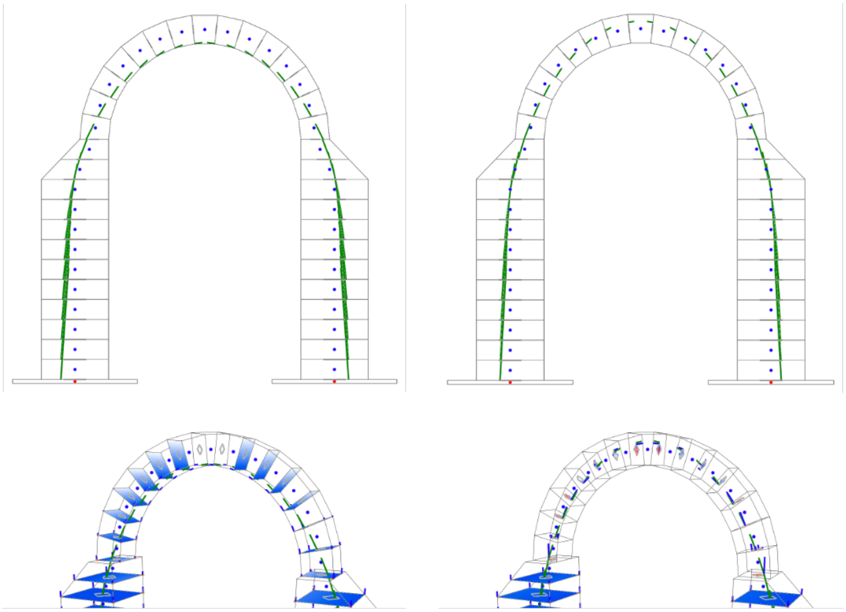
In this second study, we illustrate a procedure that can be used to explore different admissible, singular, internal, stress states. The idea is to consider new virtual interfaces, which can be obtained by tightening the original interface and writing the equilibrium equations (4.3) for these new (reduced) interfaces. Following this approach, we are looking for a thrust line solving the EP that either crosses or is as close as possible to the new, reduced interfaces. A similar procedure was adopted in Iannuzzo et al. [2020] using the PRD method. While in Iannuzzo et al. [2020] one we can get only a *Yes/No* answer, with the RBE approach, because of its penalty formulation, one can also select a solution that does not fit the prescribed requirement completely. In this case, tensile forces on the nodes of the reduced interfaces can appear, meaning that the thrust line cannot be contained within the reduced interfaces.

In Figure 6.5a, the key-stone interface is tightened to explore a maximum thrust condition for the semi-circular arch, which leads the buttresses in the worst working condition, with the interface resultants outside the kern at the bases of the buttresses. Figure 6.5b shows the results obtained by enforcing the thrust line to go through the kern of all interfaces of the arch (kern-fitting). Both the QP and LP problems return almost identical interface resultants, so we only visualise the resultant forces.

Moreover, using the *friction-net* approach, it is possible to select an internal stress state that fulfils particular requirements on the friction capacity. Indeed, if we chose a low value for the friction coefficient (thus, without adopting a threshold strategy), the problem can become infeasible. The lowest value of the friction angle for which the problem is still feasible represents the minimum friction value required by the assembly to be in static equilibrium. In this case, the minimum value is found to be 0.31. In Figure 6.6a, the solutions of the corresponding equilibrium problems are depicted both in terms of nodal forces and resultants. It can be noted that the resultants are everywhere within the reduced friction cone.

Tilting test: horizontal external forces

A common way to look for the stability of masonry structures when subjected to horizontal static forces is to perform a tilting test [Block et al., 2006b]. In this third study, we model this scenario by rotating the gravity vector until tensile forces appear. The maximum value of the tilting angle

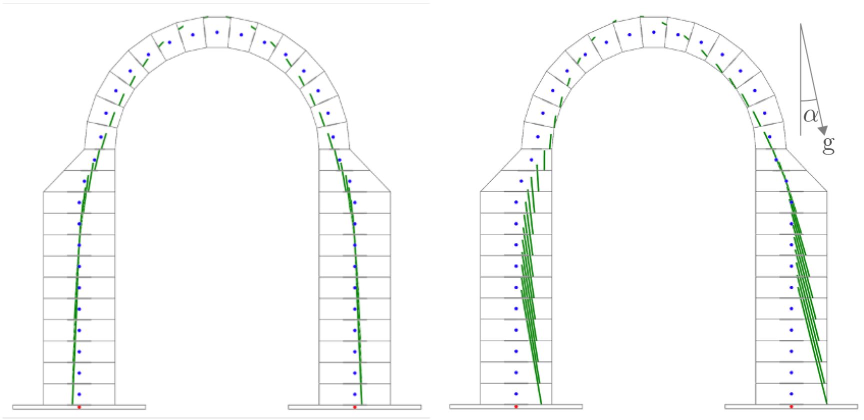


(a) Maximum thrust

(b) Arch kern-fitting

Figure 6.5: In (a), the mid-span interface of the semi-circular arch is reduced to select an admissible internal stress state for which the thrusts exerted on the buttresses are maximised. The same solution can be obtained by reducing the two base interfaces of the arch (towards the extrados). In (b), the interfaces of the semi-circular arch are virtually reduced to their kerns to seek if there is a thrust line which fully activates the arch in compression. As one can observe, the thrust is outside the mid-span interface. For both examples, the results of the QP and LP are almost identical, so we only visualise the resultant.

for which the thrust line is everywhere within the structural domain is 12 degrees, which corresponds to a horizontal static multiplier of 0.21, which is very close to the one found in Iannuzzo et al. [2020]. In Figure 6.6b, the solution is depicted: it touches the structural boundary in four points, three on the arch and one at the base on the right. These four points suggest a mechanism that is the same as the one found in Iannuzzo et al. [2020].



(a) Minimum friction

(b) Tilt test

Figure 6.6: In (a), using the *friction-net* approach, RBE returns 0.31 as the lowest value for the friction coefficient for which the assembly is still in equilibrium. The resultants are everywhere in the friction cone. In (b), the buttressed arch subjected to a tilting test: the maximum angle for which the resultants are everywhere within the structural geometry is for a tilt angle α of 12°.

Discussion

In this subsection, looking at a buttressed arch, we perform RBE analyses using both QP and LP approaches: assessing the initial configuration, exploring various internal equilibria, finding the minimum friction, and evaluating the maximum horizontal static multiplier (i.e., tilting angle) for which the assembly is still in equilibrium. All analyses show that different nodal forces can be obtained in solving the QP and LP problems, confirming what was discussed in Section 6.1.1. Except for case of stacked blocks, which is globally (in terms of interface resultants) statically determined, the buttressed arch is globally statically indeterminate, and for this reason, when we look at the initial configuration of the QP and LP solutions, in terms of resultants, are slightly different. Nonetheless, the QP and LP solutions are the same when approaching a limit state (e.g., at the collapse during tilting test) since the solution is unique and using the *friction-net* approach is dictated by the feasibility of the problem. In the following subsection, we will show how increasing the global indeterminacy of the system, the differences between QP and LP can get bigger. Nonetheless, in any case, solutions coming from both optimisations represent two statically admissible stress fields, and in this sense, safe solutions in the spirit of the Safe Theorem.

6.1.3 Hemispherical dome

In this subsection, we look at a hemispherical dome with an oculus. It illustrates how to apply RBE to assess three-dimensional unreinforced masonry structures. The centre-line radius R of the dome is 5 m while the thickness t is assumed to vary linearly from the bottom base (0.5 m) to the top part (0.25 m). The radius r of the oculus is 1 m.

The dome is discretised using 14 meridian slices and 10 parallel rings; therefore, the number of blocks is 140. The bottom ring blocks are assumed to be supports (Figure 6.7). The structural complexity of the dome is bigger than the one of the buttressed arch since, in the dome case, the graph $G(V, E)$ describing the data structure is two-dimensional. Similar to Subsection 6.1.2, we perform three studies. First, the initial geometry is analysed using both QP and LP. Second, we use kern-fitting and inner/outer base-fitting analyses to explore different equilibrium solutions. Third, we assess the dome subjected to horizontal action and define the maximum value of the horizontal multiplier.

Initial configuration: LP vs QP solutions

In the first study, we propose an RBE analysis of the reference configuration without considering any further constraints. Figure 6.8 shows results from the QP and LP optimisations. The interfaces are still coloured using a blue the gradient colour map. However, the white interfaces denote sections without any nodal force, or with forces less than a threshold value fixed as 10^{-2} of the maximum compressive force. Even though RBE can provide stress

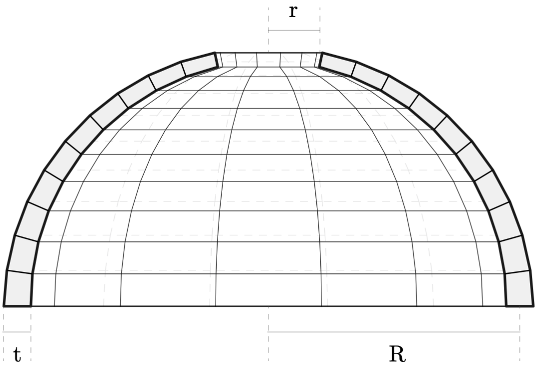
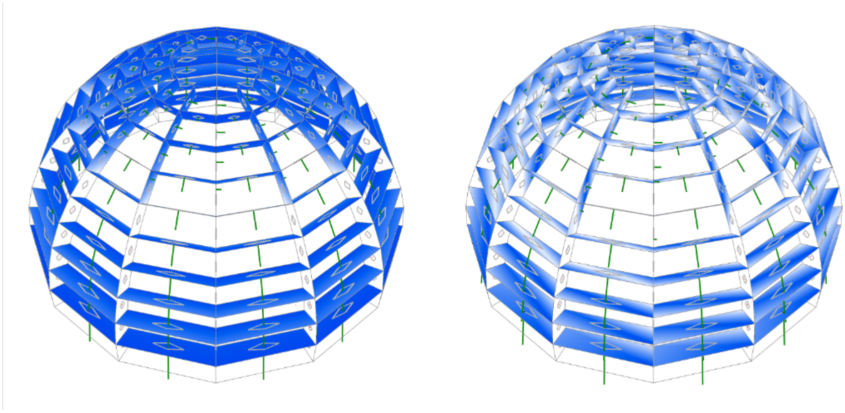


Figure 6.7: Cross-section and discretisation of the hemispherical dome: main dimensions. The mean radius R is 5 m; the radius of the oculus r is 1 m while the thickness t is assumed to linearly vary from 0.5 m (base) to 0.25 m (oculus).

solutions with tensile forces, it returns a purely compressive internal stress state for which part of the meridian interfaces are affected by zero hoop forces, which is the to-be-expected behaviour, as shown in Heyman [1997]. Nonetheless, the QP and LP solutions show different nodal force distributions, which illustrates how the difference in terms of interface resultants is more evident in the present case than in the previous ones (Figure 6.8). Moreover, the QP resultants are radial-symmetric, while the one from LP is not.

Exploring different equilibrium states

In this second study, we explore different internal stress states. The first analysis is what we call a kern-fitting analysis; that is, we look for an internal admissible stress field everywhere within the kern of interfaces. After the kern-fitting analysis, we look for inner and outer base-fitting analysis to measure the dome's stability under its self-weight. The results relate to the geometric safety factor [Huerta, 2006b]. In Figure 6.9a, the results of a



(a) QP

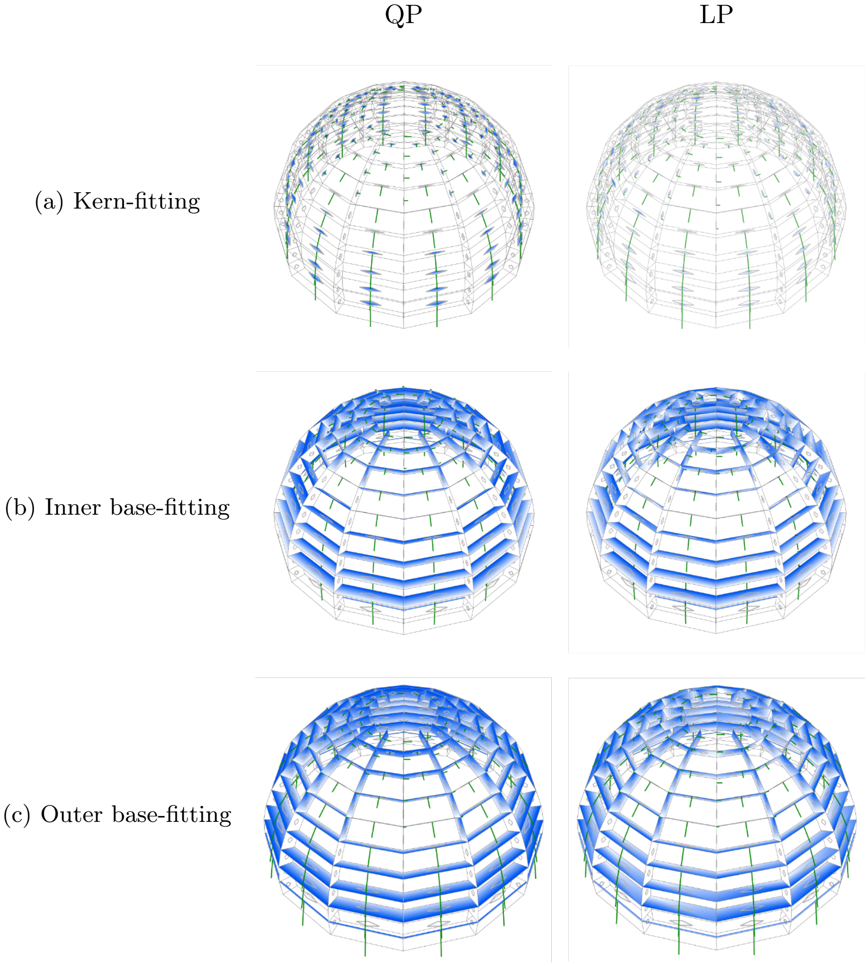
(b) LP

Figure 6.8: RBE analysis of the reference configuration: QP (a) and LP (b) solutions. The difference in interface gradients shows that these two solutions are locally different. Moreover, also the interface resultants are different, with the QP solution being radial-symmetric. The white interfaces denote zero hoop-force regions.

kern-fitting RBE analysis are depicted: the interfaces are virtually reduced to their kern to determine if an internal stress field is found, so one that fully activates all contacts. Figures 6.9b and c show the results of two analyses aimed at exploring admissible stress fields (e.g., min/max thrust states of the dome) that are as close as possible to the inner and outer parts of the supports, respectively. These results are obtained by virtually shrinking the bottom interfaces towards either the inner or the outer surface.

Tilting test: horizontal external forces

In the third study, as for the buttressed arch, we perform a tilting test to explore the maximum allowable capacity of the dome subjected to horizontal actions. The maximum tilting angle found with RBE on both LP and QP analyses is 32°. Figure 6.10 shows the flow of the resultants within the structure. With the *friction-net* approach, the problem becomes infeasible



QP

LP

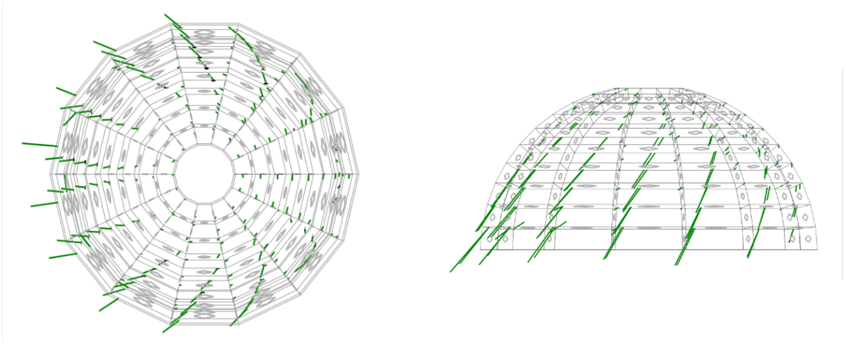
(a) Kern-fitting

(b) Inner base-fitting

(c) Outer base-fitting

Figure 6.9: In (a), the results of the kern-fitting RBE analysis. The interface resultants are enforced to go through all interface kerns. In (b), the resultants are enforced to go through the inner part of the supports. In (c), the resultants are enforced to go through the outer part of the supports. In all cases, QP and LP problems return two very similar solutions, particularly if one looks at the non-zero hoop forces areas.

before tensile stresses appear (we adopt 0.6 as the local friction value to be consistent with the threshold used in all analyses).



(a) Top view

(b) Side view

Figure 6.10: Horizontal capacity analysis: the maximum tilting angle is 32°, obtained with both QP and LP approaches.

Discussion

In this subsection, looking at a dome with an oculus we have explored different equilibrium solutions on the reference configuration and also checked its maximum capacity under increasing horizontal static actions. First, looking at the RBE analyses proposed in the first and second studies, it is worth noting that even though RBE is based on a penalty formulation, which allows tensile forces, it always returns (when admissible) solutions part of the meridian interfaces having zero contact forces, so meaning that there are no hoop forces in those parts. Secondly, we should point out that the structural complexity of a dome is higher than the one of a buttressed arch, because the network of the graph $G(V, E)$ is two-dimensional. From this aspect, the hemispherical dome represents a problem of higher statical indeterminacy, which also affects the output of the QP and LP analyses, showing larger differences if compared to the buttressed arch. We have shown how different singular, internal stress states can be explored by using virtual interfaces in RBE. This aspect can be addressed by changing the objective function to one that takes the value of the force into account only in certain points (e.g., at the supports) and/or directions (e.g., horizontal, vertical, tangential, etc.). We have explored the capacity of the dome subjected to increasing horizontal static actions. We end the incremental analysis when the problem becomes infeasible. The objective function does not play any role in this case because only the optimisation constraints determine the feasibility.

6.1.4 Discussion

Throughout these RBE studies, we understand the structural behaviour of discrete assembly better. These findings can also be applied to the CRA method, as CRA can be seen as an improved and corrected version of RBE. This section summarises the primary outcome of these findings into two main discussion points.

Firstly, we have proposed and compared two objective functions, a linear (LP) and the original quadratic (QP) one, to illustrate the RBE features. The primary outcome is that both provide statically admissible solutions in the spirit of Limit Analysis, meaning there should not be a reason to prefer one over the other. To illustrate this concept, one can look at Figure 6.1, in which the two blocks are in contact at four points. It is possible to interpret the QP and LP results by making a parallelism with a four-legged stool supported on the ground (an example often used to explain the limit analysis approach [Heyman, 1997, 2019]). The four-legged stool is a statically indeterminate system; thus, infinite solutions are possible depending on the actual contacts between the table legs and the ground, which are unknowable in reality. The QP solution reported in Figure 6.1a represents an ideal elastic solution, meaning that the contact is assumed to happen in four points. All nodal forces are as distributed as possible since the objective function is quadratic, which can be correlated to linear-elastic interface energy.

Conversely, the LP solution shown in Figure 6.1b is just one of the infinite admissible solutions. The equilibrium is guaranteed using only two contact points. Interesting to note that this is the "real" solution of a four-legged stool that is rocking, see Heyman and Hambly [1996] Hambly's Paradox. Furthermore, in Subsection 6.1.1, the solution provided by the LP objective function does not minimise the QP problem. On the other hand, QP solutions are always solutions to the LP problem. In this sense, the LP objective function further increases the solution space provided by the QP optimisation. Despite the force distribution being indeterminate, the structure is statically determined if we look at the global assembly of the two blocks. Therefore, the solution in terms of interface resultants has to be the same regardless of the objective function. As the structural indeterminacy grows, the solutions of the QP and LP approaches start getting different from each other (Subsection 6.1.2 and 6.1.3). The higher the indeterminacy, the greater the differences are. Indeed, in the dome case, the LP and QP solutions show the largest differences, even in terms of resultants. Finally, suppose the aim is to select a "smooth" solution. In that case, the QP provides elastic solutions that are as distributed as possible, which also reflects in fully symmetric solutions for symmetric models. If the aim is to englarge the QP solution space, the LP can provide admissible solutions consistent with a perfectly rigid model.

Secondly, we have shown that the RBE method can be used to explore different equilibrated states of an indeterminate structure. We utilise the concept of new virtual interfaces, which can be obtained by tightening the original interface and writing the equilibrium constraints for these virtual interfaces. By introducing the correct definition of the kern of a section, we can look for admissible solutions that fully activate a set of selected interfaces. This aspect is crucial when assessing a structure's stability through estimating its geometric safety factor. Finally, even if RBE is based on a penalty formulation, we have demonstrated that this does not affect the search for a limit state. Specifically, when the structure is in a limit condition, the space of solutions may include only one element if the solution is unique (see the third study of tilting test analyses in Subsections 6.1.2 and 6.1.3).

6.2 Numerical benchmarks

In Section 6.1, we discussed how to understand the RBE results and used RBE-based approaches to assess some simple masonry structures. However, as discussed in Chapter 5, RBE fails to give correct analysis results for specific geometries, especially for sharp *wedges* and complex shapes. Therefore, we perform several benchmark examples in this section to demonstrate CRA's accuracy.

We implemented CRA in a Python-based code using Pyomo as open-source modelling language [Hart et al., 2017] and IPOPT as solver for the nonlinear constrained optimisation problem [Wächter and Biegler, 2006]. Additionally, we utilised COMPAS, an open-source computational framework for collaboration and research in Architecture, Engineering, Fabrication, and Construction [Van Mele et al., 2021], as our base for the data structure such that our method can be easily integrated into any desired CAD software. The results are visualised with OpenGL. All examples and benchmarks were performed on a MacBook Pro with a 2.9 GHz 6-Core Intel Core i9 Processor and 16 GB memory. All implementation details will be discussed in Chapter 8.

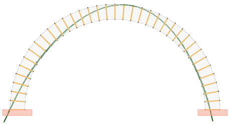
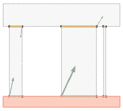
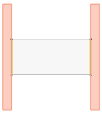
In this section, looking at geometric models with increasing complexity, we benchmark the CRA method against specialised engineering software such as 3DEC for the DEM analysis [Itasca, 2013] and Sofistik for the nonlinear FE analysis [SOFiSTiK, 2012]. In particular, the benchmarks will consider the limit state scenarios, such as finding the maximum tilting angle for which a structure is still stable. We will show how the use of CRA allows a correct evaluation of the limit state and aligns with the results obtained with commercial software.

6.2.1 Arch, 3-legged π, and model H

The first benchmarks regard the three simple examples shown in Yao et al. [2017] and Shin et al. [2016]. Table 6.1 reports the results from these previous works along with the ones obtained using CRA, 3DEC and an in-house implementation of RBE. All numerical methods, except for Variational Static Analysis (VSA) proposed by Yao et al. [2017], return the same limit tilting angle obtained from analytic solutions. For the 3-legged π example, the RBE results proposed by Shin et al. [2016] are a bit lower (0.4°) than the one obtained from our RBE implementation, which might relate to small differences due to remodelling. While VSA is still conservative compared to other numerical methods, Yao et al. [2017] justified its conservative results as they are closer to small-scale model results, and they attributed the RBE overestimation to the improper treatment of the sliding. Conversely, Shin et al. [2016] stated that the difference between physical and numerical results was related to the imperfections for the small-scale 3D-printed models, which might be taken into account through an arch thickness reduction of 13% [DeJong et al., 2008]. Small-scale physical models are always affected by imperfections. According to their size, these can drastically alter the internal stress state and consistently lower the stability values [Dell'Endice et al., 2021; Atar et al., 2020]. In Remark 2, we will show that the failure of the semicircular arch in Table 6.1 is not due to the improper treatment of sliding in RBE. In fact, the RBE result is theoretically correct in that specific case, i.e., for the perfect geometry. In addition, the RBE results for the maximum tilting angle of the masonry arch are aligned with well-known ones available in the literature [Huerta, 2006a; Blasi and Foraboschi, 1994; Heyman, 1969; Oppenheim, 1992; Como, 2013]. For evaluation of the theoretical limit state, CRA provides accurate numerical results that align well with engineering software 3DEC.

Remark 2. Here, we discuss the *arch thick* model reported in Table 6.1. The semicircular arch (Figure 6.11a), with a thickness-to-radius ratio $t/r = 0.15$, is discretised in 36 voussoirs, and the friction angle is assumed equal to

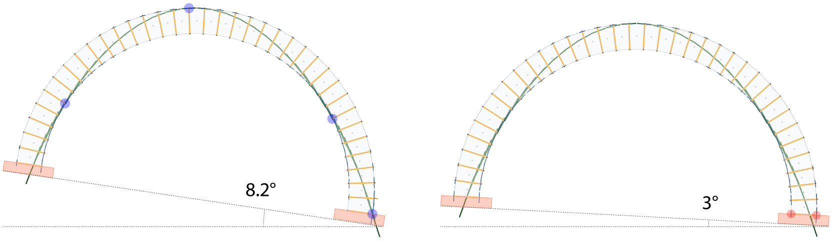
Table 6.1: Benchmark table provided in Yao et al. [2017] adding Shin et al. [2016], here extended to also include 3DEC, CRA, and an in-house version of RBE as solvers. We use the same friction angle of 43° as in all previous works. In these three examples, CRA aligns well with both analytic and engineering software solutions.

				
		Arch thick (t/r = 0.150)	3-legged π	Model H
	# Free blocks	36	4	1
	# Fixed blocks	2	1	2
Stability at tilt angle 0°	Analytic	Stable	Stable	Unstable
	Experiment (Shin et al.)	Stable	Stable	-
	RBE (Shin et al.)	Stable	Stable	Stable
	Inverse FEM (Shin et al.)	Stable	Stable	-
	VSA (Yao et al.)	Stable	Stable	Unstable
	3DEC	Stable	Stable	Unstable
	Our RBE implementation	Stable	Stable	Stable
	CRA	Stable	Stable	Unstable
Critical tilt angle θ°	Analytic	8.2	-	N/A
	Experiment (Shin et al.)	4.7 ± 0.2	14.3 ± 0.2	N/A
	RBE (Shin et al.)	8.2	19.1	N/A
	Inverse FEM (Shin et al.)	8.2	19.1	N/A
	VSA (Yao et al.)	5.3	16.4	N/A
	3DEC	8.2	19.5	N/A
	Our RBE implementation	8.2	19.5	N/A
	CRA	8.2	19.5	N/A

43°, as in Shin et al. [2016]. The RBE tilting test shows a maximum tilting angle of 8.2°. In green, the internal stress state is represented through resultant forces, which combine the nodal forces of each interface. Note that at this point a 4-hinge mechanism is formed where the “thrust line” touches the structure’s envelope alternatively at the intrados and extrados. We label interfaces with red spheres when the nodal forces are in the limit state, i.e., lying on the boundary of the Mohr-Coulomb friction cone. The resultants are everywhere within the friction cone as no interface is labelled in red. Thus, the limit friction capacity of the arch is not reached, and sliding does not occur. To show how the friction capacity can influence the result, we consider a reduced friction angle equal to 21.8° in Figure 6.11b. In this case, the RBE analysis shows that the maximum tilting angle is 3.0°. In this case, the bottom right interface of the arch with the support is labelled in red, meaning that the maximum friction capacity is reached, and sliding occurs. The lower friction capacity restricts the solution space resulting in a lower maximum tilting angle. Indeed, sliding could occur before a 4-hinge tilting collapse mechanism can form (as in Figure 6.11a) These analyses were carried out also with CRA and 3DEC obtaining the same results. Note that for these specific cases, RBE is able to correctly capture the failure mode. The difference with physical models is only due to imperfections and tolerances always affecting small-scale models.

6.2.2 The shelf model

To show how our methodology can correctly capture infeasible solutions and identify the unstable regions, we perform the analysis of the shelf example provided in Yao et al. [2017] (Figure 6.12). Similarly to 3DEC, VSA correctly captures the sliding behaviour of two unstable elements. RBE, however, incorrectly predicts the shelf as a stable structure (it is a mathematically correct but *physically unrealisable* solution, see our discussion in Section 5.1). Differently from RBE, CRA without penalty formulation (Eq. 5.6) correctly classifies the model as unstable, as the optimisation prob-



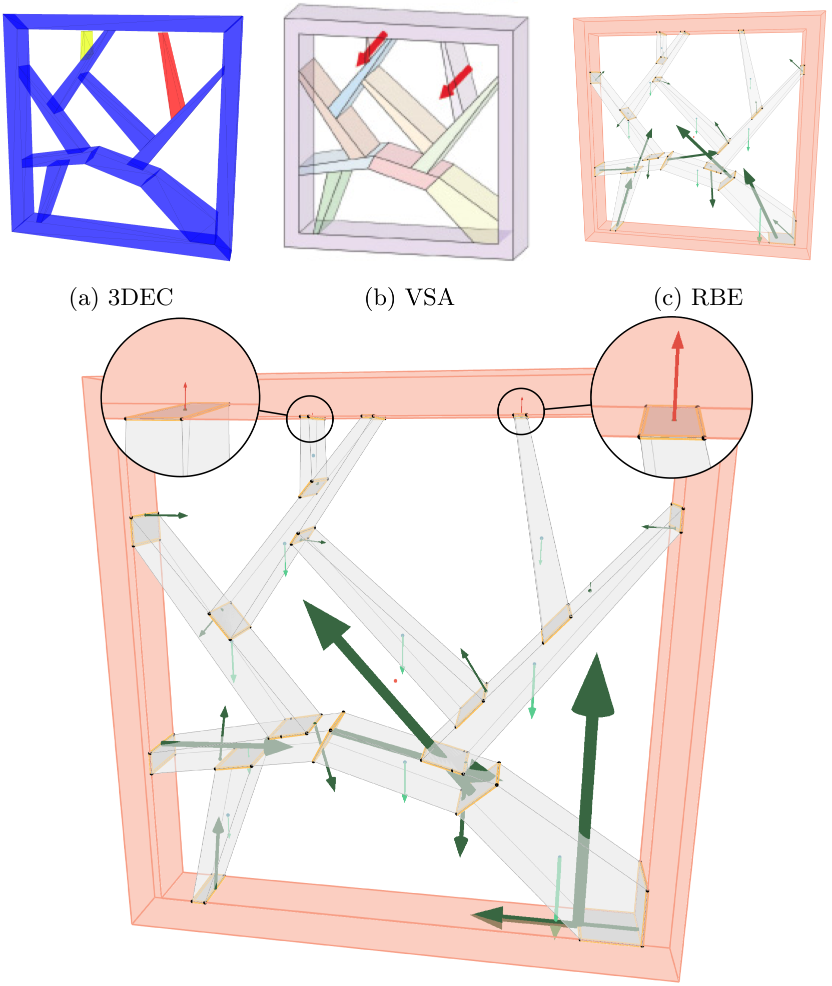
(a) (b)

Figure 6.11: *Arch thick* model: (a) with a friction angle equal to 43°, RBE returns a maximum tilting angle of 8.2°; while (b) using a reduced friction angle of 21.8°, the maximum tilting angle is 3°. The red spheres denote the interface where the friction cone constraint is violated, so where a potential sliding motion is allowed. The blue spheres denote the hinge mechanism where potential crack openings are allowed.

lem is infeasible. Applying the CRA penalty formulation (Eq. 5.8), the model becomes feasible. Additionally, the obtained result indicates the instability regions, identified by the extra tensile forces required to make the structure stable. Figure 6.12d shows that by providing a given amount of tensile capacity, represented by the red vectors, the existing shelf design is in equilibrium. Therefore, CRA provides: i) a correct assessment that the assembly is unstable, unlike RBE; and, ii) in addition to VSA, quantitative information on what could be done to stabilise the design thanks to its penalty formulation.

6.2.3 The wedge model

Classic rigid-block models have been developed for single-planar interfaces and have not been applied to non-planar or multi-planar interfaces, particularly for *sharp wedge* connections. To the best of the authors' knowledge, these scenarios were not previously addressed in any study. However, real structures are often designed with complex non-planar or multi-planar inter-



(a) 3DEC

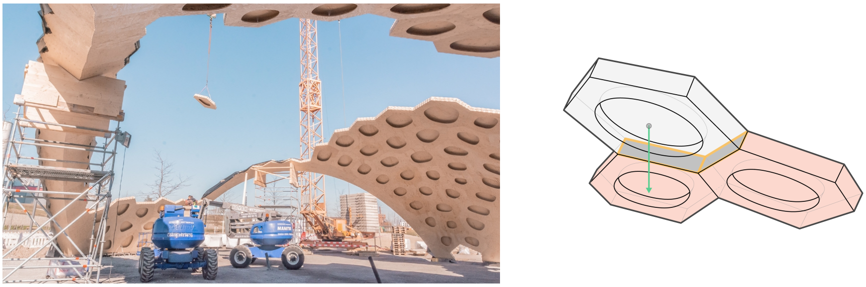
(b) VSA

(c) RBE

(d) CRA with penalty forces

Figure 6.12: *Shelf* model. (a) The 3DEC analysis shows two unstable pieces highlighted with their directions of motion. (b) VSA shows similar results as 3DEC (image taken from Yao et al. [2017]). (c) RBE incorrectly identifies the structure as stable. (d) The CRA not only correctly identifies unstable parts but also proposes the minimally required extra forces needed to make the assembly stable.

faces, see Figure 6.13a. With the third benchmark case, we want to address this problem. Figure 6.14 shows four types of *3-block wedge* models, similar to the one presented in Figure 6.13b: *type-a* and *type-b* have sharper wedge angles compared to *type-c* and *type-d*. Respectively, *type-a* and *type-d* have a lower centre of mass compared to *type-b* and *type-c*. We perform tilting tests with respect to three different axes to get the maximum tilting angles in Table 6.2. All models are rotated counterclockwise along the axis, and results are provided considering two different friction coefficients ($\mu = 0.2$ and $\mu = 0.84$). We want to point out that wedges with sharp angles (e.g., *type-a* and *type-b*) cannot be analysed with classic rigid-block models such as RBE, as they always return a feasible solution even in an upside-down configuration (e.g., model A in Section 5.1).

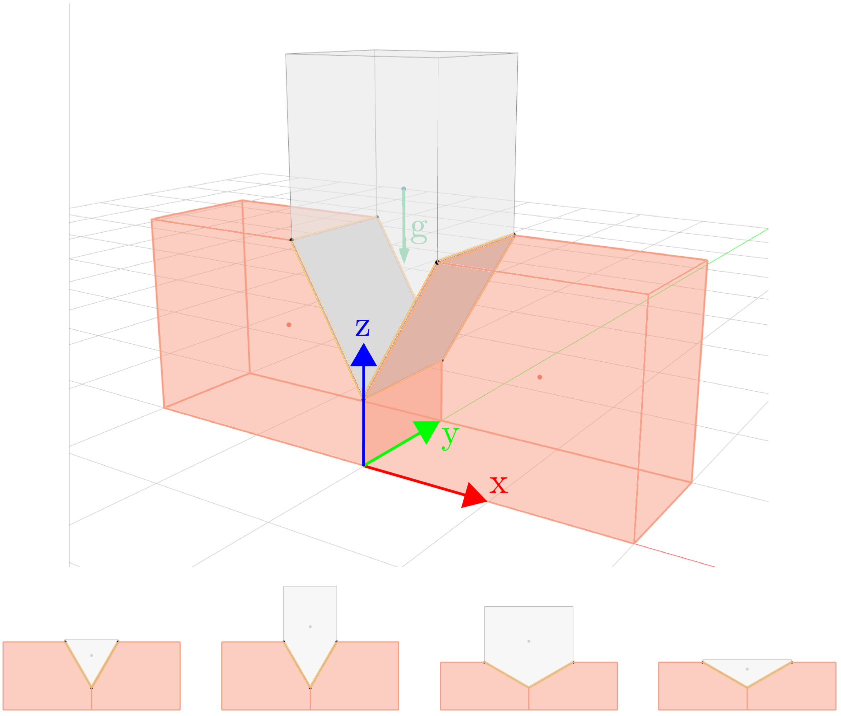


(a) (b)

Figure 6.13: Assembly during construction using *wedge* geometry. (a) Buga wood pavilion during construction Wagner et al. [2020]; in (b) an assembly that can be used to assess the local stability of the Buga wood pavilion (the two pink blocks are assumed fixed).

In the comparison table, CRA aligns well with 3DEC results except for *type-c rotate-xy30*. We observe that the different limit tilting angles come from the fundamental difference at the base of these two solvers. 3DEC is a dynamic solver that can capture the equilibrium on newly deformed configurations. Specifically, in the 3DEC analysis, between 40° and 48°, the free block slides slightly and settles, reaching equilibrium in a new, deformed configuration.

Performing a CRA analysis of the 3DEC deformed configuration, we get the same results, confirming that it is stable. Alternatively, we consider the first movement 40° as a failure tilting angle. In this sense, CRA is slightly conservative because it only considers the static equilibrium on the initial configuration. RBE, in contrast, seems to have bad predictions for *sharp wedge* angles and standard friction coefficients, as also pointed out in Section 5.1.1 with the *A* model and Section 5.1.2.



(a) type-a (b) type-b (c) type-c (d) type-d

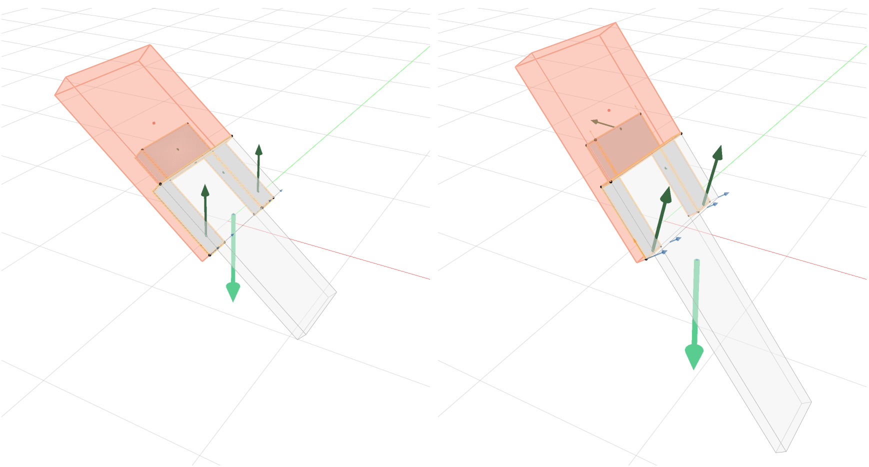
Figure 6.14: *3-blocks wedge* models. (a)–(d) are models with the XZ-plane view.

Table 6.2: Benchmark table for *3-blocks wedge* models in Figure 6.14. *rotate-x*, *rotate-y*, and *rotate-xy30* represent the results of the tilting tests using $(1, 0, 0)$, $(0, 1, 0)$, and $(\sqrt{3}, 1, 0)$ as rotation axes.

	Rotation axis	μ=0.20			μ=0.84		
Type	Angle (deg)	3DEC	CRA	RBE	3DEC	CRA	RBE
type-a	rotate-x	21.8	21.8	25.0	59.2	59.2	180.0
	rotate-y	71.3	71.3	71.3	100.0	100.0	180.0
	rotate-xy30	24.8	24.7	28.0	62.7	62.9	180.0
type-b	rotate-x	21.8	21.8	25.0	58.7	58.5	180.0
	rotate-y	67.4	67.4	67.4	109.1	109.1	180.0
	rotate-xy30	24.8	24.7	28.0	63.1	62.0	180.0
type-c	rotate-x	13.0	13.0	13.0	44.1	44.1	47.0
	rotate-y	41.3	41.3	41.3	64.5	64.4	65.7
	rotate-xy30	14.9	14.9	15.0	48.1	40.2	51.0
type-d	rotate-x	13.0	13.0	13.0	44.1	44.1	50.0
	rotate-y	41.3	41.3	41.3	70.0	70.0	70.0
	rotate-xy30	14.9	14.9	15.0	49.0	48.5	54.0

6.2.4 Interlocking connections

This subsection looks at simple assemblies with concave interfaces in order to benchmark CRA when the extended version of the data structure is used (Section 5.4). Figure 6.15a and b show two models with the same concave-shaped joint geometry but with different cantilever lengths. We compare the CRA results against 3DEC in a tilting test considering two friction values, with friction coefficients 0.10 and 0.84 corresponding to friction angles equal to 5.71° and 40.03°, respectively. As from Table 6.3, a very good agreement can be noted. In particular, the positions of the centre of mass of these two models affect the results. Indeed, while in the *concave-short* model, the failure is due to the friction capacity on the two lateral contact sides, the *concave-long* can activate an additional contact on the back that makes it more stable. Moreover, as soon as the model is tilted to the limit angle, the low-friction-angle configuration starts to slide, as expected.



(a) *concave-short* (b) *concave-long*

Figure 6.15: The maximum tilt of two concave shape joint geometries with different lengths of the cantilever. (a) and (b) are CRA results corresponding to the friction coefficient $\mu = 0.84$ in (c).

Table 6.3: Benchmark table for concave joints in Figure 6.15.

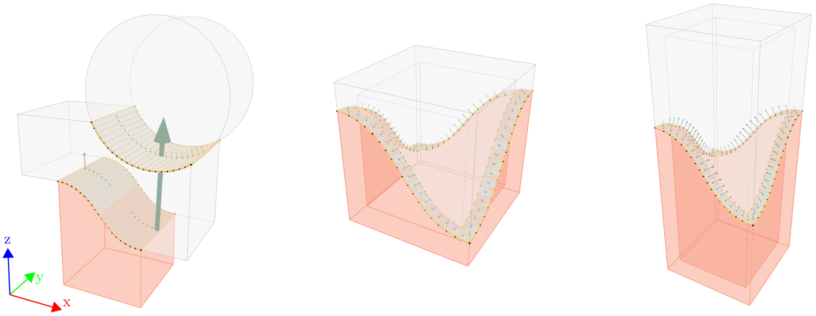
	Rotation axis	μ=0.84		μ=0.10	
Type	Angle (deg)	3DEC	CRA	3DEC	CRA
concave-short	rotate-x	40	40	5	5
concave-long	rotate-xy	56	54	10	9

6.2.5 Freeform connections

Lastly, we benchmark CRA on three types of freeform assemblies with curved interfaces against 3DEC and Sofistik. The models and the corresponding results are reported in Figure 6.16a, b, and Table 6.4. CRA directly discretises the original curved interface as described in Section 5.4 and Figure 5.4. Conversely, 3DEC cannot take the curved interfaces into account directly. To analyse assembly with curved interfaces, we proceeded with a preprocessing modelling phase cutting each block into convex sub-blocks, which were joined together later. As this is not a common problem to model in

3DEC, we decided to benchmark CRA also against the FE software Sofistik. Sofistisk similarly can handle curved interfaces but needs additional meshing steps to generate triangular surface and tetrahedral volume elements. Moreover, Sofistik considers only one normal versor for each vertex averaging the normal versors of the neighbouring faces. Table 6.4 illustrates the tilting tests performed considering three qualitatively different models and assuming various distinct tilting directions.

In Figure 6.16a, the *curve-3-blocks* model consists of three vertically stacked blocks with interfaces singly curved along the X-axis. In this case, we perform five tilting tests around different axes, and all CRA results align well with both 3DEC and Sofistik. In Figure 6.16b and c, we model curved interfaces in both X-axis and Y-axis. Both *cube-curve-short* and *cube-curve-tall* have the same interface and discretisation. They are different in terms of their centre of mass positions. Due to the symmetry of the shape, we perform only three different tilting directions for both *cube-curve* models. For the *cube-curve-short* model, CRA seems to align well with 3DEC results. On the other hand, Sofistik seems slightly more conservative, especially for the *rotation-xy* axis. For the *cube-curve-tall model*, CRA gives a more conservative angle in the *rotation-xny* axis.



(a) *curve-3-blocks* (b) *cube-curve-short* (c) *cube-curve-tall*

Figure 6.16: The maximum tilting angle of three types of non-standard assembly shapes with curved interfaces.

Table 6.4: Benchmark table for non-standard assembly with curved interface in Figure 6.16. All tilting angles are rotated anticlockwise according to a specific rotational axis, e.g., the *rotate-xy* represents the rotational vector $(1, 1, 0)$ and *rotate-xny* represents the rotational vector $(1, -1, 0)$.

	Rotation axis	μ=0.84			
Type	Angle (deg)	3DEC	Sofistik	CRA	RBE
curve-3-blocks	rotate-y	8	8	8	8
	rotate-ny	53	53	53	53
	rotate-x	24	24	24	26
	rotate-xy	11	12	12	12
	rotate-xny	46	47	47	48
cube-curve-short	rotate-x	72	70	70	180
	rotate-xy	68	64	69	180
	rotate-xny	80	79	79	180
cube-curve-tall	rotate-x	41	38	40	180
	rotate-xy	39	38	39	180
	rotate-xny	59	58	51	180

In general, CRA nicely captures the limiting state of different geometries and aligns well with commercial engineering software. The correct prediction of stability greatly enlarges the design space of discrete-element assemblies. In the next section, we will show examples of using CRA in a forward design process.

6.3 Stability-aware design process

As stated in previous sections, CRA can assess stability correctly, and its penalty formulation can provide users with additional information. Section 6.3.1 shows how CRA can be integrated into an interactive stability-aware design workflow, while Section 6.3.2 shows its potential to guide the designer towards a self-supporting shape. Lastly, Section 6.3.3 shows that using the stability-aware design process enables more extreme structural design that is impossible to achieve with a single-planar interface.

6.3.1 Workflow

We propose a stability-aware design process as summarised in the workflow in Figure 6.17. As input, the user-provided assembly includes geometry, mechanical parameters, boundary conditions, etc. The definition/choice of the boundary condition means selecting which blocks are considered as supports. In this sense, its use enables the possibility of locally assessing stability. After set up, the central analysis consists of two CRA formulations, with and without the penalty formulation, i.e., using optimisations (Eq. 5.6) and (Eq. 5.8) respectively. The algorithm first assesses the stability of the assembly using the optimisation problem (Eq. 5.6). If the assembly is unstable (i.e., infeasible), the algorithm further analyses it with the CRA penalty form (Eq. 5.8) to identify unstable regions. Once the unstable parts have been defined, the user can change the geometry accordingly to remove tensile forces until a satisfactory degree. The following section will show how to use a stability-aware design workflow to redirect design choices.

6.3.2 Interactive process towards stability

In this section, looking at the shelf model, we demonstrate the effectiveness of the proposed workflow. Optimisation problem (5.6) is infeasible, meaning that the shelf is unstable. CRA with penalty formulation (Eq. 5.8) shows two unstable elements (Figure 6.12). After that, we apply CRA on each local element individually. E.g., looking at the top-right element, we set all its neighbouring parts as support. The tensile forces at the upper interface indicate that the region is not activated (Fig 6.18a). In addition, the virtual displacement (light grey rectangle) shows the potential sliding direction. To prevent the element from sliding down and provide the uplifting force, we provide three possible modifications: adding an element as support (Fig 6.18b) or changing the bottom (Fig 6.18c) or the upper (Fig 6.18d) interface geometry to enforce the resultant inside the friction cone. For each modification, CRA provides real-time feedback about the stability of the new configuration. The users can learn from the force feedback and modify

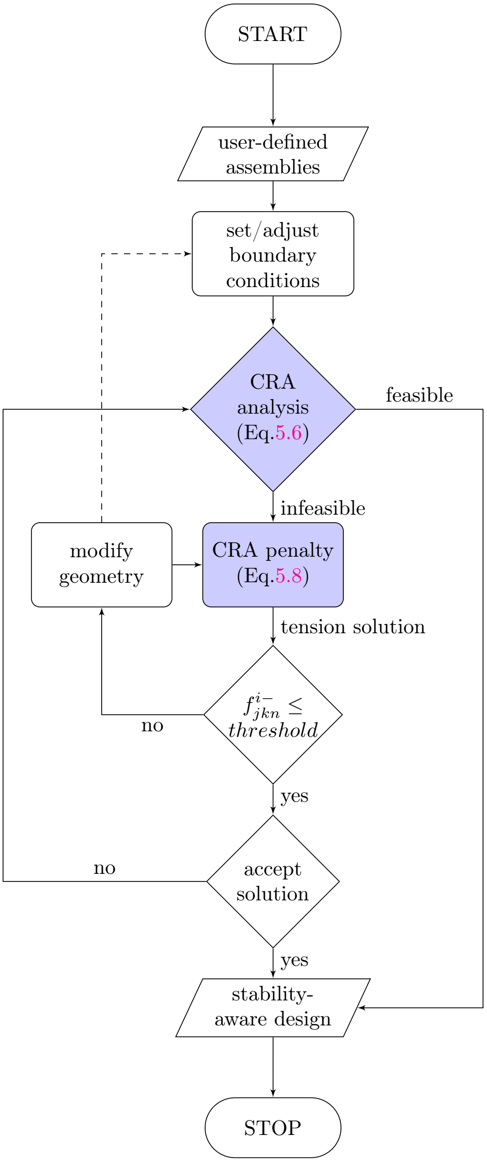


Figure 6.17: Stability-aware design workflow with CRA.

their initial design by choice toward better local stability-aware design. After the instability of a specific local element is resolved, the user can resolve different unstable regions locally one by one. Nevertheless, satisfying local stability does not guarantee a global structural equilibrium. The user must rerun the CRA to ensure the entire structure reaches global equilibrium.

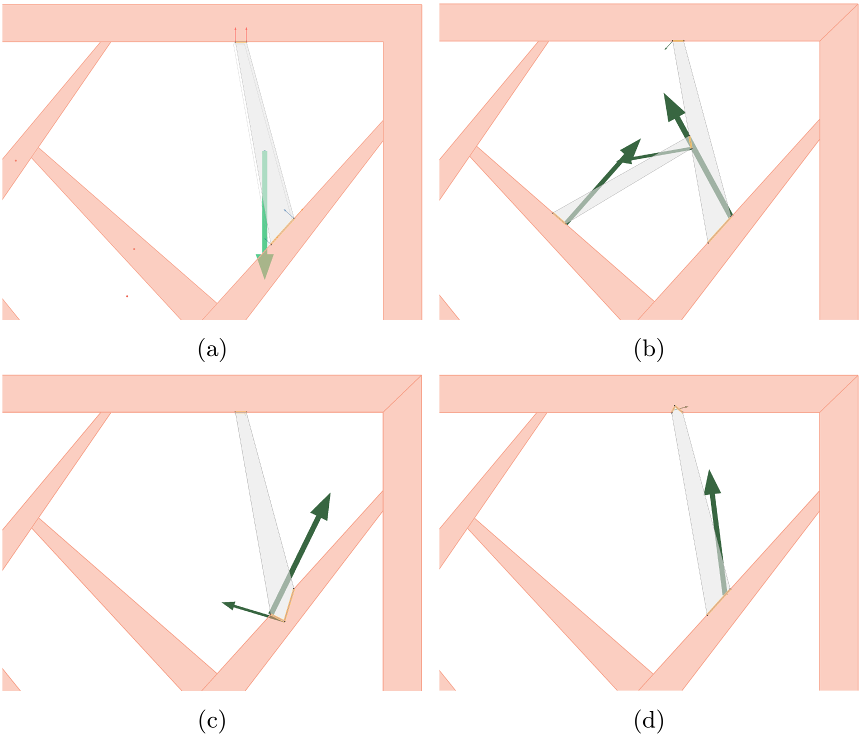


Figure 6.18: The local instability can be understood better by isolating the unstable parts from the shelf model (Figure 6.12). Tension forces on the top indicate the specific interface is not loaded, and the light grey rectangle shows the virtual displacement as potential sliding movement (a). The unstable element can be resolved by adding a supporting element (b), changing bottom interface geometry (c), or changing upper interface geometry (d).

6.3.3 Wedge arch

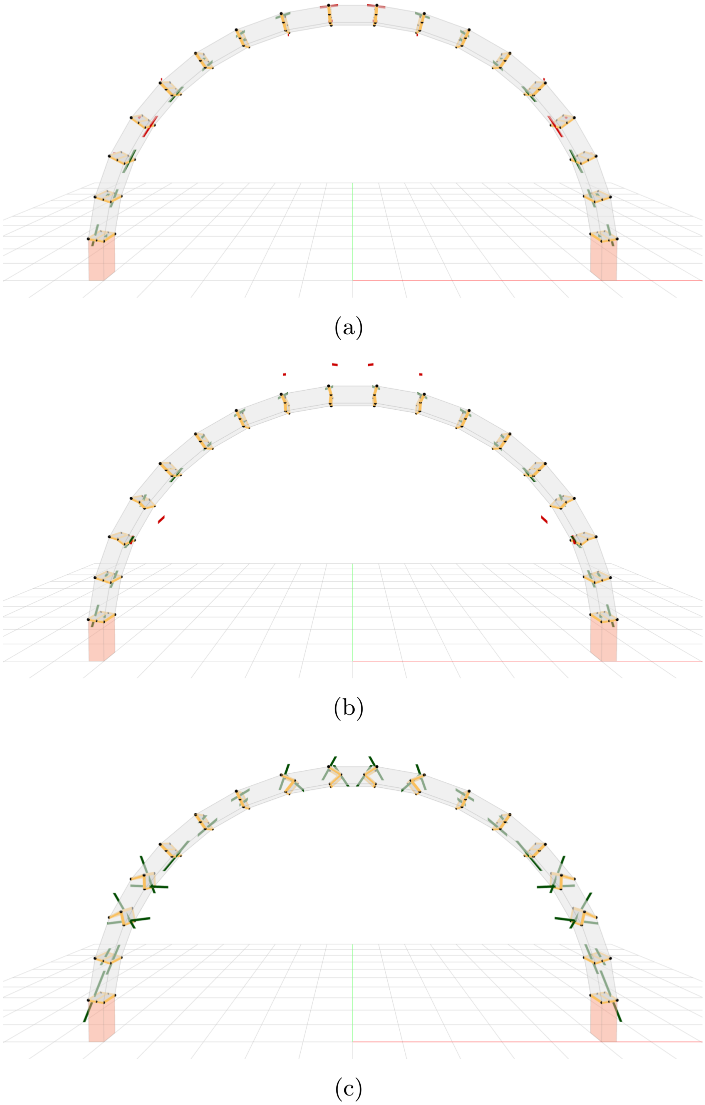
In Subsections 6.2.3 and 6.3.2, we demonstrated that a *wedge* could provide better stability than a single-planar interface. Therefore, using the workflow provided in Subsection 6.3.1, we model an unstable thin arch, see Figure 6.19a, and divide every single-planar interface with two-planar interfaces to create sharp wedges. By checking the resultant tension values between the interfaces, we iteratively increase the sharpness of the wedge angle in which the tension appears, Figure 6.19b, until all tension forces disappear, Figure 6.19c.

6.4 Analysing complex structures using CRA

Besides using CRA to analyse a few complex connections locally, it can also be applied for the global assessment of complex structures and the stability checks during construction. Subsection 6.4.1 shows the analysis of a complex shell structure, and Subsection 6.4.2 shows the use of CRA for assessing stability during shell assembly.

6.4.1 Armadillo Vault

Figure 6.20 shows the CRA analysis of the Armadillo Vault presented in Figure 1.2a. The model consists of 399 blocks and 1014 interfaces. The built Armadillo Vault structure has a 16 metres span with a minimum block thickness of 5 cm [Block et al., 2018, 2017a]. In our numerical model, we approximate non-planar ruled surfaces to locally planar interfaces and identify the first layer of blocks as supports. Importantly, CRA correctly identifies the Armadillo Vault as a stable structure, even with such thin structural thickness. In addition, the internal resultant forces obtained from CRA nicely capture the arching and hoop actions as illustrated in Figure 6.20.

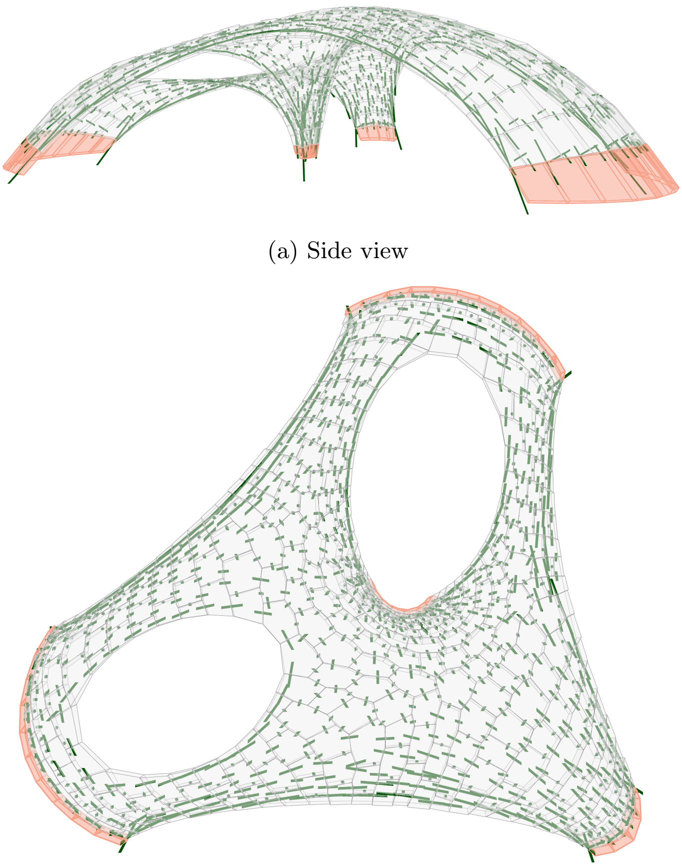


(a)

(b)

(c)

Figure 6.19: We iteratively increase the sharpness of the wedge angle on those interfaces where tension forces appear to stabilise the thin arch.



(a) Side view

(b) Top view

Figure 6.20: CRA correctly identifies the Armadillo Vault (Figure 1.2a) under its self-weight as a stable structure. It also identifies arching and hoop forces correctly.

6.4.2 Robotic assembly of shell structures

The CRA method can also be applied to correctly assess structural stability during construction using robotic fabrication [cite to AAG paper with students Wang et al. work in progress]. Figures 6.21a and b show both CRA simulations and robotic assembly of physical shell models. We perform CRA at each assembly step by giving the simulation a predefined assembly sequence to assess its stability during robotic construction. Our designs are validated through scaled physical models that are 3D-printed with a Prusa i3 MK2 3D Printer, and we applied sandpapers to the interfaces and verified that the friction value is above 0.66. Additionally, we assembled the shell structure using two Universal Robots UR5 with customised end effectors. CRA correctly predicts the stability of each assembly step of the complex shell structure. In this robotic assembly of shell structure, we utilise the assembly strategies proposed by Kao et al. [2017]. If the structure is unstable, we assemble it with neighbouring blocks and with the assistance of another robotic arm. In this sense, we can efficiently assemble shell structures with limited scaffolding, which reduces a tremendous amount of material waste.

6.5 Summary

In this Chapter, we demonstrated the use of the solvers described in Part II for the stability assessment of various structures.

In Section 6.1, we performed several studies to understand the RBE-based approach better. Firstly, we applied the RBE analysis to two simple benchmark cases to show the meaning of the nodal contact forces and to illustrate how the friction capacity is handled. Secondly, we analysed the same buttressed arch as in Iannuzzo et al. [2020] to benchmark and illustrate RBE's potential. Thirdly, we looked at a dome with an oculus to show how an improved RBE-based approach can also be used to assess 3D structures. The primary outcome of Section 6.1 is that both QP and LP approaches provide statically admissible solutions in a Limit Analysis sense. Since, in reality, it is hopeless to try to understand the actual contact conditions, there is no reason to prefer one over another, each solution being a possible admissible stress state. In general, LP solutions are more affordable (i.e., less time-consuming) for large problems. In this light, RBE is a powerful tool that

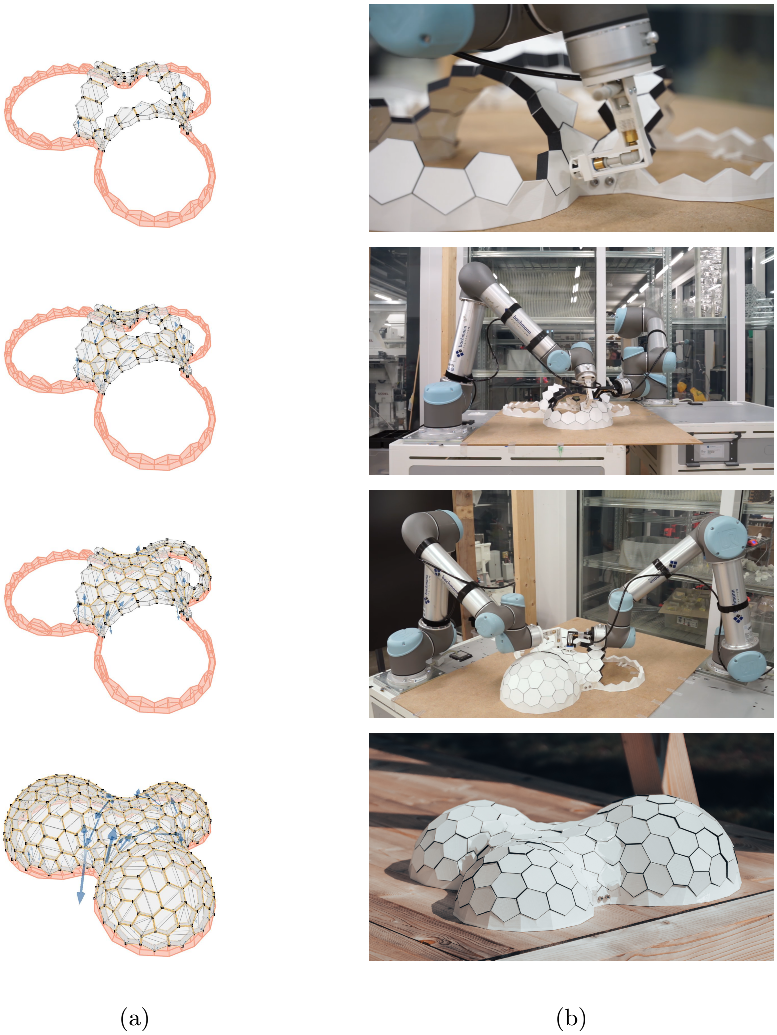


Figure 6.21: Robotic assembly of a *tri-dome shell* structure Wang et al. [2023]: (a) CRA assessment, identifying the structure is stable during the intermediate step of assembly; and, (b) 3D-printed model assembled with two Universal Robots (UR5).

can be used to explore a wide range of equilibrium states and corresponding stress solutions fast and efficiently. Additionally, their use allows for tackling typical masonry assessment problems properly.

The equilibrium approach is a common strategy for analysing the stability of rigid block assemblies. Nevertheless, existing methods are either too conservative or do not provide realistic or accurate results, even for some simple scenarios. RBE is particularly unreliable when analysing more complex geometries. Therefore, Section 6.2 performed a wide range of CRA benchmarks to demonstrate its accuracy and robustness. Benchmarking with commercial software 3DEC (DEM) and Sofistik (nonlinear FEM), CRA has been proven to handle various geometrical cases accurately, from classical masonry arches to more complex shapes with interlocking and freeform joints. Notably, CRA provides an explicit optimisation formulation with fewer parameters to tune, enabling early design feedback on the assessment problems and future improvements. Compared to other explicit methods, such as RBE and Variational Static Analysis (VSA) proposed by Yao et al. [2017], CRA aligns much better with analytical solutions and commercial software results commonly used in actual engineering practice. In addition to accurate, quick, and explicit solutions, CRA provides a penalty force solution when the structure is unstable, which gives users helpful feedback for improving a discrete structure towards a more stable solution.

After showing CRA's accurate solutions and the method's ability to be used for a wide range of problems, in Section 6.3, we presented a stability-aware design workflow that utilises CRA formulations to assist the user in adjusting their designs' geometry. For unstable structural configurations, the CRA penalty formulation gives users additional tensile forces, supplemented with virtual displacement information, to iteratively guide and improve their geometrical design intentions toward a more stable solution.

Finally, Section 6.4 demonstrated CRA's assessment usage for complex shell structures. We showed that CRA is robust enough to be applied to the practical assessment of complex shell structures. Taking the Armadillo Vault as an example, CRA correctly showed its internal stresses and captured its arching and hoop behaviour. Furthermore, we exhibited that CRA is a practical tool for assessing stability during robot-assembled shell construction. With the help of CRA, we used two robots for accurately picking and placing 3D-printed blocks to assemble a discrete shell structure entirely without any scaffolding.

Chapter 7

Assembly-Aware Design (AAD)

After demonstrating the CRA solver's robustness in Chapter 6, this chapter presents a computational framework for Assembly-Aware Design (AAD) to design discretised shell structures that can be constructed with minimal formwork. Section 7.1 illustrates the benefit of considering assembly in the design workflow and proposes embedding a stability-aware design workflow. Section 7.2 demonstrates the design process of using wedge geometry to achieve two surprising scaffolding-free design examples. We use CRA to simulate, guide the design process, and validate the design with physical models. Section 7.3 proposes the Assembly-Aware Design process and explains it in a flowchart. Section 7.4 shows the process of generating the geometries for shell structures. Finally, in Section 7.5, we demonstrate structural designs of shells that can be assembled with limited scaffolding and summarise this chapter in Section 7.6.

7.1 Introduction

In architectural practice, shell structures are challenging to construct and the process is highly inefficient, especially for complicated forms. Typically, building such structures requires massive amounts of (often single-use) material for temporary scaffolds during assemblage to keep the partially assembled structure in equilibrium during erection. Those scaffolds require extra time, cost, and labour, often producing material waste. Generally speaking, discrete shell structures are hard to fabricate and assemble; they need more structural and assembly awareness in the early design phases.

7.2 Scaffolding-free assembly designs

In Section 6.3, we showed how to apply CRA in a stability-aware design process. Based on CRA's information, designers can intuitively improve their assembly design's stability, see Figure 6.17. However, more than a stability-aware design process is required to efficiently consider the assembly process for the discrete shell structures in the early design phase. Specifically, we need at least one assembly strategy and use the stability-aware design process in each assembly step to determine if the structure is stable during the assembly process. Therefore, to simplify the problem, this section looks at some two-dimensional problems and demonstrates the design process of scaffolding-free structures. Subsection 7.2.1 presents a strategy to discretise a structure and prescribe an assembly sequence. Subsection 7.2.2 applies CRA in the stability-aware design process and checks the substructure's stability for every assemblage step. Finally, Subsections 7.2.3 and 7.2.4 present two surprising scaffolding-free assembly designs and validate the results with physical models.

7.2.1 Discretisation and assembly sequence

Suppose we have a leaning but stable tower (Figure 7.1a), want to discretise it, and then be able to assemble it without using much scaffolding. Since the tower is built from the ground level, against gravity, a good strategy is to perform "cuts" from the top to the bottom. In Figure 7.1b–g, we iteratively make such "cuts" to discretise the tower, at each step, defining all blocks below the new "cutting" interface as supports. Furthermore, we apply CRA with the stability-aware design workflow (Figure 6.17) to ensure each specific "cut" does not cause the structure above that "cut" to no longer be in equilibrium. If performing a "cut" makes the sub-structures unstable, the entire structure will for sure not be stable for further "cuts".

7.2.2 Stability during assembly

After successfully performing all the "cuts" and ensuring all sub-structures in all steps are stable, we can start disassembling the structure one by one to obtain a feasible disassembly sequence, see Figure 7.1h-o. After that, we reverse the disassembly sequence to assemble the structure. Note that our discretisation strategy steps from Figure 7.1b to Figure 7.1g do not guarantee that the assembly sequence from Figure 7.1o to Figure 7.1h is completely scaffolding-free. Suppose the structure could not be stable in the specific disassembly/assembly step; adapting the strategy from Kao et al. [2017], we can provide support or assemble with other parts simultaneously. Another valid strategy to overcome instability during disassembly is to change the interface discretisation (i.e., the "cut") below the specific unstable block locally and rerun that block's "cutting" process to ensure that the cut does not disturb the overall stability. For example, if removing the top block in Figure 7.1j causes the top triangle block in Figure 7.1k to be unstable, we can locally change the interface below that triangle block in Figure 7.1k and recheck the "cutting" step in Figure 7.1e.

7.2.3 Snake design

Following the leaning tower example, we show two simple but meaningful design cases where CRA and the stability-aware design workflow are used to assess and design self-supporting structures that can be assembled scaffold-free. We used the same procedure described in the previous subsections to update interfaces/"cuts" iteratively and get a feasible step-by-step assembly sequence. Finally, our designs are validated using scaled physical models that are 3D-printed using a Stratasys Objet500 Connex3 3D Printer.

The following example looks at a *snake* design, which can be assembled entirely scaffolding-free. Its discretisation and assembly process is summarised

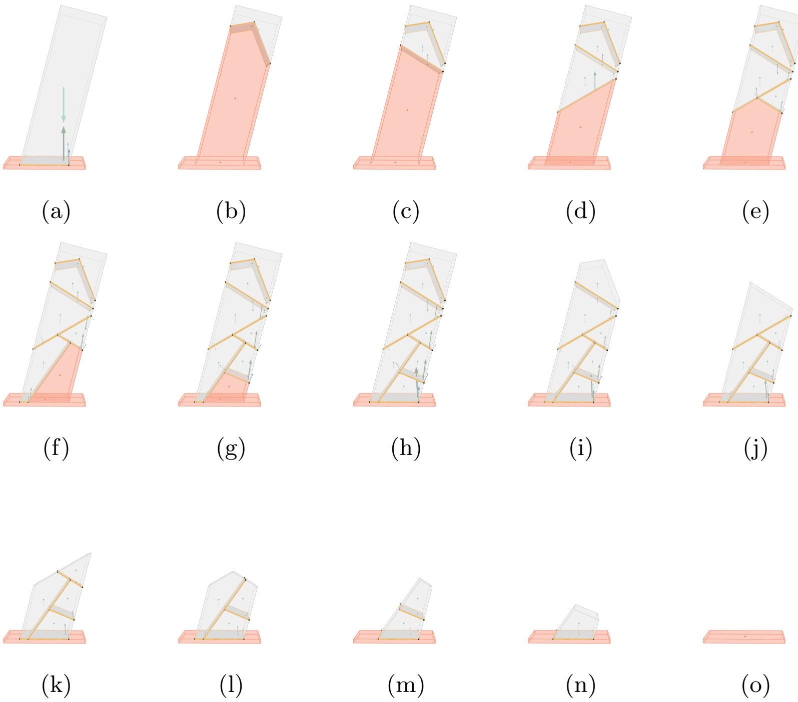


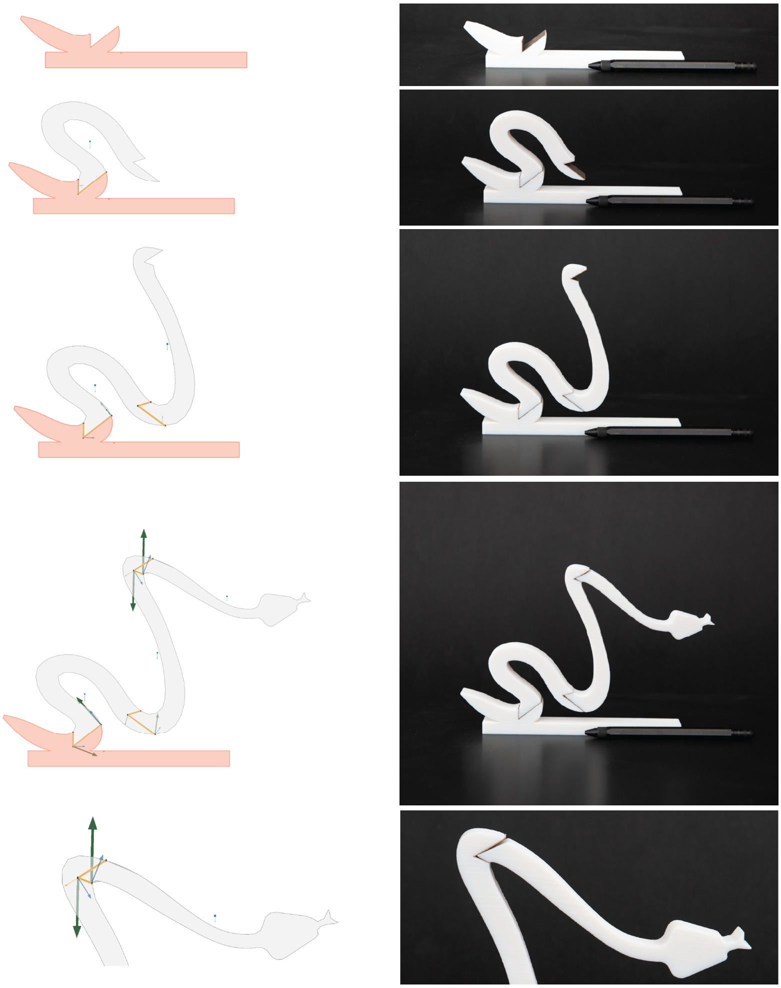
Figure 7.1: The step-by-step discretisation design process of a leaning tower that can be assembled scaffolding-free. From (a) to (g), the designer iteratively designs “cuts” (interface between the grey free block and pink fixed support) and ensures the “cuts” do not compromise the structure. The structure can be disassembled safely step-by-step from (h) to (o) with verification of the CRA method.

in Figure 7.2, The snake is composed of three pieces plus a supporting base. The ability to accurately model sharp interfaces with friction enables the user to reach cantilevered solutions that are not possible with only planar interfaces. Note the parallels between the physical model and CRA results in Figure 7.2. Because of the small tolerances, the head of the snake moves slightly. However, the small movement emphasises that equilibrium is reached through the transmission of reaction forces on two contact points.

This matches the force couple, proving the “clamping” moment, found with the CRA analysis.

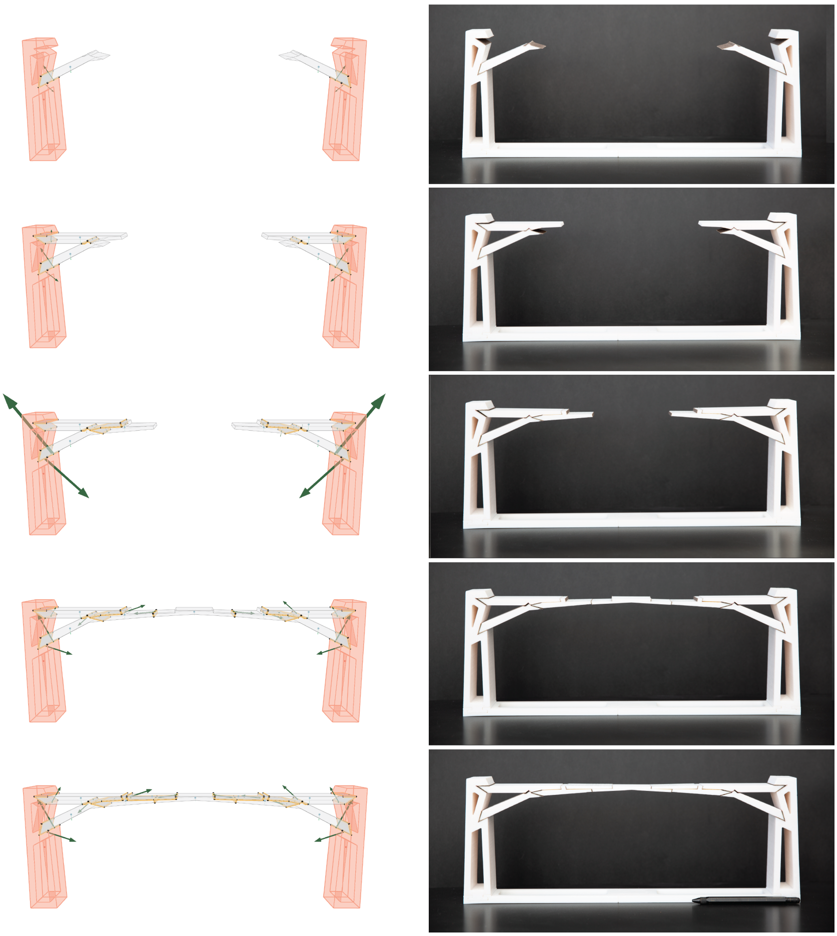
7.2.4 Bridge design

The second design-by analysis example regards a bridge design, whose assembly process, together with the CRA results, are summarised in Figures 7.3 and 7.4. The bridge has been specifically designed to avoid interlocking joints but still has an assembly sequence such that each piece can stably stand after placement without any additional scaffolding. To limit the effect of tolerances in the physical model, a connection (red circles in Figure 7.4) was designed to have the first two pieces self-registered in the desired position. The friction coefficient adopted in the analysis is 0.66. A parametric CRA analysis showed that for friction values lower than 0.35, the bridge would not be able to be constructed scaffolding-free. As the friction coefficient of the printing material is around 0.3, we applied sandpaper to the interfaces to guarantee that the actual friction value is above 0.66. The sandpaper, applied with double-sided tape, has a thickness of 0.58 mm, which was carefully considered in the 3D-printed geometry. The bridge stands under its self-weight, and as Figure 7.4 shows, it can support additional loads. The bridge’s net mass (i.e., without supports) is 319g, while the mass of the external distributed loads is 1208g. Figure 7.4 shows the CRA results obtained considering these external, additional loads (using a higher material density for the additional blocks). Lastly, looking at the physical model in Fig 7.4b, the keystone is touching and transmitting forces only through the two upper interfaces while the lower ones are not in contact, as highlighted with green circles. This phenomenon is well captured by the CRA analysis in Figure 7.4a, where the solution returns only two resultants affecting those interfaces.



(a) (b)

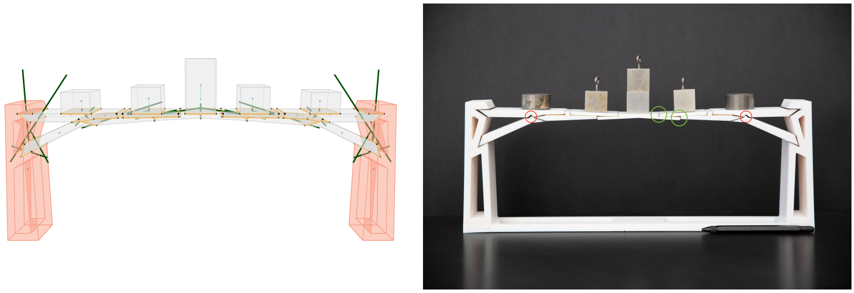
Figure 7.2: Snake model: step-by-step construction sequence of both CRA analysis (a) and corresponding physical model (b). Friction value $\mu = 0.66$ is applied in the CRA analysis. Two images on the right highlight the detailed interface and forces of the assembled model.



(a)

(b)

Figure 7.3: Sequential, scaffolding-free construction of the bridge. CRA analyses (a) and corresponding construction phases of the small-scale physical models (b).



(a) (b)

Figure 7.4: (a) Bridge in its final configuration and subjected to additional loads. (b) The span of the 3D printed model is 40 cm, the net mass of the bridge (excluding the supports) is 319g, while the mass of the externally applied loads is 1208g. The red circles denote two *sharp wedges* designed to guarantee that the two external elements are in the right position.

7.3 AAD workflow

As described in Section 7.2, we design scaffolding-free structures and verify them with physical models. Our tools and workflow open up the enormous potential for designers to consider their designs' stability and assembly sequence in the earlier design phase. Our method increases stability and assembly awareness for users. Therefore, we propose an Assembly-Aware Design (AAD) process and summarise it as an algorithm workflow in Figure 7.5.

The algorithm's input is a user-defined shape, which can be either a volumetric solid (e.g., Figure 7.1a) or a surface that still needs further materialisation. We will discuss the detailed aspects of surface materialisation next in Subsection 7.4.3. Note that the input shape should be initially stable because the final structure must be stable. With a specific disassembly strategy in mind, e.g., disassembling the structure from top to bottom, the user discretises the shape into some smaller units. After that, the user evaluates and changes the design through our proposed stability-aware design

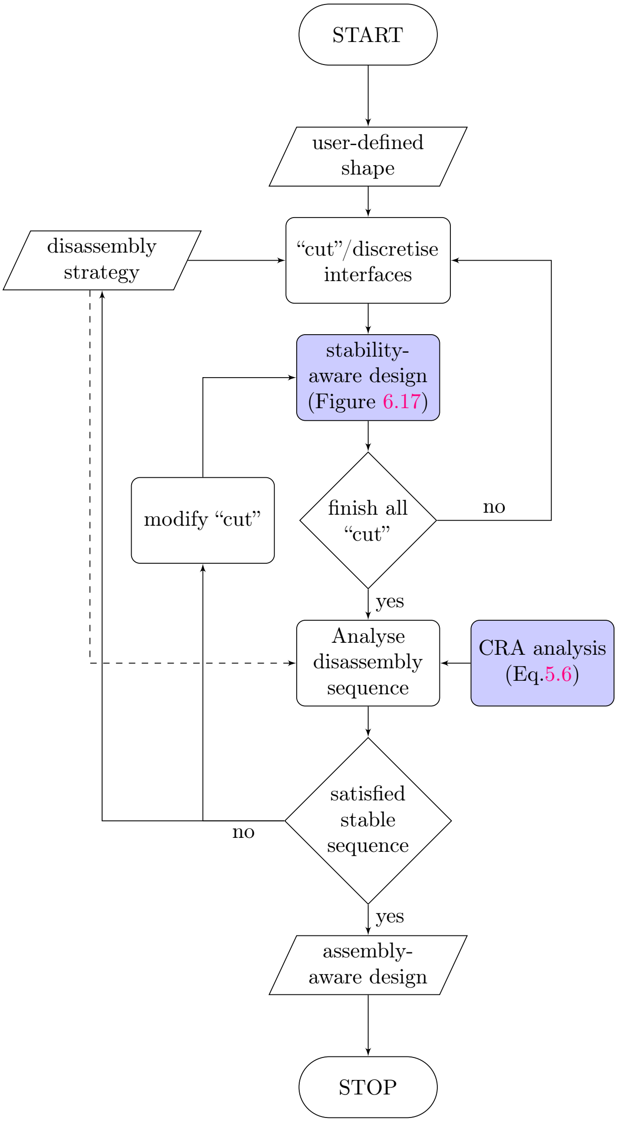


Figure 7.5: Assembly-Aware Design workflow that embeds the stability-aware design and CRA method.

workflow, in Figure 6.17, to ensure that the substructure is still in equilibrium after each discretisation step. The user repeats the process until generating all desired interfaces and blocks. Once a desired discretisation is generated, based on the user disassembly strategy, the user analyses each disassembly-sequence step's structural stability using CRA analysis, Eq. 5.6. If there is no stable disassembly sequence, the user returns to modify the discretisation or change the disassembly strategy and repeats the workflow until finding a desired stable disassembly sequence. Finally, the user can assemble the structure by reversing the disassembly sequence, as described in Kao et al. [2017]. Next Section 7.4 proposes a process to generate discrete shell structures that can apply and integrate into the AAD workflow.

7.4 Generation of discrete shell structures

Learning from traditional scaffolding-free masonry structures [Fitchen, 1981; Allen and Zalewski, 2009] and our literature discussion in Chapter 2, a good general strategy to stably construct discrete shell structures is to form stable sections, e.g., arches and rings, during the assemblage. Forming stable sections provides opportunities for incomplete and unstable parts to temporarily be supported by or lean on the stable parts, enabling forces to transmit to the boundaries, thus reducing the need for scaffold during construction. Inspired by Rippmann [2016] and Kao et al. [2017], this section demonstrates a procedure to generate discrete shell structures. In Subsection 7.4.1, we describe the procedure to generate masonry courses that form stable sections from a surface. Section 7.4.2 demonstrates how to parametrically generate staggered masonry patterns based on those courses. After this tessellation, Section 7.4.3 shows the process and algorithm to materialise the staggered patterns into blocks and planarise interfaces between blocks. Finally, Section 7.4.4 suggests one possible dis/assembly sequence for building such a discrete staggered shell structure. Additionally, Section 7.4.5 casts light on some interlocking block designs to overcome instability during assembly.

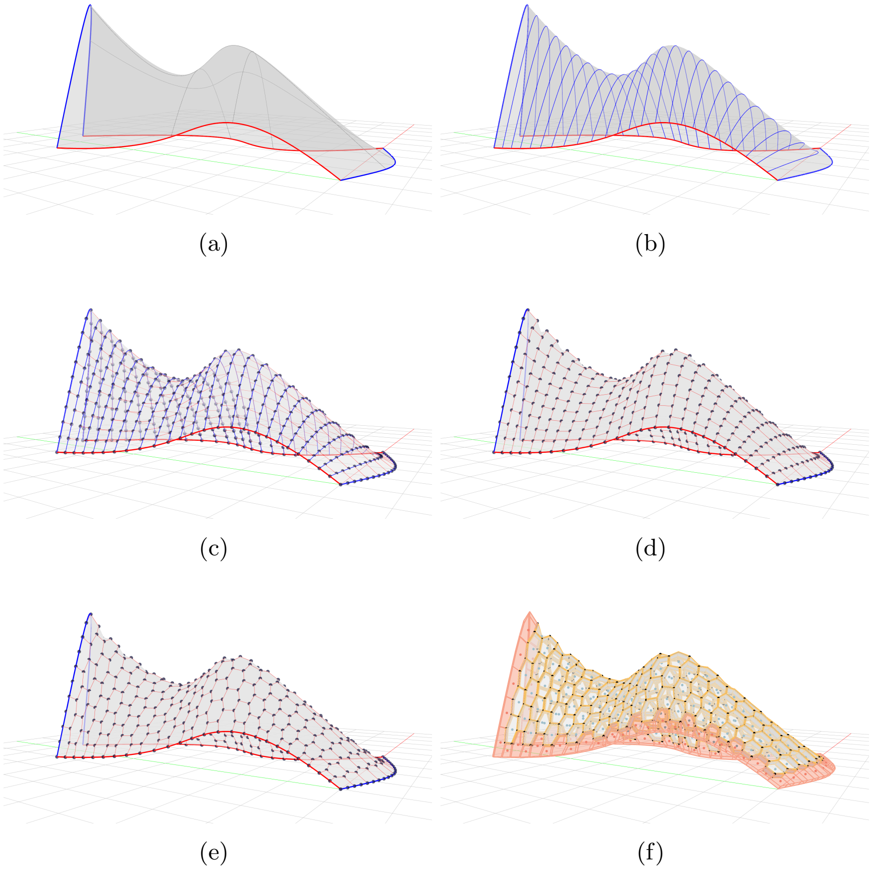
7.4.1 Courses generation

Rippmann [2016] and Oval [2019] show that shell structures can be decomposed into some smaller four-sided patches or a coarse quad mesh. The quad mesh's strips and surface's UV curves can be used to generate courses. Those courses can act like arches or rings during assembly. Figure 7.6a shows a four-sided surface with two direction edges, u (blue curves) and v (red curves). We generate the arches in the u direction, Figure 7.6b, because it creates arch curves, which provide stable sections after each arch course is completed.

7.4.2 Tessellation pattern

For the masonry voussoirs, Rankine [1872] and Rippmann [2016] mentioned that staggered configurations are tessellations that result in good structural behaviour for masonry vaults. To generate a staggered pattern, see Figure 7.6c, we divide the course curves into points and connect every other point with its neighbour point in the previous course. A similar procedure developed by Rippmann [2016] added partially inserted rows when one side is much longer than the other. In the next course, we shift the connectivity to one unit to create the stagger pattern. With this procedure, we create the staggered pattern as a polygonal mesh. Each mesh face has six vertices and edges except for particular faces along the boundaries, see Figure 7.6d. For a more thorough and sophisticated discussion of staggered tessellations for shell structures, we also refer the reader to Rippmann [2016].

Additionally, Figure 7.6e and Figure 7.7 show that each vertex can move a certain distance dv in the direction of its adjacent vertex in the neighbouring course to create a *wedge* geometry and increase stability, as discussed in Subsection 6.2.3, 6.3.3, and Section 7.2. Distance dv can vary as a parameter individually on each course based on the stability needs during structural assemblage.



(a)

(b)

(c)

(d)

(e)

(f)

Figure 7.6: A geometrical procedure to generate discrete shell structures. (a) The process starts from a four-sided surface with the u direction in blue and the v direction in red. (b) Create course curves in the u direction. (c) Divide the course curves into segments. (d) Generate a staggered pattern mesh surface as tessellation. (e) Modify the wedge angles to increase stability. (f) Materialise the tessellation into block assemblies.

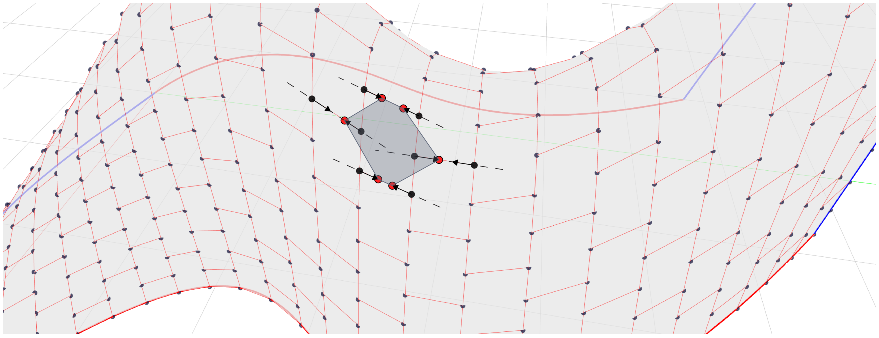


Figure 7.7: Each vertex in the staggered tessellation can be manipulated and moved in the direction of the corresponding vertex of its neighbouring course to change the block shapes and their wedge angles. The original mesh vertices are labeled in black, and the new vertex positions in red.

7.4.3 Materialisation

To materialise a shell surface with the staggered pattern, we extrude the polygonal face vertices along their mesh vertex normal vectors, see Figure 7.8a. Since shell structures typically have doubly-curved surfaces, two normal vectors for the interface extrusion usually are not in the same plane, i.e., the interfaces are non-planar.

As discussed in Section 5.4, CRA can deal with complex shapes with freeform interfaces. We only need an additional step to discretise the freeform interface into planar sub-interfaces. Note that using freeform interfaces generally increases CRA's solving time since the number of interfaces increases. Therefore, we planarise all interfaces between blocks to simplify the problem. Using the method proposed in Bouaziz et al. [2012], Figure 7.8b shows that we calculate the vertex's adjacent planar interfaces and iteratively move the vertex toward all neighbouring projection vectors' averages. For a more detailed interface planarisation, we refer the reader to the work of Bouaziz et al. [2012], Bouaziz et al. [2014], and Deuss et al. [2015]. Finally, Figure 7.6f shows the generated assembly design that can be applied to AAD.

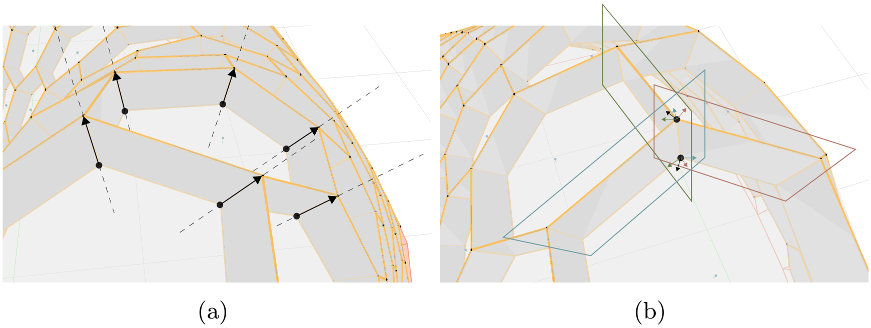


Figure 7.8: Materialise the shell surface with blocks. (a) Offsetting the mesh face from its vertex normals. (b) Planarisation of three adjacent interfaces of a vertex. The algorithm calculates the projection vector individually to the vertex's adjacent interfaces and then moves the vertex in the three vectors' average direction.

7.4.4 Assembly sequence

Our assembly strategy for staggered-pattern masonry generated from a four-sided surface is to assemble the discrete shell structure course by course (i.e., forming stable sections through arches). Additionally, we start from the lower arches/courses closer to the support so that the later constructed unfinished course can lean on the previously constructed, stable courses. Figure 7.9 shows the starting support arch in red curves and the assembly sequence direction in the black arrows.

With a dis/assembly strategy in mind, we can apply the AAD workflow and iterate back to the steps mentioned in Subsections 7.4.1, 7.4.2, and 7.4.3 to modify the design's discretisation accordingly. We iteratively remove blocks from the top until reaching the supporting blocks to obtain a feasible assembly sequence and highlight the "problem blocks" that require scaffolding. We aim to embed assembly awareness in the design process to reduce scaffolding instead of creating an entirely scaffolding-free design. Sometimes, some blocks are just geometrically impossible to assemble without scaffolding or too risky in practice.

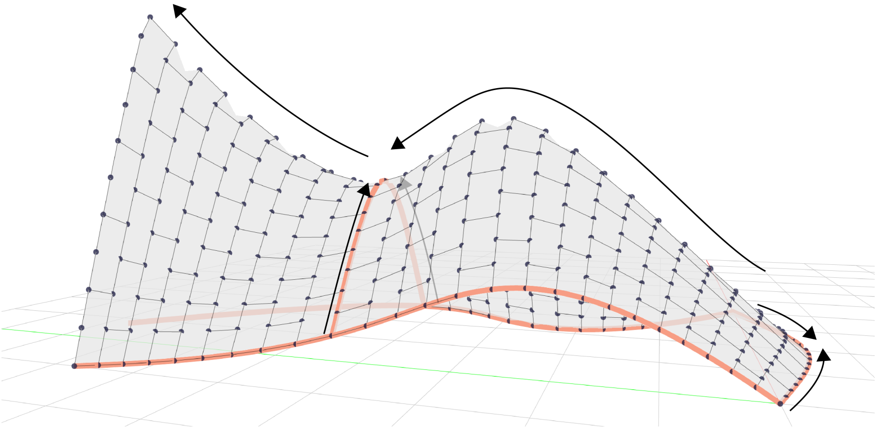
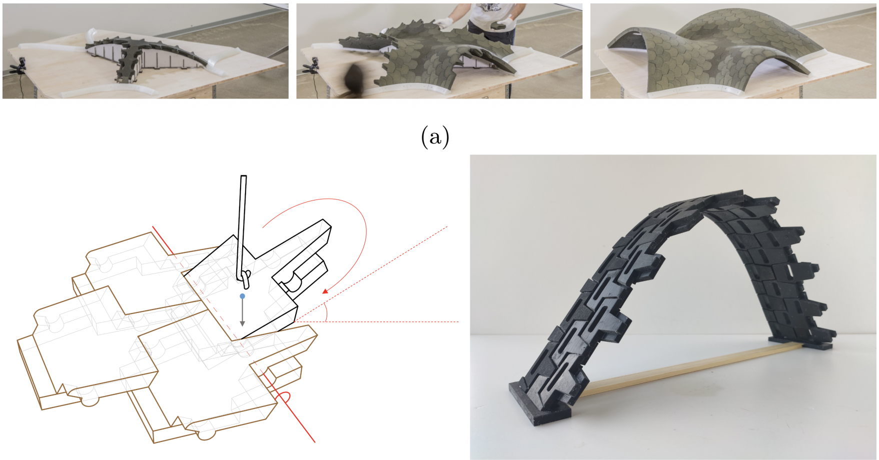


Figure 7.9: Example of assembly strategy of a shell surface. The red lines indicate the boundary conditions as starting points. Our proposed strategy is to build arch courses from the lower level toward the higher ones, such that later arch courses can lean on the stable and finished courses.

7.4.5 Block design to provide temporary stability

With the CRA method, stability and assembly-aware design processes can be used to design simple convex blocks and complex block geometries. Although complex block shapes require more sophisticated procedures to fabricate and assemble, they can sometimes be helpful for practical reasons or can be used to construct shell structures using fewer scaffolds.

Inspired by scaffolding-free shell designs (Figure 7.10a) developed by Alexander Kobald, Matthias Rippmann, and Andrei Jipa at ETH Zurich in 2017–2018, Figure 7.10b shows one of our experiments of interlocking block design that can overcome instability during assemblage. Each new block of the current assembly course/arch can be safely placed on the previous course as the interlocking block geometry is designed to prevent translational and rotational movement, which causes instability. Additionally, the block is designed to be easily rotatable along an axis so all blocks can be self-registered



(a)

(b)

(c)

Figure 7.10: Some experimental interlocking block designs enable each block to be assembled entirely scaffolding-free: (a) a scaffolding-free shell design developed by Alexander Kobald, Matthias Rippmann, and Andrei Jipa at ETH Zurich in 2017–2018; (b) inspired by (a), an interlocking block developed by the author to provide temporary stability until the stable row/arch is complete. The block is designed to be easily rotatable along an axis so all blocks can be self-registered during assembly; and, (c) a physical prototype of (b).

during assembly. Figure 7.10c shows the physical prototype model of the interlocking block design. Except for the first arch, no scaffolding is needed during the assemblage. Every block in the new course is stabilised through the previous, stable sections.

7.5 Assembly-aware discrete shell designs

This section demonstrates the AAD workflow of discrete shell structures through three examples. Subsection 7.5.1 shows the AAD workflow in a simple vault design. After that, Subsection 7.5.2 shows that the dome can be assembled entirely scaffolding-free. Additionally, we verify the design with a physical model and assemble the dome with two robots. Finally, Subsection 7.5.3 will demonstrate how to design freeform discrete shell structures using an AAD workflow.

7.5.1 Vault designs

A barrel vault is one of the simplest shapes for discrete shell structures. As discussed in literature Section 2.3 and Figure 2.5, a barrel vault has various ways to be constructed with limited scaffolding. In this section, learning from ancient barrel vaults, we demonstrate designing and simulating a simple vault assembly design using an AAD workflow.

In Figure 7.11a, suppose that we have a four-sided shell surface with set boundary conditions along three edges that connect to the ground, see Figure 7.11b. Firstly, we suggest that users can use methods such as TNA [Block, 2009] with optimisation methods [Maia Avelino et al., 2021] to form find a surface and find a reasonable thickness to ensure the final complete discrete shell structure is stable. Secondly, we create stable sections through arch courses, see Figure 7.11c. Thirdly, we generate the staggered masonry pattern in Figure 7.11d and iteratively adjust each course with a specific parameter for the wedge angles based on stability during assemblage, Figure 7.11e. Finally, in Figure 7.11f, we materialise all blocks for assembly planning. Those "cut" parameters and wedge angles can be changed based on the AAD disassembly planning with CRA analyses in order to achieve block equilibrium during assembly.

After setting up the geometric procedures to generate the discrete shell structure, we start the AAD workflow (Figure 7.5) to interactively modify the shell discretisation and find a feasible assembly sequence. Figures 7.12a and b show the first iteration of the discrete shell structure and its CRA analysis, verifying that the final structure is stable. We start with a staggered pattern with very shallow wedge angles (Figure 7.12a) because sharp corners are more fragile and easier to be broken or lead to stress concentrations.

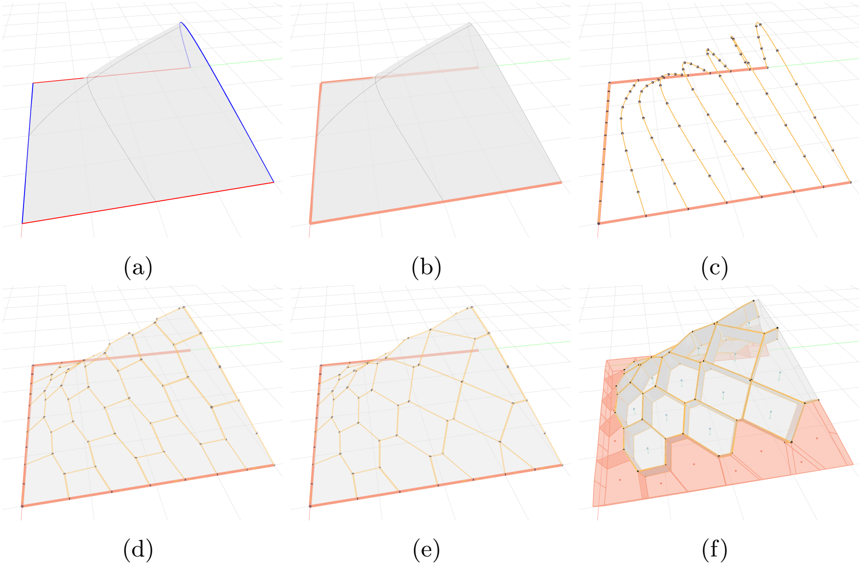


Figure 7.11: The geometrical procedure to generate a simple discrete vault structure. (a) The process starts from a four-sided surface with the u direction in blue and the v direction in red. (b) Setting up boundary conditions in pink curves. (c) Create course curves in the u direction. (d) Generate a staggered-pattern mesh as tessellation. (e) Iteratively modify the wedge angles based on CRA analyses to increase the stability of each course. (f) Materialise the tessellation into an assembly of blocks.

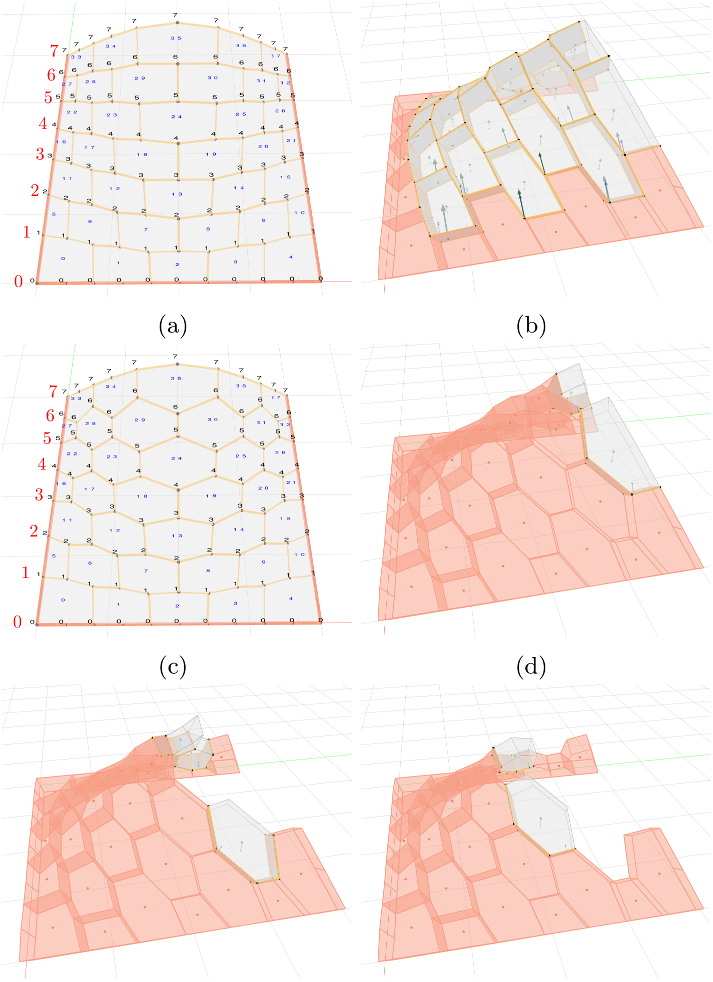
In Figure 7.12a, we label each block with an identification number in blue, and each vertex has a course number in black. The course number on the vertex indicates the neighbouring upper block's course number, e.g., blocks course 6 are those blocks between vertex courses 6 and 7. Changing the interfaces with the same vertex course number affects the stability of blocks with the same course number during assembly, e.g., modifying the interfaces and wedge angles of vertex course number 5 affects the blocks with numbers 22 to 26.

Our disassembly strategy for this simple vault is straightforward and as follows:

- We disassemble course by course and start with an open edge, so, e.g., the course sequence 6, 5, 4, 3, 2, 1.
- For each arched course, we remove blocks from the top level, e.g., 35, 36, 34, (33 and 37 are supports), 29, 30, 28, 31, . . . and so on.

Therefore, we iteratively remove blocks based on our disassembly strategy and utilise the AAD workflow to adjust the "cutting" interfaces and block shapes, Figure 7.12c. We first set all other blocks with a lower course number as boundary conditions, so fixed, to disassemble one course and adjust its interfaces, it is necessary. Figure 7.12d shows the removal of the first course, adjusting the "cut", and analysing the blocks using stability-aware design simultaneously. After being satisfied with the block shapes and verifying that all blocks can be removed safely with or without scaffoldings, we move to the subsequent courses of blocks and repeat the processes, see Figure 7.12e and f.

Finally, after finalising the discretisation design of the shell, the last step in the AAD workflow is to generate the assembly sequence and visualise the "problem areas". Figure 7.13a visualises the blocks in different colours to indicate their state for assembly. Light grey indicates the block can be assembled without any scaffolding, dark grey (blocks 7 and 8) indicates the block needs to be assembled with its neighbouring blocks simultaneously or with neighbouring blocks fixed to be stable during assemblage, and black (blocks 6 and 9) indicates the block needs scaffolding during assemblage. Figures 7.13b, c, and d show the CRA analysis of the disassembly sequence. The disassembly sequence is as follows: 35–36–34–30–29–31–28–24–25–23–19–18–20–17–13–14–12–Support(9, 6)–(8–7)–9–6, with the brackets (a–b) meaning that blocks need to be a and b need to be dis/assembled together. Figure 7.13b shows all remaining blocks are stable after the removal of block 30. Similarly, Figure 7.13c shows removing block 19 is a safe disassembly step. However, not all blocks in this shell geometry configuration can be disassembled entirely scaffolding-free. Figure 7.13d shows that the removal of blocks 7 or 8 causes neighbouring blocks 6 and 9 to become unstable.



(a)

(b)

(c)

(d)

(e)

(f)

Figure 7.12: The Assembly-Aware Design of a simple vault structure is shown previously in Figure 7.11. The yellow lines indicate the “cuts” and interfaces. (a) The first iteration of the staggered pattern has a shallow wedge angle. (b) Our CRA method shows that the final assembly structure is stable. (c) The final tessellation staggered pattern after using the AAD. (d)–(f) The user iteratively removes each course and adjusts the “cuts” and the wedge angles through the geometry manipulation discussed in Figure 7.7. The structure stability is verified through CRA.

Indeed, according to the generated disassembly sequence and our visualisation in Figure 7.13a, almost all blocks can be disassembled individually, except blocks 6 and 9 require support and blocks 7 and 8 need to be removed simultaneously.

It is worth mentioning that going back to the previous step in the geometry generation of the discrete shells and changing the interface "cuts" can potentially enable the vault to be constructed and assembled completely scaffolding-free. We chose the specific block and interface geometries for the purpose of demonstrating the AAD workflow.

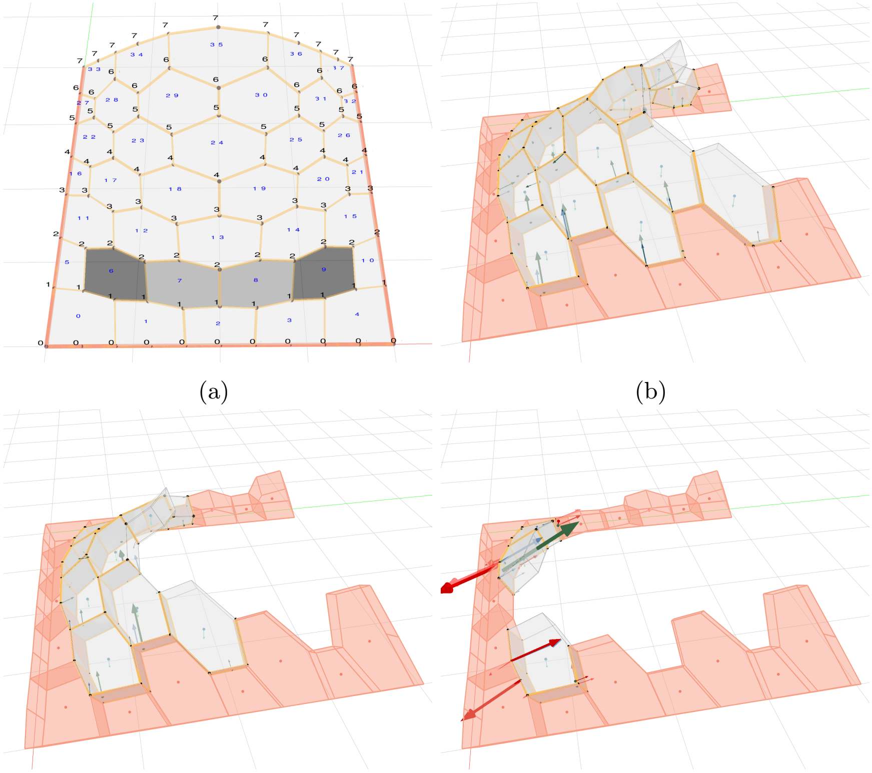
7.5.2 Dome designs

A dome or a cupola is a historically significant architectural element [Mark and Hutchinson, 1986; MacDonald, 2002; Holzer, 2021]. Some domes are constructed centring-free without any scaffolding [King, 2013]. This subsection demonstrates the design and construction process of such a scaffoldings-free dome using the AAD workflow.

Figure 7.14 shows the geometrical generation process of a discrete dome design. Similar to the discrete vault four-sided surface in Figure 7.11a, a dome topologically also has four edges, u in blue and v in red, just with two boundary edges in v overlapping and connected, see Figure 7.14a. Firstly, we set up the boundary conditions on the ground level, see Figure 7.14b, and generate stable sections, e.g., ring courses in Figure 7.14c. Secondly, we generate the staggering masonry pattern in Figure 7.14d and adjust each course's wedge angle, Figure 7.14e, using the procedure mentioned in Figure 7.7. Finally, in Figure 7.14f, we materialise all blocks to generate a discrete dome assembly.

Similarly to the AAD process of vault design in Subsection 7.5.1, we iteratively remove blocks and change the courses' parameters until a satisfactory dis/assembly sequence is obtained.

Figure 7.15 shows the assembly sequence steps, with CRA simulations, ver-



(a) (b) (c) (d)

Figure 7.13: The Assembly-Aware Design result of the vault structure. (a) Different colour schemes show different block assembly statuses. Black blocks require scaffolding during assembly, dark grey ones need to be assembled with neighbours, and light grey blocks can be assembled without scaffoldings. (b)–(d) The disassembly steps analysis using CRA.

ifying that every step is structurally stable. The friction coefficient adopted in the analysis is 0.66 for matching the sandpaper's coefficient in physical models. The dome can be assembled entirely without scaffolding. Some blocks, especially the first blocks (blocks 25 and 37 from the third and fourth courses, highlighted red in Figures 7.15d and f) must be assembled together

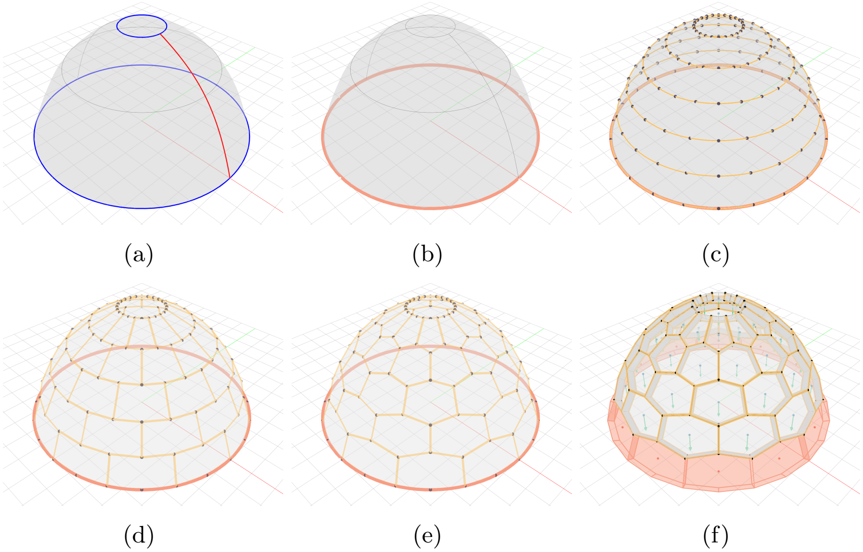


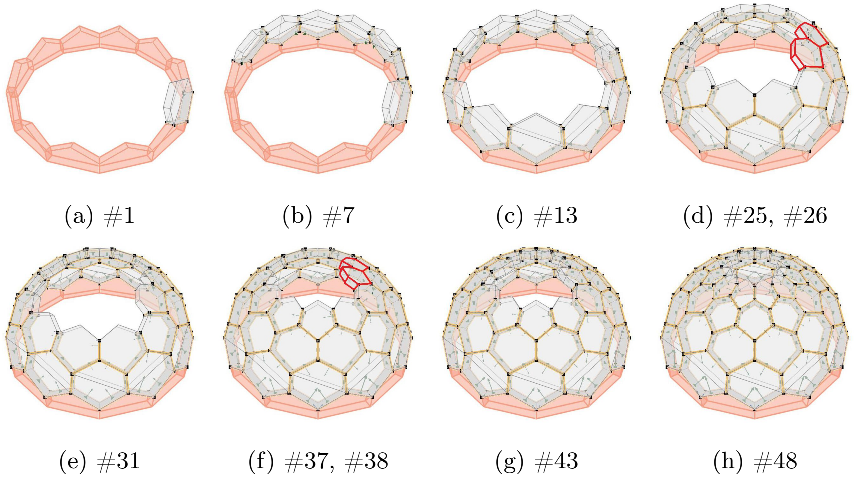
Figure 7.14: The geometrical procedure is similar to the step described in Figure 7.11 to generate a dome structure.

with neighbouring blocks 26 and 38.

To verify our AAD workflow and simulation results, we fabricated the physical models with Prusa i3 MK2 3D Printer and apply sandpaper to the blocks' interfaces to ensure the friction coefficient is above 0.66 as our settings in the CRA simulations. Additionally, in Figure 7.16, we used two Universal Robots UR5 with customised end effectors to assemble the dome to demonstrate the possibility of embedding robotic planning with the AAD workflow. We use two robots to simultaneously place two blocks stably, as shown in the CRA simulations in Figure 7.15d and f.

7.5.3 Free-form discrete shell designs

In architectural practice, a surface generated through form-finding methods can be more complicated than a simple vault or a dome shape. Rippmann

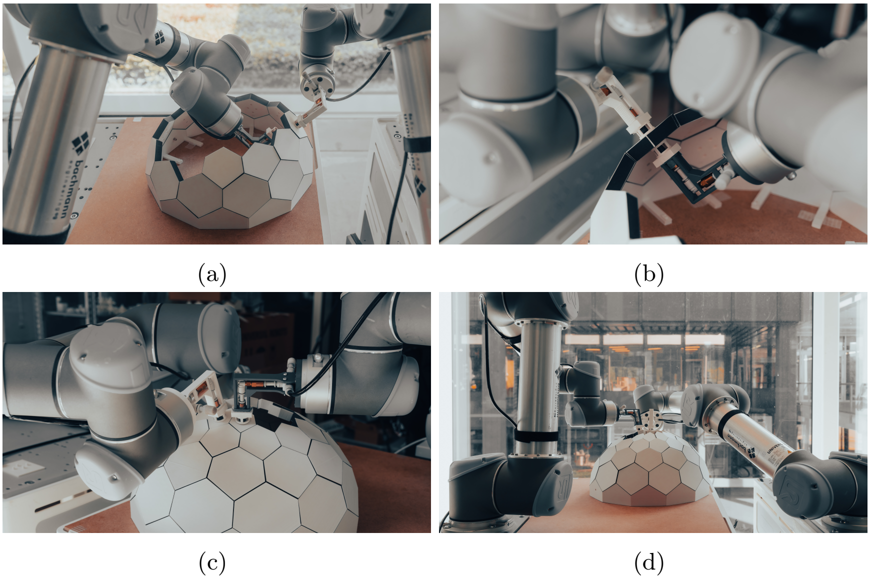


(a) #1 (b) #7 (c) #13 (d) #25, #26

(e) #31 (f) #37, #38 (g) #43 (h) #48

Figure 7.15: The stable assembly steps of a dome structure Wang et al. [2023]. The dome is designed using the AAD workflow and can be assembled entirely scaffolding-free. Some steps must be assembled simultaneously, such as block numbers 25 and 26 in (d) or 37 and 39 in (f), highlighted in red.

[2016] and Oval [2019] show that complex shell structures can be decomposed into sub-patches. The Armadillo vault, shown in Figure 6.20, also has a patch topology and staggered configuration [Block et al., 2017c]. For specific shell structures, it is straightforward for users to recognise the underlying basic shapes. For example, Figure 7.17a shows that for a freeform surface, the user can intuitively decompose it into four-sided patches, Figure 7.17b, and generate courses accordingly, Figure 7.17c. In this specific case, Figure 7.17d, a discrete freeform shell structure is generated through a three-vault topology. With the help of the AAD workflow presented in the previous Subsection 7.5.1, the vault can be designed and assembled through arches. Additionally, by adjusting block interfaces between courses, AAD can help users become aware of the assembly processing and guide them toward a discrete shell design using limited scaffoldings.



(a) (b) (c) (d)

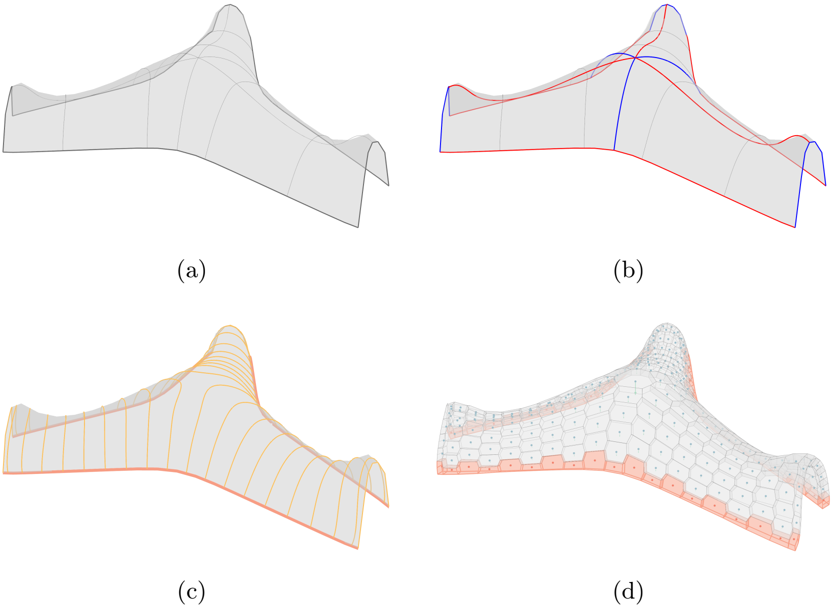
Figure 7.16: The dome design through AAD can be 3D-printed and assembled scaffolding-free with two robotic arms Wang et al. [2023].

7.6 Summary

This section presented an Assembly-Aware Design (AAD) workflow for discrete shell structures. The AAD is built upon the CRA method (Chapter 5) and the stability-aware design workflow (Section 6.3).

Section 7.1 briefly reiterated the urgent need to develop such a workflow. To help and give the reader details for understanding our proposed workflow process, in Section 7.2, we looked at more intuitive 2D examples step by step:

1. We propose one possible dis/assembly sequence.
2. We iteratively create the discretisation "cuts". At each iteration step, the discretisation should not compromised the structure. Therefore, the "cut" designs are evaluated and redirected through CRA and the


(a)

(b)

(c)

(d)

Figure 7.17: A more complex freeform shell structure can be decomposed into simpler shapes. (a) A freeform shell surface is designed through form-finding. (b) Dividing freeform surface into sub-patches. (c) Creating course curves for all patches. (d) Materisation of the shell structure.

stability-aware design workflow.

3. We reverse the disassembly process to obtain a feasible assembly sequence.

4. We design, simulate and build two scale models to demonstrate that our method is physically feasible.

After demonstrating the workflow with detail examples, in Section 7.3, we formalised the AAD workflow in a flowchart, giving the reader a clear overview of the process. Shell structures are three-dimensional structures; it is more complicated to consider both discretisation and assembly sequence simultaneously. Section 7.4 presented a procedure to generate discrete shell structures from shell surfaces. Learning from historical masonry's staggered patterns, we discretise the shell surface into course curves that can predefine the assembly sequence. Moreover, we demonstrate the procedure to materialise the structure and relate our discretisation to the AAD workflow. Additionally, at the end of the section, we show an interlocking block design. The AAD workflow allows designers to design block assemblies that can overcome stability during the shell assembly process. Finally, Section 7.5 presented three designs, a vault, a dome, and a freeform shell structure, to demonstrate the potential of AAD workflow applications.

There are many tessellation methods for shell structures; this chapter presented one for the purpose of demonstrating the AAD workflow. It is important to note that this research did not provide an optimal tessellation algorithm so that any shell structures could be applied and assembled entirely scaffolding-free. Alternatively, our AAD aimed to propose a workflow that helps users make informed design choices in the early design phases and be aware of possible assembly sequences simultaneously. The AAD workflow provides a framework for which each step in its workflow can be replaced with more automation or improved in the future.

Part IV

Implementation

Chapter 8

COMPAS CRA

This chapter presents COMPAS CRA (compas_cra) — our open-source implementation of Coupled Rigid-Block Analysis (CRA) solvers presented in Part II.

8.1 Introduction

This section provides basic information about our open-source library, an overview of the library structure, online documentation, and public release information.

8.1.1 Open source infrustructures

COMPAS by Van Mele et al. [2021] is an open-source Python-based computational framework for collaboration and research in architecture, engineering, fabrication, and construction. COMPAS CRA — Coupled Rigid-Block Analysis (CRA) for the COMPAS framework [Kao, 2022] — builds upon and utilises the framework's basic data structures, algorithm, geometry, and other functions to provide users with our research tools to solve structural equilibrium problems. Our implementation uses Python programming language [vanRossum, 1995] with some scientific libraries, such as NumPy [Oliphant, 2006] and SciPy [Virtanen et al., 2020]. Our provided solvers, CRA and RBE, uses Pyomo as a modelling language [Hart et al., 2017] and IPOPT as a solver of the nonlinear constrained optimisation problem [Wächter and Biegler, 2006]. Except for those popularly used and well-maintained Python libraries, our implementation is developed independently of any CAD software functionality, enabling portability in different platforms and efficiently used by various users in different research fields. Additionally, we utilised and extended some in-house COMPAS libraries, e.g., COMPAS Assembly [BlockResearchGroup, 2018] for our extended assembly data structure base to handle geometrical information; COMPAS View2 [BlockResearchGroup, 2021] for graphical visualisation with OpenGL.

COMPAS CRA is hosted publically on the GitHub repository:

https://github.com/BlockResearchGroup/compas_cra

8.1.2 Online documentation

The complete documentation is available online at:

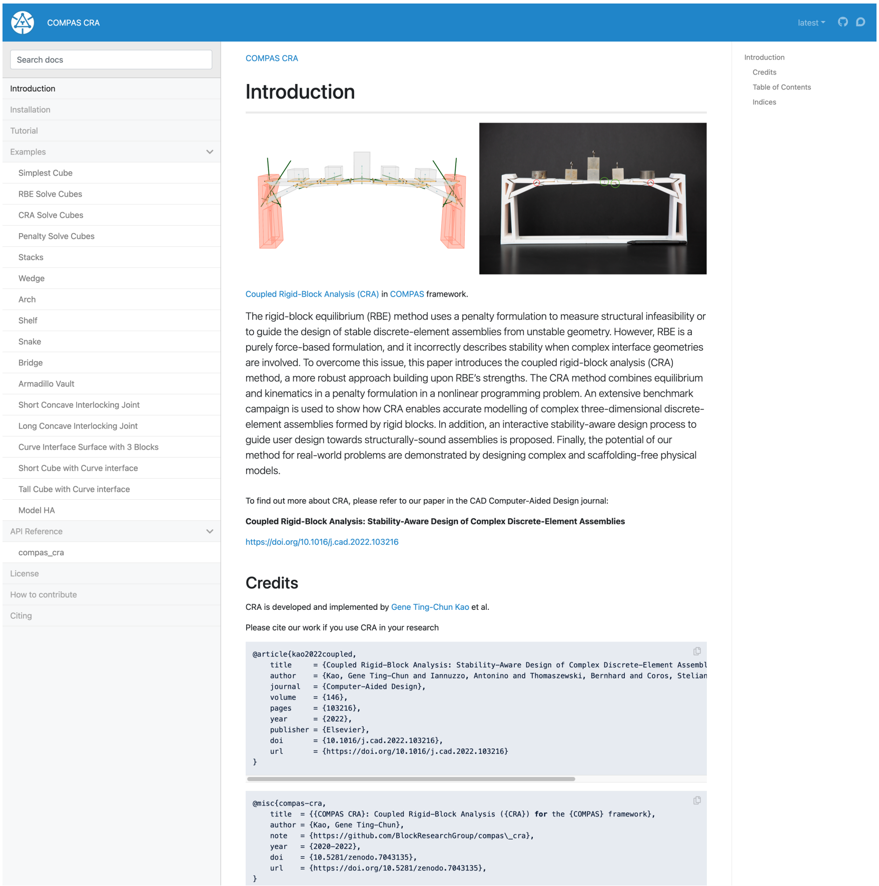
https://blockresearchgroup.github.io/compas_cra

The online documentation served as a reference and user manual. It contains various details that are helpful for the user, including an introduction, an installation guide, a step-by-step tutorial, examples, API references, the software license, contribution guides, and citing information.

8.1.3 Release

The first public release of COMPAS CRA was announced in September 2022 with version v0.2.0. There were a few minor changes and updates. The current version is v0.3.0. For the latest version and development, we refer readers to the DOI link:

https://doi.org/10.5281/zenodo.7043135



Introduction

Coupled Rigid-Block Analysis (CRA) in COMPAS framework.

The rigid-block equilibrium (RBE) method uses a penalty formulation to measure structural infeasibility or to guide the design of stable discrete-element assemblies from unstable geometry. However, RBE is a purely force-based formulation, and it incorrectly describes stability when complex interface geometries are involved. To overcome this issue, this paper introduces the coupled rigid-block analysis (CRA) method, a more robust approach building upon RBE's strengths. The CRA method combines equilibrium and kinematics in a penalty formulation in a nonlinear programming problem. An extensive benchmark campaign is used to show how CRA enables accurate modelling of complex three-dimensional discrete-element assemblies formed by rigid blocks. In addition, an interactive stability-aware design process to guide user design towards structurally-sound assemblies is proposed. Finally, the potential of our method for real-world problems are demonstrated by designing complex and scaffolding-free physical models.

To find out more about CRA, please refer to our paper in the CAD Computer-Aided Design journal:

Coupled Rigid-Block Analysis: Stability-Aware Design of Complex Discrete-Element Assemblies

https://doi.org/10.1016/j.cad.2022.103216

Credits

CRA is developed and implemented by Gene Ting-Chun Kao et al.

Please cite our work if you use CRA in your research

```
@article{kao2022coupled,
    title     = {Coupled Rigid-Block Analysis: Stability-Aware Design of Complex Discrete-Element Assembl
    author    = {Kao, Gene Ting-Chun and Iannuzzo, Antonino and Thomaszewski, Bernhard and Coros, Stelian
    journal   = {Computer-Aided Design},
    volume    = {146},
    pages     = {103216},
    year      = {2022},
    publisher = {Elsevier},
    doi       = {10.1016/j.cad.2022.103216},
    url       = {https://doi.org/10.1016/j.cad.2022.103216}
}
```

```
@misc{compas-cra,
    title  = {{COMPAS CRA}: Coupled Rigid-Block Analysis ({CRA}) for the {COMPAS} framework},
    author = {Kao, Gene Ting-Chun},
    note   = {https://github.com/BlockResearchGroup/compas\_cra},
    year   = {2020-2022},
    doi    = {10.5281/zenodo.7043135},
    url    = {https://doi.org/10.5281/zenodo.7043135},
}
```

Figure 8.1

8.1.4 Library structure

The content of COMPAS CRA in the current version 0.3.0 is structured as follows:

- **Data structure**

  This folder contains our extended assembly data structure, namely CRA_Assembly, which was inherited from the Assembly data structure from COMPAS Assembly. CRA_Assembly provides several handy functionalities, such as transforming assembly geometrically and storing multiple free-form interface data.

- **Algorithms**

  This folder contains algorithms such as scripts identifying contact interfaces between convex blocks.

- **Equilibrium**

  This folder contains the essential solvers with several helper functions, which are the core of this dissertation research work. We implemented three solvers rbe_solve, cra_solve, and cra_penalty_solve in the COMPAS CRA. The details of provided solvers are discussed in Section 8.2.

- **Geometry**

  This folder contains scripts to generate geometry, such as a parametric arch generation for the analysis.

- **Viewers**

  This folder contains our visualisation functionalities extended from COMPAS View2. The Viewers provides two handy functions, cra_view to launch an individual viewer and cra_view_ex to visualise assembly information to an existing viewer. They both provide several functional parameters, giving the user options to visualise or not the assembly information, such as nodal forces, resultant, interface, adjusting the scale of the models, and more.

8.2 Equlibrium solvers

In version v0.3.0, COMPAS CRA provides three solver functions: rbe_solve, cra_solve, and cra_penalty_solve. Three solvers exist in the equilibrium namespace and can be used interchangeably in different scenarios. Their first three function parameters from all solvers are the same: the first input is the assembly for checking the stability, the second is the friction coefficient value, and the third one is material density. We discuss three solvers in the following sections. For the latest API reference, we refer the reader to our API reference in COMPAS CRA documentation:

`https://blockresearchgroup.github.io/compas_cra/latest/api.html`

8.2.1 rbe_solve

The compas_cra.equilibrium.rbe_solve provides the solver using the Rigid-Block Equilibrium (RBE) method, as discussed in formulation (4.7) in Chapter 4.

The rbe_solve input parameters are as follows:

- assembly — the rigid block assembly for analysis, can be Assembly or CRA_Assembly,
- mu — friction coefficient value μ,
- density — density ρ of the block material,
- verbose — true to print information during the execution of the solving process,
- timer — true to print optimisation solving time,

and it returns:

- the assembly is updated in place, and this function also returns (Assembly or CRA_Assembly) for other COMPAS packages, such as COMPAS RPC and COMPAS Cloud.

8.2.2 cra_solve

The compas_cra.equilibrium.cra_solve provides the solver using the Coupled Rigid-Block Analysis (CRA) method, as discussed in formulation (5.6) in Chapter 5.

Compared to the rbe_solve, cra_solve has two additional inputs as follows:

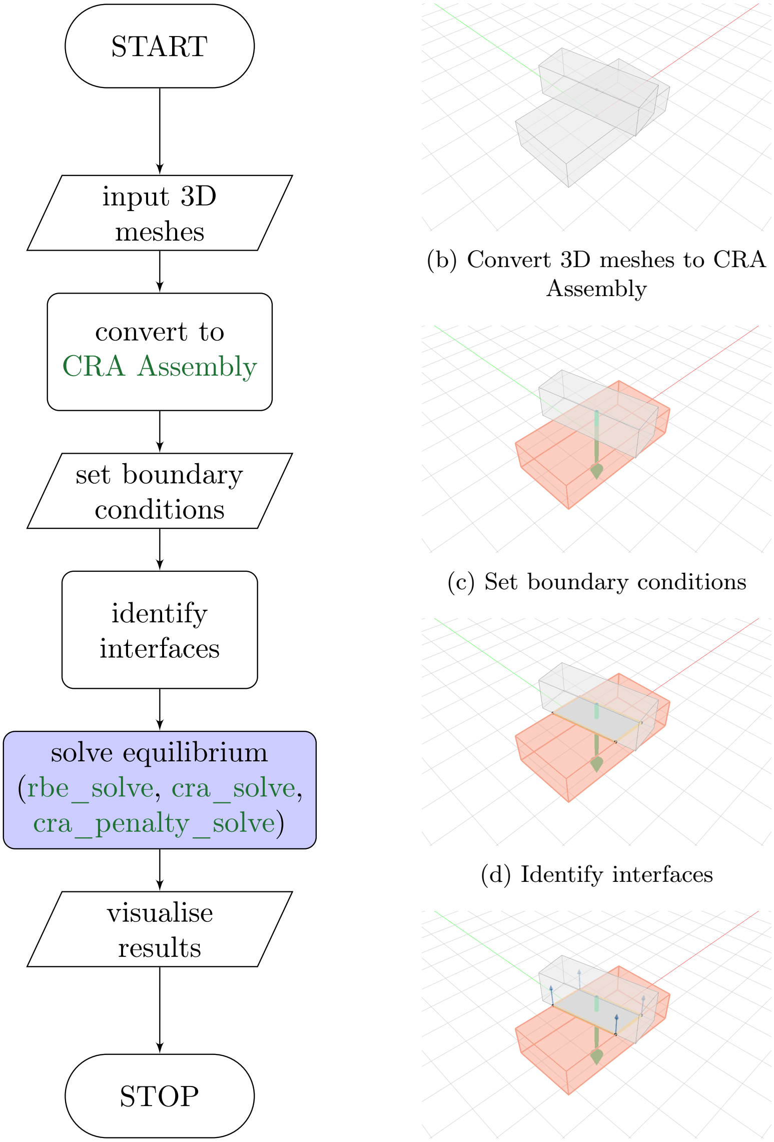
- d_bnd (float, optional) — the bound of virtual displacement η,
- eps (float, optional) — contact overlapping parameter ε.

8.2.3 cra_penalty_solve

The compas_cra.equilibrium.cra_penalty_solve provides the solver using the Coupled Rigid-Block Analysis (CRA) method with penalty formulation, as discussed in formulation (5.8) in Chapter 5. All its inputs and outputs are identical with the cra_solve.

8.3 Workflow

In addition to the core solvers, COMPAS CRA provides several handy tools for users to operate efficiently. Figure 8.2a shows the workflow of using COMPAS CRA tools. Firstly, in Figure 8.2b, the user inputs 3D meshes and uses scripts in the compas_cra.datastructures namespace to convert meshes into CRA_Assembly. After boundary conditions are correctly set in Figure 8.2c, the scripts in the compas_cra.algorithms namespace can automatically identify interfaces between blocks, see Figure 8.2d. Note that for our algorithms to be properly functional, all input meshes must be watertight and 2-manifold. Once all geometries and properties are set in the CRA_Assembly data structure, the user can use one of the three provided solvers (rbe_solve, cra_solve, and cra_penalty_solve) for the equilibrium analysis and visualise our results using functions provided in the compas_cra.viewers namespace, see Figure 8.2c.



(b) Convert 3D meshes to CRA Assembly

(c) Set boundary conditions

(d) Identify interfaces

(a) Workflow

(e) Solve equlibrium

Figure 8.2: COMPAS CRA Workflow. (a) The step-by-step flowchart shows the workflow processing of using COMPAS CRA. (b)-(e) Screenshots of the standalone viewer showing the individual step of the workflow process.

8.4 Examples

This section provides some example code snippets to express our workflow (Figure 8.2a) in the previous section. Firstly, we create two boxes for the analysis (Figure 8.2b):

```
from compas.geometry import Box, Frame, Translation

support = Box(Frame.worldXY(), 4, 2, 1)  # supporting block
free1 = Box(
    Frame.worldXY().transformed(
        Translation.from_vector([0, 0, 1])
        * Rotation.from_axis_and_angle([0, 0, 1], 0.2)
    ), 1, 3, 1
)  # block to analyse
```

Secondly, we convert box geometries into CRA_Assembly:

```
from compas_assembly.datastructures import Block
from compas_cra.datastructures import CRA_Assembly

assembly = CRA_Assembly()
assembly.add_block(Block.from_shape(support), node=0)
assembly.add_block(Block.from_shape(free1), node=1)
```

Thirdly, we set the bottom box as the boundary condition (Figure 8.2c):

```
assembly.set_boundary_conditions([0])
```

Fourthly, we call the function to identify planar interfaces between two blocks automatically (Figure 8.2d):

```
from compas_cra.algorithms import assembly_interfaces_numpy
assembly_interfaces_numpy(assembly)
```

Fifthly, we use cra_solve to find structure equilibrium (Figure 8.2d):

```
from compas_cra.equilibrium import cra_solve
cra_solve(assembly, verbose=True, timer=True)
```

Lastly, we run the viewer to visualise the result:

```python
from compas_cra.viewers import cra_view
cra_view(assembly, resultant=False, nodal=True, grid=True)
```

Additionally, COMPAS CRA provides several examples for users to reproduce our paper results. They can be found in our online documentation:

https://blockresearchgroup.github.io/compas_cra/latest/examples.html

8.5 Performance

Examples and benchmarks in this dissertation were performed on a MacBook Pro with a 2.9 GHz 6-Core Intel Core i9 Processor and 16 GB memory. Table 8.1 reports the computational burden listed for the performed analyses.

Table 8.1: The performance of all examples and all solving times of models are analysed at the rest position without applying any tilting.

Fig	Model	#Blocks	#(Sub)Interfaces	Solving time (sec)
6.11	Arch thick	38	37	1.02
6.12	Shelf	11	19	0.5
6.14a	type-a	3	2	0.05
6.14b	type-b	3	2	0.05
6.14c	type-c	3	2	0.05
6.14d	type-d	3	2	0.05
6.15a	concave-short	2	7	0.23
6.15b	concave-long	2	7	0.15
6.16a	curve-3-blocks	3	40	2.38
6.16b	cube-curve-short	2	72	1.15
6.16c	cube-curve-tall	2	72	1.04
6.20	Armadillo Vault	399	1014	2424.44
7.1h	Leaning tower	8	11	2.43
7.2	Snake	4	7	0.35
7.3	Completed bridge	11	33	1.27
7.4	Bridge with load	16	38	1.26

Part V

Conclusion

Chapter 9

Conclusion

Discrete shell structures have various structural benefits and are aesthetically ravishing. Discrete shells, such as ancient masonry structures, have been standing for thousands of years but have yet to be built often recently compared to concrete or steel structures due to a lack of construction efficiency and proper tools. Modern computational technics and construction technology have opened a new opportunity to design and build discrete structures more efficiently. Therefore, this dissertation has developed a new structural solver and workflow for designing discrete shell structures. The various chapters provided the motivations for the problem, looked at some critical challenges with some literature background studies, reviewed the state-of-the-art Rigid-Block Equilibrium (RBE) method thoroughly with some extended tools, formulated the new robust Coupled Rigid-Block Analysis (CRA) method, showed CRA for assessment, proposed a new Stability and Assembly-Aware Design workflow for the discrete shell structures, and also contributed our software implementation — COMPAS CRA in the open-source community.

This final chapter presents the concluding summary statement and some relevant discussions for this research and reflects the problem statement and research objectives outlined in Chapter 3. Additionally, the limitations and potential future research directions are also listed. At the end of this chapter, we conclude this research with some final reflections.

9.1 Contributions

This dissertation contributes various aspects which relate to the research objectives outlined in Chapter 3.

- **Better understanding of the mechanical behaviour of RBE results**
    - ○ We introduced the kern of a generic polygonal interface as a primal variable in the optimisation process. Its use allows us to assess and define the range of external actions for which the structure or its part is fully working in compression;
    - ○ We highlighted how to use RBE to explore different internal admissible stress states by virtually reducing the interface (e.g., to its kern). This method is a crucial aspect of the RBE method since it leaves the problem feasible, and the virtual reduction does not affect the feasibility of the problem, and in this sense, RBE returns as a solution the one that best matches the prescribed requirement;
    - ○ Furthermore, since RBE implicitly takes the thickness of the assembly into account, it provides a more extensive range of statically admissible solutions when compared to methods that use compressive 2D or 1D elements;
    - ○ We highlighted the main differences in using a *friction-net* and *friction+* approach and show how the *friction-net* approach with a global threshold can be helpful when the aim is to evaluate ultimate states ruled by the friction capacity; and,
    - ○ To understand the rule of the nodal forces, we performed and compared all analyses using the original quadratic objective func-

tion and a linear one. They both provide admissible stress states, but the difference between them increases as the dimension of the graph network, and thus also indeterminacy of the problem increases. The quadratic formulation looks at distributing the nodal forces as much as possible, and it comes from an elastic interpretation of the contact among blocks. In contrast, the linear one looks at the interface as a rigid element.

- **New robust structural solver**
  - ○ We recognise and overcome issues of the state-of-the-art RBE implementation. To have an accurate tool for designing structurally informed assemblies with complex shapes and interfaces, we propose a new mathematical formulation that allows for the correct assessment of complex assemblies and excludes RBE's *physically unrealisable* solutions;
  - ○ We formulated a new CRA solver which enables correct stability assessments of connections with sharp wedge interfaces, which are not possible with existing equilibrium methods;
  - ○ We gave consistent notation and visualisation throughout this research to aid the reader in comprehending the RBE optimisation flaws and insight into resolving the broken formulation;
  - ○ Our approach is explicit and does not need complicated parameter tuning, which is critical when detailed information about material properties and their translation into numerical parameters is not available;
  - ○ Additionally, it is worth mentioning that our method is *static* and does not require simulating the movement of the objects, such that there is no additional time parameter to tune;
  - ○ We extend our optimisation with a penalty formulation, which enables the formulation to go beyond non-feasible solutions and

    provides meaningful information about which part of the assembly is not in equilibrium;

  - The penalty formulation allows the user to redirect design choices. In our physical models, we also used the penalty formulation to obtain meaningful information about the discretisation process. CRA with penalty formulation not only suggests if the structure is stable but also improves our understanding of the structural response;
  - An accurate description of the assembly internal stress state when coupled with corresponding small virtual displacements represents a robust overall strategy to understand the structural behaviour during the design process; and,
  - Even though it is based on a nonlinear programming formulation, the computation time of CRA required for local stability analysis is satisfactory. Taking the Armadillo Vault — a real structure — as an example, we demonstrated that CRA is able to solve problems of practically relevant sizes and complexity. Table 8.1 reports the computational burden listed for all the performed analyses.

- **Benchmark solver results**
  - We benchmarked the CRA method with commercial solvers such as 3DEC (DEM) and Sofistik (nonlinear FEM) analysis;
  - Our numerical results are accurate and align well with analytical (whenever these are available or possible) and engineering software solutions;
  - We verified our numerical results with physical models; and,
  - In order to consider fabrication defects or assembly tolerances, we can use a lower friction value or reduced interface to obtain conservative results.

- **Improve assembly data structure**
    - We demonstrate that our approach is flexible and can be used beyond simple convex shapes. Specifically, we extend an existing graph-based data structure to handle information for complex discrete-element assemblies with concave shapes and freeform interfaces; and,
    - Compared to tedious manual convex decomposition or additional tetrahedralisation algorithm steps, the extended assembly data structure only stores sub-interfaces in the graph edge, which is much more straightforward to automate the discretisation process in the future.
- **Study of geometry and stability**
    - We explored many kinds of block geometries and interfaces, such as wedge angles, curve interfaces, and interlocking joints. We performed block stability analyses and tilt tests to find their limit states;
    - Compared to convex blocks with a planar interface, blocks with wedge interfaces have higher limit tilting angles, i.e., more stable — the sharper the angle for the wedge, the more stable the block during assembly; and,
    - Changing the wedge interface parameter enables some structures, e.g., dome structures, to be assembled entirely scaffolding-free.
- **Efficient construction sequencing strategy**
    - We developed a discretisation strategy that requires the user to explore the "cuts" sequence simultaneously with the dis/assembly sequence. The "cut" is iteratively generated by the user and adjusted through CRA formulation to ensure stable sub-structures; and,

  - Learning from the traditional scaffold-free masonry structures, we discretise the shell structures with a staggered configuration with a notable dis/assembly sequence and apply the AAD workflow.

- **Workflow for efficient design and construction of discrete shell structures**
  - Our formulation can be easily integrated into an interactive design workflow to help designers understand structural stability on the global and local scale;
  - The potential of CRA formulation is demonstrated with a proposed stability-aware design process with several design examples, showing that utilising CRA in the design workflow can help designers improve their design towards a more structurally sound solution;
  - In addition to the stability-aware design workflow, we proposed the Assembly-Aware Design (AAD) workflow to help the designer find the stable assembly sequence during the early design phase; and,
  - The potential of our proposed workflows is demonstrated in several design examples, such as the leaning tower, snake, cantilever bridge design, vault and domes, where the interfaces were iteratively designed using the proposed stability and assembly-aware design workflow to achieve stable solutions, both for the final structure and during assembly.
- **Open-source software implementation**
  - Using the COMPAS framework, we released an open-source COMPAS CRA (compas_cra) python software package [Kao, 2022], the Coupled Rigid-Block Analysis (CRA) implementation. The code is accessible publically and can be reused and further developed;

- COMPAS CRA is implemented completely standalone from CAD software. It uses pure CPython libraries that can be easily embedded in various CAD software environments; and,
- We also included thorough documentation of the COMPAS CRA implementation, including an introduction to the related theoretical backgrounds in research publications, an installation instruction, a step-by-step tutorial on geometrical set-ups and using multiple solvers, examples to reproduce our experiment results, API references to all public functions, software license, contribution guides for future collaborators, and citing information.

9.2 Discussions

This section discusses the potential contributions and impacts on relevant practice and research fields: architectural design, structural engineering, and computational fabrication.

9.2.1 Contribution related to architectural design

As discussed in Chapter 1, shell structures are a particular type of architecture that is difficult to build and requires massive amounts of scaffoldings during construction. This dissertation formulated a robust solver and proposed workflow, which aids designers in efficiently designing and building discrete shell structures. Through the open-source software library COMPAS CRA (compas_cra), the user can design discrete element assemblies and check the stability simultaneously, which can ensure their designs are structurally sound in the early design phase.

In addition to shell structures, the CRA method can also be applied to any discrete shape design. With the advancement of additive manufacturing and digital fabrication technics in architectural design, more and more customised prefab elements are designed, 3D printed and built. CRA can contribute and have an impact in those fields. Furthermore, in the early architectural design phase, the designers typically do not have the material or the structural behaviour of their design shape in mind. Therefore, compared to engineering software that requires detailed material properties for structural analysis, e.g., FEM analysis, CRA provides the designer with an easier, faster, more accurate, and more intuitive way of checking only the static equilibrium of rigid blocks. Without the need for detailed material properties for numerical models, CRA is a tool that can assist architectural design without restraining the designer's creativity.

9.2.2 Contribution related to structural engineering

In masonry structures, typically, engineers use DEM tools such as 3DEC software code to assess their mechanical behaviours, such as cracks and settlements. Compared to those typical toolsets, CRA provides an alternative solution to the masonry assessment. Furthermore, CRA is easier to use and completely open-source. Compared to commercial software requiring costly licence fees, CRA provides an alternative solution which is free of charge and open to modification and improvement.

For non-trivial discrete element assessments, we showed that much equilibrium-based research does not work well with elements with sharp wedges. In the worst cases, they give completely wrong results and falsely claim unsafe structures to be stable, which is extremely dangerous in practice. CRA formulation precisely describes the structural mechanics for assessing those structures.

9.2.3 Contribution related to computational fabrication

In addition to architectural design and structural engineering, CRA can be applied to computational fabrication in robotics research. CRA is suitable for assessing structural stability in digital fabrication or robotic assembly planning. Furthermore, CRA can be used in a wide range of fabrication for furniture, equilibrium sculptures, toys, or other industrial designs.

To conclude, the potential of CRA and its contribution to computational fabrication has been demonstrated by designing some physical scale models in surprising equilibrium configurations, where the physical contacts of 3D-printed blocks match the CRA interface forces. Also, in some examples, we assembled blocks with robots to demonstrate the practical usage of our research in computational fabrication and robotics.

9.3 Limitation and future works

This dissertation presented a robust solver and workflow for designers to design discrete shell structures with assembly awareness in mind. While many contributions and advantages have been highlighted, we also summarise some limitations and future research direction as follows:

- **Computational time and convergence**

  The presented formulation CRA is based on nonlinear constrained optimisation. Thus, it does not guarantee global convergence, and different starting points may result in different local optima. Some additional future enhancements include parallel programming and other nonlinear solving algorithms, which can significantly speed up the optimisation solving time and convergence rate.

- **Shape optimisation**

  RBE is a gradient-based optimisation, which has been used for automatic shape optimisation in various research works [Whiting et al., 2012; Wang et al., 2018, 2021a], e.g., using sensitivity analysis to modify block geometry to reduce the tension. However, shape optimisation relying on RBE may converge to an incorrect solution, as RBE failed to assess stability in some instances. Completely removing friction leads to over-conservative solutions. Therefore, with its penalty formulation, CRA represents an ideal basis for automatic shape optimisation of initially infeasible configurations.

- **Dynamic analysis to check settlements**

  CRA is a static analysis method. All displacements are entirely virtual, only coupled with the force solution. However, with prospective modification, CRA is also suitable for using a sequential optimisation procedure to superimpose small displacements at each step to capture equilibrated solutions on a deformed but safe configuration.

- **Automatic discretisation**

  The current implementation requires users manually input the discretisation of the sub-interfaces for the freeform block assembly. Therefore, some future enhancements include some algorithmic procedures to discretise the freeform interface automatically.

- **Better CAD integration and GUI with user interaction**

  We provided CRA implementation with APIs in open-source software libraries. Future developments include more CAD software integration and GUI to help users interact with CRA results' information more conveniently. Some valuable features, such as allowing users to apply external loads to the structure, can significantly assist users in better understanding the structural behaviours of discrete-element assembly.

- **Professional usage with a combination of other tools**

  The CRA provides a valuable tool to assess the stability of rigid blocks. However, it does not give any information related to material strain, and it is hard to tell if the material will fail. A more thorough study of stress distribution or combination with other advanced engineering software, e.g., FEM analysis, can become a more powerful toolset.

- **More automation involved in AAD workflow**

  Our proposed discrete shell generation in the AAD workflow is neither the only nor the optimal discretisation. Finding the optimal discretisation for shell construction with any scaffolding is a challenging research topic beyond our scope. We aimed to demonstrate the workflow's potential, allow the user to explore the structure's geometry, and enhance their stability and assembly awareness during the design process.

Additionally, some of the steps in the AAD workflow can involve more automation to assist designers further. For example, further implementation of a more automated shell course generation or a more sophisticated tessellation algorithm can be included.

- **Uncertainty and tolerances during robotic assembly**

  CRA can be applied in robotic research to assess stability during assembly planning. CRA gives theoretically correct solutions which do not consider any tolerance or uncertainty in the real-world scenario. Further research in this aspect is beneficial in practice.

- **Production ready implementation**

  We publically released our open-source implementation to help designers and researchers to explore and reproduce our results. Our naive implementation does not intend to be applied directly in practice. One of our implementation limitations is that COMPAS CRA uses the IPOPT solver, whose variable range and scales affect the results' accuracy. The solver with our current implementation sometimes needs manual density tweaking to find a solution, especially when a large-scale optimisation problem with the wrong scale. This issue must be further overcome for our solver to be adequately used in practice. One potential fix is to use RBE's result as a proper scaling factor to preprocess the model into the correct scale. For COMPAS CRA to be safely used in practice, some standard software engineering practices should be applied, including more detailed unit tests for corner case usage.

9.4 Final reflections

Achieving entirely scaffold-free assembling of discrete shell structures is incredibly challenging and the holy grail research area in shell structures. Many aspects must be fully addressed and overcome to reach such a target. This dissertation addressed the two essential aspects of the problem and presented: i) a robust solver that can accurately assess discrete-element assembly to help designers creates stability-aware design. ii) an assembly-aware design workflow to assist in designing and building discrete shell structures with limited scaffoldings. Importantly, our research intends to open up various research directions to reduce wasteful formwork/falsework/support-/scaffolding in the construction industry. Our open-source library enables researchers to build debates upon our findings to achieve such goals. Such research debate is only possible and meaningful if a robust solver CRA is publicly available. Finally, the ultimate expectancy of this dissertation is to see more contributions and extensions to the Assembly-Aware Design (AAD) frameworks and our Coupled Rigid-Block Analysis (CRA) be widely spread and improved over time across individual researchers and institutions. We hope this dissertation and its future extensions can significantly help designers increase their stability and assembly awareness and design better architecture and structures.

List of Figures

List of Tables

List of Symbols

Symbol	Description
$\mathcal{A}$	Assembly set.
$\mathcal{B}$	Block set.
$\mathcal{C}$	Interface set.
$\mathcal{C}_{jk}$	Polygonal planar interface between block j and k.
$\mathbf{c}^i_{jk}$	i^{th} vertex of $\mathcal{C}_{jk}$ interface.
$G(V, E)$	Graph representing the assembly $\mathcal{A}$ data structures.
V	Vertices of the graph $G(V, E)$.
E	Directed edges of the graph $G(V, E)$.
$\mathbf{w}$	Self-weight of block.
μ	Friction coefficient.
$\mathbf{r}^i_{kj}$	The torque arm vector pointing from mass centre of $\mathcal{B}_k$ to $\mathbf{c}^i_{jk}$.
$\mathbf{A}_{eq}$	Equilibrium matrix.
$\mathbf{A}_{fr}$	Matrix enforcing the linearised Mohr–Coulomb yield criterion.
$(\hat{\mathbf{u}}_{jk}, \hat{\mathbf{v}}_{jk}, \hat{\mathbf{n}}_{jk})$	Local reference system of interface $\mathcal{C}_{jk}$.
m_{jk}	Total number of vertices $\mathbf{c}^i_{jk}$ of interface $\mathcal{C}_{jk}$.

$\mathbf{f}^i_{jk}$ Unknown nodal force vector acting on the vertex $\mathbf{c}^i_{jk}$ of the interface $\mathcal{C}_{jk}$.

$\mathbf{f}^i_{jku}$ Interface's force $\mathbf{f}^i_{jk}$ tangential component in $\hat{\mathbf{u}}_{jk}$ direction.

$\mathbf{f}^i_{jkv}$ Interface's force $\mathbf{f}^i_{jk}$ tangential component in $\hat{\mathbf{v}}_{jk}$ direction.

$\mathbf{f}^i_{jkn}$ Interface's force $\mathbf{f}^i_{jk}$ normal component in $\hat{\mathbf{n}}_{jk}$ direction.

$\mathbf{f}^i_{jkt}$ Interface's tangential force on $\mathbf{c}^i_{jk}$.

$\mathbf{p}$ Vector collecting the external forces as acting on the blocks' centroids.

f^{i+}_{jkn} Positive component of the normal force $\mathbf{f}^i_{jkn}$.

f^{i-}_{jkn} Negative component of the normal force $\mathbf{f}^i_{jkn}$.

$\mathbf{f}$ Vector collecting all interface force components wrt the $(\hat{\mathbf{n}}_{jk}, \hat{\mathbf{u}}_{jk}, \hat{\mathbf{v}}_{jk})$ local reference system.

$\tilde{\mathbf{f}}$ Vector collecting all interface force components wrt the $(+\hat{\mathbf{n}}_{jk}, -\hat{\mathbf{n}}_{jk}, \hat{\mathbf{u}}_{jk}, \hat{\mathbf{v}}_{jk})$ local reference system.

$\mathbf{B}$ Matrix mapping $\tilde{\mathbf{f}}$ to $\mathbf{f}$.

$\mathbf{H}$ Diagonal square matrix collecting the force components' weights.

$\delta\mathbf{q}_k$ Vector collecting the six Lagrangian parameters of $\mathcal{B}_k$.

$\delta\mathbf{d}^i_{jku}$ Tangential component in the $\hat{\mathbf{u}}_{jk}$ of the relative virtual displacement between block j and k on $\mathbf{c}^i_{jk}$.

$\delta\mathbf{d}^i_{jkv}$ Tangential component in the $\hat{\mathbf{v}}_{jk}$ of the relative virtual displacement between block j and k on $\mathbf{c}^i_{jk}$.

$\delta\mathbf{d}^i_{jkn}$ Tangential component in the $\hat{\mathbf{n}}_{jk}$ of the relative virtual displacement between block j and k on $\mathbf{c}^i_{jk}$.

$\delta\mathbf{d}^i_{jkt}$ Interface's nodal tangential virtual displacement on $\mathbf{c}^i_{jk}$.

α^i_{jk} Non-negative parameter to align friction force $\mathbf{f}^i_{jkt}$ direction with relative sliding displacement $\delta\mathbf{d}^i_{jkt}$.

ε	Small numerical overlap parameter (i.e. 10^{-5} - 10^{-4} of the block dimension).
η	Limit value defining the max allowable displacement.
γ	Weighting factor for penalising negative normal force.
ρ	Material density.
$\boldsymbol{\varsigma}_{jk}^{i}$	The position vector of the i-th vertex in the local coordinate system of interface $\mathcal{C}_{jk}$.
$\boldsymbol{I}^{(n)}$	The n^{th}-order area moments of an polygon.
I_u, I_v	Components of the first-order area moment of the polygon.
$I_{uu}, I_{vv}, I_{uv}, I_{vu}$	Components of the second-order area moment of the polygon.

Bibliography

Edward Allen and Waclaw Zalewski. 2009. *Form and forces: designing efficient, expressive structures*. John Wiley & Sons. (↑ p. 27, 28, 128, 174)

Maurizio Angelillo, Luca Cardamone, and Antonio Fortunato. 2010. A numerical model for masonry-like structures. *Journal of Mechanics of Materials and Structures* 5, 4 (2010), 583–615. (↑ p. 82)

Maurizio Angelillo and Antonio Fortunato. 2004. Equilibrium of masonry vaults. In *Novel approaches in civil engineering*. Springer, 105–111. (↑ p. 25)

Maurizio Angelillo, Antonio Fortunato, Antonio Gesualdo, Antonino Iannuzzo, and Giulio Zuccaro. 2018. Rigid block models for masonry structures. *International Journal of Masonry Research and Innovation* 3, 4 (2018), 349–368. (↑ p. 40)

Tolga Atar, Theresa Mccabe, Ece Erdogmus, and Bora Pulatsu. 2020. Dynamic Analysis of Semi-Circular Dry-Joint Masonry Arches: Small-Scale Experiment and Discrete Element Modeling. *Wseas Transactions on Computers* 19, 19 (2020), 137–142. (↑ p. 97)

David Baraff. 1991. Coping with friction for non-penetrating rigid body simulation. *ACM SIGGRAPH computer graphics* 25, 4 (1991), 31–41. (↑ p. 24)

David Baraff. 1993. Issues in computing contact forces for non-penetrating rigid bodies. *Algorithmica* 10, 2 (1993), 292–352. (↑ p. 24)

David Baraff. 1994. Fast contact force computation for nonpenetrating rigid bodies. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*. 23–34. (↑ p. 24)

Javier Barrallo and Santiago Sánchez-Beitia. 2011. The geometry of organic architecture: the works of Eduardo Torroja, Felix Candela and Miguel Fisac. In *Proceedings of Bridges 2011: Mathematics, Music, Art, Architecture, Culture*. 65–72. (↑ p. 14)

Simon Bechert, Daniel Sonntag, Lotte Aldinger, and Jan Knippers. 2021. Integrative structural design and engineering methods for segmented timber shells-BUGA Wood Pavilion. In *Structures*, Vol. 34. Elsevier, 4814–4833. (↑ p. 21, 22, 174)

Dimitris Bertsimas and John N Tsitsiklis. 1997. *Introduction to linear optimization*. Vol. 6. Athena Scientific Belmont, MA. (↑ p. 45)

Lukas Beyeler, Jean-Charles Bazin, and Emily Whiting. 2015. A graph-based approach for discovery of stable deconstruction sequences. In *Advances in Architectural Geometry 2014*. Springer, 145–157. (↑ p. 26, 30)

Shajay Bhooshan, Vishu Bhooshan, Alessandro Dell'Endice, Jianfei Chu, Philip Singer, Johannes Megens, Tom Van Mele, and Philippe Block. 2022. The Striatus bridge: Computational design and robotic fabrication of an unreinforced, 3D-concrete-printed masonry arch bridge. *Architecture, Structures and Construction* (2022), 1–23. (↑ p. 10, 11, 21, 173)

Shajay Bhooshan, Tom Van Mele, and Philippe Block. 2018. Equilibrium-aware shape design for concrete printing. In *Humanizing Digital Reality*. Springer, 493–508. (↑ p. 26)

Carlo Blasi and Paolo Foraboschi. 1994. Analytical approach to collapse mechanisms of circular masonry arch. *Journal of structural engineering* 120, 8 (1994), 2288–2309. (↑ p. 97)

Philippe Block. 2009. *Thrust Network Analysis: Exploring Three-dimensional Equilibrium*. Ph.D. Dissertation. Cambridge, MA, USA. PhD dissertation. (↑ p. 135)

Philippe Block, Thierry Ciblac, and John Ochsendorf. 2006a. Real-time limit analysis of vaulted masonry buildings. *Computers & structures* 84, 29-30 (2006), 1841–1852. (↑ p. 24, 25)

Philippe Block, Matt DeJong, and John Ochsendorf. 2006b. As hangs the flexible line: Equilibrium of masonry arches. *Nexus Network Journal* 8, 2 (2006), 13–24. (↑ p. 86)

Philippe Block and Lorenz Lachauer. 2014. Three-dimensional funicular analysis of masonry vaults. *Mechanics Research Communications* 56 (2014), 53–60. (↑ p. 25)

Philippe Block and John Ochsendorf. 2007. Thrust network analysis: a new methodology for three-dimensional equilibrium. *Journal of the International Association for shell and spatial structures* 48, 3 (2007), 167–173. (↑ p. 20, 25)

Philippe Block, Matthias Rippmann, Tom Van Mele, and David Escobedo. 2017a. The Armadillo Vault: Balancing computation and traditional craft. *Fabricate* (2017), 286–293. (↑ p. 111)

Philippe Block, Arno Schlueter, Diederik Veenendaal, Jack Bakker, Moritz Begle, Illias Hischier, Johannes Hofer, Prageeth Jayathissa, Iain Maxwell, T Méndez Echenagucia, et al. 2017b. NEST HiLo: Investigating lightweight construction and adaptive energy systems. *Journal of Building Engineering* 12 (2017), 332–341. (↑ p. 9)

Philippe Block, Tom Van Mele, Andrew Liew, Matthew DeJong, David Escobedo, and John A Ochsendorf. 2018. Structural design, fabrication and construction of the Armadillo vault. *The Structural Engineer: journal of the Institution of Structural Engineer* 96, 5 (2018), 10–20. (↑ p. 111)

Philippe Block, Tom Van Mele, M Rippmann, and Noelle Paulson. 2017c. *Beyond Bending - Reimagining Compression Shells*. Edition DETAIL, Munich. (↑ p. 11, 142, 173)

BlockResearchGroup. 2018. COMPAS Assembly. `https://blockresearchgroup.github.io/compas_assembly` (↑ p. 55, 72, 150)

BlockResearchGroup. 2020. rhinoVAULT 2: Funicular Form Finding for Rhinoceros 6+. `https://github.com/BlockResearchGroup/compas-RV2` (↑ p. 20, 174)

BlockResearchGroup. 2021. COMPAS View2. `https://github.com/compas-dev/compas_view2` (↑ p. 150)

Geoffrey Boothroyd. 1994. Product design for manufacture and assembly. *Computer-Aided Design* 26, 7 (1994), 505–520. (↑ p. 13, 14, 30, 173)

Geoffrey Boothroyd and Leo Alting. 1992. Design for assembly and disassembly. *CIRP annals* 41, 2 (1992), 625–636. (↑ p. 30)

Sofien Bouaziz, Mario Deuss, Yuliy Schwartzburg, Thibaut Weise, and Mark Pauly. 2012. Shape-up: Shaping discrete geometry with projections. In *Computer Graphics Forum*, Vol. 31. Wiley Online Library, 1657–1667. (↑ p. 21, 131)

Sofien Bouaziz, Sebastian Martin, Tiantian Liu, Ladislav Kavan, and Mark Pauly. 2014. Projective dynamics: Fusing constraint projections for fast simulation. *ACM transactions on graphics (TOG)* 33, 4 (2014), 1–11. (↑ p. 21, 131)

Edvard PG Bruun, Rafael Pastrana, Vittorio Paris, Alessandro Beghini, Attilio Pizzigoni, Stefana Parascho, and Sigrid Adriaenssens. 2021. Three cooperative robotic fabrication methods for the scaffold-free construction of a masonry arch. *Automation in Construction* 129 (2021), 103803. (↑ p. 31)

Tan-Trung Bui, Ali Limam, Vasilis Sarhosis, and Mohammed Hjiaj. 2017. Discrete element modelling of the in-plane and out-of-plane behaviour of dry-joint masonry wall constructions. *Engineering Structures* 136 (2017), 277–294. (↑ p. 25)

Lucrezia Cascini, Raffaele Gagliardo, and Francesco Portioli. 2018. LiABlock_3D: a software tool for collapse mechanism analysis of historic masonry structures. *International Journal of Architectural Heritage* (2018). (↑ p. 69)

Mario Como. 2013. *Statics of historic masonry constructions*. Vol. 1. Springer. (↑ p. 25, 97)

E Coumans. 2013. Bullet collision detection and physics library. (↑ p. 56)

Peter A Cundall. 1971. A computer model for simulating progressive, large-scale movement in blocky rock system. In *Proceedings of the International Symposium on Rock Mechanics, 1971*. (↑ p. 25, 56)

Elena De Chiara, Claudia Cennamo, Antonio Gesualdo, Andrea Montanino, Carlo Olivieri, and Antonio Fortunato. 2019. Automatic generation of statically admissible stress fields in masonry vaults. *Journal of Mechanics of Materials and Structures* 14, 5 (2019), 719–737. (↑ p. 25)

Borja García de Soto, Isolda Agustí-Juan, Jens Hunhevicz, Samuel Joss, Konrad Graser, Guillaume Habert, and Bryan T Adey. 2018. Productivity of digital fabrication in construction: Cost and time analysis of a robotically built wall. *Automation in Construction* 92 (2018), 297–311. (↑ p. 11)

Matthew Justin DeJong. 2009. *Seismic assessment strategies for masonry structures*. Ph.D. Dissertation. Massachusetts Institute of Technology. (↑ p. 24, 25)

Matthew J DeJong, Laura De Lorenzis, Stuart Adams, and John A Ochsendorf. 2008. Rocking stability of masonry arches in seismic regions. *Earthquake Spectra* 24, 4 (2008), 847–865. (↑ p. 97)

Alessandro Dell'Endice, Antonino Iannuzzo, Matthew J DeJong, Tom Van Mele, and Philippe Block. 2021. Modelling imperfections in unreinforced masonry structures: Discrete Element simulations and scale model experiments of a pavilion vault. *Engineering Structures* 228 (2021), 111499. (↑ p. 25, 97)

Mario Deuss, Anders Holden Deleuran, Sofien Bouaziz, Bailin Deng, Daniel Piker, and Mark Pauly. 2015. ShapeOp—a robust and extensible geometric modelling paradigm. In *Modelling Behaviour*. Springer, 505–515. (↑ p. 21, 131)

Mario Deuss, Daniele Panozzo, Emily Whiting, Yang Liu, Philippe Block, Olga Sorkine-Hornung, and Mark Pauly. 2014. Assembling self-supporting structures. *ACM Trans. Graph.* 33, 6 (2014), 214–1. (↑ p. 20, 26, 31)

J. DREW. 2013. United Lock-Block Ltd. `https://lockblock.com/` (↑ p. 31)

D D'Ayala and C Casapulla. 2001. Limit state analysis of hemispherical domes with finite friction. *Historical constructions, possibilities of numerical and experimental techniques* (2001), 617–626. (↑ p. 25)

Michael Eigensatz, Martin Kilian, Alexander Schiftner, Niloy J Mitra, Helmut Pottmann, and Mark Pauly. 2010. Paneling architectural freeform surfaces. In *ACM SIGGRAPH 2010 papers*. 1–10. (↑ p. 21)

Kenny Erleben. 2007. Velocity-based shock propagation for multibody dynamics animation. *ACM Transactions on Graphics (TOG)* 26, 2 (2007), 12–es. (↑ p. 24)

MC Ferris and F Tin-Loi. 2001. Limit analysis of frictional block assemblies as a mathematical program with complementarity constraints. *International Journal of Mechanical Sciences* 43, 1 (2001), 209–224. (↑ p. 69, 70)

John Fitchen. 1981. *The construction of Gothic cathedrals: a study of medieval vault erection.* University of Chicago Press. (↑ p. 28, 128, 174)

Fernando Fraternali. 2010. A thrust network approach to the equilibrium problem of unreinforced masonry vaults via polyhedral stress functions. *Mechanics Research Communications* 37, 2 (2010), 198–204. (↑ p. 25)

Fernando Fraternali, Maurizio Angelillo, and Antonio Fortunato. 2002. A lumped stress method for plane elastic problems and the discrete-continuum approximation. *International journal of solids and structures* 39, 25 (2002), 6211–6240. (↑ p. 25)

Ursula Frick, Tom Van Mele, and Philippe Block. 2015. Decomposing three-dimensional shapes into self-supporting, discrete-element assemblies. In *Modelling Behaviour*. Springer, 187–201. (↑ p. 9, 26, 27, 40, 41, 46, 173, 174)

Ursula Frick, Tom Van Mele, and Philippe Block. 2016. Data management and modelling of complex interfaces in imperfect discrete-element assemblies. In *Proceedings of IASS Annual Symposia*, Vol. 2016. International Association for Shell and Spatial Structures (IASS), 1–9. (↑ p. 23, 27, 40, 41, 55, 72, 174)

Chi-Wing Fu, Peng Song, Xiaoqi Yan, Lee Wei Yang, Pradeep Kumar Jayaraman, and Daniel Cohen-Or. 2015. Computational interlocking furniture assembly. *ACM Transactions on Graphics (TOG)* 34, 4 (2015), 1–11. (↑ p. 23)

Antonio Gesualdo, Giuseppe Brandonisio, Antonello De Luca, Antonino Iannuzzo, Andrea Montanino, and Carlo Olivieri. 2019. Limit analysis of cloister vaults: The case study of Palazzo Caracciolo di Avellino. *Journal of Mechanics of Materials and Structures* 14, 5 (2019), 739–750. (↑ p. 25)

Somaye Ghandi and Ellips Masehian. 2015. Review and taxonomies of assembly and disassembly path planning problems and approaches. *Computer-Aided Design* 67 (2015), 58–86. (↑ p. 30)

M Gilbert, C Casapulla, and HM Ahmed. 2006. Limit analysis of masonry block structures with non-associative frictional joints using linear programming. *Computers & structures* 84, 13-14 (2006), 873–887. (↑ p. 26)

Melbourne Gilbert and C Melbourne. 1994. Rigid-block analysis of masonry structures. *Structural engineer* 72, 21 (1994). (↑ p. 25)

Yukiko Goda and Graeme W Davis. 2003. Mechanisms of synapse assembly and disassembly. *Neuron* 40, 2 (2003), 243–264. (↑ p. 30)

Suresh Goyal, Andy Ruina, and Jim Papadopoulos. 1991. Planar sliding with dry friction part 1. limit surface and moment function. *Wear* 143, 2 (1991), 307–330. (↑ p. 24)

Eran Guendelman, Robert Bridson, and Ronald Fedkiw. 2003. Nonconvex rigid bodies with stacking. *ACM transactions on graphics (TOG)* 22, 3 (2003), 871–878. (↑ p. 24)

Maximilian Haas-Heger and Matei Ciocarlie. 2020. Accurate Energetic Constraints for Passive Grasp Stability Analysis. *IEEE Transactions on Robotics* 36, 4 (2020), 1079–1095. (↑ p. 24)

David Hally. 1987. *Calculation of the Moments of Polygons.* Technical Report. DEFENCE RESEARCH ESTABLISHMENT SUFFIELD RALSTON (ALBERTA). (↑ p. 49, 50)

William E Hart, Carl D Laird, Jean-Paul Watson, David L Woodruff, Gabriel A Hackebeil, Bethany L Nicholson, John D Siirola, et al. 2017. *Pyomo-optimization modeling in python.* Vol. 67. Springer. (↑ p. 96, 150)

Jacques Heyman. 1966. The stone skeleton. *International Journal of solids and structures* 2, 2 (1966), 249–279. (↑ p. 25)

Jacques Heyman. 1969. The safety of masonry arches. *International Journal of Mechanical Sciences* 11, 4 (1969), 363–385. (↑ p. 85, 97)

Jacques Heyman. 1997. *The stone skeleton: structural engineering of masonry architecture*. Cambridge University Press. (↑ p. 90, 94)

Jacques Heyman. 2019. The structural engineer's view of ancient buildings. *Journal of Mechanics of Materials and Structures* 13, 5 (2019), 609–615. (↑ p. 94)

J Heyman and EC Hambly. 1996. HAMBLY'S PARADOX: WHY DESIGN CALCULATIONS DO NOT REFLECT REAL BEHAVIOUR.. In *Proceedings of the Institution of Civil Engineers-Civil Engineering*, Vol. 114. Thomas Telford-ICE Virtual Library, 161–166. (↑ p. 95)

Stefan M Holzer. 2021. How to Build a Dome. In *The History of Building Trades and Professionalism. Proceedings of the Eighth Conference of the Construction History Society*. Construction History Society, 197–206. (↑ p. 139)

Shu-Wei Hsu and John Keyser. 2012. Automated constraint placement to maintain pile shape. *ACM Transactions on Graphics (TOG)* 31, 6 (2012), 1–6. (↑ p. 26)

Santiago Huerta. 2006a. Galileo was wrong: the geometrical design of masonry arches. *Nexus Network Journal* 8, 2 (2006), 25–52. (↑ p. 25, 97)

Santiago Huerta. 2006b. Geometry and equilibrium: The gothic theory of structural design. *Structural Engineer* 84, 2 (2006), 23–28. (↑ p. 25, 90)

Antonino Iannuzzo. 2019. Energy based fracture identification in masonry structures: the case study of the church of "Pietà dei Turchini". *Journal of Mechanics of Materials and Structures* 14, 5 (2019), 683–702. (↑ p. 84)

Antonino Iannuzzo, Philippe Block, Maurizio Angelillo, and Antonio Gesualdo. 2021a. A continuous energy-based numerical approach to predict fracture mechanisms in masonry structures: CDF method. *Computers & Structures* 257 (2021), 106645. (↑ p. 61)

Antonino Iannuzzo, Antonello De Luca, Antonio Fortunato, Antonio Gesualdo, and Maurizio Angelillo. 2018. Fractures detection in masonry constructions under horizontal seismic forces. *Ingegneria Sismica* 35, 3 (2018), 87–103. (↑ p. 84)

Antonino Iannuzzo, Alessandro Dell'Endice, Tom Van Mele, and Philippe Block. 2021b. Numerical limit analysis-based modelling of masonry structures subjected to large displacements. *Computers & Structures* 242 (2021), 106372. (↑ p. 25)

Antonino Iannuzzo, Alessandro Dell'Endice, Ricardo Maia Avelino, Gene Ting-Chun Kao, Tom Van Mele, and Philippe Block. 2021c. COMPAS masonry: a computational framework for practical assessment of unreinforced masonry structures. In *Proceedings of the SAHC Symposium*. (↑ p. 20)

Antonino Iannuzzo, Tom Van Mele, and Philippe Block. 2020. Piecewise rigid displacement (PRD) method: a limit analysis-based approach to detect mechanisms and internal forces through two dual energy criteria. *Mechanics Research Communications* 107 (2020), 103557. (↑ p. 56, 61, 82, 84, 86, 87, 114)

U Itasca. 2013. 3DEC—3 Dimensional Distinct Element Code. *Itasca Consulting Group Inc., Minneapolis* (2013). (↑ p. 25, 96)

DI Kadzhaya. 1966. *Precast spherical cupola roof for sport palace in tbilisi and its erection by overhang method.* (↑ p. 29, 175)

Gene Ting-Chun Kao. 2022. COMPAS CRA: Coupled Rigid-Block Analysis (CRA) for the COMPAS framework. `https://doi.org/10.5281/zenodo.7043135` https://github.com/BlockResearchGroup/compas_cra. (↑ p. 149, 166)

Gene Ting-Chun Kao, Antonino Iannuzzo, Stelian Coros, Tom Van Mele, and Philippe Block. 2021. Understanding the rigid-block equilibrium method by way of mathematical programming. *Proceedings of the Institution of Civil Engineers-Engineering and Computational Mechanics* 174, 4 (2021), 178–192. (↑ p. 15)

Gene Ting-Chun Kao, Antonino Iannuzzo, Bernhard Thomaszewski, Stelian Coros, Tom Van Mele, and Philippe Block. 2022. Coupled Rigid-Block Analysis: Stability-Aware Design of Complex Discrete-Element Assemblies. *Computer-Aided Design* 146 (2022), 103216. `https://doi.org/10.1016/j.cad.2022.103216` (↑ p. 15)

Gene Ting-Chun Kao, Axel Körner, Daniel Sonntag, Long Nguyen, Achim Menges, and Jan Knippers. 2017. Assembly-aware design of masonry shell structures: a computational approach. In *Proceedings of IASS Annual Symposia*, Vol. 2017. International Association for Shell and Spatial Structures (IASS), 1–10. (↑ p. 31, 114, 121, 128, 175)

Efim N Kaplunovich and Christian Meyer. 1982. Shell construction with precast elements. *Concrete International* 4, 4 (1982), 37–43. (↑ p. 29, 175)

Danny M Kaufman, Shinjiro Sueda, Doug L James, and Dinesh K Pai. 2008. Staggered projections for frictional contact in multibody systems. In *ACM SIGGRAPH Asia 2008 papers*. 1–11. (↑ p. 24)

Ross King. 2013. *Brunelleschi's dome: how a Renaissance genius reinvented architecture*. Bloomsbury Publishing USA. (↑ p. 139)

Oliver David Krieg, Tobias Schwinn, Achim Menges, Jian-Min Li, Jan Knippers, Annette Schmitt, and Volker Schwieger. 2015. Biomimetic lightweight timber plate shells: computational integration of robotic fabrication, architectural geometry and structural design. In *Advances in architectural geometry 2014*. Springer, 109–125. (↑ p. 21, 22, 174)

L Laperriere and HA ElMaraghy. 1992. Planning of products assembly and disassembly. *CIRP annals* 41, 1 (1992), 5–9. (↑ p. 30)

JV Lemos. 1995. Assessment of the ultimate load of a masonry arch using discrete elements. *Computer methods in structural masonry* 3 (1995), 294–302. (↑ p. 25)

Jian-Min Li and Jan Knippers. 2015. Segmental timber plate shell for the landesgartenschau exhibition hall in schwäbisch gmünd—the application of finger joints in plate structures. *International Journal of Space Structures* 30, 2 (2015), 123–139. (↑ p. 21, 22, 174)

Zhengdao Li, Geoffrey Qiping Shen, and Xiaolong Xue. 2014. Critical review of the research on the management of prefabricated construction. *Habitat international* 43 (2014), 240–249. (↑ p. 9)

Robert Kenneth Livesley. 1978. Limit analysis of structures formed from rigid blocks. *International journal for numerical methods in engineering* 12, 12 (1978), 1853–1871. (↑ p. 25, 26, 40, 41)

Robert Kenneth Livesley. 1992. A computational model for the limit analysis of three-dimensional masonry structures. *Meccanica* 27, 3 (1992), 161–172. (↑ p. 25, 26, 40)

Ena Lloret, Amir R Shahab, Mettler Linus, Robert J Flatt, Fabio Gramazio, Matthias Kohler, and Silke Langenberg. 2015. Complex concrete structures: merging existing casting techniques with digital fabrication. *Computer-Aided Design* 60 (2015), 40–49. (↑ p. 11)

David López López, Marta Domènech-Rodríguez, Josep Brazo Ramírez, and Philippe Block. 2014. Thin-tile vault for the Seventh World Urban Forum in Medellin. In *Proceedings of IASS Annual Symposia*, Vol. 2014. International Association for Shell and Spatial Structures (IASS), 1–8. (↑ p. 20, 174)

Sheng-Jie Luo, Yonghao Yue, Chun-Kai Huang, Yu-Huan Chung, Sei Imai, Tomoyuki Nishita, and Bing-Yu Chen. 2015. Legolization: Optimizing lego designs. *ACM Transactions on Graphics (TOG)* 34, 6 (2015), 1–12. (↑ p. 23, 174)

William Lloyd MacDonald. 2002. *The Pantheon: design, meaning, and progeny.* Harvard University Press. (↑ p. 139)

Ricardo Maia Avelino, Antonino Iannuzzo, Tom Van Mele, and Philippe Block. 2021. Assessing the safety of vaulted masonry structures using thrust network analysis. *Computers & Structures* 257 (2021), 106647. (↑ p. 135)

Robert Mark and Paul Hutchinson. 1986. On the structure of the Roman Pantheon. *The Art Bulletin* 68, 1 (1986), 24–34. (↑ p. 139)

Matthew T Mason and Yu Wang. 1988. On the inconsistency of rigid-body frictional planar mechanics. In *Proceedings. 1988 IEEE International Conference on Robotics and Automation.* IEEE, 524–528. (↑ p. 24)

D. Matthews. 1955. Factory at Denton, Manchester for Matthews & Mumby, Ltd. *The Architect and Building News* (1955), 195–198. (↑ p. 29)

Raju Mattikalli, David Baraff, and Pradeep Khosla. 1996. Finding all stable orientations of assemblies with friction. *IEEE Transactions on Robotics and Automation* 12, 2 (1996), 290–301. (↑ p. 24, 65)

Raju Mattikalli, David Baraff, Pradeep Khosla, and Bruno Repetto. 1995. Gravitational stability of frictionless assemblies. *IEEE Transactions on Robotics and Automation* 11, 3 (1995), 374–388. (↑ p. 24)

Michael Grab. 2008. Gravity Glue. `https://gravityglue.com/` Accessed: 2022-10-21. (↑ p. 8, 9, 173)

Ioanna Mitropoulou, Mathias Bernhard, and Benjamin Dillenburger. 2020. Print Paths Key-framing: Design for non-planar layered robotic FDM printing. In *Symposium on Computational Fabrication.* 1–10. (↑ p. 21)

Jean Jacques Moreau. 1966. Quadratic programming in mechanics: dynamics of one-sided constraints. *SIAM Journal on control* 4, 1 (1966), 153–158. (↑ p. 65)

Heiko Mosemann, Frank Rohrdanz, and Friedrich M Wahl. 1997. Stability analysis of assemblies considering friction. *IEEE Transactions on Robotics and Automation* 13, 6 (1997), 805–813. (↑ p. 24)

Elham Mousavian and Claudia Casapulla. 2020a. The role of different sliding resistances in limit analysis of hemispherical masonry domes. *Frattura ed Integrità Strutturale* 14, 51 (2020), 336–355. (↑ p. 25)

Elham Mousavian and Claudia Casapulla. 2020b. Structurally informed design of interlocking block assemblages using limit analysis. *Journal of Computational Design and Engineering* 7, 4 (2020), 448–468. (↑ p. 26)

Pier Luigi Nervi. 1953. Precast Concrete Offers New Possibilites for Design of Shell Structures. In *Journal Proceedings*, Vol. 49. 537–548. (↑ p. 30)

Nvidia. 2013. Nvidia Physx Library. `http://www.nvidia.com/object/physx-9.12.0213-driver.html` (↑ p. 56)

John Ochsendorf. 2010. *Guastavino vaulting: the art of structural tile.* Princeton Architectural Press. (↑ p. 28)

John Allen Ochsendorf. 2002. *Collapse of masonry structures.* Ph.D. Dissertation. University of Cambridge. (↑ p. 25)

Travis E Oliphant. 2006. *A guide to NumPy*. Vol. 1. Trelgol Publishing USA. (↑ p. 150)

Toru Omata and Kazuyuki Nagata. 2000. Rigid body analysis of the indeterminate grasp force in power grasps. *IEEE Transactions on Robotics and Automation* 16, 1 (2000), 46–54. (↑ p. 62, 66)

Irving J Oppenheim. 1992. The masonry arch as a four-link mechanism under base motion. *Earthquake engineering & structural dynamics* 21, 11 (1992), 1005–1017. (↑ p. 97)

A Orduña and Paulo B Lourenço. 2005. Three-dimensional limit analysis of rigid blocks assemblages. Part I: Torsion failure on frictional interfaces and limit analysis formulation. *International Journal of Solids and Structures* 42, 18-19 (2005), 5140–5160. (↑ p. 26)

Robin Oval. 2019. *Topology Finding of Patterns for Structural Design*. Ph.D. Dissertation. Université Paris-Est. (↑ p. 129, 142)

Richard S Palmer. 1989. *Computational complexity of motion and stability of polygons*. Technical Report. Cornell University. (↑ p. 24)

J-S Pang and J Trinkle. 2000. Stability characterizations of rigid body contact problems with coulomb friction. *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik* 80, 10 (2000), 643–663. (↑ p. 24, 65)

Daniele Panozzo, Philippe Block, and Olga Sorkine-Hornung. 2013. Designing unreinforced masonry models. *ACM Transactions on Graphics (TOG)* 32, 4 (2013), 91. (↑ p. 20)

Konstantinos A Papadopoulos. 2021. Seismic stability assessment of an ancient dry stone defensive wall. *Bulletin of Earthquake Engineering* 19, 1 (2021), 463–482. (↑ p. 24)

Sergio Pellegrino. 1993. Structural computations with the singular value decomposition of the equilibrium matrix. *International Journal of Solids and Structures* 30, 21 (1993), 3025–3035. (↑ p. 66)

F Portioli, C Casapulla, and L Cascini. 2015. An efficient solution procedure for crushing failure in 3D limit analysis of masonry block structures with non-associative frictional joints. *International Journal of Solids and Structures* 69 (2015), 252–266. (↑ p. 26, 54)

F Portioli, C Casapulla, M Gilbert, and L Cascini. 2014. Limit analysis of 3D masonry block structures with non-associative frictional joints using cone programming. *Computers & Structures* 143 (2014), 108–121. (↑ p. 26, 69)

Francesco PA Portioli. 2020. Rigid block modelling of historic masonry structures using mathematical programming: a unified formulation for non-linear time history, static pushover and limit equilibrium analysis. *Bulletin of Earthquake Engineering* 18, 1 (2020), 211–239. (↑ p. 61)

Francesco PA Portioli, Michele Godio, Chiara Calderini, and Paulo B Lourenço. 2021. A variational rigid-block modeling approach to nonlinear elastic and kinematic analysis of failure mechanisms in historic masonry structures subjected to lateral loads. *Earthquake Engineering & Structural Dynamics* 50, 12 (2021), 3332–3354. (↑ p. 61)

Helmut Pottmann, Alexander Schiftner, Pengbo Bo, Heinz Schmiedhofer, Wenping Wang, Niccolo Baldassini, and Johannes Wallner. 2008. Freeform surfaces from single curved panels. *ACM Transactions on Graphics (TOG)* 27, 3 (2008), 1–10. (↑ p. 21)

Helmut Pottmann and Johannes Wallner. 2008. The focal geometry of circular and conical meshes. *Advances in Computational Mathematics* 29, 3 (2008), 249–268. (↑ p. 21)

Alfonso Ramírez Ponce and Rafael Ramírez Melendez. 2015. Curves of Clay: Mexican Brick Vaults and Domes. In *Architecture and Mathematics from Antiquity to the Future*. Springer, 309–324. (↑ p. 27)

Francesco Ranaudo, Tom Van Mele, and Philippe Block. 2021. A low-carbon, funicular concrete floor system: design and engineering of the HiLo floor. In *Proceedings of IABSE Congress 2021*. Ghent, 2016–2024. https://doi.org/10.2749/ghent.2021.2016 (↑ p. 15, 173)

William John Macquorn Rankine. 1872. *A manual of civil engineering*. Charles Griffin. (↑ p. 129)

Matthias Rippmann. 2016. *Funicular Shell Design: Geometric approaches to form finding and fabrication of discrete funicular structures*. Ph.D. Dissertation. ETH Zurich. (↑ p. 9, 20, 28, 29, 128, 129, 141, 174, 175)

Matthias Rippmann, Lorenz Lachauer, and Philippe Block. 2012. Interactive vault design. *International Journal of Space Structures* 27, 4 (2012), 219–230. (↑ p. 20, 174)

Matthias Rippmann, Andrew Liew, Tom Van Mele, and Philippe Block. 2018. Design, fabrication and testing of discrete 3D sand-printed floor prototypes. *Materials Today Communications* 15 (2018), 254–259. https://doi.org/10.1016/j.mtcomm.2018.03.005 (↑ p. 15, 173)

Matthias Rippmann, Tom Van Mele, Mariana Popescu, Edyta Augustynowicz, Tomás Méndez Echenagucia, Cristian Javier Calvo Barentin, Ursula Frick, and Philippe Block. 2016. The Armadillo Vault: Computational design and digital fabrication of a freeform stone shell. *Advances in architectural geometry 2016* (2016), 344–363. (↑ p. 10, 11, 12, 173)

Giovanni Romano. 2002. *Scienza delle costruzioni.* Hevelius. (↑ p. 49, 52)

M. Sanchez-Arcas. 1961. Form und Bauweise der Schalen. (↑ p. 30)

Vasilis Sarhosis, Katalin Bagi, José V Lemos, and Gabriele Milani. 2016. *Computational modeling of masonry structures using the discrete element method.* IGI Global. (↑ p. 25)

Vasilis Sarhosis, Daniel V Oliveira, José V Lemos, and Paulo B Lourenço. 2014. The effect of skew angle on the mechanical behaviour of masonry arches. *Mechanics Research Communications* 61 (2014), 53–59. (↑ p. 25)

Yuliy Schwartzburg and Mark Pauly. 2013. Fabrication-aware design with intersecting planar pieces. In *Computer Graphics Forum*, Vol. 32. Wiley Online Library, 317–326. (↑ p. 23)

Tianjia Shao, Aron Monszpart, Youyi Zheng, Bongjin Koo, Weiwei Xu, Kun Zhou, and Niloy J Mitra. 2014. Imagining the unseen: Stability-based cuboid arrangements for scene understanding. *ACM Transactions on Graphics* 33, 6 (2014). (↑ p. 26)

Hijung V Shin, Christopher F Porst, Etienne Vouga, John Ochsendorf, and Frédo Durand. 2016. Reconciling elastic and equilibrium methods for static analysis. *ACM Transactions on Graphics (TOG)* 35, 2 (2016), 1–16. (↑ p. 24, 26, 97, 98, 99, 185)

AG SOFiSTiK. 2012. SOFiSTiK analysis programs version 2012. *Oberschleißheim: SOFiSTiK AG* (2012). (↑ p. 96)

Peng Song, Bailin Deng, Ziqi Wang, Zhichao Dong, Wei Li, Chi-Wing Fu, and Ligang Liu. 2016. CofiFab: coarse-to-fine fabrication of large 3D objects. *ACM Transactions on Graphics (TOG)* 35, 4 (2016), 1–11. (↑ p. 23)

Daniel Sonntag, Simon Bechert, and Jan Knippers. 2017. Biomimetic timber shells made of bending-active segments. *International Journal of Space Structures* 32, 3-4 (2017), 149–159. (↑ p. 21, 22, 174)

David E Stewart and Jeffrey C Trinkle. 1996. An implicit time-stepping scheme for rigid body dynamics with inelastic collisions and coulomb friction. *Internat. J. Numer. Methods Engrg.* 39, 15 (1996), 2673–2691. (↑ p. 24)

Alan Song-Ching Tai. 2012. *Design for assembly: a computational approach to construct interlocking wooden frames.* Ph.D. Dissertation. Massachusetts Institute of Technology. (↑ p. 23, 30)

Romain Pierre Testuz, Yuliy Schwartzburg, and Mark Pauly. 2013. *Automatic generation of constructable brick sculptures.* Technical Report. (↑ p. 23)

Tom Van Mele, Juney Lee, and Philippe Block. 2022. Lightweight cable-net and fabric formwork system for the HiLo unit at NEST. In *FABRIC[ated]*, T. Stonorov (Ed.). Routledge. (↑ p. 9, 11)

Tom Van Mele, Andrew Liew, Tomas Mendez, Matthias Rippmann, et al. 2017-2021. COMPAS: A framework for computational research in architecture and structures. https://doi.org/10.5281/zenodo.2594510 http://compas.dev. (↑ p. 96, 149)

Guido vanRossum. 1995. Python reference manual. *Department of Computer Science [CS]* R 9525 (1995). (↑ p. 149)

Pauli Virtanen, Ralf Gommers, Travis E Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, et al. 2020. SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature methods* 17, 3 (2020), 261–272. (↑ p. 150)

Etienne Vouga, Mathias Höbinger, Johannes Wallner, and Helmut Pottmann. 2012. Design of self-supporting surfaces. *ACM Transactions on Graphics (TOG)* 31, 4 (2012), 1–11. (↑ p. 20)

Andreas Wächter and Lorenz T Biegler. 2006. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical programming* 106, 1 (2006), 25–57. (↑ p. 96, 150)

Hans Jakob Wagner, Martin Alvarez, Abel Groenewolt, and Achim Menges. 2020. Towards digital automation flexibility in large-scale timber construction: integrative robotic prefabrication and co-design of the BUGA Wood Pavilion. *Construction Robotics* 4, 3 (2020), 187–204. (↑ p. 102, 179)

Jingwen Wang, Wenjun Liu, Gene Ting-Chun Kao, Ioanna Mitropoulou, Francesco Ranaudo, Philippe Block, and Benjamin Dillenburger. 2023. Multi-robotic Assembly of Discrete Shell Structures. *Advances in Architectural Geometry 2023* (2023). (↑ p. 115, 142, 143, 180, 182, 183)

Ziqi Wang, Peng Song, Florin Isvoranu, and Mark Pauly. 2019. Design and structural optimization of topological interlocking assemblies. *ACM Transactions on Graphics (TOG)* 38, 6 (2019), 1–13. (↑ p. 26)

Ziqi Wang, Peng Song, and Mark Pauly. 2018. DESIA: A general framework for designing interlocking assemblies. *ACM Transactions on Graphics (TOG)* 37, 6 (2018), 1–14. (↑ p. 23, 30, 169)

Ziqi Wang, Peng Song, and Mark Pauly. 2021a. MOCCA: modeling and optimizing cone-joints for complex assemblies. *ACM Transactions on Graphics (TOG)* 40, 4 (2021), 1–14. (↑ p. 26, 169)

Ziqi Wang, Peng Song, and Mark Pauly. 2021b. State of the Art on Computational Design of Assemblies with Rigid Parts. In *Computer Graphics Forum*, Vol. 40. Wiley Online Library, 633–657. (↑ p. 26, 56)

C Weißenfels and P Wriggers. 2015. Methods to project plasticity models onto the contact surface applied to soil structure interactions. *Computers and Geotechnics* 65 (2015), 187–198. (↑ p. 61)

David Wendland. 2005. Vaults built without formwork: comparison of the description of a traditional technique in building manuals with the results of practical observations and experimental studies. (2005). (↑ p. 28)

David Wendland. 2007. Traditional vault construction without formwork: Masonry pattern and vault shape in the historical technical literature and in experimental studies. *International Journal of Architectural Heritage* 1, 4 (2007), 311–365. (↑ p. 28)

Emily Whiting, John Ochsendorf, and Frédo Durand. 2009. Procedural modeling of structurally-sound masonry buildings. In *ACM SIGGRAPH Asia 2009 papers*. 1–9. (↑ p. 26, 40, 41, 45, 46, 49, 52)

Emily Whiting, Hijung Shin, Robert Wang, John Ochsendorf, and Frédo Durand. 2012. Structural optimization of 3D masonry buildings. *ACM Transactions on Graphics (TOG)* 31, 6 (2012), 1–11. (↑ p. 26, 40, 41, 45, 46, 49, 56, 169)

Emily Jing Wei Whiting. 2012. *Design of structurally-sound masonry buildings using 3d static analysis*. Ph.D. Dissertation. Massachusetts Institute of Technology. (↑ p. 24)

Yong-Liang Yang, Yi-Jun Yang, Helmut Pottmann, and Niloy J Mitra. 2011. Shape space exploration of constrained meshes. *ACM Trans. Graph.* 30, 6 (2011), 124. (↑ p. 21)

Jiaxian Yao, Danny M Kaufman, Yotam Gingold, and Maneesh Agrawala. 2017. Interactive design and stability analysis of decorative joinery for furniture. *ACM Transactions on Graphics (TOG)* 36, 2 (2017), 1–16. (↑ p. 23, 34, 56, 60, 97, 98, 99, 101, 116, 174, 179, 185)

K Yokota and DR Brough. 1992. Assembly/disassembly sequence planning. *Assembly Automation* (1992). (↑ p. 30)