

CONFIDENTIAL

Methodology and Tools for Testing, Numerical Analysis and Design of the 3D Printed Moulds

Paweł Krzysztof Baran

Master of Science Thesis

CONFIDENTIAL

Methodology and Tools for Testing, Numerical Analysis and Design of the 3D Printed Moulds

MASTER OF SCIENCE THESIS

For the degree of Master of Science in Civil Engineering at Delft
University of Technology

Paweł Krzysztof Baran

October 4, 2016

Faculty of Civil Engineering and Geosciences · Delft University of Technology



The work in this thesis was supported by Tentech BV and DUS Architects. Their cooperation is hereby gratefully acknowledged.



Copyright © Delft University of Technology
All rights reserved.

DELFT UNIVERSITY OF TECHNOLOGY
FACULTY OF CIVIL ENGINEERING AND GEOSCIENCES
DEPARTMENT OF STRUCTURAL ENGINEERING

The undersigned hereby certify that they have read and recommend to the Faculty of
Civil Engineering and Geosciences for acceptance a thesis entitled
METHODOLOGY AND TOOLS FOR TESTING, NUMERICAL ANALYSIS AND DESIGN
OF THE 3D PRINTED MOULDS

by

PAWEŁ KRZYSZTOF BARAN

student number: 4417666

pawel.baran@mail.com

in partial fulfillment of the requirements for the degree of
MASTER OF SCIENCE IN CIVIL ENGINEERING

Dated: October 4, 2016

Supervisor(s):

Prof. ir. R. Nijse, TU Delft
r.nijse@tudelft.nl

Dr. ir. J.L. Coenders
TU Delft / White Lioness Technologies
jeroencoenders@white-lioness.com

Dr. ir. F.A. Veer, TU Delft
f.a.veer@tudelft.nl

Dr. ir. P.C.J. Hoogenboom, TU Delft
p.c.j.hoogenboom@tudelft.nl

Ir. Rogier Houtman, Tentech BV
rogier@tentech.nl

Preface


Today's world is changing in staggering pace. Rapid development of the information technology affects every aspect of our lives, while constant Internet connection becomes a virtual must. Lately, the revolution enters the architecture and engineering. Not only on conceptual level, by the means of freedom of expression or unprecedented computational power, but also with computer-controlled, digital fabrication methods. One of them, large-scale 3D printing, is researched in this graduation project. With an attempt to develop a framework for testing, analysis and structural design of moulds produced with this technology, I am making my humble contribution to the common good of science and engineering.

This document constitutes the final report of a Master's thesis research in Building Engineering at the Delft University of Technology. It has been created in collaboration with and under supervision of Tentech BV, an engineering practice being a partner in the project. Inspiring, innovation-driven environment created by these organizations shaped my mindset and encouraged me to explore new fields with faith and sense of purpose.

Leaving judgement of the content to the reader, I would like to thank all people and parties involved in the process. Firstly, the owners of Tentech, R. Houtman and H. Werkman, who put great trust in me, letting me find my own way through the project while providing me with almost infinite support. I pay highest respects to my daily supervisor, J. Coenders, as well as to other members of my Graduation Committee: R. Nijssen, F. Veer, P. Hoogenboom and mentioned R. Houtman – this work would not have been possible without them. The same applies to DUS Architects, owners of the fabrication facility and providers of all prints used in the research.

Sincere word of thanks goes to other persons actively involved in the project: M. van Leeuwen, for countless amounts of concrete cast in great atmosphere fuelled by his irresistible wit, and B. Gorte, for selfless, enthusiastic help with my weakest point, photogrammetry.

Finally, I would like to thank all people, who define me and my life. My Mother, whose strength and serenity is an inexhaustible source of inspiration, my family, whom I cannot see as often as I should, and friends, who help me not to forget what is truly important.



Utrecht, October 2016

Abstract

The large-scale 3D printing technology created by a group of partners led by DUS Architects allows fabricating components made of thermoplastic polymers with *Fused Deposition Modelling* technique. These components can be used as moulds for concrete elements, providing freedom of form and unprecedented optimization opportunities. In this report the author investigates general applicability of the concept and makes an attempt to establish the relationship between laboratory tests, computational modelling and the design of an end product. This is achieved by developing a streamlined framework for processing the experimental outcome and using it in finite element simulation to find mould geometry, which deforms into the desired shape after casting the infill.

The content of this report is contained in three factual sections. First of them covers preparation of the experiments and interpretation of their results. Main focus is put to mechanical response of the 3D printed cross sections in property tests (tensile, flexural, shear, creep, thermal sensitivity) and scale mould setups. Systematization of the output includes introduction of the print quality classes, which is a way to accommodate high variability in geometrical precision of the printer. Second part of the document describes and validates the proposed numerical approach using a triple-layer composite shell element with adjustable layer thicknesses. Implementation of the latter is driven by the inconsistency in cross-sectional stiffness parameters of an extruded wall in each direction. Next, the design method utilizing mentioned modelling technique is proposed. It is based on the principle of applying initial negative deformation, which compensates the excessive deflections that occur after casting the concrete. The above solutions are implemented with Python programming language and wrapped together in Rhinoceros 5.0 software with Grasshopper plugin. Finally, applicability of the framework is validated by the case study experiments on predeformed moulds.

The discussion and conclusions argue that the use of the 3D printed components as forms for small concrete elements is feasible, but it is likely to exhibit issues with scalability. The test scheme is considered sufficient yet the print quality issues are highlighted. Likewise, numerical modelling technique and design method are acknowledged along with their drawbacks being mentioned. In the end, a list of potential improvements is given, including extension of the test scheme, development of a custom element formulation, changes in software implementation or alteration of the design procedure to enable its application to other problems.

Table of Contents

1	Introduction	1
1.1	Problem definition	2
1.2	Scope of the research	2
1.3	Research method	3
2	Large scale 3D printing	5
2.1	3D printing in architecture	6
2.2	3D Print Canal House	7
3	Material Introduction	11
3.1	Polymers – overview	11
3.2	Mechanical properties of thermoplastics	13
3.3	Investigated material	15
4	Test Description	19
4.1	Coordinate systems	19
4.2	Selection of the properties	20
4.3	Time dependency	20
4.4	Material tests	21
4.5	3D printed mould tests	27
4.6	Concrete property tests	29
5	Test Results	31
5.1	Material tests	31
5.2	3D printed mould tests	53
5.3	Concrete property tests	59
5.4	Summary	62

6 Numerical analysis	63
6.1 Modelling of solid polymers - overview	63
6.2 Modelling of 3D printed cross sections	63
6.3 Implementation	69
7 Validation of the numerical model	79
7.1 Tensile test	80
7.2 Flexural test	81
7.3 Single layer moulds	83
7.4 Truss-like walled mould	83
7.5 Summary	84
8 Design methodology	87
8.1 Design criteria	88
8.2 Design procedure	90
8.3 Software implementation	93
8.4 Summary	96
9 Case study	99
9.1 Evaluation method	99
9.2 Prismatic blocks	99
9.3 Parametric column	100
9.4 Wall panel	100
9.5 Summary	101
10 Discussion	107
10.1 Testing and mechanical properties	107
10.2 Numerical analysis	108
10.3 Design and implementation	109
11 Conclusions	111
11.1 Testing and mechanical properties	111
11.2 Numerical analysis	112
11.3 Design and implementation	112
12 Recommendations	113
12.1 Testing and mechanical properties	113
12.2 Numerical analysis	113
12.3 Design and implementation	114
12.4 Other aspects	114
A Mechanical properties of thermoplastics - overview	115

B	Raw material test output	117
C	Multiplication factors for cross section reclassification	139
D	Effective length of the tensile specimen	143
E	Statistical foundations for derivation of the material characteristics	145
F	Single layer mould test setup	147
G	Standard concrete mixture specification	149
H	Self-compacting concrete mixture specification	151
I	Photogrammetric method for measuring surfaces	155
J	Determination of layer thicknesses t_1, t_2 for modelling of element's stiffness	159
K	Source code: generator of global input model	161
L	Source code: stress-strain data generator	169
M	Source code: predeformation component	173
N	Source code: post-processing	183
	Bibliography	189

“The machines are to practically everybody what the white men were to the Indians. People are finding that, because of the way the machines are changing the world, more and more of their old values don’t apply any more. People have no choice but to become second-rate machines themselves, or wards of the machines.”

—*Kurt Vonnegut, Player Piano*

Chapter 1

Introduction

Construction industry has historically been linked to manual labour and human-controlled machinery. Driven by safety and reliability, it was always resistant to major technological changes. A chance for radical shift in paradigm appeared when the computers became available several decades ago. Experiments with digital fabrication, initially unsuccessful, grew in popularity as the robotics advanced, leading to a wide variety of techniques being developed nowadays (Gramazio et al., 2014; McGee and de Leon, 2014). One of them, 3D printing, has potential to be introduced into the market in relatively near future, providing numerous possibilities for customization, optimization and elimination of the human error. Being equipped with a large scale printer capable of extruding polymers, *Kamermaker*, 3D Print Canal House is one of the most remarkable undertakings related to this technology (DUS Architects, 2014). Originally starting with the aim to actually fabricate the whole building, in time it shifted to production of single components and moulds. Structural aspects of the latter are the subject of this report, with particular focus on material testing and translation of the experimental data into a prediction model, followed by development of a design method.



Figure 1-1: 3D Print Canal House. Left: 3D printed member. Right: Printing process.

1.1 Problem definition

Until the start of this research, several attempts had been made to systematize the knowledge about characteristics of the material that is used by *Kamermaker*. They included mainly laboratory tests and hand calculations (van der Veen, 2014; Peulen, 2015). Despite relatively large amount of experiments, the information collected during previous works lacked coherence due to selective test schemes combined with evolution of the material formulation over time. Consequently, only qualitative conclusions could be made with enough confidence. Having gathered general knowledge, it has been decided to make an attempt to set a framework describing mechanical behaviour of the material with quantitative values as well as to define a practical way to use the data for design purposes. These tasks were assigned to the author as a research objective, which can be defined by three core questions:

1. *What are the engineering properties of the extruded cross-sections and how to test them?*
2. *How to model mechanical behaviour of the 3D printed moulds based on the experimental results?*
3. *How to design the moulds with the use of numerical analysis?*

The answers to these questions were provided in the form of an experimentally validated, software-enhanced framework for interpretation of the results and performing the analysis, supplemented by a set of design recommendations and a design module. Starting from material property investigation, through computational activities up to large-scale tests an endeavour was made to create a coherent methodology with potential to be used in practice.

1.2 Scope of the research

Structural behaviour of thermoplastic polymers is complex and depends on numerous parameters. Taking into account logistical constraints and practical objective of the project, it has been decided to focus on chosen, most relevant aspects of possible research in this field. Range of activities declared in Section 1.1 has been adjusted to the scope of final product and available resources. Following framework has been applied:

- Material investigation has been limited to relevant mechanical properties related to cross sectional strength and stiffness, generalized creep behaviour and thermal sensitivity. Only uniaxial tests have been performed because of difficulties associated with building biaxial setups.
- Formwork experiments were held in controlled climate and involved relatively small concrete volumes, which led to practical negligence of the temperature influence.
- Numerical tools were developed by the means of writing custom algorithms that link and utilize third party software. Provided functionalities include translation of experimental force-displacement data into cross-sectional properties of the 3D printed walls, numerical analysis of the members based on these properties and prediction of predeformation needed to achieve the desired shape after casting.

The second point is particularly important from the engineering point of view. It clearly defines the scope of application, limiting it to constant temperature environment and low heat of hydration. However, the proposed method is meant to be extendable and adjustable. For instance, adaptation to other (relatively mild) climate conditions is expected to be only a matter of further testing and validation. Also codewise, space is left for development: algorithms are structured in a way that allows for application of new functionalities and modules. All scripting is done in Python as it is one of the most popular programming languages in today's scientific community (Reitz and Schlusser, 2016).

1.3 Research method

The project has been divided into three main stages:

Stage 1 – Introduction to the material and technology, laboratory tests: chapters 2-5.

Stage 2 – Numerical modelling and validation: chapters 6-7.

Stage 3 – Design methodology and case study: chapters 8-9.

Stage 1 starts from the summary of existing knowledge about the material characteristics and fabrication technique. Based on the collected information a test scheme is defined for both material property tests as well as for the experiments involving 3D printed moulds and concrete, with geometry of the former complying with restraints concerning wall span/stiffness relationship explained in Section 8.2.2. Test results are presented and evaluated with regard to their correctness and practical usefulness.

Next, data gathered in the first stage is translated into a numerical model. This is done by converting the force-displacement experimental curves into stress-strain relationships, which are then utilized to define the stiffness of each individual, triple-layer shell element. The modelling approach developed for the purpose of this project is then validated by comparing the FE analysis output with laboratory measurements.

The last stage listed in this section is focused on design aspects. Firstly, the observations made during testing and modelling are discussed with particular focus on strength, stiffness and practical issues. Later, a computational design tool is introduced based on the findings. The tool allows for prediction of mould's behaviour under the loads as well as for determination of the geometry that needs to be printed in order to achieve the desired shape after casting the concrete and consequent deflection of the walls.

Finally, the last three chapters of this document contain discussion, conclusions and recommendations. These, besides covering the scope of the research, go a step further by highlighting possible directions that can be taken by future developers of the technology.

As can be noticed, literature review has not been defined as a separate part of the project. Instead, relevant literature is studied in the beginning of each chapter. The reason for such structuring is wide spectrum of problems covered by this report. Referring the literature directly to respective subject allows reading certain parts of the document independently, which makes it more intelligible.

Flowchart showing the research process presented in this section is shown in Figure 1-2.

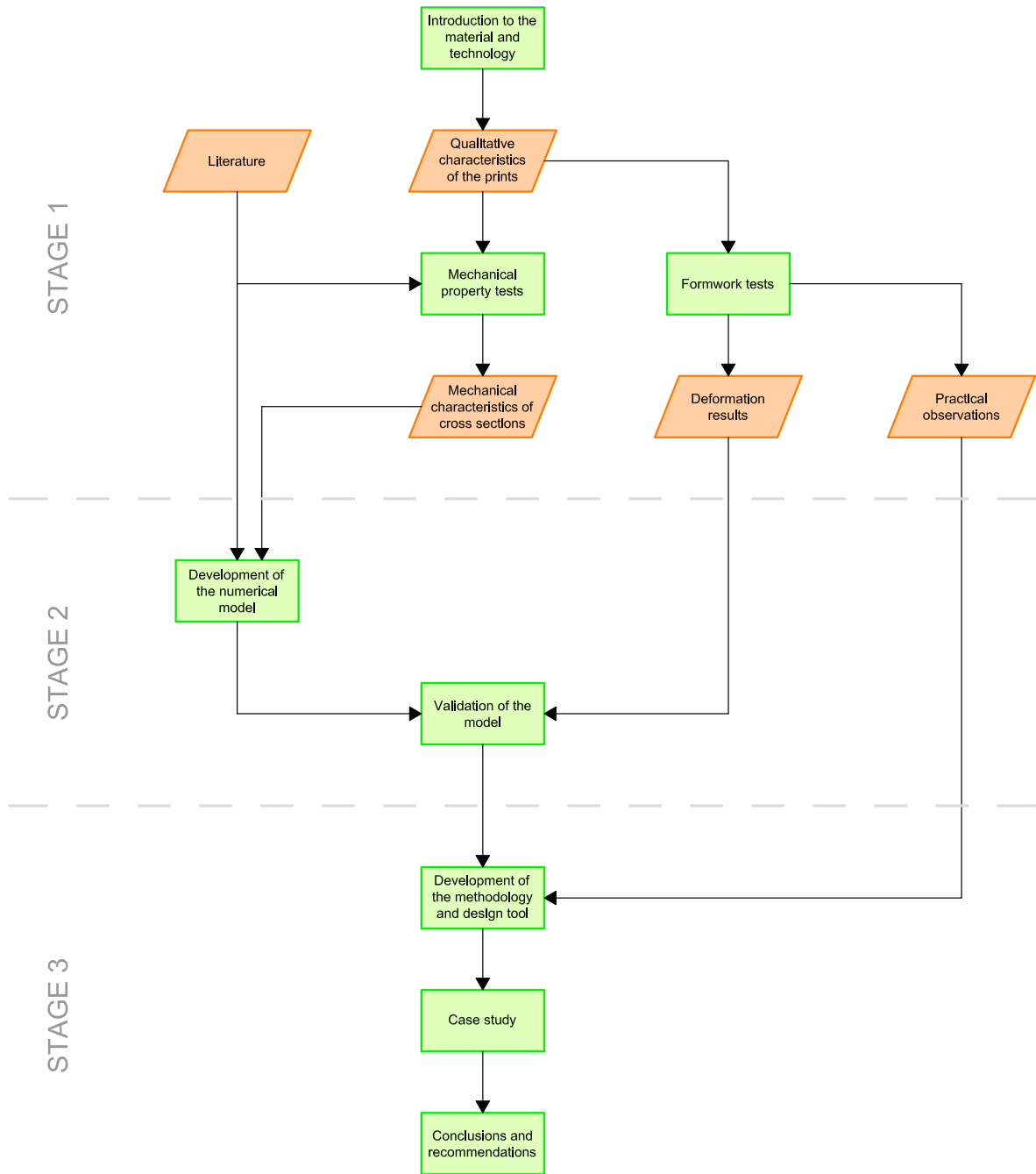


Figure 1-2: Research method.

Chapter 2

Large scale 3D printing

Research on additive manufacturing started in 1970's with initial focus on stereolithography - a process in which chosen parts of polymeric matter are solidified with light treatment. This method was commercialized in 1986 (Hull, 1986; Bártolo and Gibson, 2011). Extrusion-based solutions independent of radiation were introduced in the next decade, with first FDM (Fusion Deposition Modelling) device announced in 1992 (Chua et al., 2003). Since then the industry evolved in scope and size, today offering a wide range of materials and devices. However, despite the growth of both consumer and industrial market, the technology itself hardly rose in scale and did not find its way into construction until recently. Situation changed in last years with the appearance of various undertakings attempting to develop 3D printers capable of fabricating architectural components. This chapter gives an overview of the existing market with particular focus on the master project of this research, 3D Print Canal House.



Figure 2-1: Buildings reported to be printed by WinSun. Left: multi-storey apartment building in Suzhou (Sevenson, 2015). Right: office building in Dubai (DiStasio, 2016).

2.1 3D printing in architecture

Nowadays, large scale 3D printing is being realized with different types of materials used as building matter. The most popular one is concrete and its equivalents: the number of organizations running research on cement-based substances is constantly growing, with some of them only starting and others already having developed products. Highest publicity so far has been gained by the Chinese company WinSun, which claims to have built multiple houses of various size and architecture with the use of extruded concrete (Sevenson, 2015). The reported success resulted in a collaboration with other parties, which led to fabrication of a 3D printed office building in Dubai (DiStasio, 2016). Other pioneers of using extruded mixture of aggregate and binder are Enrico Dini, who started developing his technology in 2007 (D-Shape, 2014) and Andrey Rudenko, the maker of a 3D printed concrete castle (Rudenko, 2014). Many others follow this trend.



Figure 2-2: D-Shape binder-based printer (D-Shape, 2014).

However, there is a common feature linking the ventures mentioned above: they are privately held and do not publicly share all their knowledge. Among the academic institutions involved in the research on large scale additive manufacturing with concrete only a few managed to reach beyond preliminary tests or speculative publications. Quantitative results are provided e.g. by Malaeb et al. (2015), Gosselin et al. (2016), Perrot et al. (2016) or Le et al. (2012). There is also an ongoing attempt to introduce second of the most popular modern construction materials, steel. An Amsterdam-based company, MX3D, is currently working on a bridge manufactured with the use of an industrial robot equipped with a welding tool (MX3D, 2015). Similarly to commercial solutions in concrete, there is no existing scientific coverage of this project. Arup, a global engineering consultancy, approached the problem from different angle by using additive manufacturing technique to produce an individual, complex steel node, which has been documented in a conference paper by Galjaard et al. (2015). Besides concrete and steel, experiments are performed with alternative materials, such as clay-based mixtures and polymeric filaments. On academic level, the former group is investigated by the Institute for Advanced Architecture of Catalonia with the project named Pylos (Institute for Advanced Architecture of Catalonia, 2015), while in the private sector an attempt is being made to fabricate a sustainable village made of soil-straw blend (World's Advanced Saving Project, 2016).

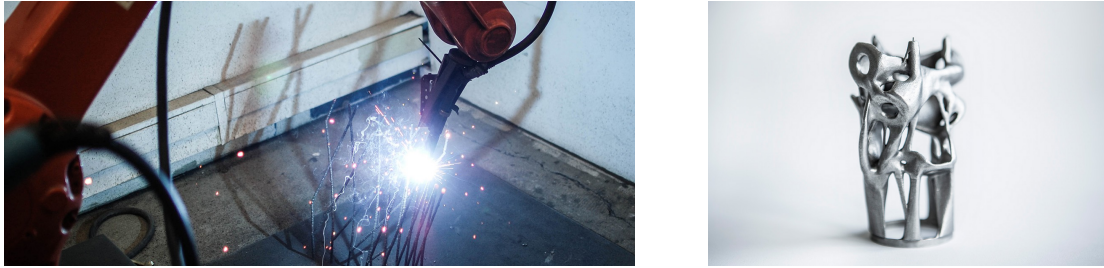


Figure 2-3: 3D printing with steel. Left: printing arm, MX3D (MX3D, 2015). Right: 3D printed node, Arup (Arup, 2015).

Thermoplastic polymers, as a wide group, are used in most consumer 3D print applications. With extensive knowledge openly available, a number of parties is trying to upscale the technology and introduce it to the architecture. Expertise in this field has been gathered both in the industry and academia led by i.a. Gramazio Kohler research group at the ETH Zurich and Oak Ridge National Laboratory, who are working on Mesh-Mould technology (Hack and Lauer, 2014) and AMIE project (Oak Ridge National Laboratory, 2015), respectively. The team coordinated by DUS Architects, founders of the 3D Print Canal House, can also be considered as one of the key players in this category.



Figure 2-4: Mesh-Mould technology, ETH Zurich (Hack and Lauer, 2014).

2.2 3D Print Canal House

History of the project starts in 2012, when DUS Architects and a 3D printer manufacturer, Ultimaker, established cooperation with an objective to build a device capable of fabricating components in building scale. Soon after the kick off, in 2013, a core group of participants was formed, including Tentech, an engineering consultancy responsible for structural aspects of the technology and employer of the author. Starting with an initial goal to build a modular house room after room, the concept gradually shifted towards production of single, custom-made elements. Currently, the main field of interest within the team is a combination of the extruded polymer with concrete. One of the most promising applications of this type is production of custom-shaped moulds, which is the subject of this report.

2.2.1 Technology

The 3D printing technology developed by DUS and Ultimaker utilizes continuous extrusion of a fiber-reinforced bio-based thermoplastic compound, which is heated up to temperature around 180°C prior to processing. The molten material is placed in layers in global XY plane, based on the path defined by a CAD/CAM input file. While finishing each layer, the movable frame equipped with a nozzle and processing machinery moves up along the Z axis compensating the growth in component's height. With maximum dimensions of single element equal to $2.2 \times 2.2 \times 3.5$ m and maximum usable speed of around 10000 mm/min, such method can be considered as a large-scale application of Fused Deposition Modeling (FDM) technique.

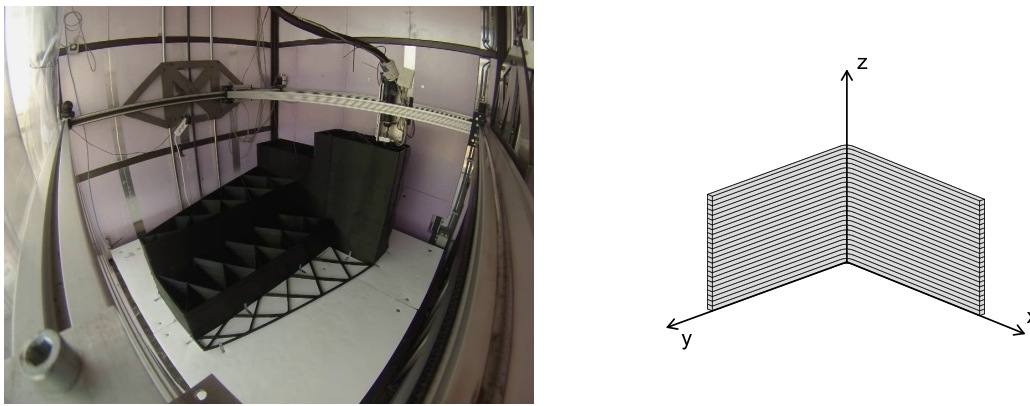


Figure 2-5: Kamermaker 3D printer. Left: device during operation. Right: Global coordinate system.

Although being relatively simple and robust, it bears numerous limitations, among which the most serious are:

- inability to print horizontally above the zero level
- geometry of the component strictly limited by the printer size
- anisotropic, potentially irregular structure of the product
- high thermal sensitivity and viscoelasticity of the extruded material

Taken these drawbacks into account (with particular emphasis on long-term behaviour and temperature issues), focusing on non-structural systems such as formwork is justifiable. Originally, the moulding technique was introduced by DUS in 2014, but the only tests done until the start of this research project involved arbitrary castings supported by no calculations or measurements (Figure 2-6, left). Despite the lack of systematized knowledge, one spectacular application of the 3D printed polymer combined with concrete has already been shown in the public space: it was used as topping for the bench seats installed at the Europe Building in Amsterdam (Figure 2-6, right).

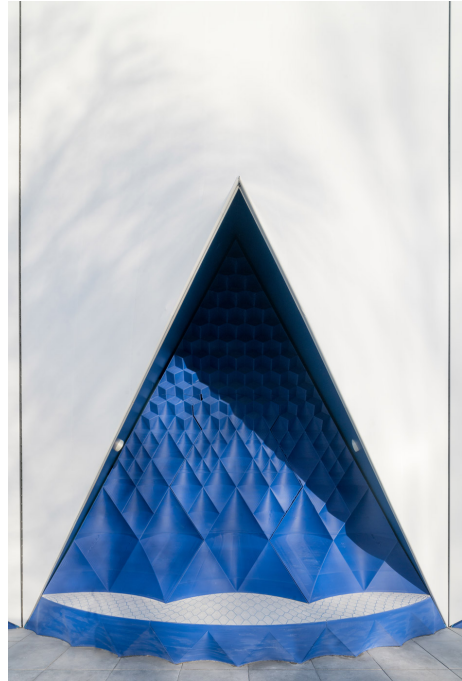
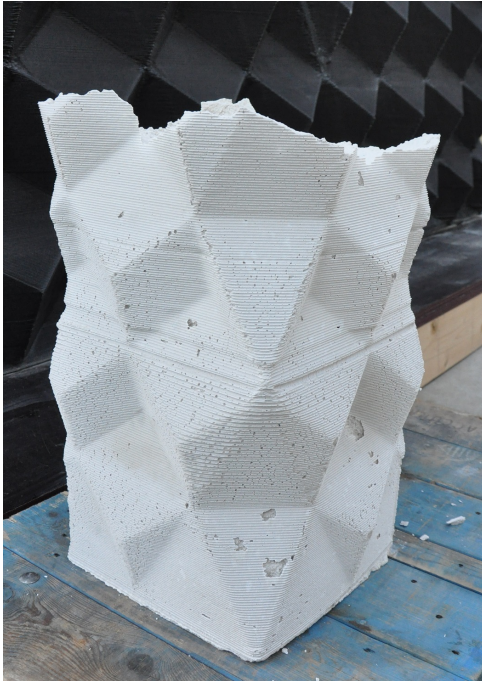


Figure 2-6: 3D Print Canal House: application of concrete. Left: casting test on site. Right: Bench in the Europe Building in Amsterdam (DUS Architects, 2016).

Chapter 3

Material Introduction

Understanding of material typology, formulation and characteristics is core of any research project in the field of structural and mechanical engineering. Chapter 3 contains an overview of mentioned aspects, starting from describing polymers as a general group and narrowing down to the particular compound used as a print material for the 3D Print Canal House. Facing lack of detailed knowledge about its chemical structure after extrusion, the description is mainly qualitative, with only few value estimates. The information presented in Section 3.3 is a starting point, based on which the test scheme has been designed and prepared.

3.1 Polymers – overview

Polymers are a broad group of substances that are composed of large molecules consisting of multiple repetitive subunits (from Greek *poly-*, "many" + *-meros*, "parts", American Heritage Dictionary, 2011). With their complex structure and very high molecular masses they play a fundamental role in both nature and human-made part of the world. They can be found in various forms and applications, starting from the organic ones, for example as floral and body tissues (cellulose, proteins, DNA), through half-synthetic, chemically modified natural fabrics such as leather or cellophane, up to a wide range of fully synthetic products.

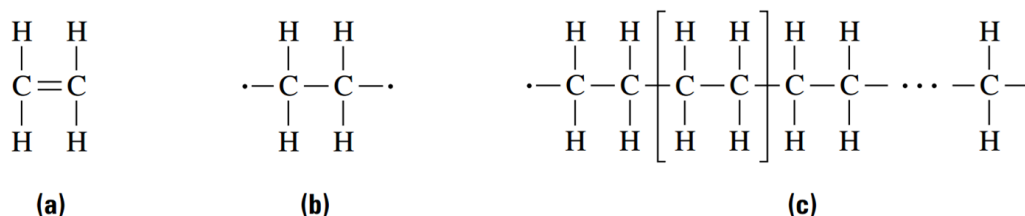


Figure 3-1: The structure of a simple polymer, polyethylene (PE). **(a)** single monomer of C_2H_4 **(b)** "opening" of the double bond in a monomer **(c)** many monomers linked together to form a repetitive polymer chain. Schaffer et al. (1999).

The last group, synthetic polymers, consists of compounds usually based on components that originate from fossil fuels: coal, oil and natural gas. Their different variations are used in numerous industries such as construction, automotive, electrics, toys, consumer goods or aerospace. Most common materials are:

- high/low density polyethylene (HDPE/LDPE)
- polypropylene (PP)
- polystyrene (PS)
- polyvinylchloride (PVC)
- polytetrafluoroethylene (PTFE, brand name: Teflon)
- polyimide (PI, brand name: Nylon)
- thermoplastic polyurethanes (TPU)
- thermosetting polyurethanes (PU)
- epoxy resins (EP)
- synthetic rubbers

The substances above (as well as many others) exist in multiple configurations that differ in composition and processing.

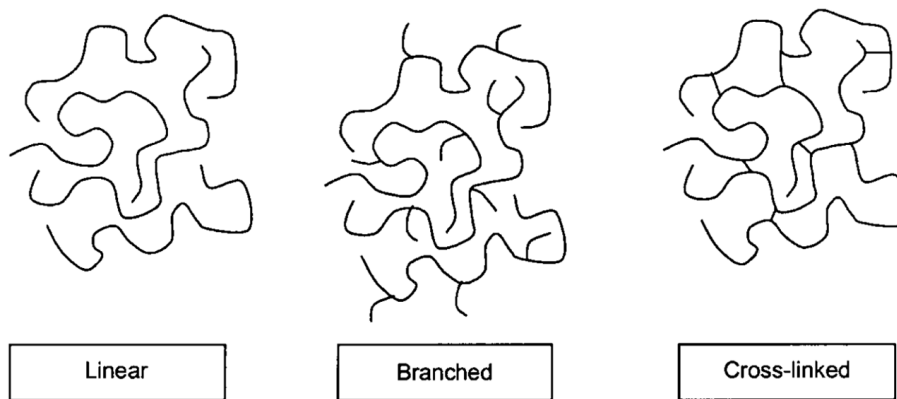


Figure 3-2: Linear, branched and crosslinked polymer (Harper, 2002).

Polymers are created in the process of synthesis of small molecules that is called polymerization. Depending on chemical reactions that take place, the resulting product can consist of either separate linear macromolecules (chains) or a chemically bonded network. In the former case, the chains have a form of long, sometimes branched coils that are entangled with each other. The chains are bonded together by weak interaction forces, which provide the system with relatively low stiffness. On the other hand, due to the entanglement of the coils, the material shows high resistance to flow (it has high viscosity). Properties of chemically bonded networks are significantly different. As the chains are joined together, they in fact create a single body. Thanks to strong links between the molecules, stiffness of the system is high and flow is generally not possible.

In general, polymers can be divided into three main groups with regard to their chemical structure:

Thermoplastics – non-crosslinked systems that change their mechanical properties along the temperature (glass-solid while cooled, they start to flow while heated up).

Synthetic elastomers – initially showing flow, they transform into solid, highly elastic materials after formation of a loose intermolecular network (e.g. vulcanized rubbers).

Thermosets – relatively stiff solid systems that are products of a reaction, during which a tight network is created at (in most cases) elevated temperature. Once cured, the material does not change its mechanical properties while heated.

Most synthetic and half-synthetic polymers are mixed with each other as well as with a variety of fillers and additives. Technical name for such group of materials is composite plastics or just plastics, which brings necessary clarification to the common understanding of that word. Blending polymers together or combining them with other substances can be done for various reasons. Most popular of them are processing (e.g. lubricants, antioxydants, vulcanization, acceleration of the reactions) and widely understood change of properties: reinforcement, plasticizers, antistatic agents, flame retardants etc. (van der Vegt, 2006).

Since the material investigated in this report is a composite based on a thermoplastic, further information presented in this chapter is mostly limited to this certain group of polymers.

3.2 Mechanical properties of thermoplastics

From the mechanical point of view, thermoplastics are a specific group of materials that exhibit a variety of properties depending on their chemical structure, temperature, age or external conditions. Behaviour of a certain compound may differ from a brittle solid, through elastic rubber up to a viscous liquid. In each state it will show entirely different features, many of them nonlinear and nonexistent in other states. What is more, thermoplastics are usually highly viscoelastic in the temperature of operation. Since complex explanation of all these phenomena cannot be contained in several pages, only the principles are described in this chapter, while curious reader is referred to the literature (Flory, 1953; Harper, 2002; van der Vegt, 2006; Ward and Sweeney, 2013).

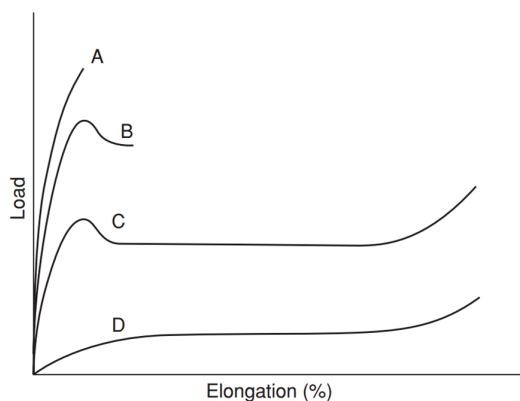
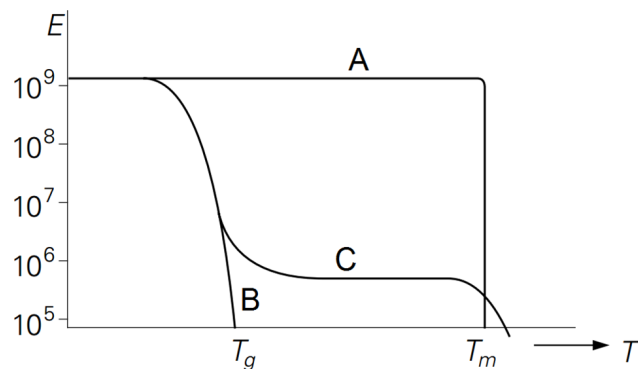


Figure 3-3: Load–elongation curves of a semi-crystalline thermoplastic at different temperatures. Curve A: brittle fracture (T below T_g). Curve B: ductile failure (T below, but close to T_g). Curve C: cold drawing (T around T_g). Curve D: rubber-like behaviour (T above T_g). Ward and Sweeney (2013).

As already mentioned, each of the thermoplastics and their mixes can exist in three different states, ranging from brittle glass-solid when cool, through rubbery at intermediate temperatures up to liquid after melting. The phase change occurs at two thresholds: glass-rubber transition temperature T_g and melting temperature T_m . Usually materials have only one of them, either the former (amorphous ones, e.g. glass) or the latter (crystalline compounds). Existence of both of the values is a unique feature of polymers, which is possible due to their high molecular mass and dual nature consisting of both amorphous and crystalline components. A practical consequence of this phenomenon is shown in Figure 3-4.

Figure 3-4: Temperature/stiffness relationship of different types of materials. Curve A: perfect crystalline material. Curve B: amorphous non-polymer (low-molecular matter). Curve C: amorphous polymer (length of the plateau depending on length of the molecular chains). van der Vegt (2006).



What is interesting, even in a rare case when the polymer is fully amorphous, it does not turn into liquid directly after reaching T_g , but retains a rubbery, highly elastic state for some time. This refers to the high chain length and the energy that is needed to untangle the complex molecules even after breaking all the interaction forces between them. The exact values of T_g and T_m for each material may differ depending on the cooling rate as well as other conditions such as the external environment or thermal history and molecular structure of a certain object.

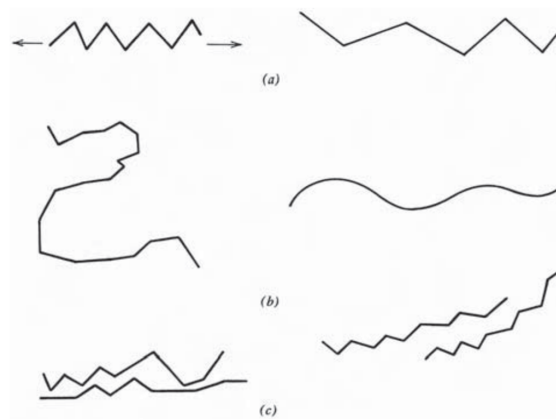


Figure 3-5: Polymer extension modes. a) bond bending; b) uncoiling; c) slippage. Shah (2007).

Another complex property of thermoplastics is their viscoelasticity. Looking at Figure 3-4 one can notice that stiffness of the polymer is stable below the glass transition temperature. This derives from the fact that in this range the sliding motion between the macromolecules is fully restrained, so the only possible deformation refers to bending and stretching of the interatomic

bonds between their atoms. This type of movement is fully reversible and almost temperature independent. After reaching T_g a highly temperature dependent, viscous sliding is activated. It consists of two modes: reversible and irreversible one (primary and secondary creep, caused by uncoiling of the molecular chains and by actual slippage, respectively). Consequently, at temperatures above the glass transition point there are three different components of deformation working simultaneously, which results in viscoelastic behaviour. Figure 3-5 explains the mechanics of each of the components, while resultant stress-strain relationship is presented on graphs in the Figure 3-6 (Shah, 2007).

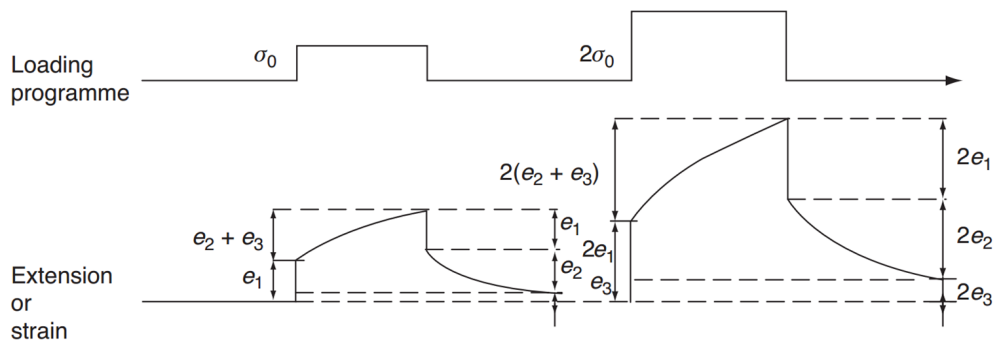


Figure 3-6: Elastic response of a viscoelastic body. e_1 , reversible deformation due to breaking the weak interaction forces and uncoiling the chains; e_2 , reversible deformation due to viscous sliding; e_3 , irreversible deformation due to viscous sliding (permanent set). Ward and Sweeney (2013).

In addition to creep, constant loading of thermoplastics induces stress relaxation – the phenomenon in which the body subjected to constant strain releases part of its stress along time. This, in total, leads to high complexity of the material model. Possible approaches to modelling of such behaviour are introduced in Section 6.1.

It is hard to clearly define the stiffness and strength of thermoplastics as a group of substances. As already explained, the values for each compound rely strongly on several factors, and the properties of different materials might differ from each other by tens or even hundreds of times. In general, however, thermoplastics need to be regarded as much weaker and more elastic than the regular materials used in the construction industry: steel, concrete or timber. A brief overview of strength and stiffness of thermoplastics is presented in Appendix A.

3.3 Investigated material

Material used in the 3D Print Canal House project is a compound based on a thermoplastic adhesive. Although its full recipe is confidential, the core constituents are presented in Table 3-1. In general, the mixture can be considered as a reinforced plastic composite with a flexible thermoplastic matrix and very stiff and strong glass fibers.

Figure 3-7 shows the distribution of fibers within a broken cross-section. As can be seen, they are positioned almost uniaxially along the print direction, which means that the longitudinal and transverse properties differ significantly. Across the print, the material behaviour resembles pure matrix, while along it presents much higher strength and stiffness.

The exact values depend on the length, amount and distribution of the fibers as well as on

the adhesion between matrix and reinforcement. Derivation of extruded polymer's mechanical characteristics based on its chemical composition and microstructure requires extensive effort and knowledge, while it still yields relatively high risk of providing numbers that are inapplicable on macro scale. Instead, a mechanical test scheme has been designed and implemented in order to determine the properties that are relevant to the purpose of this project. However, not all parameters could be measured in the laboratory. The missing ones are estimated: Poisson's ratio of the matrix is taken directly from specification of Macromelt 6900E, while the analogue value in the fiber direction is calculated using an estimate method named Halpin-Tsai semi-empirical equation (Fu et al., 2009).

Name	Fraction (weight)	Description
Macromelt 6900E	>60%	<p>Thermoplastic hot melt adhesive based on polyamide, with high viscosity, low modulus and good flexibility. Properties according to <i>Technical information: MACROMELT 6900E</i> (2009):</p> <ul style="list-style-type: none"> • glass transition temperature: 5°C • softening point: 130-145°C • tensile strength: 30 MPa • tensile yield: 12 MPa • max. elongation: 600% • Young's Modulus: 140 MPa • shear modulus: 49 MPa • Poisson's ratio: 0.43 • density: 1.0 g/cm²
Ethylene-vinyl acetate	N/A	<p>Coupling agent, copolymer of ethylene and vinyl acetate, an elastomeric polymer with cross-linked structure. Used in hot melt mixtures, provides additional intermolecular bonding that enhances melt strength while still enabling low melt processing temperatures (Harper, 2002).</p>
Glass fibers	12%	<p>Short E-glass fibers oriented unidirectionally along the print line, significantly improve strength and stiffness in that direction:</p> <ul style="list-style-type: none"> • Young's Modulus: 70 Gpa • Shear Modulus: 30 GPa • Poisson's ratio: 0.22 • density 2.5 g/cm²

Table 3-1: Composition of the material used in the project.

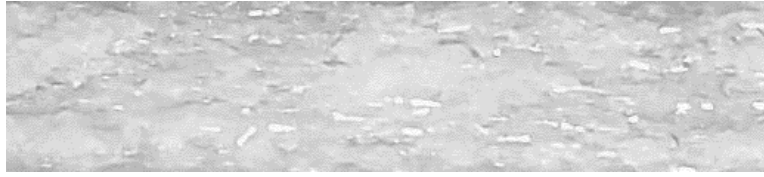


Figure 3-7: Fiber distribution within the cross-section.

The method uses following expression to directly calculate the missing value:

$$\nu_{12} = \nu_f v_f + \nu_m v_m \quad (3.1)$$

Where:

- ν_f - Poisson's ratio of the fibers
- ν_m - Poisson's ratio of the matrix
- $v_f = \frac{M_f/\rho_f}{M_f/\rho_f + M_m/\rho_m}$ - volumetric fraction of the fibers
- $v_m = 1 - v_f$ - volumetric fraction of the matrix

Having derived the above values from Table 3-1, this yields:

$$\nu_{12} = 0.42$$

Halpin-Tsai equations 3.2 and 3.3 are used to determine the shear modulus in fiber direction based on known shear modulus of the matrix (G_m , calculated from transverse tensile test results based on Hooke's isotropic formulation) and fibers (G_f):

$$G_{12} = \frac{1 + \eta_G v_f}{1 - \eta_G v_f} G_m \quad (3.2)$$

Where:

$$\eta_G = \frac{G_f/G_m - 1}{G_f/G_m + 1} \quad (3.3)$$

Equations 3.1 to 3.3 combined with the test output presented in Chapter 5 enable definition of the stiffness matrix of a 3D printed cross section (see Chapter 6).

As already mentioned, the material, with changing form and composition, has been investigated by the architects from DUS and engineers from Tentech since 2013 (van der Veen, 2014; van Baarsen et al., 2015; Peulen, 2015). Consequently, it was possible to compose a qualitative description of mechanical behaviour of the prints before the actual tests had started.

The most important issue is the print quality. Depending on the material type, external environment, printer settings and geometry of the component, wall cross section may differ significantly within a single piece (Figure 3-8). For the scope and conditions applicable to this graduation project, the quality was believed to be partially controlled by limiting the

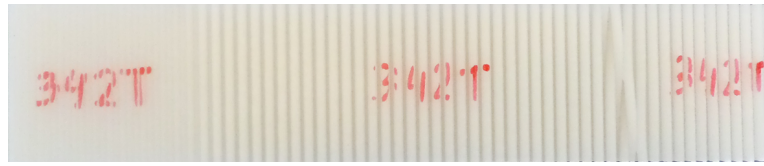


Figure 3-8: Varying print quality within a single element.

maximum wall span – with the length of 300 mm the walls were expected to remain stable while applying new layers on top of them (Figure 3-9).

Other known material characteristics can be summarized in following points (under the assumption of being used in the room temperature):

- The material is a highly temperature-dependent soft thermoplastic in its intermediate state ($T_g < T < T_m$).
- It is highly viscoelastic and can withstand large strains.
- Tensile, compressive and shear strength is expected to be in range 5-20 MPa.
- Material properties change dynamically in the first days after printing, they are expected to stabilize after ca. 14 days.
- The material degrades steadily while exposed to sun radiation and temperature amplitudes.
- Interface between the printed layers needs to be investigated carefully as in the past it proved to be the least reliable element of the system.

The knowledge presented in this section has been taken into account on planning stage of the research project. It strongly helped to identify both opportunities as well as limitations of the developed technology. In following chapters it can be found how accurate were the above assumptions and how did they affect the final product.

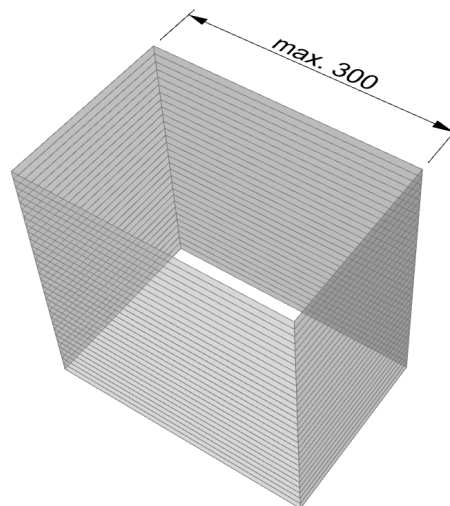


Figure 3-9: Expected maximum acceptable print span between the turns of the nozzle.

Test Description

The tests constitute a significant part of the project covered in this document. They are especially important facing the lack of literature or other resources regarding the mechanical behaviour of the 3D printed moulds and 3D printed structures in general. So far only the manufacturing and practical aspects of the technique have been documented, e.g. by Gardiner and Janssen (2014); Keating et al. (2014); Peters (2015). What is more, even mentioned sources do not relate directly to the 3D Print Canal House project as they concern technologies that either utilize other materials or do not even specify them. Therefore, a custom test scheme has been designed in order to evaluate the applicability of the 3D printed moulds as concrete formwork. Exploration of an entirely new field induced several limitations that needed to be embraced while designing the test scheme. Proposed framework is presented in the following sections.

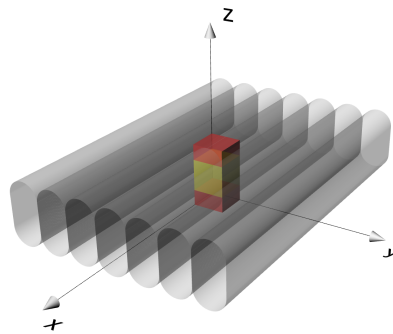


Figure 4-1: Local coordinate system of a triple-layer element.

4.1 Coordinate systems

Distinction needs to be made between the global coordinates of the printer, shown in Figure 2-5 and local coordinate system used to characterize a 3D printed wall. The latter is defined by local x and z axes of the wall, which coincide with the print direction and wall's normal, respectively (Figure 4-1). This general rule refers to all calculations presented in this report.

4.2 Selection of the properties

Prior to studying the behaviour of complex systems, a new material needs to be defined in small scale. Consequently, two independent test schemes have been designed and executed: one for investigation of the material itself and the other for checking quantitative and qualitative response of the 3D printed moulds filled with concrete:

Material tests – laboratory tests to determine mechanical properties of the material. Since the wall and column moulds consist mainly of vertical panels, focus was put to the properties of cross sections made of layers stacked along the global z axis of the printer. Compared to van der Veen (2014), the scope of investigation has been limited to only basic properties that are relevant to given application:

- **tensile strength and stiffness** – most basic and reliable measure of material behaviour – done repetitively with various strain rates
- **flexural strength and stiffness** – another core property, especially important for the project as the walls of the filled form work mainly against bending
- **shear strength** – parameter defining the susceptibility of the element to failure mode that is possible at low strains and therefore hard to anticipate
- **tensile creep rate** – rate of deformation under constant stress, main measure defining viscoelasticity
- **temperature dependency** – reliability of the element at increased temperatures, due to e.g. sun radiation or heat release during curing of the infill

Since the printed product is highly orthotropic, strength and stiffness tests were executed for two directions – along and across the print. Biaxial tests have not been done due to time and resource limitations. In case of last two properties mentioned above, it has been assumed that it is sufficient and economically justified to test them in one direction instead of two, therefore only longitudinal tests have been performed. No compression tests have been done as this type of stress state is almost nonexistent within the formwork system.

Mould and concrete tests – casting of the predesigned moulds and investigation of their behaviour as well as testing the final concrete product for its strength and water absorption. Geometry of the specimens has been defined taking into account the stability issues explained in Section 8.2.2.

4.3 Time dependency

Polymer used to print the forms was expected to settle its properties after ca. 14 days. For practical and economical reasons, it was desirable to get an insight into the aging process in order to evaluate the possibility of using the forms sooner. Therefore, a five-step schedule has been adopted, with both basic property and formwork tests being performed after 1, 3, 7, 14 and 28 days from the manufacturing date. Knowing in advance the approximate material characteristics allowed running both schemes in parallel. As a consequence, the test stage took less than 3 months, starting in mid-January and being finished by the end of March.

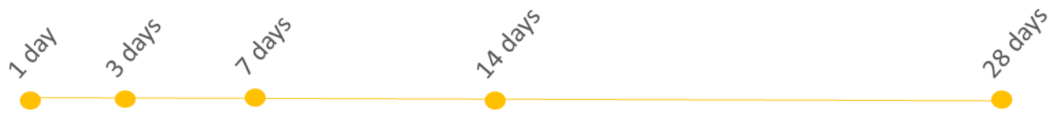


Figure 4-2: Timeframe for the test phase.

4.4 Material tests

4.4.1 Code review

As of 2016, there were no standards covering the use of structural polymers in the construction industry (although introduction of a new Eurocode has been suggested to the European Commission already in 2007 by Gutiérrez et al. (2007)). Most relevant documents defining the procedures of testing the material researched in this report were provided by the International Organization for Standardization (ISO). These, however, refer to general cases that for various reasons could not be directly applied to the purpose of this project.

First of all, it needs to be highlighted that at the current stage the print quality control is still underdeveloped, as explained in the previous chapter. This fact had a substantial impact on the sample geometry. In order to embrace the potentially variable print resolution it was desirable to test relatively large pieces, which on the other hand was limited by the maximum unsupported span of 300 mm. Consequently, a set of equally long (240 mm) specimen geometries has been introduced to provide a universal scheme based on a workable area of 240×240 mm obtained from square plates of 300×300 mm (Figure 4-3.). However, according to the observations presented in the next chapter, the quality within almost every set of tested samples was non-uniform, which questions the correctness of the assumed maximum dimensions.

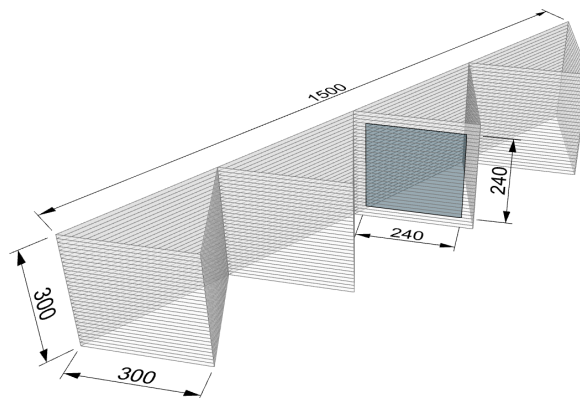


Figure 4-3: Print piece consisting of 13 square plates with workable areas of 240×240 mm.

Codewise, unification of the specimen dimensions resulted in substantial derogations from the recommendations provided by ISO (ISO 527-1 (Nederlands Normalisatie-instituut, 2012), ISO 527-4 (Nederlands Normalisatie-instituut, 1997), ISO 178 (Nederlands Normalisatie-instituut, 2010), ISO 899-1 (Nederlands Normalisatie-instituut, 2003) – used samples are in general larger and have different proportions than the standard ones. What is more, the shape applied in testing the shear strength is a custom solution first proposed by Bao and Wierzbicki (2004)

and then used for testing of the 3D prints by van der Veen (2014), but it has not yet been included in any official standard.

Additional constraints that needed to be taken into account while designing the test framework were the laboratory time and resource limitations, which in many cases resulted in numbers of specimens below the minima defined by the codes (often only 3 pieces were tested, while ISO in general requires no less than 5 measurements per each series). This of course reduces the scientific value of the research, but it is still enough to determine the characteristic values of the properties according to NEN-EN 1990 (Nederlands Normalisatie-instituut, 2008). Therefore, such approach was considered acceptable. In general, however, obtained results need to be analyzed with the utmost care as their reliability might be questionable due to low sample counts.

4.4.2 Tensile test

Axial tension test has been extensively used in this research as it provides a suitable platform for measuring and evaluating basic material properties as well as the influence of temperature or creep behaviour. A universal specimen has been defined for all tensile tests in compliance with the general considerations explained in the previous section. With 240 mm in length and a narrow part of 100×20 mm, it is an oversized and slightly distorted version of the geometry suggested in ISO 527-4 code for testing short fiber-reinforced thermoplastics.

Standard tensile tests recording the deformation under increasing load in room temperature have been done repetitively with different strain rates ranging from 2 mm/min up to 50 mm/min and maximum strain of 80 mm. Extensometers have not been used due to the concerns regarding their reliability on uneven, layered surfaces. Details of the setup that has been used are given in Table 4-1.


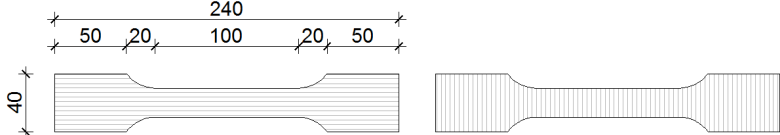
Test name:	Tensile test
Test setup:	
Specimen geometry:	
Testing equipment:	Testing machine: Zwick Z100 or Zwick Z010 (alternatively)
Testing speeds:	2 mm/min, 10 mm/min, 50 mm/min
Tested values:	Standard force, nominal displacement (between grips)

Table 4-1: Specification: tensile test.

4.4.3 Flexural test

A three point bending test has been used for measuring the flexural properties of 3D printed single layer cross-sections. Similarly as for axial tension, used specimen geometry differs from the basic one suggested by the ISO 178 standard. This, however, should not be regarded as a serious breach since the code leaves much freedom in case of rough and anisotropic materials. The dimensions have been chosen so that the samples fit in the workable area of 240×240 mm, while providing enough volume to measure the global behaviour. The 3-point out of plane bending test was performed with nominal support span of 115 mm, deformation speed of 10 mm/min and was stopped after reaching 40 mm deflection, which can be translated into 9.6% and 5.2% strain in the outer fiber of a 5 mm and 2.7 mm thick cross-section, respectively (which refers to average and minimum thicknesses of a printed wall, see Table 5-1). This provides enough data to measure both flexural stiffness as well as flexural yield strength, which, according to common practice for thermoplastics, is equal to the stress when the maximum strain in the outer fiber of the specimen has reached 5% (Shah, 2007).

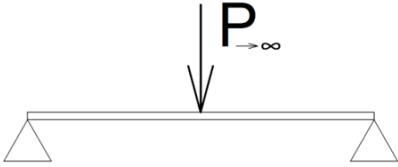
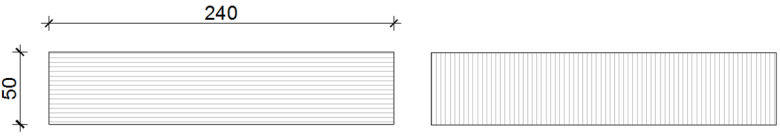
Test name:	Flexural test
Test setup:	
Specimen geometry:	
Testing equipment:	Testing machine: Zwick Z010
Testing speed:	10 mm/min
Tested values:	Standard force, displacement of the head

Table 4-2: Specification: flexural test.

4.4.4 Shear test

As previously mentioned, the shear test has been performed on specimens inspired by the work of Bao and Wierzbicki (2004). The applicability of such geometries for testing 3D printed materials has been proven van der Veen (2014). Previously used dimensions have been optimized with regard to the workable area of the printed plates. The test was performed by stretching the specimen with the speed of 2 mm/min until it either broke or delaminated.

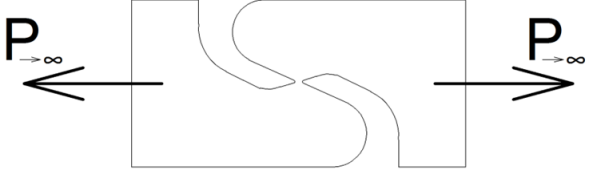
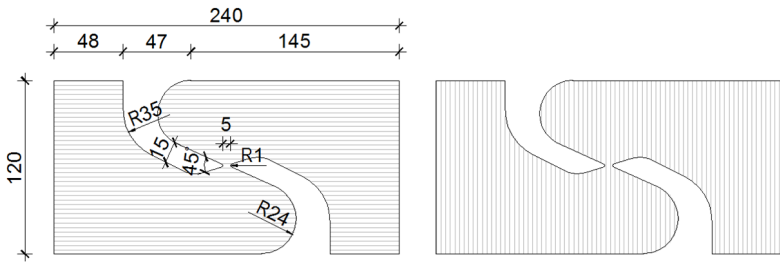
Test name:	Shear test
Test setup:	
Specimen geometry:	
Testing equipment:	Testing machine: Zwick Z010
Testing speed:	2 mm/min
Tested values:	Standard force, nominal displacement (between grips)

Table 4-3: Specification: shear test.

4.4.5 Tensile creep test

According to ISO 899-1 standard, the same specimen geometry is suggested for both tensile strength and creep under tension tests. Therefore, the dumbbell described in section 4.4.2 has been used for investigation of the viscoelastic behaviour of the material.


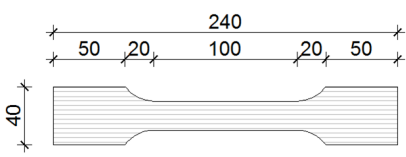
Test name:	Tensile creep test
Test setup:	
Specimen geometry:	
Testing equipment:	Testing machine: Zwick Z010
Testing time:	600 s
Tested values:	Time, nominal displacement (between grips)

Table 4-4: Specification: tensile creep test.

A relatively short timeframe of 600 s has been considered sufficient for the purpose of small scale concrete forms since in such application the pressure on the walls disappears after less than 24 hours. Besides that, no tests including unloading have been performed. This decision has been made based on the assumption that the initial load of the mixture is gradually turned into imposed deformation caused by the hardened concrete, which restrains significant reverse movement of the mould.

4.4.6 Thermal sensitivity

The influence of temperature on mechanical properties of the material has been investigated by tensile testing the specimens heated up to certain temperatures (30°C, 40°C, 50°C, 60°C, 70°C). Optimal equipment for such experiment is a testing machine fitted inside the oven. Here an alternative solution has been chosen: due to technical limitations, the samples have been first heated up in the separate oven and then tested in the universal machine in room conditions. Although in such case the material is subject to cooling down while being tested, it has been considered negligible with regard to expected precision of the output. The same specimen geometry has been used as for the tensile tests, strain rate of 10 mm/min has been applied.

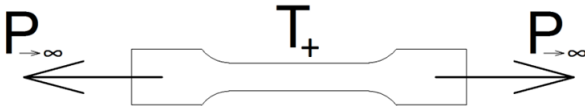
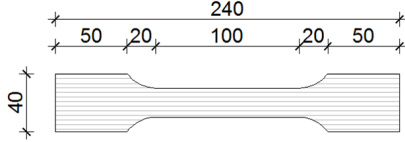
Test name:	Temperature dependency test
Test setup:	
Specimen geometry:	
Testing equipment:	Heating oven: Heraeus UT6, testing machine: Zwick Z100
Testing speed:	10 mm/min
Tested values:	Standard force, nominal displacement (between grips)

Table 4-5: Specification: thermal sensitivity test.

4.4.7 Preparation of the specimens

All samples presented in section 4.4 have been prepared in the same manner. For each batch of specimens, first a geometry consisting of 13 interconnected plates of 300×300 mm (Figure 4-3) has been printed by DUS Architects. Then it was separated into planar pieces with the use of an oscillating tool. Later, the plates were laser cut into specimens at the ProtoSpace FabLab Utrecht based on previously prepared cut layout.



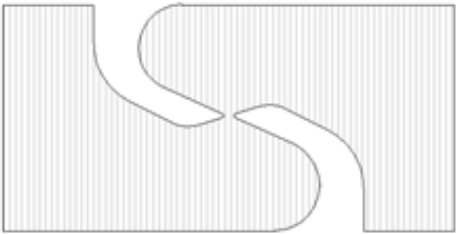


Test	Specimen	No. of batches	Standard no. of specimens in batch
Tensile		5	15L 15T
Flexural		5	3L 3T
Shear		5	3L 3T
Tensile creep		5	4L
Temperature		5	5L

Table 4-6: Specimen summary.

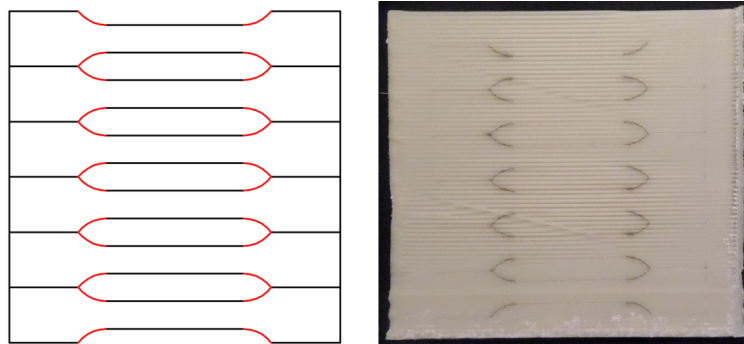


Figure 4-4: Left: optimized laser cut layout, red lines piercing through the material, black ones penetrating outer layer only, then cut with a paper knife. Right: 300 × 300 mm plate after laser processing.

4.5 3D printed mould tests

There were two main reasons for running the formwork tests. Firstly, to confirm the general applicability of the technique, and secondly to get an insight in how do the experimental properties translate into real 3D printed structures. Two types of geometries have been investigated: single layered moulds were tested in every time step (after 1, 3, 7, 14, 28 days from printing) in order to get a general overview of mechanics of the extruded forms, while a bigger structure consisting of truss-like, cross-linked walls was filled with concrete to check whether the basic rules can be applied to more complex scale.

As mentioned in Section 4.2, dimensions of the forms were designed in a way that provides necessary wall stability. This has been achieved by performing a preliminary analysis based on predicted material properties (linear elastic material model) and hydrostatic load of concrete. All tests described in this and next section have been performed at the Macrolab section of Stevin Laboratory at the TU Delft.

4.5.1 Single-layered forms

Single-layered prismatic forms of 200 × 200 × 500 mm have been tested with the use of a custom-made setup, which is presented in Appendix F. A 3D printed form was placed on the platform, sealed with clay and then filled with concrete. Out of plane deformation during casting and hardening of the concrete was initially measured by two separate means:

- on the back wall a set of four strain gauges measured the out of plane deformations 125, 250, 375 and 500 mm above the zero level of the platform
- three dimensional deformations of the front wall were measured with the use of a photogrammetric method developed by B. Gorte and explained in Appendix I.

However, as the project matured, the decision has been made to neglect the results collected with the former technique due to high uncertainty resulting from potential creep of the wooden platforms and slipping of the gauge needles on the mould surface.

The measurements were taken three times for each specimen before demoulding:

- before casting the concrete (initial setup)
- right after casting the concrete, instantaneous elastic deformation
- after over 24 hours from casting to investigate the amount of creep

After hardening of the concrete, the blocks were demoulded and examined once again without the 3D printed envelope, i.e. front and back wall were measured with photogrammetry. The output of that procedure were point grids representing walls of the final product.

4.5.2 Truss-like walled form

A prismatic block of $600 \times 800 \times 200$ mm has been cast in a 28 days old built-up mould (Figure 4-5). A setup consisting only of a base plate has been used as only the deformations of demoulded product were measured. Similar to other concrete pieces, point clouds of two largest faces of the object have been generated with photogrammetry.

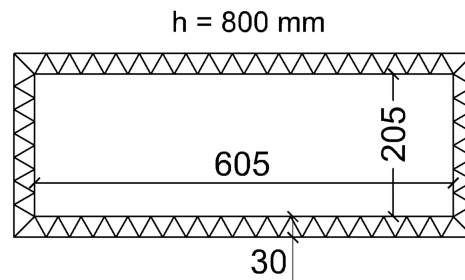


Figure 4-5: Built-up mould geometry, axial dimensions.

4.5.3 Preparation of the specimens

The specimens of $200 \times 200 \times 500$ mm were printed in batches of three up to six pieces and then cut and polished with the oscillating tool. In total 18 such forms were prepared and tested. Print quality varied between the specimens, but in general it was acceptable with only incidental small holes in the corner zones. These were either neglected or filled with clay. Only one built-up form of $600 \times 800 \times 200$ mm has actually been prepared and tested in this stage as a result of a trade-off between scientific need and amount of the resources for further stages of the research. It has been printed in one piece during an eight hour shift.

Two concrete types have been used for the experiments. A basic mixture applied to 15 out of 18 small specimens and to the big one was a standard C20/25 based on CEM III/B 42.5, compacted with a needle vibrator. Alternatively, one batch of single-layered forms has been cast with a self-compacting C28/35 mixture in order to evaluate the influence of a more fluid infill combined with no vibrations. Technical specification of each concrete type can be found in Appendix G and Appendix H.

4.6 Concrete property tests

Concrete property tests constitute the last step of the framework described in this chapter. Their objective was to confirm that quality of the final product does not differ from the standard concrete applications. Two values have been recognized as most relevant to the project: strength and water absorption (in consequence, porosity). The former reflects structural performance of the concrete, while the latter refers to its durability. The choice was dictated by the available time, equipment and resources – the scheme consists of a minimum number of simple tests necessary to get an overview of the influence of manufacturing process on the product. Along with specimens cast in the 3D printed moulds, a batch of standard cubes was prepared as a source of reference strength values.

4.6.1 Strength

A standard compressive strength test has been performed according to NEN-EN12390-3 (Netherlands Normalisatie-instituut, 2009). Cubes $150 \times 150 \times 150$ mm were compressed with the speed of 13.5 kN/s until they broke. The reference specimens were tested after 28 days from casting. Testing of the 3D printed formwork products has been delayed due to difficulties in cutting the blocks to the right size. Finally, usable (although not perfectly cubic) specimens were provided and tested after 84 days (standard mixture) and 94 days (self-compacting mixture) from casting. The age difference, although significant, has been considered acceptable for practical purposes as the expected increase in concrete strength after the 28th day from casting lies in range between 0 and 10 percent, depending on source (Price, 1951; Neville, 2012). Additionally, the Eurocode provides a formula that normalizes strength of the material based on its age (see Equation 5.13).

4.6.2 Water absorption

Water absorption has been tested according to the Belgian code NBN 15-215:1989 (Belgisch Instituut voor Normalisatie, 1989), which provides a simple method based on comparing the weight of a specimen soaked with water and its weight after drying. For that purpose, hardened (>28 days old) $150 \times 150 \times 150$ mm specimens were first immersed in water in the room temperature for at least 48 hours, then weighted, dried in the oven for over 72 hours at the temperature above 100°C and finally weighted once again.

4.6.3 Preparation of the specimens

$150 \times 150 \times 150$ mm cubic specimens have been cut out of pieces described in Section 4.5.3. As already mentioned, problems with cutting of the large blocks into smaller ones resulted in a serious delay as well as geometrical imperfections of ± 5 mm. The influence of these on the test results is explained in section 5.3.1. The reference cubes have been cast at the concrete lab using standard forms and techniques.

Chapter 5

Test Results

This chapter contains the numerical outcome of all experiments that were performed during the test stage along with the resultant property values and qualitative conclusions. Three categories of tests are covered in three separate sections: material investigation, behaviour of the moulds and finally the quality of concrete products. Since volume of the raw output is large and hardly compressible, only processed data is presented on the following pages, while the former can be found in Appendix B. Information provided here is meant to supply the reader with sufficient knowledge for understanding and critical evaluation of further chapters of this report.

5.1 Material tests

Systematization of the basic material tests proved to be a challenging task mainly due to two phenomena: time-dependent curing process and variable print quality. Because of these complexities, each specimen needed to be treated as an individual object with its own characteristics. In order to build a consistent description of mechanical behaviour of the 3D printed elements under load, the results have been evaluated for each age group separately and an attempt has been made to unify the cross-sectional properties of specimens. Since the section of force-displacement graph that is relevant for design purposes lies between its starting point and the point where the ultimate force occurs, in most cases the analysis and resultant curves are limited to this specific range. In this section the results of each test are treated independently, while the relationships between them are investigated in Chapter 6.

5.1.1 Factors influencing the results

There are multiple settings and variables that might affect the behaviour of specimens before and during the tests, and in consequence influence the results. They might be caused by various reasons, starting from the outer environment, through imprecisions of the machinery up to human mistakes.

In general, phenomena distorting the experimental output can be divided into two main groups:

- the ones that refer to the testing method and conditions
- the ones that refer to the specimens

All material tests were executed at the Materials Science and Engineering laboratory at the Faculty of Mechanical, Maritime and Materials Engineering (3mE), TU Delft. The environment there is a typical room atmosphere: $(20\pm 2)^{\circ}\text{C}$ and $(50\pm 10)\%$ relative humidity. Consequently, an assumption has been made that all tests were held under the same conditions – no additional coefficients have been applied.

In many cases, however, interference into the results was needed in order to address the distortions caused by imperfection of the testing machinery. One of the most common issues was the toe effect (Figure 5-1), but also other phenomena took place, e.g. settling of the head in bending test. Faulty sections of the graphs have been cut out and replaced with extrapolated curves based on correct data (Swallowe, 1999).

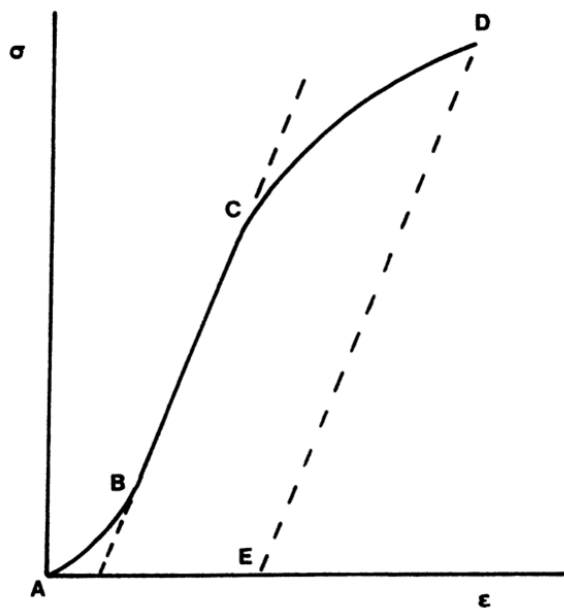


Figure 5-1: Stress-strain curve illustrating the toe effect (Swallowe, 1999).

Adaptation of the results to variability in the specimen geometry and properties was more complex. Large differences in the cross-sectional properties deriving from unstable print quality needed to be addressed. To cover this issue, a print quality classification system has been introduced with five classes depending on the cross-section of the print. Geometrical parameters of each class are shown in the Table 5-1 (for a 50 millimeter-wide cross section). A set of correction factors has been calculated for bending and tension in both directions in order to unify the characteristics of different classes. Along the print line this is simple since the deformation of the element refers directly to its cross-sectional properties: area and second moment of inertia.






Class	Cross section (50mm)	A [mm ²]	I [mm ⁴]	t _{max} [mm]	t _{min} [mm]
1		225	427	5.0	4.1
2			381	5.3	3.7
3			392	5.6	3.3
4			485	5.9	2.9
5			541	6.2	2.5

Table 5-1: Dimensional properties of cross section classes.

Modelling of the mechanics in transverse direction is more complex as the cross section varies over length. A robust simplification has been applied in order to address this issue. An assumption has been made that the effective volume that works against tension and bending is contained within the core of the cross section, as shown in Figure 5-2.

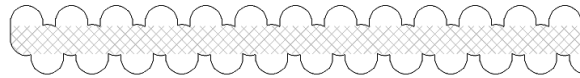


Figure 5-2: Effective cross section working against transverse tension and bending.

Following relationships based on dimensional properties of the cross sections have been derived based on linear theory for tension and bending:

Longitudinal tensile force:
$$F_{x,eq} = \frac{A_m}{A_n} F_x = F_x \quad (5.1)$$

Longitudinal bending moment:
$$M_{y,eq} = \frac{I_m}{I_n} M_y \quad (5.2)$$

Transverse tensile force:
$$F_{y,eq} = \frac{t_{min,m}}{t_{min,n}} F_y \quad (5.3)$$

Transverse bending moment:
$$M_{x,eq} = \frac{t_{min,m}^3}{t_{min,n}^3} M_x \quad (5.4)$$

Where n is quality class of the object, m the reference class and t is thickness.

Unification of the results provides better control over the mechanical behaviour of the prints. It allows comparing the properties of different classes, which can be then translated into the models of large scale objects. In the material test stage, each of the specimens has been assigned with a class based on visual examination in combination with measurements of minimum and maximum thicknesses before testing. Where applicable, the results have been reclassified to class 1, which is most stable dimensionwise and therefore most robust. Properties for other classes can be obtained by refactorization of the design values with respective coefficients. Full matrices of correction factors for each mode of deformation can be found in Appendix C.

At this point it needs to be highlighted that the approach described above, although practical, bears a risk of numerical distortion. Due to limited length of the test series and random distribution of the quality classes within them, the proposed factors could not be evaluated with high statistical reliability. Consequently, all values presented in this report should be considered as provisional and need more extensive investigation to be accepted as definitive.

5.1.2 Derivation of design values

Two groups of values defining the cross section can be distinguished with regard to their numerical interpretation for the purpose of structural analysis:

- **strength** – depending on the force-displacement curve shape it is either the maximum force (in case of nearly linear curve with a distinguishable peak) or yield force (nonlinear graph with gradually decreasing stiffness, e.g. tensile test along the print direction or double-linear graph with distinguishable yield point, e.g. shear across the print). Each statistical strength data is translated into a characteristic value according to the instructions provided by NEN-EN 1990 (Nederlands Normalisatie-instituut, 2008).
- **elasticity and thermal sensitivity** – no minimum requirements, therefore statistical means are used in the design as representative values.

Due to variable print quality and complex orthotropic behaviour, mechanical properties of the material itself are hardly applicable for engineering purposes. More robustness and efficiency in modelling can be achieved by considering all values as cross-sectional properties (per running millimeter):

$$\text{In-plane strength:} \quad \frac{F}{b} \left[\frac{N}{mm} \right]$$

$$\text{Out-of-plane strength:} \quad \frac{M}{b} [N]$$

$$\text{In-plane stiffness:} \quad \frac{EA}{b} \left[\frac{N}{mm} \right]$$

$$\text{Out-of-plane stiffness:} \quad \frac{EI}{b} [Nmm]$$

In order to derive the above values from the test results, it is crucial to determine the effective dimensional properties of the specimens. These are equal to real dimensions with one exception: the effective length of the strained zone in tensile tests has been normalized from 140 mm to 132 mm, which is explained in Appendix D.

In the graphs in subsections 5.1.3-5.1.6 each curve represents an average of the results obtained for a series of specimens. Dispersion of the values within each series as well as resultant material properties can be found in relevant tables. Statistical background of the calculations performed in this chapter is presented in Appendix E.

5.1.3 Tensile longitudinal tests

Being a universal source of information about such properties as stiffness, strength, creep or thermal sensitivity, tensile tests along the print direction were the most extensively used type of experiment run during the laboratory stage of the project. All specimens have been analyzed with the same dimensional assumptions:

$$\begin{aligned}l_{eff} &= 132mm \\ A_{L,class1} &= 90mm^2 \\ b &= 20mm\end{aligned}$$

In case of the standard tensile test under room conditions, each set of results (strain rate/age combination) has been processed according to the following procedure:

1. Force-displacement data has been cut at the lowest displacement corresponding to the ultimate force.
2. Shortest displacement domain has been divided into 200 unidistant sections, mean value in each node and global coefficient of variation have been calculated. A graph has been generated based on the resultant nodal data.
3. Stiffness modulus has been calculated according to ISO 527-1:

$$\left(\frac{EA}{b}\right)_{code} = \frac{N_{0.25\%} - N_{0.05\%}}{(0.25\% - 0.05\%) \cdot b} \quad (5.5)$$

4. Yield strain ϵ_y has been determined by offsetting the initial tangent stiffness by 2% along the strain axis and finding intersection point with the graph, which is a method often used for thermoplastics (Swallowe, 1999).
5. Mean and characteristic yield force per running millimeter has been calculated for each series.

Two different modes of failure have been observed longitudinal tensile test: freshly printed specimens tend to plastify and elongate to large extents, while most of well cured ones broke relatively shortly after plastifying.

Graphical results for strain rate of 2 mm/min:

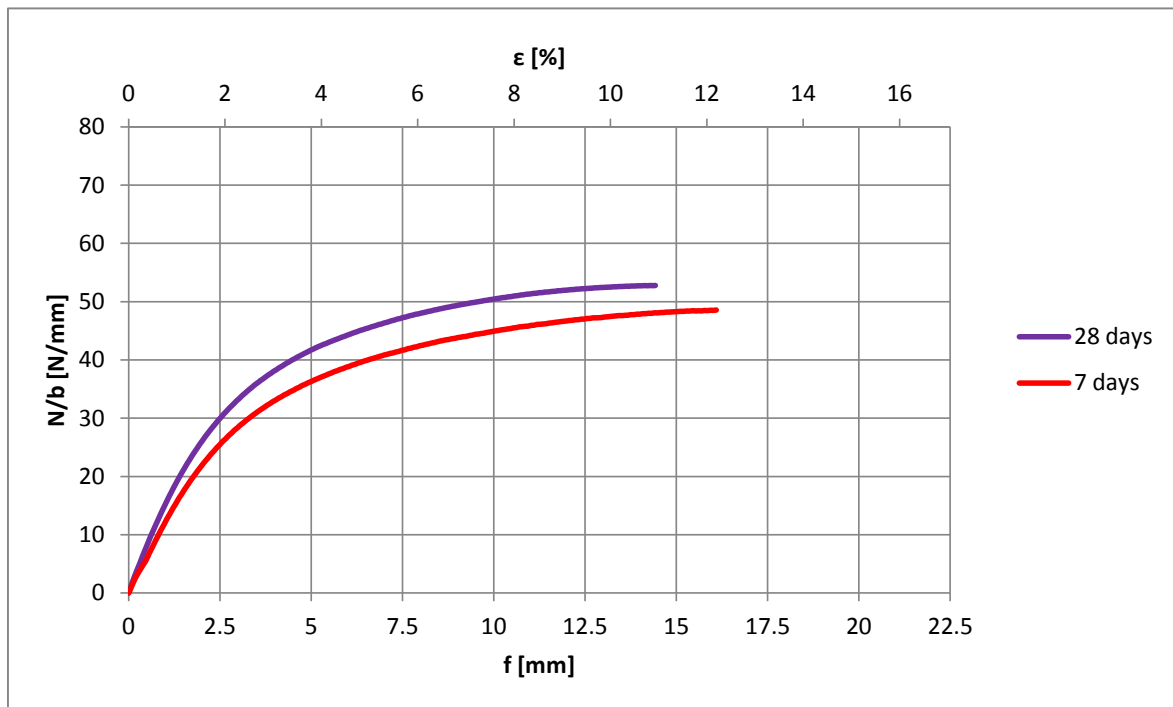


Figure 5-3: Results: longitudinal tensile test, 2 mm/min.

Tabular results for strain rate of 2 mm/min:

Age	Count	CV	ϵ_y	$\left(\frac{F_y}{b}\right)_{mean}$	$\frac{F_{yk}}{b}$	$\left(\frac{EA}{b}\right)_{code}$
[days]	[pieces]	[%]	[%]	[N/mm]	[N/mm]	[N/mm]
7	3	1.9	4.0	37.1	34.8	1610
28	5	1.2	3.7	41.3	39.5	2144

Table 5-2: Results: longitudinal tensile test, 2 mm/min.

Where:

CV - coefficient of variation

ϵ_y - yield strain

$\left(\frac{F_y}{b}\right)_{mean}$ - mean cross-sectional value of yield force

$\frac{F_{yk}}{b}$ - cross-sectional value of characteristic strength

$\left(\frac{EA}{b}\right)_{code}$ - cross-sectional stiffness according to ISO 527-1

Graphical results for strain rate of 10 mm/min:

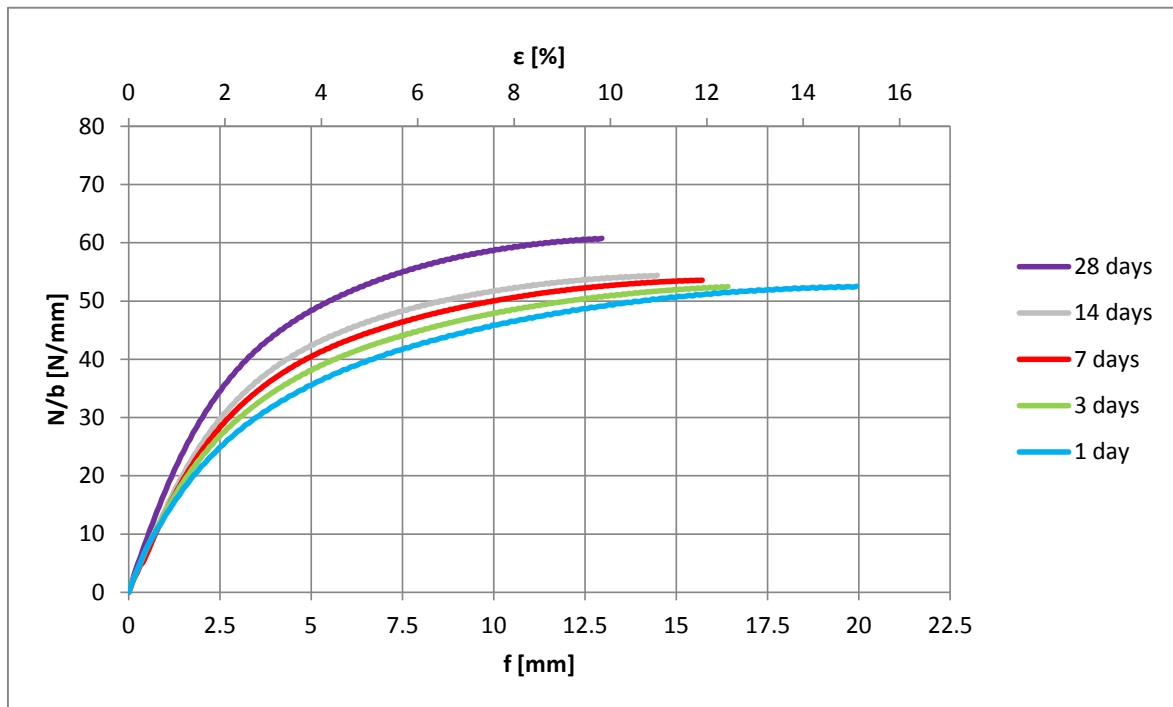


Figure 5-4: Results: longitudinal tensile test, 10 mm/min.

Tabular results for strain rate of 10 mm/min:

Age	Count	CV	ϵ_y	$\left(\frac{F_y}{b}\right)_{mean}$	$\frac{F_{yk}}{b}$	$\left(\frac{EA}{b}\right)_{code}$
[days]	[pieces]	[%]	[%]	[N/mm]	[N/mm]	[N/mm]
1	3	1.5	3.5	34.3	32.2	2053
3	6	1.8	3.4	36.3	32.8	1932
7	5	1.5	3.9	40.9	38.5	1851
14	6	1.2	3.7	41.9	39.7	1698
28	5	3.0	3.9	48.9	42.9	2440

Table 5-3: Results: longitudinal tensile test, 10 mm/min.

Graphical results for strain rate of 50 mm/min:

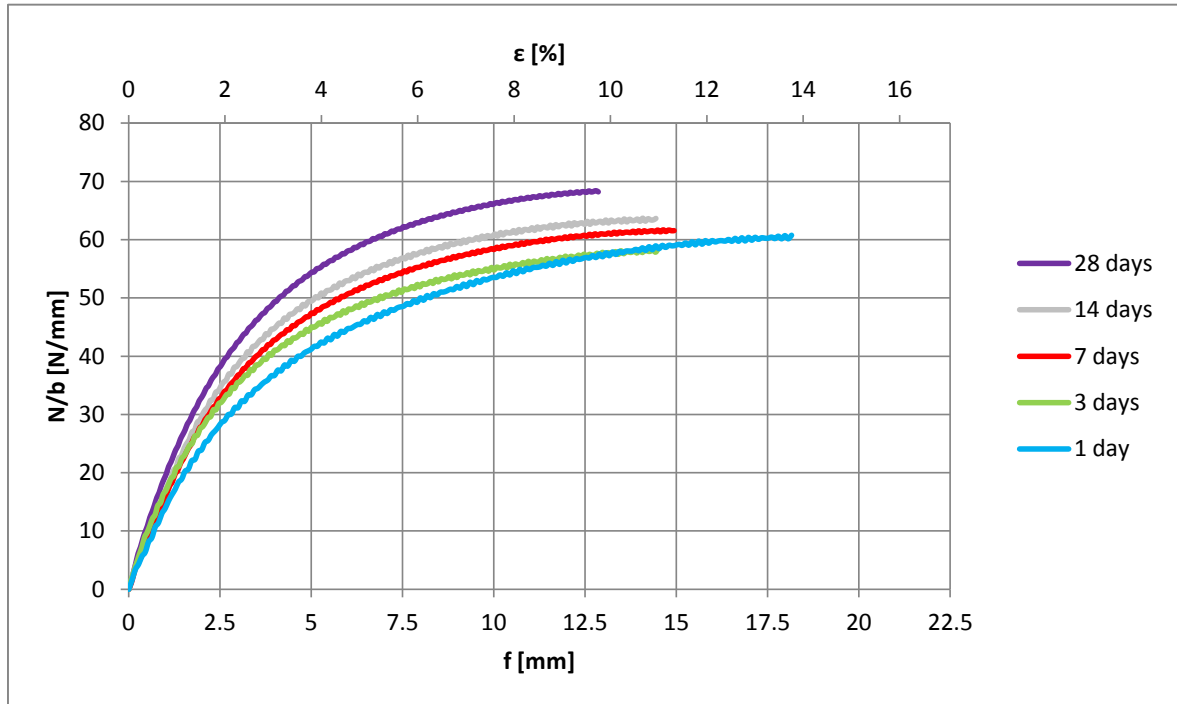


Figure 5-5: Results: longitudinal tensile test, 50 mm/min.

Tabular results for strain rate of 50 mm/min:

Age	Count	CV	ϵ_y	$\left(\frac{F_y}{b}\right)_{mean}$	$\frac{F_{yk}}{b}$	$\left(\frac{EA}{b}\right)_{code}$
[days]	[pieces]	[%]	[%]	[N/mm]	[N/mm]	[N/mm]
1	3	1.2	3.8	41.1	40.0	2022
3	3	0.7	3.4	43.0	41.8	2550
7	3	1.8	4.5	50.3	48.3	2236
14	5	2.0	3.6	48.8	44.8	2083
28	3	3.5	4.3	57.0	51.3	3056

Table 5-4: Results: longitudinal tensile test, 50 mm/min.

5.1.4 Tensile transverse tests

The analysis of tensile test data in transverse direction has been preceded by reclassification of all results into class 1 according to the method described in Section 5.1.1. Following geometrical assumptions have been used:

$$l_{eff,class1} = 132mm$$

$$A_{T,class1} = 82mm^2$$

$$b = 20mm$$

Due to high variability in layer thickness and unpredictable nature of the failure mechanism across the print line, it has been decided to fix the yield strain at the conservative value below the lowest ultimate strain observed during tests, namely $\epsilon = 5\%$. Each set of results (strain rate/age combination) has been processed as follows:

1. Data per each specimen has been reclassified to class 1.
2. Reclassified force-displacement data has been cut at the lowest displacement corresponding to the ultimate force.
3. Shortest displacement domain has been divided into 200 undistant sections, mean value in each node and global coefficient of variation have been calculated. A graph has been generated based on the resultant nodal data.
4. Stiffness modulus has been calculated based on eq. 5.5.
5. Mean and characteristic yield force per running millimetre has been calculated for each series at $\epsilon = 5\%$.

Opposite to tensile tests along the print direction, all specimens described in this section fractured suddenly under relatively low strain rates, with no visible relation to age or quality.

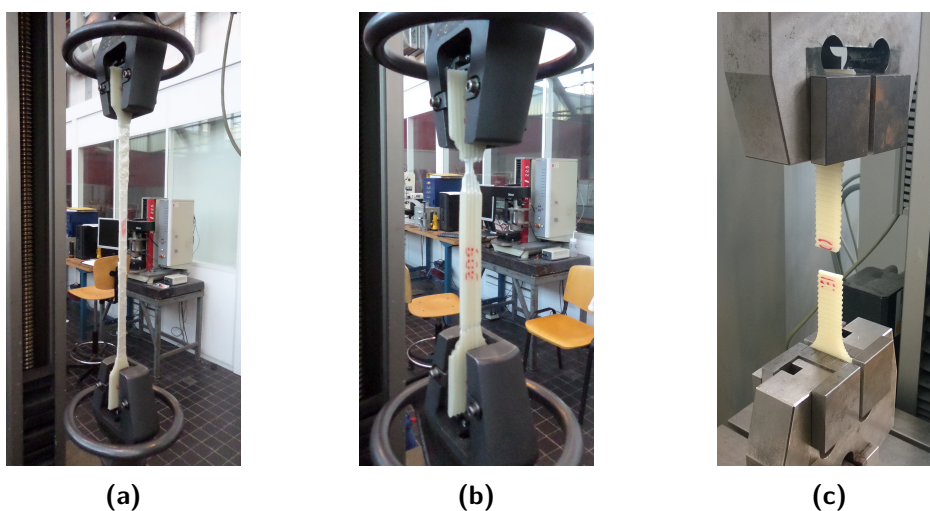


Figure 5-6: Tensile test: different failure modes. (a) plastification, high elongation (b) plastification, low elongation (c) rupture.

Graphical results for strain rate of 2 mm/min:

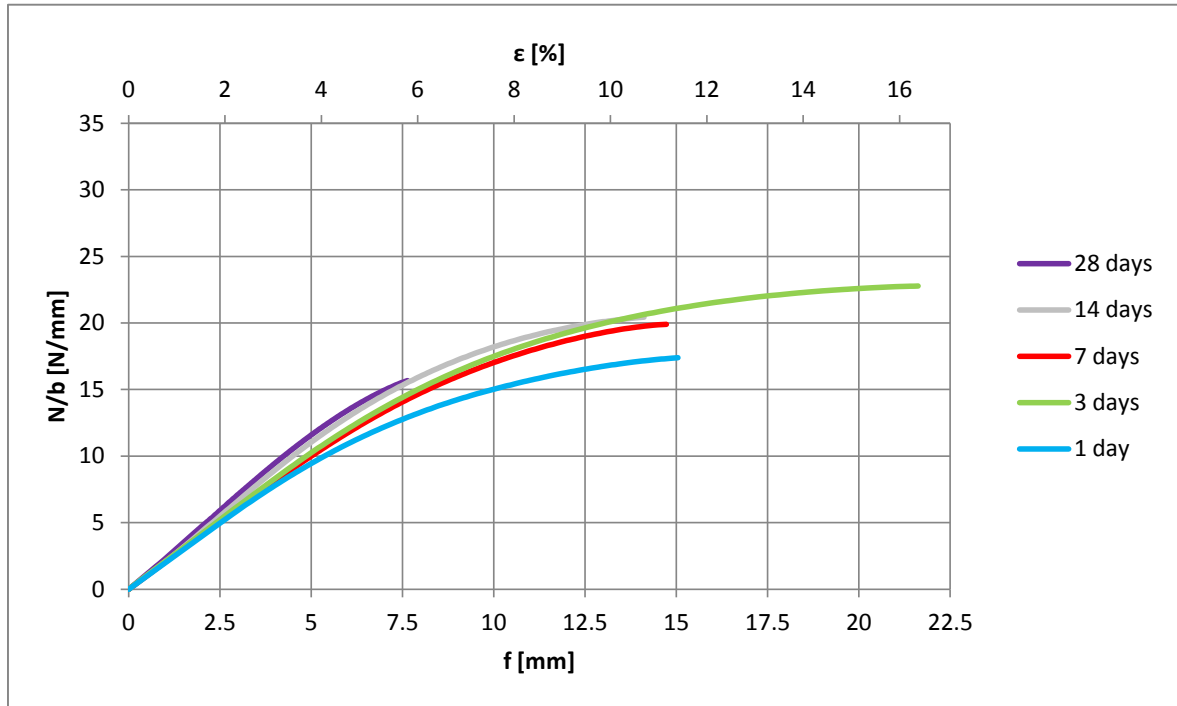


Figure 5-7: Results: transverse tensile test, 2 mm/min.

Tabular results for strain rate of 2 mm/min:

Age	Count	CV	ϵ_y	$\left(\frac{F_y}{b}\right)_{mean}$	$\frac{F_{yk}}{b}$	$\left(\frac{EA}{b}\right)_{code}$
[days]	[pieces]	[%]	[%]	[N/mm]	[N/mm]	[N/mm]
1	3	9.5	5.0	11.7	8.2	262
3	3	9.0	5.0	13.0	9.1	282
7	3	1.9	5.0	12.7	12.0	270
14	3	1.6	5.0	14.0	13.5	273
28	5	3.1	5.0	14.4	13.3	304

Table 5-5: Results: transverse tensile test, 2 mm/min.

Graphical results for strain rate of 10 mm/min:

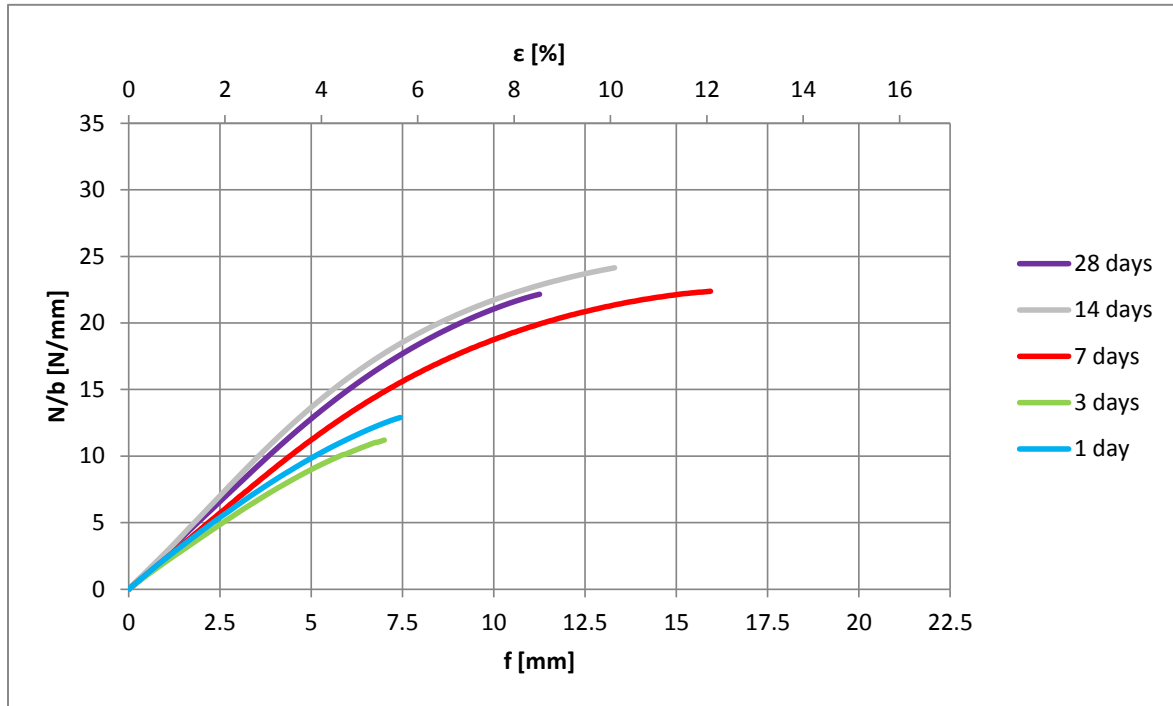


Figure 5-8: Results: transverse tensile test, 10 mm/min.

Tabular results for strain rate of 10 mm/min:

Age	Count	CV	ϵ_y	$\left(\frac{F_y}{b}\right)_{mean}$	$\frac{F_{yk}}{b}$	$\left(\frac{EA}{b}\right)_{code}$
[days]	[pieces]	[%]	[%]	[N/mm]	[N/mm]	[N/mm]
1	6	5.5	5.0	12.0	9.6	306
3	5	4.6	5.0	10.9	8.0	282
7	8	2.9	5.0	14.2	12.2	302
14	6	3.8	5.0	17.0	14.7	359
28	5	2.9	5.0	16.1	14.7	353

Table 5-6: Results: transverse tensile test, 10 mm/min.

Graphical results for strain rate of 50 mm/min:

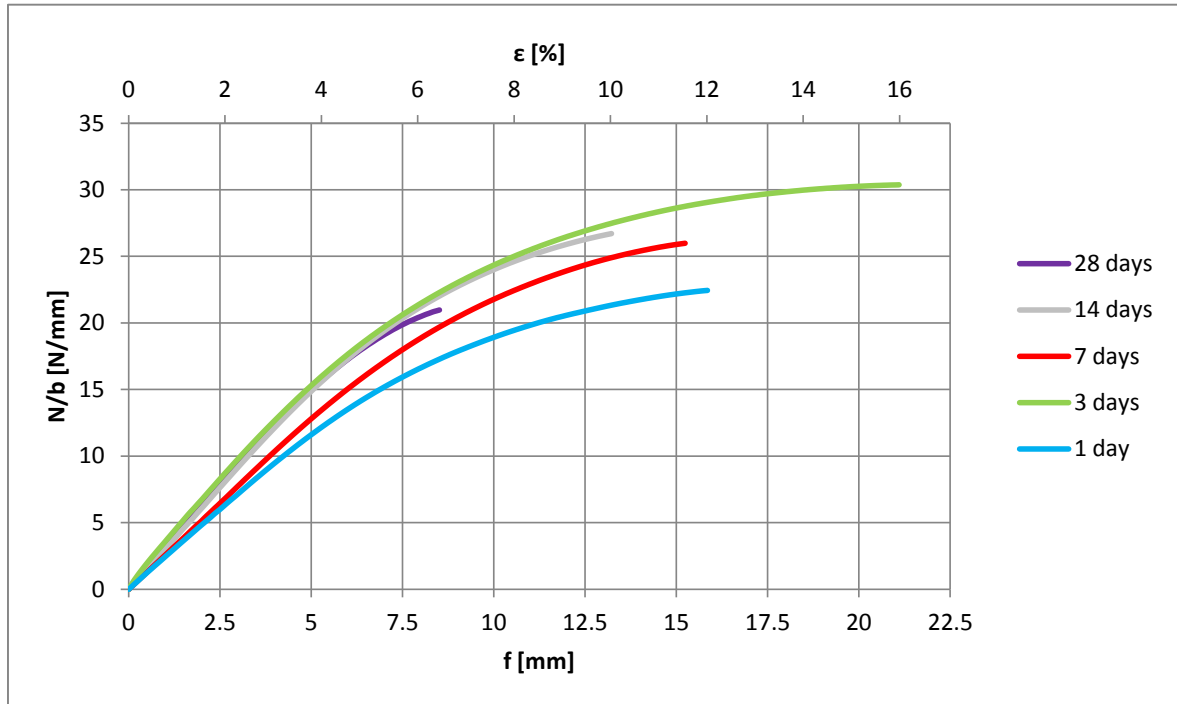


Figure 5-9: Results: transverse tensile test, 50 mm/min.

Tabular results for strain rate of 50 mm/min:

Age	Count	CV	ϵ_y	$\left(\frac{F_y}{b}\right)_{mean}$	$\frac{F_{yk}}{b}$	$\left(\frac{EA}{b}\right)_{code}$
[days]	[pieces]	[%]	[%]	[N/mm]	[N/mm]	[N/mm]
1	3	4.2	5.0	14.5	12.8	328
3	3	3.9	5.0	18.9	16.2	525
7	3	1.8	5.0	16.3	15.1	339
14	6	2.5	5.0	18.6	17.0	397
28	3	1.6	5.0	18.4	17.5	409

Table 5-7: Results: transverse tensile test, 50 mm/min.

5.1.5 Flexural longitudinal tests

Bending test results have been processed based on the assumption of linear stress distribution. Although this can be regarded as a far reaching simplification, there are at least two practical reasons for that. Firstly, correct and detailed reproduction of complex nature of the investigated material is nearly impossible based on available amount of test data, while erroneous interpretation could lead to deceiving results. What is more, for the given purpose (limited strains), chosen approach has been expected to provide sufficiently precise information – this issue has been elaborated in Section 6.3.4 and validated in Section 7.2. With the assumption of linear stress distribution, formulas describing the relationships between the standard force, moment, deformation and strain can be easily derived:

$$M = \frac{Pl}{4} \quad (5.6)$$

$$\delta = \frac{Pl^3}{48EI} \quad (5.7)$$

$$\epsilon = \frac{6\delta t}{l^2} \quad (5.8)$$

With the use of equations above, cross-sectional stiffness of the print can be calculated. Value of $\epsilon = 5\%$ has been used as yield strain, which is suggested for flexural testing of plastics (Shah, 2007). Similar to the tensile transverse test, raw data has been reclassified to class 1 prior to deriving any resultant values. The nominal span of 115 mm has been decreased to 112 mm due to its effective shortening under large deformations, which is explained in Section 7.2. Following dimensional properties of a specimen have been assumed:

$$I_{class1} = 427mm^4$$

$$t_{class1} = 5.0mm$$

$$b = 50mm$$

$$l = 112mm$$

Test output has been processed according to a procedure:

1. Data per each specimen has been reclassified to class 1.
2. Reclassified force-displacement data has been cut at the lowest displacement corresponding to the ultimate force.
3. Shortest displacement domain has been divided into 200 undistant sections, mean value in each node and global coefficient of variation have been calculated. A graph has been generated based on the resultant nodal data.
4. Stiffness modulus has been calculated according to ISO 178 with the use of equations 5.6-5.8:

$$\left(\frac{EI}{b}\right)_{code} = \frac{(Mt/2)_{0.25\%} - (Mt/2)_{0.05\%}}{(0.25\% - 0.05\%) \cdot b} \quad (5.9)$$

5. Mean and characteristic yield moment per running millimetre has been calculated for each series at $\epsilon = 5\%$.

Graphical results:

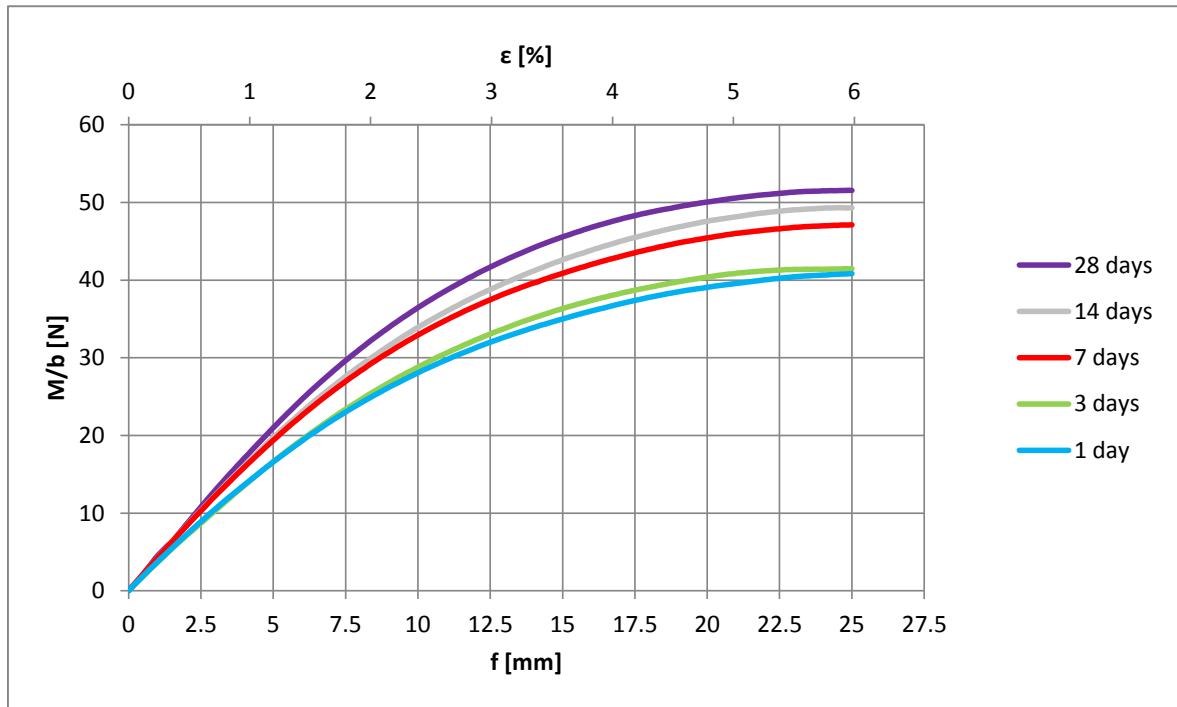


Figure 5-10: Results: longitudinal flexural test.

Tabular results:

Age	Count	CV	ϵ_y	$\left(\frac{M_y}{b}\right)_{mean}$	$\frac{M_{yk}}{b}$	$\left(\frac{EI}{b}\right)_{code}$
[days]	[pieces]	[%]	[%]	[N]	[N]	[Nmm]
1	3	4.6	5.0	39.5	33.3	3834
3	3	5.8	5.0	40.8	31.7	3780
7	3	2.5	5.0	46.0	41.1	4385
14	3	1.1	5.0	48.1	46.5	4409
28	3	1.3	5.0	50.5	47.8	4794

Table 5-8: Results: longitudinal flexural test.

5.1.6 Flexural transverse tests

Flexural test in the transverse direction proved to be a relatively unreliable source of information about the material behaviour. It provided highly variable output, which remained inconsistent even after refactorization based on the print quality. Although no failure occurred, only a limited section of the force-displacement graphs could be analysed due to slipping of the specimen that took place during the test. Consequently, no information concerning flexural strength could be derived from the results – only stiffness has been investigated. Following cross-sectional dimensions have been used in calculations after reclassification to class 1:

$$I_{class1} = 287mm^4$$

$$t_{class1} = 4.1mm$$

$$b = 50mm$$

$$l = 112mm$$

Stepwise procedure that has been used:

1. Data per each specimen has been reclassified to class 1.
2. Reclassified force-displacement data has been cut at the lowest displacement corresponding to the ultimate force.
3. Shortest displacement domain has been divided into 200 unidistant sections, mean value in each node and global coefficient of variation have been calculated. A graph has been generated based on the resultant nodal data.
4. Stiffness modulus has been calculated based on eq. 5.9.

Tabular results:

Age	Count	CV	$\left(\frac{EI}{b}\right)_{code}$
[days]	[pieces]	[%]	[Nmm]
1	3	11.2	544
3	3	27.9	462
7	4	1.4	800
14	5	5.9	1066
28	4	14.6	978

Table 5-9: Results: transverse flexural test.

Graphical results:

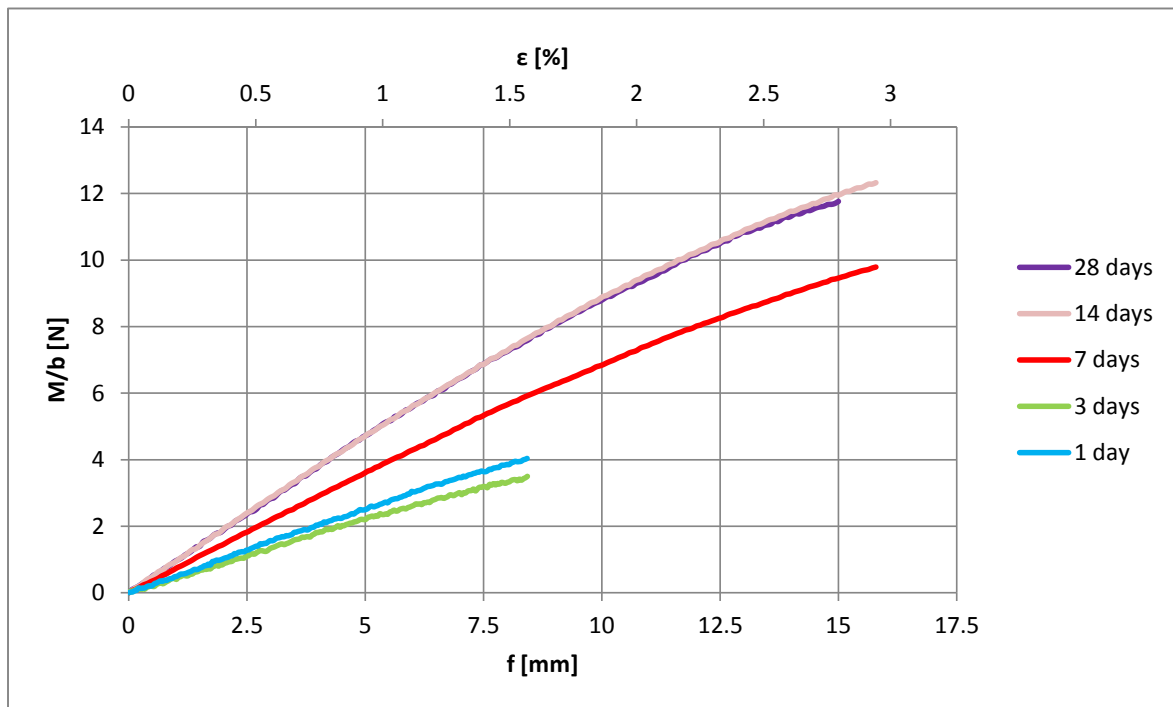


Figure 5-11: Results: transverse flexural test.

5.1.7 Shear between the layers

Due to the choice of setup used in the shear test, it was impossible to measure the strain in a reliable way. Therefore, only the strength results are presented in coming two sections. Figure 5-12 and Figure 5-13 show the force-displacement graphs for all specimens tested for shear strength between the layers during the whole test stage, with distinction into age groups and classes, respectively. It can be seen that the maximum values per each series correlate much better with the print quality than with the time that passed since extrusion. This led to neglect of the latter and application of the framework consisting of following steps:

1. Maximum force per each series has been derived from the test output data.
2. Mean and characteristic ultimate force per running millimetre has been calculated for each class based on the assumption of width of the notched section equal to 5 mm.
3. Due to the lack of specimens of class 3, an approximate method has been used to determine the characteristic strength value for this division. Firstly, a linear least square regression curve has been fitted into the known numbers, then the value corresponding to class 3 has been interpolated (Figure 5-14).

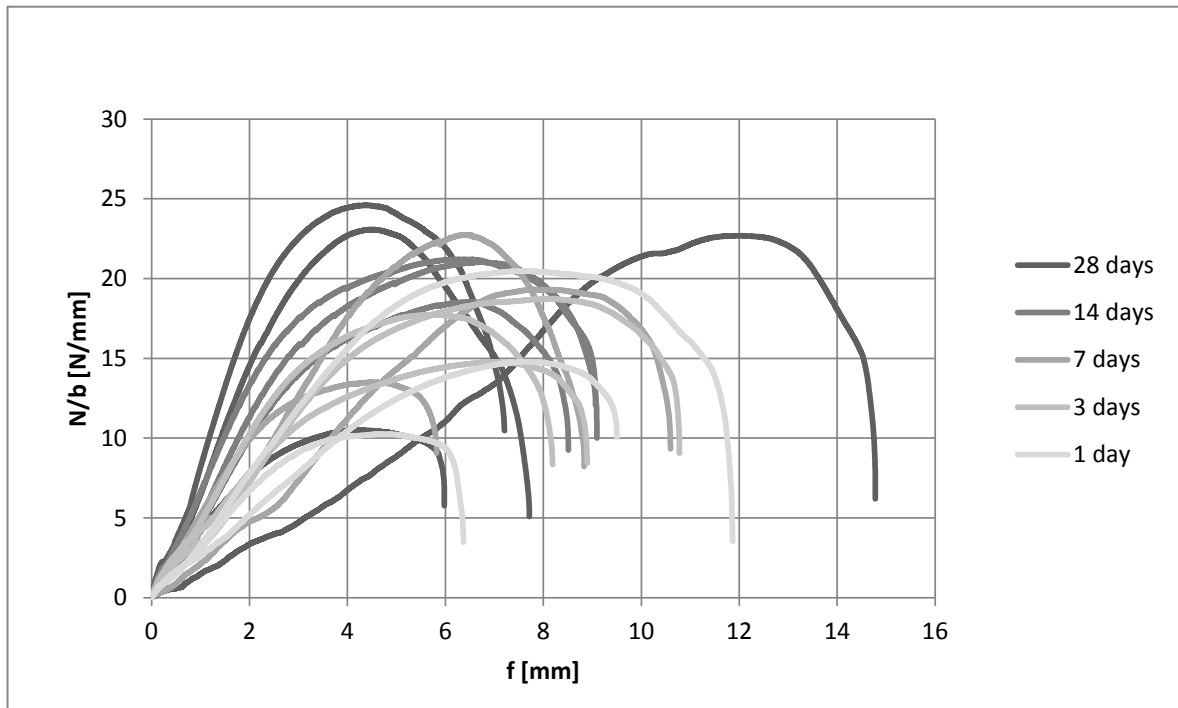


Figure 5-12: Results: shear between the layers test. Part 1.

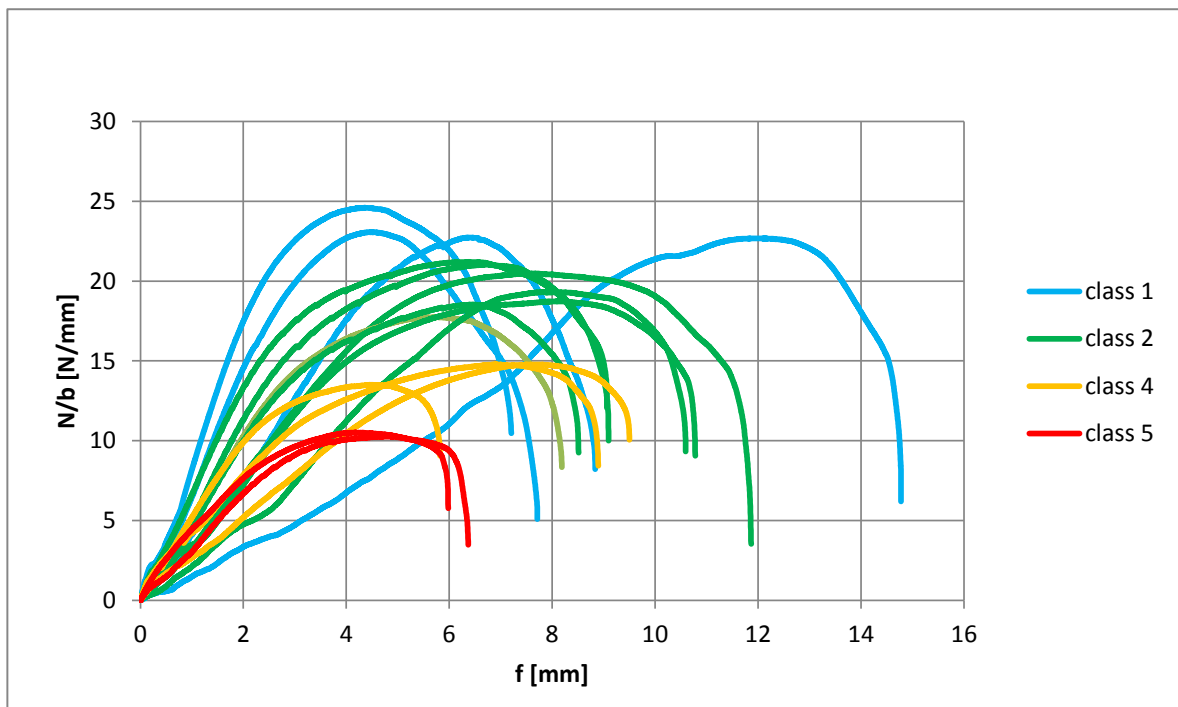


Figure 5-13: Results: shear between the layers test. Part 2.

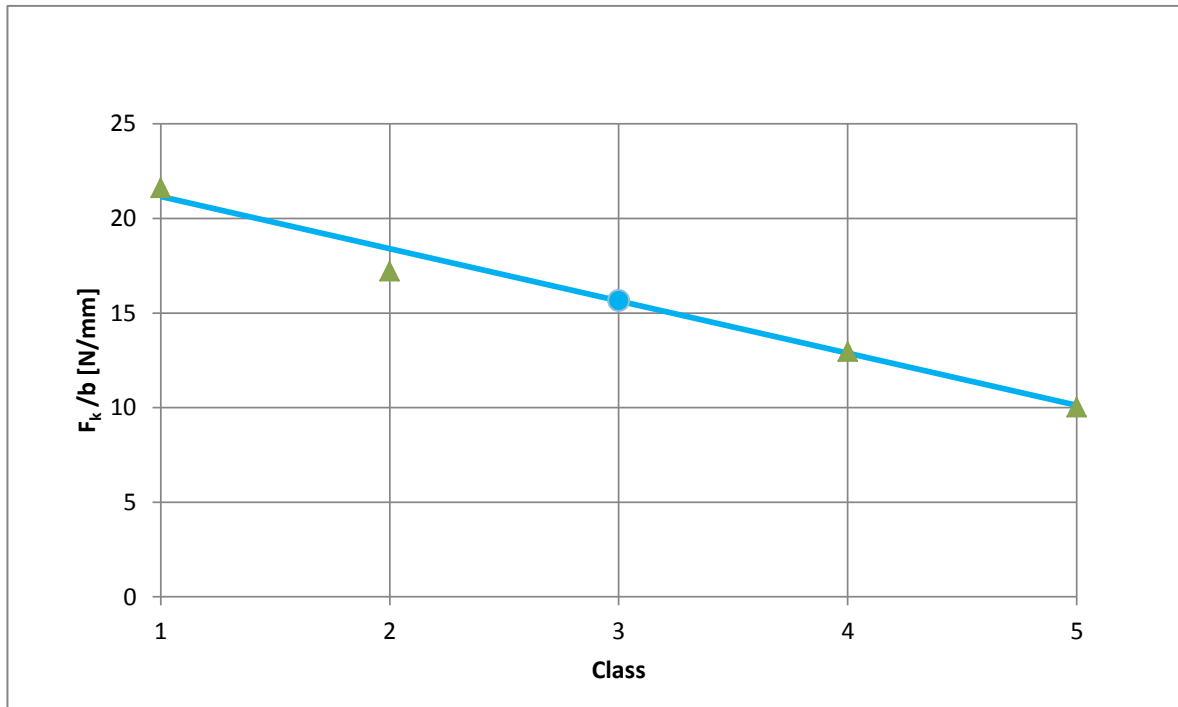


Figure 5-14: Shear between the layers test: interpolation for class 3.

Resultant values of characteristic strength are shown in table 5-10.

Class	Count	$\left(\frac{F_u}{b}\right)_{mean}$	CV	$\frac{F_k}{b}$
[-]	[pieces]	[N/mm]	[%]	[N/mm]
1	4	23.3	3.9	21.6
2	7	19.6	6.9	17.2
3	0	-	-	15.6
4	3	14.4	5.1	13.0
5	2	10.4	1.7	10.0

Table 5-10: Results: shear between the layers test.

5.1.8 Shear across the print

Two modes of failure were observed during the shear tests across the print direction. Relatively well cured specimens (14 days or older) showed predictable behaviour that ended with rupture in each test series, while the fresh ones were subject to a more complex phenomenon consisting of yield in the notched section and delamination of the layers around it. The mechanism can be distinguished in the force-displacement graph (Figure 5-15): fracture is represented by a sudden plunge of the curve, while delamination leads to a change in slope.

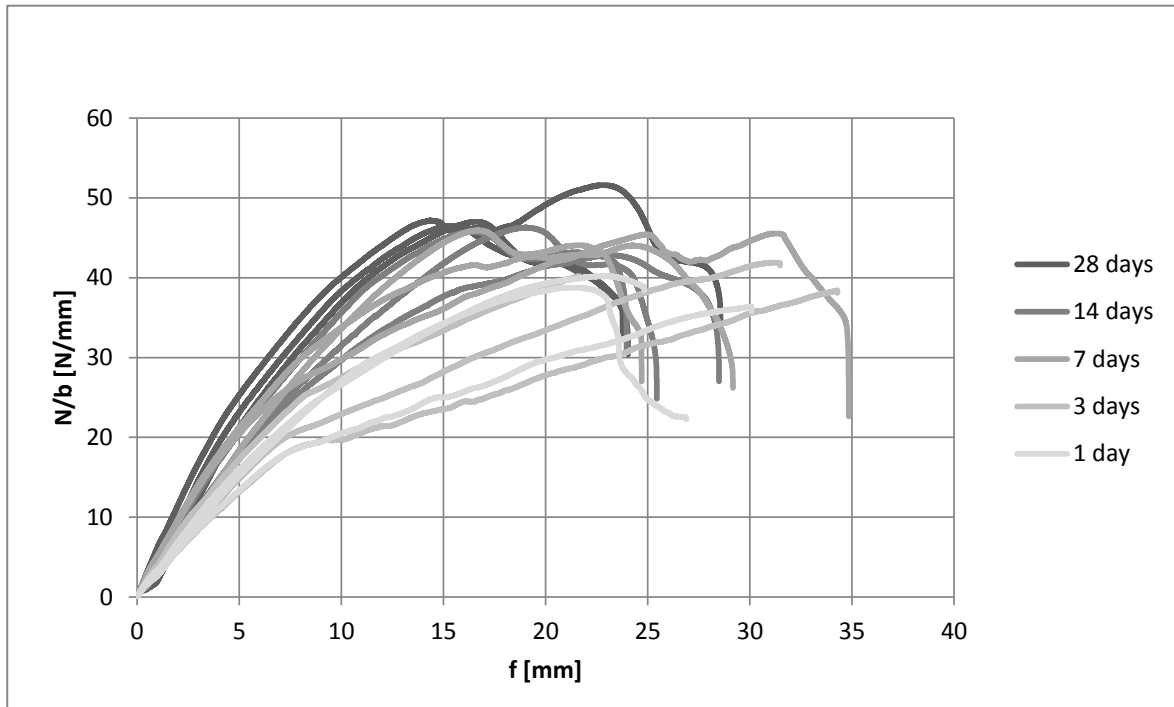


Figure 5-15: Results: shear across the print test.

As a consequence, two different methods have been applied to determine the strength value:

- For specimens that did not show signs of delamination, strength has been defined by the first peak in force-displacement graph.
- For specimens that failed due to delamination of the layers, the yield strength has been defined as a force at the yield strain, i.e. the intersection point of the elastic-plastic model lines (Figure 5-16).

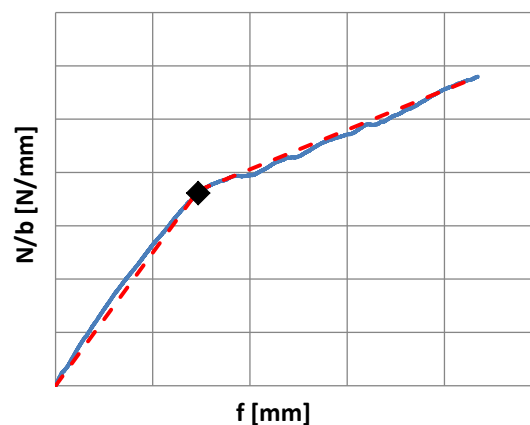


Figure 5-16: Shear test: yield due to delamination.

Table 5-11 contains the results obtained with the assumption of notched section width equal to 5 mm.

Class	Count	$\left(\frac{F_u}{b}\right)_{mean}$	CV	$\frac{F_k}{b}$
[days]	[pieces]	[N/mm]	[%]	[N/mm]
1	3	32.0	40.6	7.4
3	3	20.7	15.4	14.6
7	3	37.4	35.4	12.4
14	3	45.0	4.1	41.5
28	3	46.9	0.8	46.2

Table 5-11: Results: shear across the print test.

5.1.9 Creep test

Graphs representing the creep tests results can be found in Appendix B. In general, the material proved to creep less as it cures over time. Only the test of a 3 days old plastic does not follow this pattern – the output strains for this series are relatively higher, which raises suspicions about the dynamics of changes in viscoelasticity of the material in the first days. Power law has been chosen as a numerical scheme for representing creep as it provides robust input data for computational analysis:

$$\dot{\epsilon} = A\sigma^n t^m \quad (5.10)$$

Values of A, n, m have been determined with the use of basic stress-strain relationships for axial tension combined with least square regression, based on the dimensional assumptions similar to standard tensile tests. Relatively consistent results ($CV \approx 10\%$) are achieved with constant $n = 3.2$, $m = 0.32$ and A varying as follows:

Age	A	CV
[days]	[-]	[%]
1	$8.21 \cdot 10^{-6}$	4.4
3	$1.02 \cdot 10^{-5}$	9.2
7	$6.52 \cdot 10^{-6}$	9.8
14	$5.56 \cdot 10^{-6}$	7.2
28	$3.59 \cdot 10^{-6}$	13.2

Table 5-12: Results: creep test.

5.1.10 Thermal sensitivity tests

Being a soft thermoplastic, the investigated material is subject to changes under thermal loads. Test results clearly indicate this dependency: the specimens exhibit a drastic drop in mechanical properties as the temperature rises. Although severe in effects, the relationship is independent of the age of print. Quantitative description of the phenomenon has been provided by the means of change in tensile modulus and tensile strength. Following steps have been taken:

1. Ultimate force has been derived from each force-displacement data.
2. Force-displacement data has been cut at the lowest displacement corresponding to the ultimate force.
3. Stiffness modulus has been calculated based on eq. 5.5.
4. Both ultimate force and tensile moduli per each temperature have been compared with each other and with reference values provided by the standard tensile test (Figure 5-17, Figure 5-18).
5. Functions describing the temperature-relative stiffness and temperature-relative strength relationships have been determined with the least square fitting method.

It needs to be highlighted that the test of 14 days old material has been run on a different machine than usually (Zwick Z10 instead Zwick Z100), which resulted in distortion of the results (toe effect and possible slippage of the specimen). Therefore, the output of this test, although shown in the graphs, has not been taken into account in further analysis.

Temperature-relative stiffness function:

$$E_{rel}(T) = e^{-0.036T-0.74} \cdot E_{20} \quad (5.11)$$

$$CV = 8.4\%$$

Temperature-relative strength function:

$$f_{u,rel}(T) = (1.21 - 0.0105T) \cdot f_{u,20} \quad (5.12)$$

$$CV = 5.6\%$$

Where:

- E_{20} - Young's modulus at room temperature
- $f_{u,20}$ - ultimate strength at room temperature
- T - temperature of the material

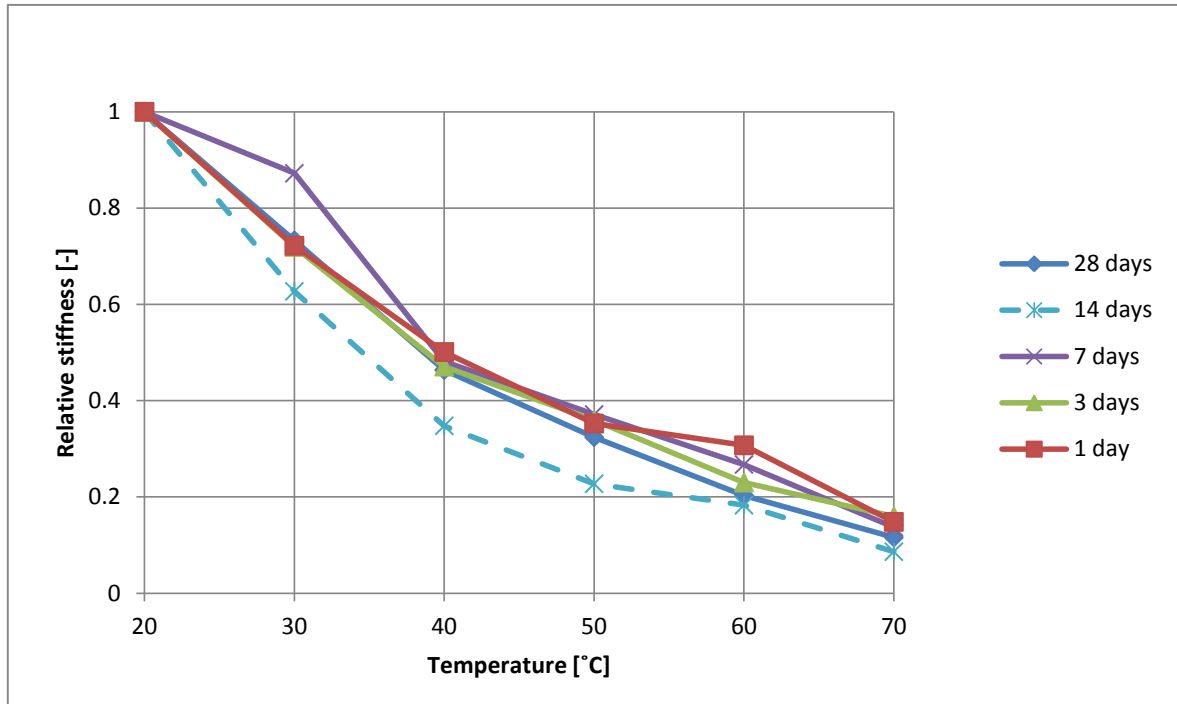


Figure 5-17: Results: influence of the temperature on tensile stiffness.

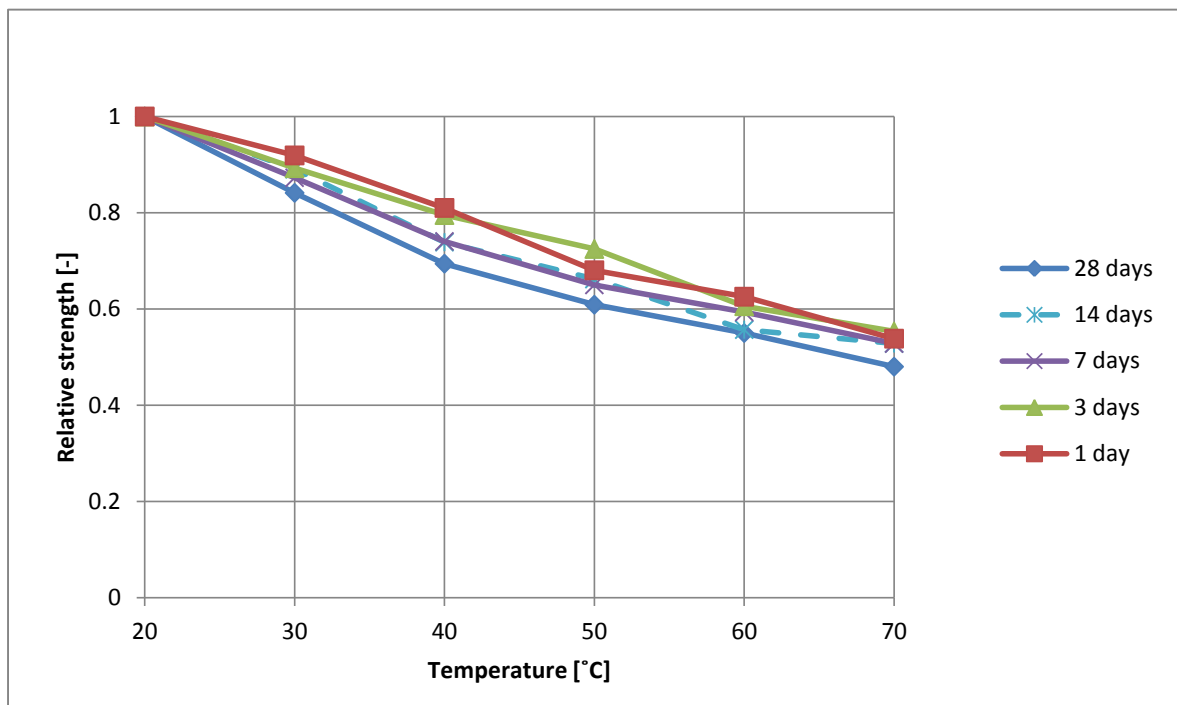


Figure 5-18: Results: influence of the temperature on tensile strength.

5.1.11 Summary

A wide range of conclusions can be drawn from the test results presented in Section 5.1. Following observations can be made besides and based on the quantitative description of the material properties elaborated on previous pages:

- The material gets stiffer, stronger and more brittle as it cures with time.
- In general, it has relatively low stiffness to strength ratio in all directions and modes of deformation.
- Highly orthotropic behaviour has been observed, with rate of orthotropy depending on the quality of printed product – the lower the quality (larger distortions of the print line), the lower cross-sectional properties in the transverse direction.
- Tensile properties of the cross sections depend on testing speed - strength and stiffness grows as the strain rate is increased.
- In longitudinal direction the tensile test results are consistent, but print quality affects bending behaviour.
- In transverse direction all results are widespread due to variable print quality and subsequent variability in geometrical properties.
- Shear strength between the layers is defined by the ultimate stress and depends mostly on the print quality, across the layers it is determined by age of the material and refers to either the ultimate stress (fracture) or yield stress (delamination of the layers).
- Viscoelastic nature of the compound is distinguishable, it decreases as the material ages.
- The material is highly sensitive to the changes of temperature – it loses around 85% of its initial tensile modulus and 50% of initial strength after heating from 20°C to 70°C.
- The quality issue has been embraced to certain extent by introduction of the cross section classification and related multiplication factors that are meant to unify the results, although larger number of specimens of each class should be tested in order to confirm the applied values statistically.

5.2 3D printed mould tests

Practical tests of the 3D printed forms yielded valuable information about the research subject. Analysis of the output data and observations made during testing resulted in both quantitative and essential qualitative conclusions. Similar to basic property tests, the influence of polymer age and print quality is visible. What is more, concrete mixture type and related casting process do also affect the final product. In this section the formwork tests are treated as completely separate from material property tests, while the relationship between both groups of experiments is investigated in further chapters of this document.

5.2.1 Single layer moulds

Test setup for single-layer forms has been explained in Section 4.5.1. In this section only the final numbers are discussed, while their derivation from the raw data is elaborated in Appendix

I. Tables 5-13 to 5-18 present the average out-of-plane deflections of the form walls analyzed separately per each age group. Three most relevant measurement steps are discussed: the outer surface of the form right after casting (instantaneous deformation), after more than 24 hours from casting (when the movement due to creep and concrete setting has stabilized), and finally, after demoulding of the hardened block.

Some global trends can be distinguished rather clearly from the tables. Firstly, the deflections after setting of the concrete are lower than right after casting by 0.3 mm on average. This suggests the existence of negative creep within the system, which can be the consequence of several factors such as gradual decrease in concrete pressure due to setting of the mixture or stabilization of the pressure that is initially distorted due to compaction. No in-depth research on this issue has been done due to scope and resource limitations. Worth highlighting is the fact that positive creep has not been observed at any stage of the tests. Digital reconstruction of the concrete blocks after removing the forms shows slightly lower (0.2 mm on average) out-of-plane deflections compared to the values observed on hardened yet unmoulded specimens. The reason for this phenomenon is hard to identify, it can only be supposed that it stems from variability in form wall thickness or differences in measurement technique (glued target points vs. projected target points, see Appendix I).

In general, the results vary per each series and do not follow a clear pattern that would refer exclusively to either of the parameters considered as most relevant in the material test stage, namely age or print quality. The latter varies not only between the specimens, but even between sections of their walls. As a consequence, it is impossible to define its representative value and systematize the influence of each factor. It can only be concluded that the global member stiffness is a resultant of them both.

Regarding the influence of infill type, the numbers in Tables 5-17 and 5-18 tend to confirm the assumption of lower pressure exerted by self-compacting concrete compared to regular mixtures, although no firm statements can be built based on one series of measurements.

Last qualitative conclusion to be discussed in this section is variability of the results and precision of the measurement method. As can be easily deduced, standard deviation is notably higher for the series consisting of freshly printed (1 day and 3 days old) forms, which correlates well with the statistical distribution of the material property test results. On the other hand, the error of measurement technique defined as a mean root mean square error is uniform and oscillates around the value of 0.1 mm.

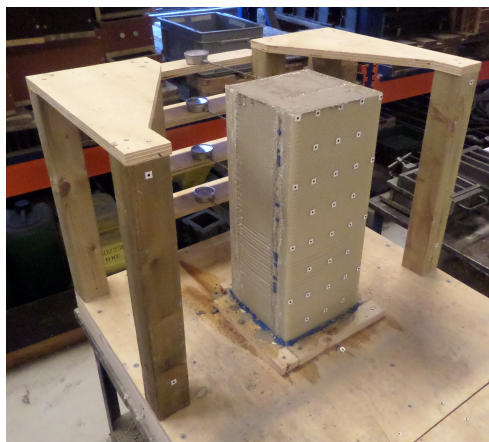


Figure 5-19: Casting of a single layer form.

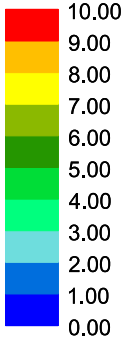
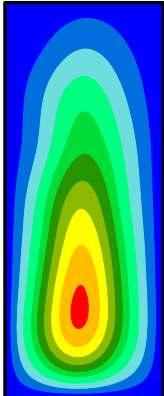
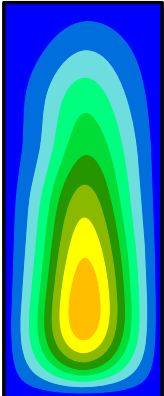
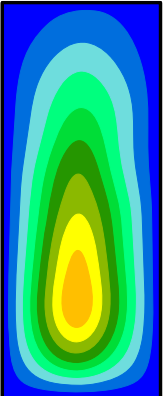
Measurement	0 hrs from casting	>24 hrs from casting	after demoulding
displacement [mm] 			
Max. deflection [mm]	9.18	8.72	8.65
Global st. dev. [mm]	0.650	0.601	0.584
Method error [mm]	0.090	0.100	0.073

Table 5-13: Results: deformations of single layer moulds, 1 day old.

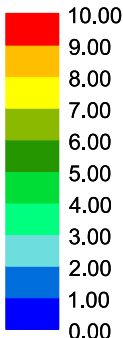
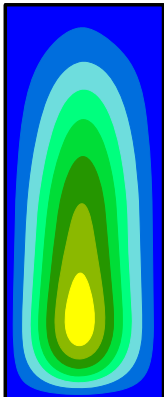
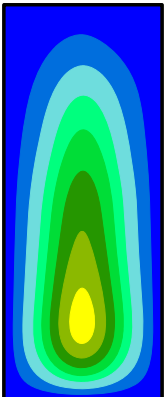
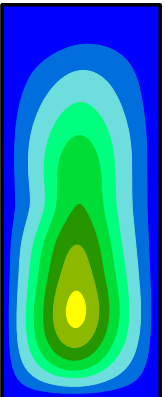
Measurement	0 hrs from casting	>24 hrs from casting	after demoulding
displacement [mm] 			
Max. deflection [mm]	7.48	7.24	7.27
Global st. dev. [mm]	0.631	0.644	0.546
Method error [mm]	0.100	0.055	0.072

Table 5-14: Results: deformations of single layer moulds, 3 days old.

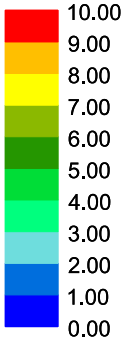
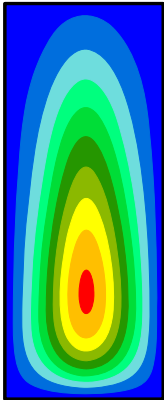
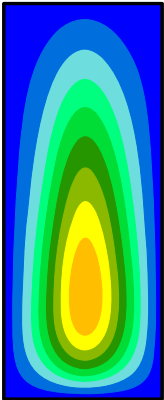
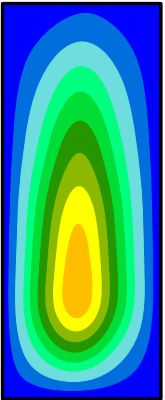
Measurement	0 hrs from casting	>24 hrs from casting	after demoulding
displacement [mm] 			
Max. deflection [mm]	9.19	8.87	8.66
Global st. dev. [mm]	0.274	0.313	0.293
Method error [mm]	0.125	0.116	0.066

Table 5-15: Results: deformations of single layer moulds, 7 days old.

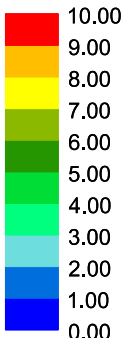
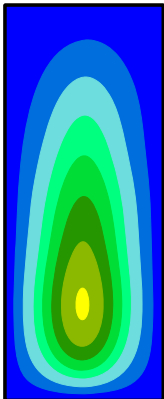
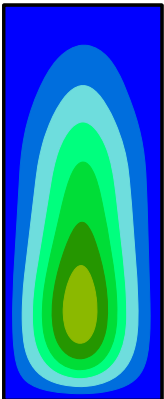
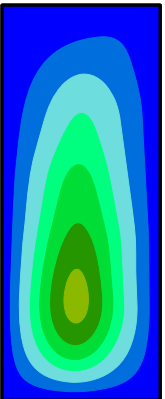
Measurement	0 hrs from casting	>24 hrs from casting	after demoulding
displacement [mm] 			
Max. deflection [mm]	7.12	6.70	6.26
Global st. dev. [mm]	0.421	0.530	0.470
Method error [mm]	0.100	0.086	0.088

Table 5-16: Results: deformations of single layer moulds, 14 days old.

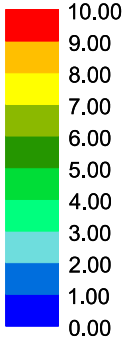
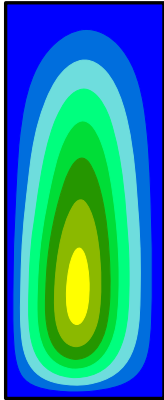
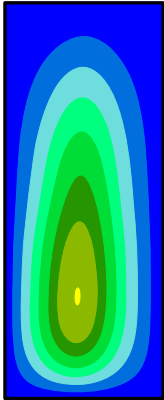
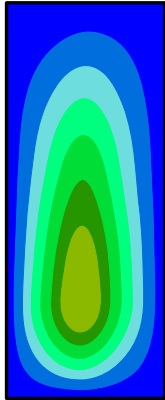
Measurement	0 hrs from casting	>24 hrs from casting	after demoulding
displacement [mm] 			
Max. deflection [mm]	7.36	7.02	6.87
Global st. dev. [mm]	0.261	0.251	0.369
Method error [mm]	0.105	0.121	0.094

Table 5-17: Results: deformations of single layer moulds, 28 days old.

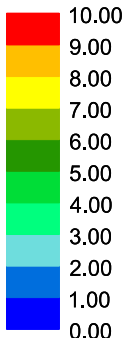
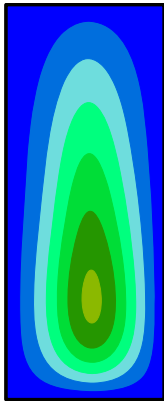
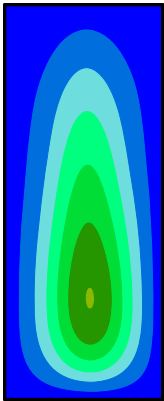
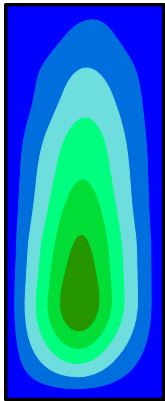
Measurement	0 hrs from casting	>24 hrs from casting	after demoulding
displacement [mm] 			
Max. deflection [mm]	6.24	6.03	5.67
Global st. dev. [mm]	0.226	0.202	0.297
Method error [mm]	0.099	0.075	0.088

Table 5-18: Results: deformations of single layer moulds, 28 days old, SCC.

Two relevant qualitative observations were made during the experiments with single layered pieces. Firstly, as already mentioned, the print quality was good enough to hold the mixture inside the form during setting of the concrete, with only incidental small leakages through the walls. On the other hand, in several cases noticeable amounts of water ran out of the setup through the bottom of the specimen, which is shown in Figure 5-19. This issue, although not investigated further in this research project, might be a subject for the future.

5.2.2 Truss-like walled mould

Table 5-19 presents measurement results of a $600 \times 800 \times 200$ mm concrete block cast in a 28 days old prismatic form consisting of 30 mm thick, built-up, multilayer walls (Figure 4-5). Only demoulded piece has been examined due to lack of technical capability to measure the deformations of the inner face of the wall during casting. The results are statistically relevant as both standard deviation within the data series as well as mean root mean square error of photogrammetry process are relatively low. As can be seen in Figure 5-20, only minor bottom leakage occurred during setting of the concrete and no holes or serious print faults have been spotted. Extraordinary effort put to demoulding needs to be mentioned: both cutting the multi-layered walls apiece as well as tearing the mould off is laborious and bears high risk of damaging the concrete surface. Finding a practical solution to this issue is recommended.

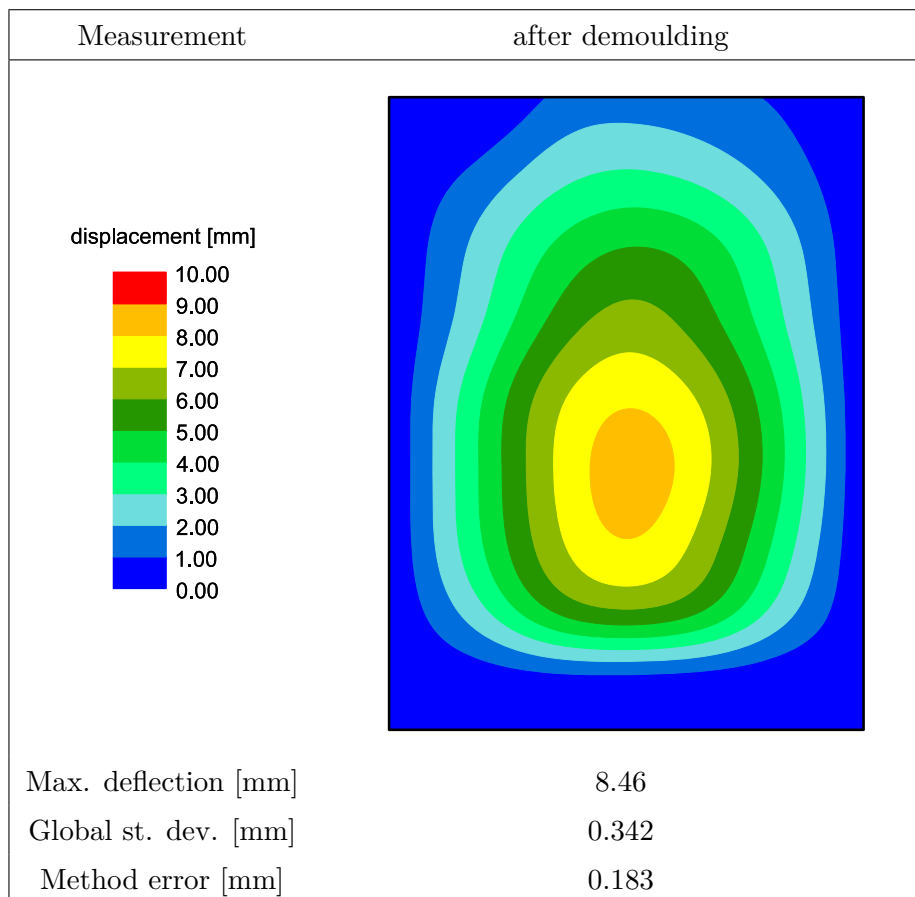


Table 5-19: Results: deformations of a built-up mould, 28 days old.



Figure 5-20: Casting of a built-up mould.

5.3 Concrete property tests

Concrete has been examined for two properties: compressive strength and water absorption. With an objective to evaluate quality of the concrete cast in the 3D printed forms, strength test output has been compared with recommendations of Bamforth et al. (2008) as well as with the results obtained for reference cubes made with the same mixture, while the latter mentioned parameter has been plotted against the data collected in work of Ployaert (2009).

5.3.1 Strength

Two types of mixtures have been tested. Tables 5-20 and 5-21 contain the results obtained for specimens made of a normal weight C20/25 concrete (Appendix G) and a self-compacting C28/35 mixture (Appendix H) respectively. The value of expected mean strength has been calculated based on the expressions provided by EN 1992-1-1, sub-clause 3.1.2(6):

$$f_{cm}(t) = (\beta_{cc}(t)) f_{cm} \quad (5.13)$$

$$\beta_{cc}(t) = \exp \left[s \left(1 - \left(\frac{28}{t} \right)^{0.5} \right) \right] \quad (5.14)$$

Where:

- f_{cm} - target mean strength, in this case $f_{cm,cube}$
- $f_{cm}(t)$ - mean compressive strength of concrete at 20°C at age of t days
- s - coefficient depending on cement type, for CEM III 42.5 $s = 0.20$

	3D printed form	Reference cubes	$f_{cm,cube(84)}$
Test series [MPa]	40.68	44.26	
	40.70	42.11	
	36.37	45.59	
	34.93		
	37.49		
	44.30		
Mean strength [MPa]	39.08	43.99	38.09
CV [%]	8.83	3.99	

Table 5-20: Results: normal weight C20/25 concrete compressive strength.

	3D printed form	Reference cubes	$f_{cm,cube(94)}$
Test series [MPa]	29.93	51.03	
	39.70	57.98	
	46.20	57.43	
	61.27		
	49.60		
	63.95		
Mean strength [MPa]	48.44	55.48	49.28
CV [%]	26.61	6.96	

Table 5-21: Results: self-compacting C28/35 concrete compressive strength.

The average compressive strength of concrete cast into 3D printed forms is in both cases almost equal to the values suggested by Eurocode and lower than the ones obtained in experiments performed on reference cubes. The source of differences between specimens made of the same mixture lies in the fabrication. Casting process for reference cubes provides high quality control, while cutting, although done as thoroughly as possible, resulted in dimensional imperfections of up to 5 mm. The influence of flaws can be clearly seen in the specimen shown in Figure 5-21, which exhibits crack pattern that is significantly different from the theoretical, double-cone shape reflecting uniaxial stress distribution. This phenomenon also explains relatively high dispersion of the results. Nevertheless, the obtained values are close to the target set by the code, which suggests that quality of the final concrete product is acceptable.



Figure 5-21: Concrete compressive strength test: crack pattern.

5.3.2 Water absorption

Table 5-22 contains the results of water absorption test. The percentage of pores within the concrete volume, expressed as *Abs* has been calculated based on the formula provided by NBN 15-215:1989 (Belgisch Instituut voor Normalisatie, 1989):

$$Abs = \frac{m_{wet} - m_{dry}}{m_{dry}} \cdot 100 \quad (5.15)$$

Where:

m_{wet} - mass of a specimen after immersion in water for at least 48 hours

m_{dry} - mass of a specimen after drying in temperature over 100°C for at least 72 hours

Similar to Section 5.3.1, the output of this test was distorted due to cutting imperfections, which in this case, however, should not affect relevance of the results. Calculated mean porosity of 4.42% lies on the outer bounds of dataset provided by Ployaert (2009), but it can be considered as plausible and justified.

	m_{wet} [g]	m_{dry} [g]	<i>Abs</i> [%]
Test series	8250	7890	4.56
	8710	8370	4.06
	7990	7655	4.38
	8330	7990	4.68
	8520	8160	4.41
Mean [%]			4.42
CV [%]			10.57

Table 5-22: Results: concrete porosity.

5.4 Summary

The outcome of repetitive material testing at five age intervals is a relatively comprehensive overview of its mechanical properties and their changes over time. Similarly, multiple mould deflection measurements provided complex information about behaviour of the formwork and infill in each stage of the test, starting from casting of the concrete and ending at the dimensions of a hardened element. In both cases, however, much confidence is lost due to varying print quality. For the time being, this phenomenon has been addressed by implementing the print quality classification system, but further, more extensive and better structured investigation would be desirable to fully embrace this issue. Nevertheless, collected information is sufficient to draw a range of quantitative and qualitative conclusions, from which the most important are:

- Mechanical properties of the 3D printed cross sections are time dependent with strength, stiffness and brittleness increasing as they cure.
- Their behaviour is highly orthotropic, with strong influence of print quality visible both in laboratory and structural tests.
- In general, ratio between cross-sectional stiffness and strength is relatively low in all directions and modes of deformation.
- Much higher strength and stiffness are found in the direction along the print line compared to transverse.
- Statistical dispersion of all results decreases as the time between fabrication and testing increases - the material becomes more stable.
- Although the material shows viscoelasticity and thermal sensitivity, for given scale, conditions and scope of application their practical influence was negligible.
- Currently, print quality within moulds is randomly distributed, which practically prevents it from being used as a design parameter (it can not be predicted prior to fabrication).
- In all concrete tests the quality was sufficient to prevent the mixture from leaking through the walls, only minor leakages through the bottom have been observed.
- Limited values of inverse creep were observed in the first hours after filling the moulds with fresh concrete - the reason of that phenomenon has not been clearly determined and should be further investigated.
- Using self-compacting concrete leads to lower mould deflections compared to the regular mixture.

Finally, based on the examination of demoulded concrete blocks it can be stated that for given scale and range of application the technology of 3D printed formwork does not affect the concrete strength and porosity to significant extent.

At this point one global conclusion regarding the process can be drawn. Since the current stage of development does not give the fabricator actual control over the print quality, class distribution within the test series was practically random. This lead to certain inconsistency within the test results, despite large number of series. Therefore, a larger, more structured test scheme would be desirable in order to embrace the influence of print quality in a more systematic manner.

Numerical analysis

Gaining the ability to predict stresses and deformations within the 3D printed moulds under loads is a main objective of this research project. As shown in the previous chapters, investigated technology provides a product with unique mechanical characteristics. Extruded cross sections, being defined by a combination of geometrical and material nonlinearities in two perpendicular directions, require a special-purpose approach. On following pages the reader can find a summary of existing knowledge concerning modelling of polymers followed by definition of given mechanical problem and detailed specification of proposed solution. The latter contains a framework for translating the material property test results into a concrete formwork prediction model.

6.1 Modelling of solid polymers - overview

Multiple material models allowing analysis of polymers have been developed over the years. Mathematicians and engineers adopted a wide range of formulations, from relatively simple, commonly known to highly sophisticated ones. Introducing all of them is virtually impossible, it would also require volume far larger than allowed for this document. Therefore, only a summary of five main types (linear elastic, hyperelastic, linear viscoelastic, plastic, viscoplastic) is presented in Table 6-1, followed by a more detailed description and usability evaluation of two models that are most relevant to the research task. For curious readers, comprehensive information on most popular material models used in numerical analysis of polymers including mathematical and mechanical foundations can be found e.g. in work of Bergstrom (2015).

6.2 Modelling of 3D printed cross sections

Chapter 5 shows that mechanical properties of the cross sections vary strongly depending on print direction, age of the material, temperature, loading time or strain rate. As a conse-

Type	Characteristics	Application in polymer modelling
Linear elastic	<ul style="list-style-type: none"> • based on Hooke's law • simple, efficient, easy to calibrate • applicable to small strains only 	solid polymers within small strain range
Hyperelastic	<ul style="list-style-type: none"> • nonlinear generalization of linear elasticity • expressed with strain energy density • suitable for large strains • efficient, rather easy to calibrate • does not capture viscoelasticity and hysteresis • starting formulation for most viscoelastic and viscoplastic models 	elastomers, foams
Linear viscoelastic	<ul style="list-style-type: none"> • extension of either elastic or hyperelastic model • captures linear forms of stress relaxation, creep and cyclic loading response 	mainly elastomers, thermoplastics in very small strain range
Plastic	<ul style="list-style-type: none"> • developed for modeling metals • numerically efficient, although sometimes difficult to calibrate • capable of capturing plastic nonlinearity • precise only for monotonic load 	thermoplastics under monotonic loading
Viscoplastic	<ul style="list-style-type: none"> • most accurate and complex group of material models • can be numerically expensive and difficult to calibrate • capable of capturing viscoelasticity and yield under small/large strains and monotonic/cyclic loading 	all polymers

Table 6-1: Overview of material models used in modelling of polymers.

quence, modelling of their behaviour was a complex task requiring a range of assumptions. As a first step, relevant simplifications in four fields have been identified and applied in order to narrow down the spectrum of possible approaches:

Structure type

With one dimension much smaller than the others, extruded walls were classified as shells. Modelling strategy has been tailored particularly to this element category.

Viscoelasticity

Based on the information presented in Section 5.2.1, it can be concluded that there is no practical reason for taking viscoelasticity into account in case of given application.

Material plasticity

Assuming negligence of the inverse creep that has been observed during the experiments, analysis of the forms can be treated as a single loading cycle. Consequently, potential yielding of the material can be considered as nonlinear elasticity.

Temperature influence

Since the scope of research is limited to relatively small scale experiments performed in controlled conditions, the influence of external climate and heat of hydration can be neglected.

With the above assumptions, the problem elaborated in this chapter can be defined as static analysis of an orthotropic, nonlinear elastic/hyperelastic shell structure under monotonic loads.

6.2.1 Stress/strain relationship within the cross section

Definition of an orthotropic shell element requires consistent information about its flexural and in-plane stiffness in each direction. This needs to be deduced from the experimental data by establishing a relationship between the results of tensile and bending tests. Following expressions relate strain rates in both deformation modes:

$$v_{\epsilon,normal} = \frac{v_{normal}}{l_{eff}} \quad (6.1)$$

$$v_{\epsilon,flexural} = \frac{6v_{flexural}t}{l^2} \quad (6.2)$$

Where:

- $v_{\epsilon,normal}$, $v_{\epsilon,flexural}$ - strain rate in tension and bending
- v_{normal} , $v_{flexural}$ - head speed in tensile and bending test
- l_{eff} - effective length of a tensile specimen
- l - distance between the supports in 3-point bending
- t - thickness of a specimen

Knowing the values of l_{eff} , l and assuming $t = 5$ mm one can find that for $v_{flexural} = 10$ mm/min, $v_{\epsilon,flexural} = 2.39$ %/min, which is closest to $v_{normal} = 2$ mm/min. Therefore, this speed is used globally in the project as it is most relevant.

Having coupled correspondent stress-strain data, an essential observation can be made. Assuming full linearity and considering a unit wide section of the 3D printed wall of any class and age, based on the results from Section 5.1 and basic continuum mechanics formulas, it can be proven that for any equivalent shell thickness the correspondent Young's moduli calculated from tensile and flexural test output are not equal in at least one direction:

$$(E_{x,normal} \neq E_{x,flexural} \vee E_{y,normal} \neq E_{y,flexural}) \forall t_{eq} \in \mathbb{R}_+ \quad (6.3)$$

Potential reasons for that are manufacturing and curing processes - further research of this issue is recommended for the future. Important fact for this project is that fourfold stiffness characteristics within a single cross section cannot be expressed by the means of a single layer shell. Consequently, application of a multilayer element is necessary.

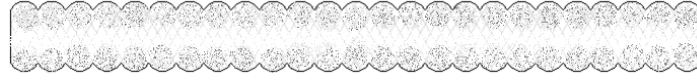


Figure 6-1: Unequal material curing throughout the cross section.

6.2.2 Hyperelastic fiber reinforced material model

As explained in Section 6.1, hyperelastic material models are suitable for modelling polymers thanks to their capability of working with large strains. However, applicability of most of them was severely limited due to their isotropic formulations. Within a narrow group of anisotropic models, maximum few are implemented into each software package. In case of CalculiX, which has been chosen for this project, the only available option was an extension of isotropic Neo-Hookean formulation complemented by exponential strenghtening terms imitating fiber reinforcement. This model, originally proposed by Holzapfel et al. (2000) for modelling of arterial walls, is given by the following mathematical form of strain energy density function:

$$U = C_{10}(\bar{I}_1 - 1) + \frac{1}{D_1}(J - 1)^2 + \sum_{i=1}^n \frac{k_{1i}}{2k_{2i}} \left(e^{k_{2i} \langle \bar{J}_{4i} - 1 \rangle^2} - 1 \right) \quad (6.4)$$

Where first two terms of the equation are the original Neo-Hookean expression with strain state represented by first invariant of the isochoric part of the right Cauchy-Green deformation tensor \bar{I}_1 and volume ratio J , with C_{10} and D_1 being independent elasticity parameters. Last term refers to n directions of fiber reinforcement with stiffness dependent on the fourth strain invariant in each fiber direction \bar{J}_{4i} and dimensionless constants k_{1i} and k_{2i} , $i = 1 \dots n$. Bracket $\langle x \rangle$ is understood as $\langle x \rangle = 0$ for $x \leq 0$ and $\langle x \rangle = x$ for $x > 0$, which in practice means that fiber reinforcement works only in tension. Since this is not true in case of the investigated material (fibers tend to work both in tension in compression), the original constitutive equation has been rewritten into:

$$U = C_{10}(\bar{I}_1 - 1) + \frac{1}{D_1}(J - 1)^2 + \sum_{i=1}^n \frac{k_{1i}}{2k_{2i}} \left(e^{k_{2i}(\bar{J}_{4i} - 1)^2} - 1 \right) \quad (6.5)$$

The change has been introduced into the material model code. Assuming single fiber direction and knowing that:

$$C_{10} = \frac{\mu}{2} \quad (6.6)$$

$$D_1 = \frac{2}{\kappa} \quad (6.7)$$

$$\bar{J}_{4i} = \lambda_{fiber}^2 \quad (6.8)$$

Where:

- μ - initial shear modulus of the composite matrix
- κ - bulk modulus of the composite matrix
- λ_{fiber}^2 - square of engineering stretch in the fiber direction, $\lambda_{fiber}^2 = (1 + \epsilon_e)^2$

Analytical solution for the uniaxial tension problem can be found rather straightforward. The method has been used for numerical reproduction of the laboratory tests based on force/displacement curve. After initial success with modelling of the specimen as a single layer entity under relatively small strains, convergence issues occurred while applying strains of $>2\%$ and implementing composite elements in order to handle the specific stiffness distribution explained in Section 6.2.1. Consequently, the material model could be considered reliable for only part of the scope investigated in this report.

6.2.3 Elastic material model

Elastic material model is the most basic formulation known and often used in finite element method. Thanks to its simplicity, robustness and universality it can be applied to various problems. It has one serious limitation with regard to scope of potential application: since it is based on linear Hooke's law, it loses physical sensibility under large strains. This could be considered as a potential drawback concerning the research assignment. However, the distortions are believed to affect mainly stress values under high compression, while for given problem category (shells with occasional large flexural strains) the effect should not be relevant (Dhondt, 2004, 2015). Consequently, the St.Venant-Kirchoff material model, as it is often called, has been recognized as applicable.

Standard elastic orthotropic formulation has been adapted to reflect mechanical characteristics of the 3D printed members. While lacking information about Young's modulus in the direction normal to the plane, extruded walls can be considered as made of an unidirectional fiber-reinforced compound. These are often treated as transverse isotropic, i.e. symmetric around an axis, which in such cases is coincident with direction of the reinforcement. This means that in plane normal to the axis all properties are uniform. The consequent advantage of this special mode of orthotropy is the ability to define the element's stiffness matrix with 5 constants instead of 9. Its theoretical derivation can be found in available literature, e.g. (Bower, 2010). From the engineering perspective the most convenient way to express transverse isotropy is the inverse Hooke's law.

Starting from the widely known inverse stiffness matrix of an orthotropic material:

$$\underline{\underline{C}}^{-x} = \begin{bmatrix} \frac{1}{E_x} & -\frac{\nu_{yx}}{E_y} & -\frac{\nu_{zx}}{E_z} & 0 & 0 & 0 \\ -\frac{\nu_{xy}}{E_x} & \frac{1}{E_y} & -\frac{\nu_{zy}}{E_z} & 0 & 0 & 0 \\ -\frac{\nu_{xz}}{E_x} & -\frac{\nu_{yz}}{E_y} & \frac{1}{E_z} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{G_{yz}} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{G_{zx}} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{G_{xy}} \end{bmatrix} \quad (6.9)$$

Assuming transverse isotropy in the plane normal to axis x, following relationships can be found for transverse Young's modulus, in-plane shear modulus, shear modulus along the polar axis and Poisson's ratio along the polar axis:

$$E_y = E_z \quad (6.10)$$

$$G_{yz} = \frac{E_y}{2(1 + \nu_{yz})} \quad (6.11)$$

$$G_{xy} = G_{xz} \quad (6.12)$$

$$\nu_{xy} = \nu_{xz} \quad (6.13)$$

Four input parameters have been eliminated, yielding simplified compliance matrix:

$$\underline{\underline{C}}^{-1} = \begin{bmatrix} \frac{1}{E_x} & -\frac{\nu_{yx}}{E_y} & -\frac{\nu_{yx}}{E_y} & 0 & 0 & 0 \\ -\frac{\nu_{xy}}{E_x} & \frac{1}{E_y} & -\frac{\nu_{yz}}{E_y} & 0 & 0 & 0 \\ -\frac{\nu_{xy}}{E_x} & -\frac{\nu_{yz}}{E_y} & \frac{1}{E_y} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{2(1 + \nu_{yz})}{E_y} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{G_{xy}} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{G_{xy}} \end{bmatrix} \quad (6.14)$$

Linear orthotropic elastic model based on five independent material constants per layer has been preliminarily verified for numerical stability by reproducing arbitrary tensile and flexural laboratory tests. It proved to be reliable with convergence being achieved for every problem.

6.3 Implementation

Due to unique approach to modelling of the wall stiffness, procedure of building the input for finite element analysis, as well as the analysis itself, is not straightforward. A set of custom algorithms has been built to directly link the experiment results with characteristics of each element under the effect of external loads.

6.3.1 Software

CalculiX finite element analysis package has been chosen as the core computational tool for this project. Main reason for that was its open source formula that allows for viewing and modifying the code according to one's needs. What is more, its fully scriptable interface was preferred over often constrained commercial products. Input mesh for the analysis has been prepared with the use of Rhinoceros 5.0 with Grasshopper plugin combined with a powerful meshing application Gmsh. Whole procedure has been wrapped up with the use of Python programming language - further information on that aspect is presented in Section 8.3.

6.3.2 Analysis mode

Static, nonlinear analysis mode has been chosen because of predicted occurrence of both large strains (usually understood as $\epsilon > 1\%$) and large deformations (out of plane wall deflection larger than half of its thickness). Length of each step has not been specified as the automatic values generated by the software proved to be sufficient for finding the final result.

6.3.3 Element type

The reason for implementing multilayered shell as a default element has been explained in Section 6.2.1. The object of choice has been further specified as a 2nd order quadrilateral with reduced integration. Such elements are considered to be most efficient and yield most precise results for given problem category (nonlinear analysis of a walled structure with smooth result and bending as dominant deformation mode). What is important, they are not sensitive to such numerical distortions as shear locking or hourglassing, which provides robustness and reliability (Dhondt, 2015; Dassault Systèmes, 2010, 2011).

For computations in CalculiX each shell is expanded into a volume, which means that in practice an original 8-node quadrilateral element is treated as a 20-node brick (see Figure 6-2). As a result, the actual number of integration points in a S8R element is equal to the value correspondent to C3D20R, i.e. 8. Coordinates of these points stem from Gaussian quadrature rule and correspond to the corners of a prism created by offsetting each face of the original element by $0.5(1 - \sqrt{1/3})$ of its full dimension (Figure 6-3). In case of a composite shell element, a separate set of integration points is defined for each layer, which means that a single planar element is expanded into n volumetric ones, where n is the number of layers.

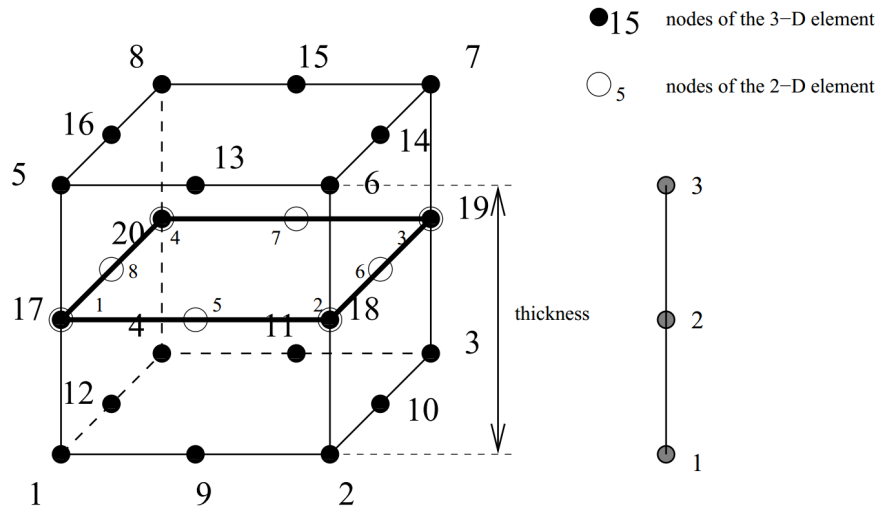


Figure 6-2: Expansion of a 2D 8-node element into a 3D brick element in CalculiX (Dhondt, 2015).

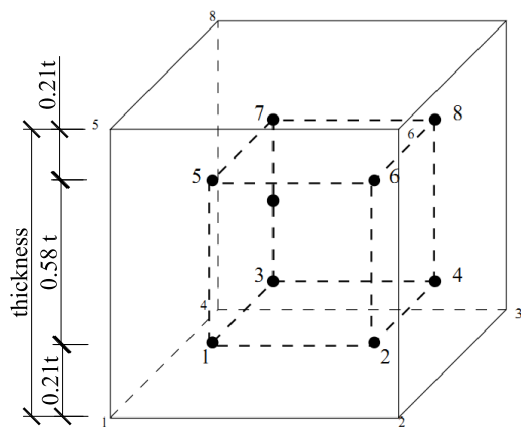


Figure 6-3: Distribution of integration points within C3D20R element. (Dhondt, 2015)

6.3.4 Material

Applicability of the two considered material models, fiber reinforced hyperelastic and elastic orthotropic is discussed in Sections 6.2.2 and 6.2.3 respectively. As explained, the former exhibits convergence issues at moderate strains, which could become troublesome during operation. Therefore, it has been decided to use the transversely isotropic extension of St. Venant-Kirchhoff approach, which proved to be simple and robust. Potential distortions related to application of linear theory to large deformations have been regarded as an acceptable trade-off for its intelligibility and adjustability in further stages of development.

To accommodate the inconsistency in cross-sectional stiffness reported in Section 6.2.1, a plane symmetric, triple layer element has been introduced (Figure 6-4). Default total thickness is set to 5 mm for a single layer wall, thicknesses t_1 and t_2 are determined individually for each element based on the relationship between values of EA , EI in longitudinal and transverse directions (the procedure is explained in Appendix J). Young's moduli per each layer are derived from cross-sectional stiffness formulas.

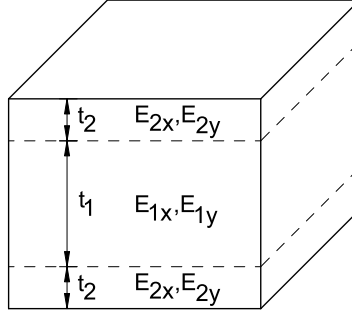


Figure 6-4: Plane symmetric, triple layer element.

Starting from the linear relationships within a plane symmetric continuum:

$$E_{normal}t = E_1t_1 + 2 \cdot E_2t_2 \quad (6.15)$$

$$E_{flexural} \frac{t^3}{12} = E_1 \frac{t_1^3}{12} + 2 \cdot E_2 \left(\frac{t_2^3}{12} + \left(\frac{t_1 + t_2}{2} \right)^2 \cdot t_2 \right) \quad (6.16)$$

Transformations lead to:

$$E_1 = E_{normal} \cdot \frac{t_1 + 2t_2}{t_1} - E_{flexural} \cdot \frac{(t_1 + 2t_2)^2 - t_1^2}{2t_1(t_1 + t_2)} \quad (6.17)$$

$$E_2 = \frac{E_{flexural} \cdot (t_1 + 2t_2)^2 - E_{normal} \cdot t_1^2}{4t_2(t_1 + t_2)} \quad (6.18)$$

Knowing Young's moduli and Poisson's ratios in all directions, missing values of shear moduli can be calculated from Equations 6.11 and 3.2-3.3.

Since orthotropic nonlinear elastic material model is not provided with the CalculiX package, a custom material nonlinearity implementation has been written on top of the existing linear elastic one. It iteratively solves given model and updates each element's E_{normal} and $E_{flexural}$ (consequently E_{1x} , E_{1y} , E_{2x} , E_{2y}) based on its strain state until the difference in the results of two subsequent iterations lies within tolerance. Default values of strain convergence criteria were set to $\Delta\epsilon \leq 0.01\%$ on average and $\Delta\epsilon \leq 0.05\%$ maximum in each direction:

$$\frac{1}{n} \sum_{i=1}^n |\epsilon_{k+1,i} - \epsilon_{k,i}| \leq 0.0001 \quad (6.19)$$

$$\max(\{|\epsilon_{k+1,i} - \epsilon_{k,i}| : i = 1 \dots n\}) \leq 0.0005 \quad (6.20)$$

Where n is the number of elements and $\epsilon_{k,i}$ is strain in i -th element after k -th iteration.

The relationship between Young's modulus and strain within each element is treated separately per each direction and can be expressed as:

$$E_{normal} = f_{normal}(\epsilon) \quad (6.21)$$

$$E_{flexural} = f_{flexural}(\epsilon) \quad (6.22)$$

Where $f_{flexural}(\epsilon)$ and $f_{normal}(\epsilon)$ should be understood as procedures applied in order to associate strain value ϵ with Young's moduli $E_{flexural}$ and E_{normal} derived from relevant force-displacement graphs presented in sections 5.1.3-5.1.6. Since CalculiX utilizes *total Lagrangian approach* for all calculations, force and displacement data is translated into second Piola-Kirchoff stress and Lagrangian strain. Based on the linear theory:

$$\sigma_{eng,normal} = \frac{F}{bt} \quad (6.23)$$

$$\epsilon_{eng,normal} = \frac{\delta}{L} \quad (6.24)$$

$$\sigma_{eng,flexural} = \frac{6FL}{4bt^2} \quad (6.25)$$

$$\epsilon_{eng,flexural} = \frac{6\delta t}{L^2} \quad (6.26)$$

Second Piola-Kirchoff stress and Lagrangian strain:

$$\sigma = \sigma_{eng}(1 + \epsilon_{eng}) \cdot (1 - \epsilon_{eng})^2 \quad (6.27)$$

$$\epsilon = \epsilon_{eng}\left(1 + \frac{\epsilon_{eng}}{2}\right) \quad (6.28)$$

Where:

- F - force in given node of the domain
- δ - displacement in given node of the domain
- t - assumed element thickness in the numerical model
- b - specimen width
- L - specimen length or span

The secant Young's modulus E_{normal} or $E_{flexural}$ at strain ϵ is interpolated from resultant stress/strain curve.

Any additional material calibration has been considered unnecessary as the analytical solution provided sufficiently precise values. This has been proved by reproducing the laboratory bending and tensile tests with satisfying results (see Sections 7.1, 7.2).

6.3.5 Material data

As shown in Section 6.2.1, the most relevant tensile test strain rate is 2 mm/min. In longitudinal direction, the material has been tested with such speed only after 7 and 28 days from printing. Missing data has been generated by multiplying the output collected under 10 mm/min by the mean value of stiffness and characteristic strength ratio between both mentioned speeds, which is equal to 0.87 and 0.91, respectively (calculated in Table 6-2).

Age [days]	$\frac{E_2}{E_{10}}$	$\frac{F_{yk,2}}{F_{yk,10}}$
7	0.87	0.90
28	0.88	0.92
Mean	0.87	0.91

Table 6-2: Results: longitudinal tensile test, influence of the strain rate.

6.3.6 Boundary conditions and loads

The moulds have been modelled as fixed with full translation and rotation restraint. This reflects the real setup, in which the specimens were held in place by a fixed perimeter (see Figure 5-19), assuming that the walls do not move inwards.

True values of concrete pressure were impossible to evaluate because of their dependence on various parameters ranging from composition and admixtures, through placing rate and compaction method up to formwork dimensions and material. What is more, most of research that has been done in this field is focused on larger scale and industrial casting methods. Consequently, an estimation has been made based on available resources. According to Proske and Graubner (2002), for vibrated concrete and cast height below 1 meter the load is hydrostatic. Assaad and Khayat (2006) discusses two particularly important factors related to self-compacting mixture. Firstly, it can be concluded that for slump flow values in the range comparable to SCC used in this research (~700 mm), pressure is nearly hydrostatic. However, it is shown that low filling rate can decrease maximum initial pressure by 15%. Taking into account these conditions, concrete loads have been defined as shown in Figure 6-5, with $\gamma_b = 24 \text{ kN/m}^3$. They have been translated into numerical input by dividing mesh vertically into 10 slices and calculating the pressure values for each of them.

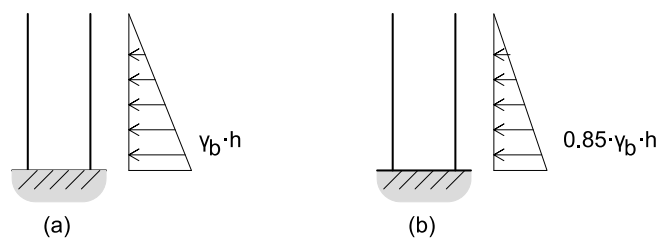


Figure 6-5: Modelling of the concrete pressure. a) normal concrete mixture b) self-compacting concrete.

6.3.7 Interpretation of the output

CalculiX generates the output for integration points that have been evaluated in the analysis. Since they are located *inside* the element, the results need to be translated into values that correspond to the convention used in this project.

Strain in each direction is derived from the output of every FE iteration. To accommodate the combination of normal and bending action within the element, ϵ for determination of normal stiffness is approximated by an average strain in the inner layers (in red on Figure 6-6), while ϵ for determination of flexural stiffness is taken as half of the difference between average strains in the outer faces of the element. These are calculated from the average strains in two outermost groups of integration points (in blue on Figure 6-6) multiplied by the factor related to their distance from the outer face.

$$\epsilon_{normal} = \left| \frac{\epsilon_{L3} + \epsilon_{L4}}{2} \right| \quad (6.29)$$

$$\epsilon_{flexural} = \left| \frac{\epsilon_{L1} - \epsilon_{L6}}{2} \cdot \frac{t_1 + 2t_2}{t_1 + \left(1 + \frac{\sqrt{3}}{3}\right)t_2} \right| \quad (6.30)$$

Where:

$$\epsilon_{Li} = \frac{\sum_{j=4i-3}^{4i} \epsilon_j}{4} \quad (6.31)$$

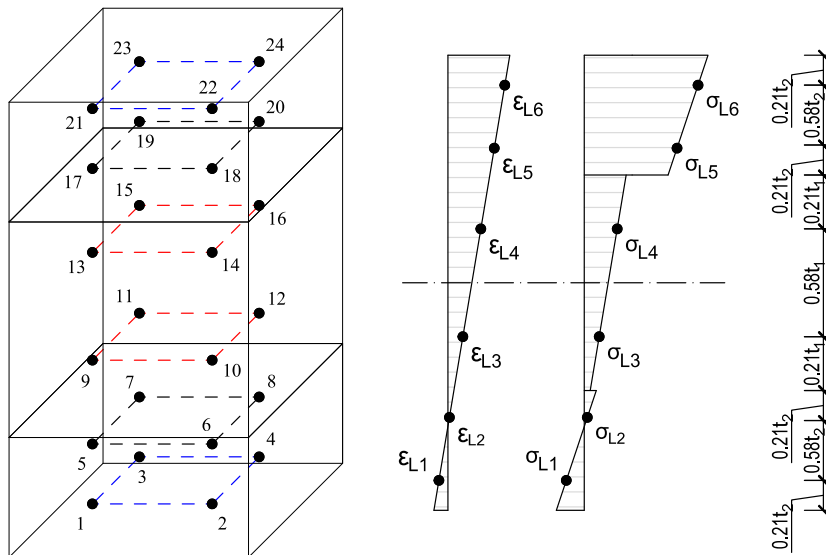


Figure 6-6: Distribution of integration points, strains and stresses over a triple layer shell. Left: isometry, inner layers of integration points in red, outermost layers in blue. Center: example strain distribution. Right: example stress distribution.

Resultant cross-sectional forces within the elements are calculated after the final iteration. Normal forces and moments are retrieved from the average stresses in point groups presented in Figure 6-6. Following universal formulas can be derived based on standard relationships for a triple layer plane-symmetric cross section under tension and bending:

$$\frac{N}{b} = \frac{\sigma_{L3} + \sigma_{L4}}{2} t_1 + \frac{\sigma_{L1} + \sigma_{L2} + \sigma_{L5} + \sigma_{L6}}{2} t_2 \quad (6.32)$$

$$\begin{aligned} \frac{M}{b} = & \frac{(\sigma_{L3} - \sigma_{L4}) \cdot \sqrt{3} \cdot t_1^2}{12} + \frac{(\sigma_{L1} + \sigma_{L2} - \sigma_{L5} - \sigma_{L6})(t_1 + t_2)t_2}{4} + \\ & + \frac{(\sigma_{L1} - \sigma_{L2} + \sigma_{L5} - \sigma_{L6}) \cdot \sqrt{3} \cdot t_2 \cdot \left(\frac{t_1}{2} + \frac{2t_2}{3}\right)}{2} \end{aligned} \quad (6.33)$$

Where:

$$\sigma_{Li} = \frac{\sum_{j=4i-3}^{4i} \sigma_j}{4} \quad (6.34)$$

Derivation of cross-sectional shear forces is a more complex task. Firstly, introduction of layered structure into an element results in virtual shear stresses between the layers. Additionally, directions of shell element's edges do not necessarily comply with its local coordinate axes. Taking these facts into account, precise calculation of shear stresses on expanded element's faces would be difficult and computationally expensive. Therefore, a simplified, conservative approach is proposed. Shear force in each direction is calculated for each set of integration points lying on the same (x, y) local coordinates (Picture 6-7):

$$\frac{S_{ij}}{b}(n) = \frac{(\sigma_{ij,12-n} + \sigma_{ij,16-n})t_1 + (\sigma_{ij,4-n} + \sigma_{ij,8-n} + \sigma_{ij,20-n} + \sigma_{ij,24-n})t_2}{2} \quad (6.35)$$

Where $\sigma_{ij,k}$ refers to shear stress in ij direction at k-th integration point. Representative value is taken as maximum of four resultant forces:

$$\frac{S_{ij}}{b} = \max \left(\left\{ \frac{S_{ij}}{b}(n) : n = 0...3 \right\} \right) \quad (6.36)$$

With print line coincident with local x axis, shear across the layers (symbolized by S_x) refers to S_{xy} and S_{xz} , while shear between the layers (S_y) is represented by S_{yx} and S_{yz} . Finally, knowing that for an orthotropic linear elastic formulation $\sigma_{ij} = \sigma_{ji}$, extreme forces in each shear mode can be defined as:

$$\frac{S_x}{b} = \max \left(\frac{S_{xy}}{b}, \frac{S_{xz}}{b} \right) \quad (6.37)$$

$$\frac{S_y}{b} = \max \left(\frac{S_{xy}}{b}, \frac{S_{yz}}{b} \right) \quad (6.38)$$

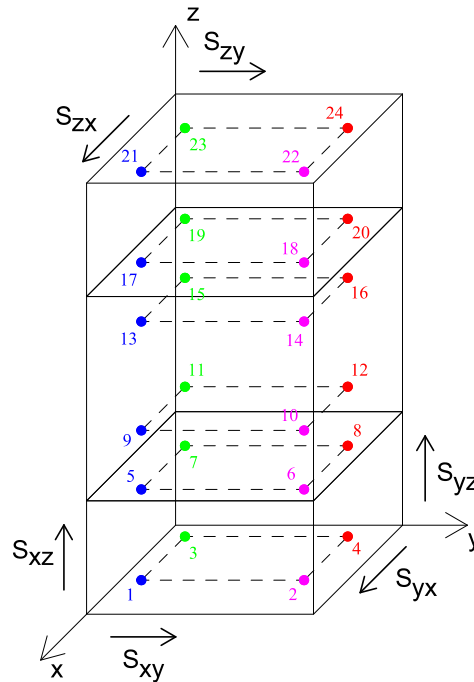


Figure 6-7: Element scheme for derivation of shear forces. Coloured: grouped integration points.

6.3.8 Summary

Information presented in this chapter constitutes a comprehensive base for the design/engineering activities reported on further pages. Preceded by listing of applied assumptions and explanation of the unique cross-sectional characteristics of the 3D printed walls, a choice has been made regarding:

Software

CalculiX FE package supported by Gmsh (mesher) and Rhino/Grasshopper (parametric geometry), wrapped together with Python language.

Analysis mode

Static, nonlinear analysis (large strains, large deformations) with automatic step length.

Element type

Composite, triple-layer 2nd order quad shell element with reduced integration (S8R) and automatically adjusted layer thicknesses t_1, t_2 .

Material model

Transversely isotropic mode of St.Venant-Kirchoff material model (Hooke's formulation for large strains), with strain-dependent nonlinear Young's moduli in each direction and deformation mode based directly on the experimental output.

Boundary conditions and loads

Fixed bottom supports, hydrostatic concrete pressure over the whole height of the mould ($\gamma_b = 24 \text{ kN/m}^3$), $0.85\gamma_b$ in case of self-compacting mixture.

Finally, a method for translating the simulation output into cross-sectional forces has been provided with necessary formulas. Flowchart explaining the process of numerical analysis proposed in this chapter is presented in Figure 6-8.

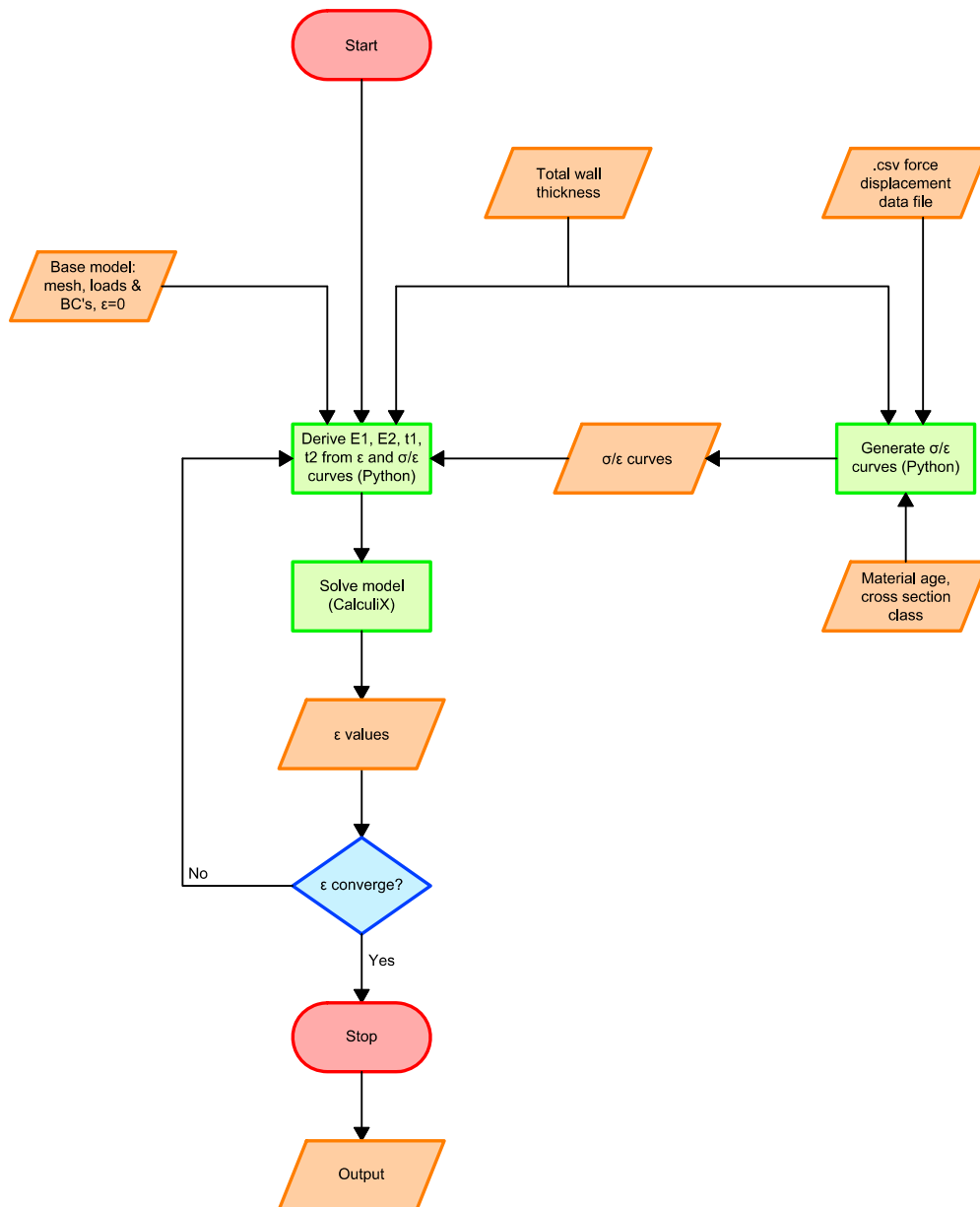


Figure 6-8: Flowchart of an individual finite element simulation.

Validation of the numerical model

In this chapter, virtual considerations presented in Sections 6.1-6.3 are verified against the experimental output. Four setups have been modelled for validation purposes. First two are standard property tests: tensile and three point bending, covered in Sections 7.1 and 7.2, respectively. They are followed by a single layer and trussed-wall mould simulation, which can be found in two further sections. All models have been built based on the foundations elaborated in previous chapter, with a triple-layered cross section being applied as standard for all cases. Deformations are used as main mean of comparison between each numerical model and experiment. Scope of investigation per setup was based on the expected range of application and input data limitations (strain range from 0 to 4 percent for most of property tests, full hydrostatic load in mould tests).

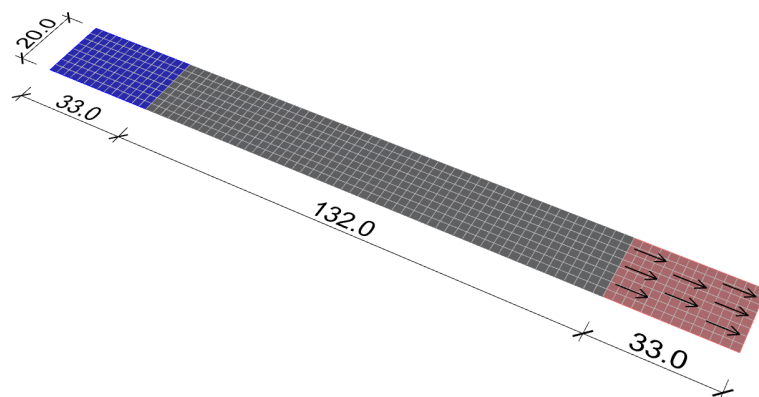


Figure 7-1: Numerical model of a tensile test. Blue: clamped zone. Grey: specimen area. Red: loaded zone.

7.1 Tensile test

Tensile test has been modelled as a narrow part of the specimen with loads and supports distributed over additional components attached to its both ends, which imitates clamps of the testing machinery (Figure 7-1). Chosen combinations of direction/age/class have been simulated, each combination in four load cases correspondent to expected strain of 1%, 2%, 3% and 4%. The results, expressed as relative displacement of specimen edges, are presented in Tables 7-1 to 7-3 together with comparison to the experimental stress-strain data. As can be seen, the difference between numerical output and expected values read from relevant curves does not exceed 5%.

Theoretical		Model, 7 days old		Model, 28 days old	
Strain [%]	Deflection [mm]	Deflection [mm]	Difference [%]	Deflection [mm]	Difference [%]
1.0	1.32	1.34	+1.60	1.34	+1.64
2.0	2.64	2.67	+1.18	2.67	+1.09
3.0	3.96	4.02	+1.45	4.01	+1.20
4.0	5.28	5.09	-3.54	5.35	+1.32

Table 7-1: Validation: longitudinal tensile test, classes 1-5.

Theoretical		Model, 7 days old		Model, 28 days old	
Strain [%]	Deflection [mm]	Deflection [mm]	Difference [%]	Deflection [mm]	Difference [%]
1.0	1.32	1.34	+1.52	1.30	-1.40
2.0	2.64	2.65	+0.41	2.59	-1.82
3.0	3.96	3.98	+0.48	3.89	-1.88
4.0	5.28	5.31	+0.49	5.14	-2.62

Table 7-2: Validation: transverse tensile test, class 1.

Theoretical		Model, 7 days old		Model, 28 days old	
Strain [%]	Deflection [mm]	Deflection [mm]	Difference [%]	Deflection [mm]	Difference [%]
1.0	1.32	1.30	-1.81	1.32	+0.18
2.0	2.64	2.56	-2.91	2.64	-0.16
3.0	3.96	3.85	-2.66	3.96	+0.03
4.0	5.28	5.15	-2.42	5.29	+0.11

Table 7-3: Validation: transverse tensile test, class 5.

7.2 Flexural test

Numerical representation of the 3-point bending setup has been built as an assembly of three components: a specimen of real dimensions and two halfpipes that represent the testing machinery (Figure 7-2). A boundary condition imitating sliding contact between the elements has been applied. A set of direction/age/class combinations has been compared with experimental force/displacement output under strains dependent on the length of each stress-strain dataset. The resultant differences do not exceed 10%, with only few values higher than 5%.

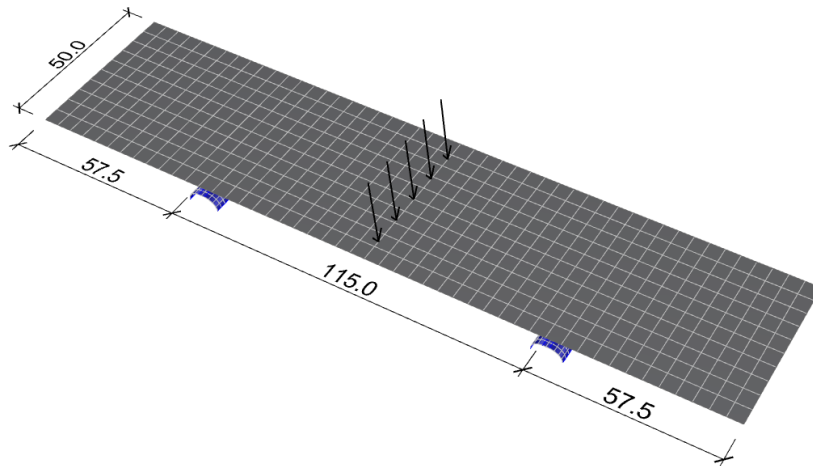


Figure 7-2: Numerical model of a flexural test. Blue: supports. Grey: specimen.

Theoretical		Model, 7 days old		Model, 28 days old	
Strain [%]	Deflection [mm]	Deflection [mm]	Difference [%]	Deflection [mm]	Difference [%]
1.0	4.18	4.19	+0.32	4.25	+1.75
2.0	8.36	8.15	-2.51	8.24	-1.43
3.0	12.54	12.06	-3.84	12.16	-3.09
4.0	16.73	17.31	+3.50	17.30	+3.43

Table 7-4: Validation: longitudinal flexural test, class 1.

Theoretical		Model, 7 days old		Model, 28 days old	
Strain [%]	Deflection [mm]	Deflection [mm]	Difference [%]	Deflection [mm]	Difference [%]
1.0	3.37	3.53	+4.63	3.58	+6.30
2.0	6.74	7.00	+3.80	7.09	+5.15
3.0	10.12	10.46	+3.39	10.57	+4.51
4.0	13.49	14.40	+6.75	14.54	+7.82

Table 7-5: Validation: longitudinal flexural test, class 5.

Theoretical		Model, 7 days old		Model, 28 days old	
Strain [%]	Deflection [mm]	Deflection [mm]	Difference [%]	Deflection [mm]	Difference [%]
0.5	2.55	2.60	+2.14	2.56	+0.36
1.0	5.10	5.21	+2.11	5.14	+0.83
1.5	7.65	7.87	+2.95	7.77	+1.58
2.0	10.20	10.60	+3.97	10.42	+2.19

Table 7-6: Validation: transverse flexural test, class 1.

Theoretical		Model, 7 days old		Model, 28 days old	
Strain [%]	Deflection [mm]	Deflection [mm]	Difference [%]	Deflection [mm]	Difference [%]
0.5	4.18	4.14	-0.88	4.26	+1.90
1.0	8.36	8.33	-0.45	8.52	+1.83
1.5	12.54	12.84	+2.37	13.29	+5.94

Table 7-7: Validation: transverse flexural test, class 5.

It needs to be highlighted that the assumption of span length equal to the axial distance between supports yields inaccurate results. As can be seen in Figure 7-3, loaded specimen undergoes deformation which results in shift of effective support towards the center. In case of given setup, this shift is equal to approximately 1.5mm at each side. Therefore, a corrected global value of span length used in this report is 112mm. Assuming linearity and knowing from Equation 5.7 that deflection under bending is proportional to the third power of span, this can be translated into roughly 7.6% reduction in flexural response:

$$\frac{EI_{112}}{EI_{115}} = \frac{112^3}{115^3} = 0.924$$

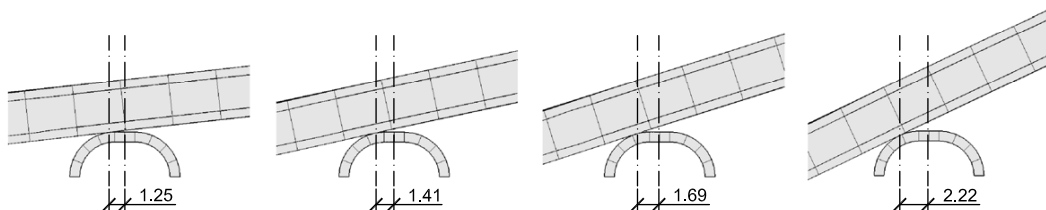


Figure 7-3: Specimen slippage during 3-point bending. From left to right: deformation corresponding to strains from 1% to 4%, distance in millimeters.

7.3 Single layer moulds

Figure 7-4 shows the mesh used to simulate casting of a standard single layer prismatic mould. Boundary conditions and loads have been applied according to the description in Section 6.3.6. Six cases presented in Section 5.2.1 have been analysed with an assumption of class 1 print quality, which is used as a default parameter facing the random class distribution explained in Section 5.2.1. Table 7-8 contains the information about absolute deflections retrieved from the numerical output, followed by the comparison to relevant experimental observations after demoulding. Differences between both sets are within range from -2mm up to +2mm. One pattern can be noticed: in the bottom part of each specimen real deformations are smaller than the ones resulting from computations.

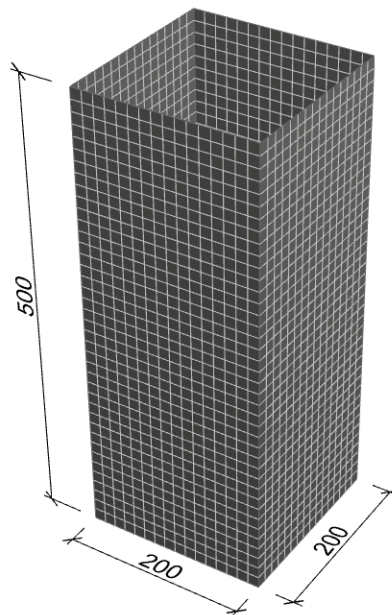


Figure 7-4: Numerical model of a single layer mould test.

7.4 Truss-like walled mould

Last setup modelled for the purpose of validation was a truss-walled form presented in Section 4.5.2. Similar to single-layered moulds, deformation measurements after demoulding of the specimen were compared with the FE analysis output. Figure 7-5 shows meshed geometry, the results can be found in Table 7-9. Correlation between the experimental and numerical results is rather high, although a discrepancy over 1 mm can be found in the bottom zone, similar to single-layered moulds.

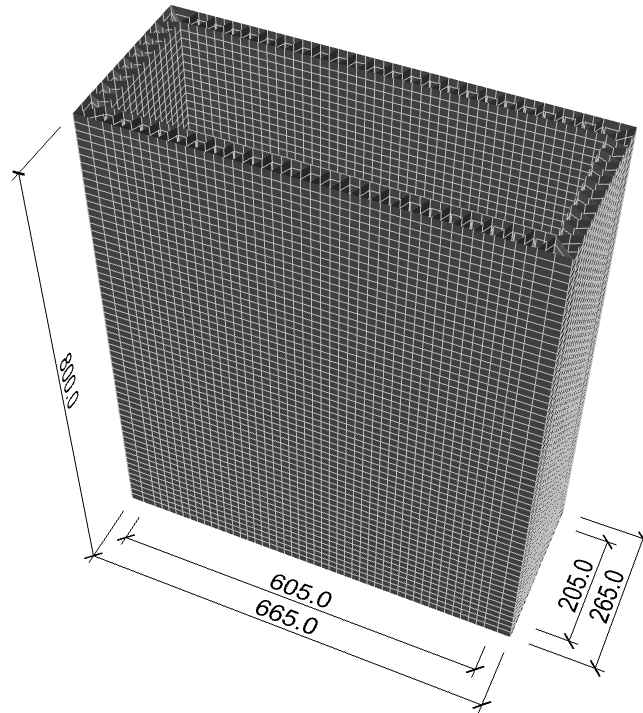


Figure 7-5: Numerical model of a built-up mould test.

7.5 Summary

Validation of the computational method based on reconstruction of the material property tests proves its accuracy of more than 95% in almost all cases. The results have been considered precise enough not to implement additional calibration.

Reproduction of mould tests generally proves correctness of the numerical approach, although locally the differences between model and reality are relatively significant (over a millimeter). This can be caused by the influence of print quality class (a general assumption of class 1 has been used for validation purpose), statistical distribution of the results or measurement errors. It needs to be highlighted that all comparisons involving concrete-filled components exhibit a pattern related to prediction of excessive deflection in the bottom zone of the specimen, which turns to be smaller in reality. The potential reasons for that are either the abovementioned modelling and measurement issues or overrestriction of the mould by perimeter fixings.

Mould age	1 day	3 days	7 days	14 days	28 days	28 days, SCC
Absolute deflection [mm]						
Max. deflection [mm]	8.18	8.20	7.18	6.88	6.71	5.90
Compared to experimental results [mm]						
Max. deviation (+) [mm]	0.99	1.76	1.37	1.36	1.26	1.31
Max. deviation (-) [mm]	-1.52	-1.23	-2.43	-1.05	-1.24	-0.98
Avg. deviation [mm]	-0.52	0.11	-0.81	-0.06	-0.30	-0.20

Table 7-8: Validation: deformations of single layer forms.

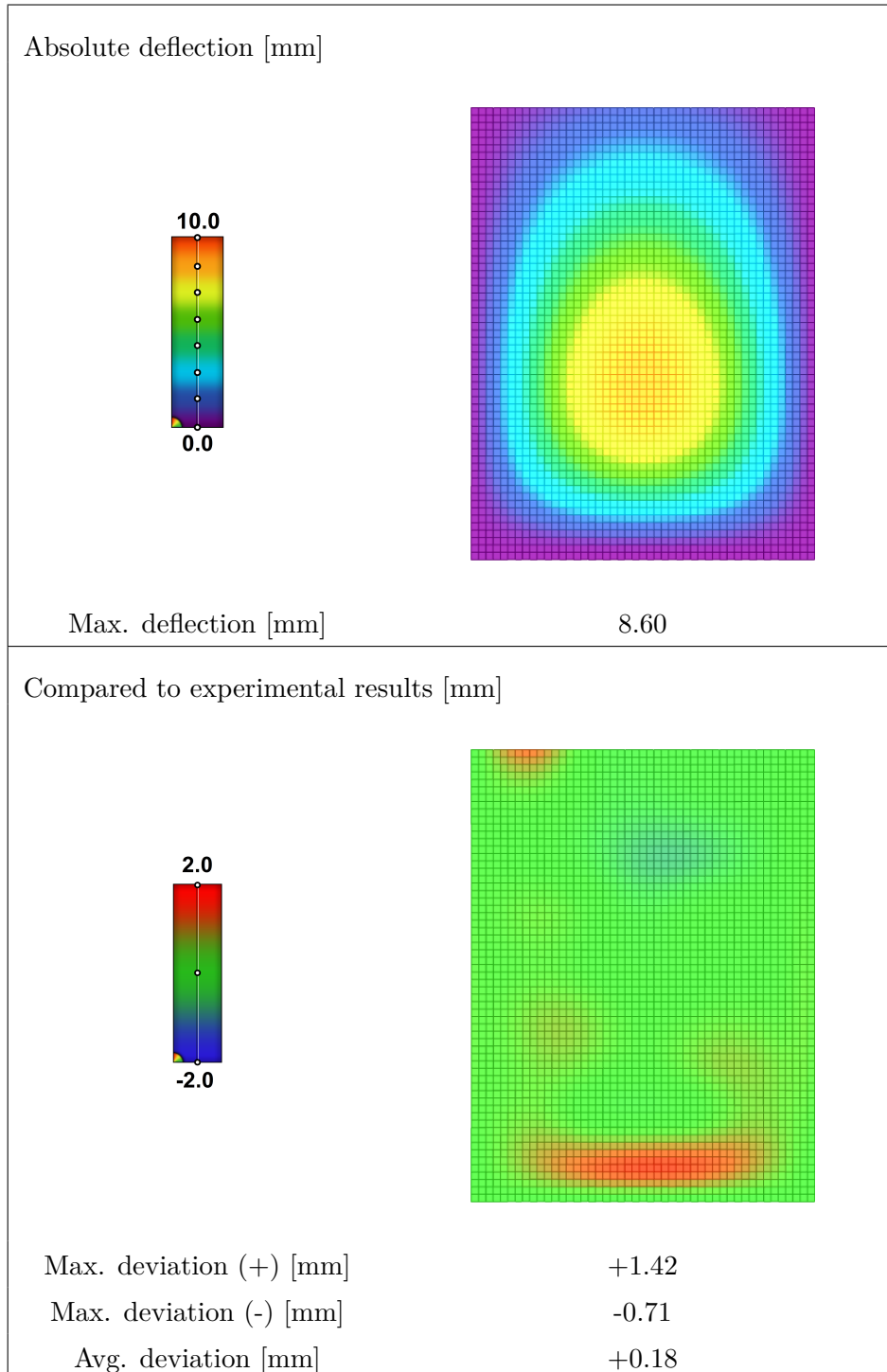


Table 7-9: Validation: deformations of built-up mould.

Chapter 8

Design methodology

3D printed moulds, due to their specific manufacturing process and mechanical characteristics, hardly comply to any existing design codes or manuals. A special-purpose approach is proposed in this chapter, which takes into account the unique properties of the product with particular focus on embracing its low stiffness to strength ratio. Its main objective is to achieve the desired concrete shape by creating the components with predefined negative deformation, which transforms into expected geometry after casting (Figure 8-1). Proposed methodology is based on the experimental results and computational technique presented in Chapters 5 and 6. This chapter has been divided into two main parts: Section 8.1 discusses the aspects that need to be taken into account as design criteria, while Sections 8.2 and 8.3 describe the design procedure itself and its implementation into the Rhino/Grasshopper software package. With particular interest in prediction of deformations, most attention is paid to this issue, while the other parameters are considered in terms of minimum requirements.

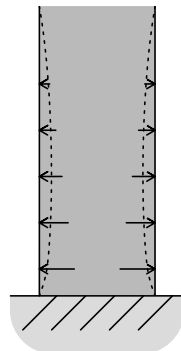


Figure 8-1: Predeformation-based design method principle. Dotted: predeformed mould shape. Continuous: geometry after casting.

8.1 Design criteria

8.1.1 Strength

Strength is a basic design criterion that needs to meet the minimum requirements. In case of 3D printed moulds modelled as composite shells this can be checked by translating stress state within the element into a set of cross-sectional forces (see Section 6.3.7) and comparing them with allowable design values:

$$\frac{F_{Ed}}{b} < \frac{F_{Rd}}{b} \quad (8.1)$$

Every design value $\frac{F_{Rd}}{b}$ can be expressed as a combination of a characteristic value $\frac{F_{yk}}{b}$ and safety factor γ_F :

$$\frac{F_{Rd}}{b} = \frac{F_{yk}}{b} \cdot \gamma_F \quad (8.2)$$

Characteristic strength values correspondent to class 1 cross section for each mode can be found in Sections 5.1.3-5.1.8. Full design value matrices for possible age/class combinations (based on Appendix C) are presented in Tables 8-1 to 8-5. No information about flexural strength in transverse direction is provided because of specimen slippage happening before yield during the experiments (see Section 5.1.6). Due to lack of insight into combinations of different stress modes, they are meant to be considered separately. However, investigation of this issue is recommended in future stages of the research.

	1 day	3 days	7 days	14 days	28 days
	$\left[\frac{N}{mm} \right]$	$\left[\frac{N}{mm} \right]$	$\left[\frac{N}{mm} \right]$	$\left[\frac{N}{mm} \right]$	$\left[\frac{N}{mm} \right]$
classes 1-5	29.3	29.8	34.8	36.1	39.5

Table 8-1: Characteristic strength: tensile longitudinal $\frac{F_{x,k}}{b}$.

	1 day	3 days	7 days	14 days	28 days
	$\left[\frac{N}{mm} \right]$	$\left[\frac{N}{mm} \right]$	$\left[\frac{N}{mm} \right]$	$\left[\frac{N}{mm} \right]$	$\left[\frac{N}{mm} \right]$
class 1	8.2	9.1	12.0	13.5	13.3
class 2	7.4	8.2	10.8	12.2	12.0
class 3	6.6	7.3	9.7	10.9	10.7
class 4	5.8	6.4	8.5	9.5	9.4
class 5	5.0	5.5	7.3	8.2	8.1

Table 8-2: Characteristic strength: tensile transverse $\frac{F_{y,k}}{b}$.

	1 day	3 days	7 days	14 days	28 days
	[N]	[N]	[N]	[N]	[N]
class 1	33.3	31.7	41.1	46.5	47.8
class 2	28.0	26.7	34.6	39.1	40.2
class 3	27.3	26.0	33.7	38.1	39.2
class 4	32.1	30.5	39.6	44.8	46.0
class 5	34.0	32.4	42.0	47.5	48.8

Table 8-3: Characteristic strength: flexural longitudinal $\frac{M_{y,k}}{b}$.

	1 day	3 days	7 days	14 days	28 days
	$\left[\frac{N}{mm} \right]$	$\left[\frac{N}{mm} \right]$	$\left[\frac{N}{mm} \right]$	$\left[\frac{N}{mm} \right]$	$\left[\frac{N}{mm} \right]$
classes 1-5	7.4	14.6	12.4	41.5	46.2

Table 8-4: Characteristic strength: shear across the layers $\frac{S_{x,k}}{b}$.

	1-28 days
	$\left[\frac{N}{mm} \right]$
class 1	21.6
class 2	17.2
class 3	15.6
class 4	13.0
class 5	10.0

Table 8-5: Characteristic strength: shear between the layers $\frac{S_{y,k}}{b}$.

Design safety factors are determined based on multiple criteria including i.a. prediction accuracy of the imposed loads, material strength estimate, exposure to the environmental effects, consequences of failure and the cost of over-dimensioning to achieve certain factor. For the scale and scope of application covered by this research project, including controlled production and testing environment, potential low damages and virtually full human safety in case of failure, it is considered acceptable to use a global safety factor equal to unity:

$$\gamma_F = 1.0$$

However, in case of using the technology under more hazardous conditions than covered in this report it is recommended to apply a design safety factor higher than 1. The exact value should be determined based on given circumstances.

8.1.2 Stiffness and creep

For the design solution proposed in Section 8.2.1, deformation control is a crucial aspect defining quality of the final product. In fact, it is used to shape the ultimate geometry, therefore it is not justifiable to set minimum requirements concerning member deflections. Instead, the initial, predeformed mould shapes are allowed to deflect freely as long as the yield forces and strains are not exceeded and predeformation is found. Creep is not taken into account as it is considered negligible.

8.1.3 Temperature factor

For geometries and ambient conditions considered in this project, influence of the temperature could be neglected. However, in case of using the moulds in outside environment or designing a concrete mass with significant heat release, it is suggested to decrease the yield strength of material according to Eq. 5.12 and material stiffness according to Eq. 5.11.

8.1.4 Practical aspects

Constructability needs to be considered during the design stage. Firstly, hardware limitations regarding component size and printing directions explained in Section 2.2.1 need to be taken into account. Concrete needs to be cast in a controlled way, including such aspects as placement speed (and consequent additional pressure on the mould walls), compaction technique or fluidity (potential leakage). What is more, design and provision of the support conditions is necessary: currently this only involves linear supports on the zero level, but there is potential for supplementing them with additional, three-dimensional framework made of other materials, e.g. timber or steel. Since such improvement would increase the design possibilities almost infinitely, it is particularly recommended to investigate this aspect in the future.

8.2 Design procedure

As already mentioned, the main challenge regarding usability of the 3D printed moulds is their low stiffness/strength ratio. In practice, this prevents applying traditional design methods known from the construction industry. A predeformation-based solution is proposed to embrace this issue and allow using extruded polymers as formwork materials. It can be contained in three main steps:

1. Problem definition

Definition of desired geometry, cross-sectional input data, expected print age and class (in case of the latter not being known *a priori*, class 1 is recommended), loads and boundary conditions, predicted temperature. Evaluation of practical aspects mentioned in Section 8.1.4. A preliminary strength analysis can be included in case of concerns with regard to this aspect.

2. Predeformation

Finding an initially deformed shape, which turns into the desired one after application of

the load. Numerical explanation of the process can be found in Section 8.2.1, flowchart is presented in Figure 8-4.

3. Strength verification

Translation of the numerical output into internal cross-sectional forces with the use of Equations 6.32-6.38, checking the component for strength with the use of formulas and tables provided in Section 8.1.1.

In case of lack of confidence concerning predicted quality of the product, it is suggested to use print class 1 as a default value for predeformation combined with lowest possible characteristics strength values. It also needs to be noted that the scope of application of given method is currently limited to the problem category described in this report. More information about limiting factors can be found in Section 8.2.2.

8.2.1 Predeformation algorithm

Numerical model explained in Chapter 6 includes two types of nonlinearities: large deformations and strain-dependent variability of cross-sectional parameters. Analytical response prediction of such system prior to solving is a particularly challenging mathematical problem that reaches beyond the scope of a Master's thesis. Instead, a straightforward, numerical iterative approach has been chosen for determination of the necessary amount of negative deformation. The method is based on iterative update of the input mesh coordinates: starting from the input mesh equal to the desired one, after each solution the vector between coordinates of the desired and output mesh nodes is added to the coordinates of input mesh for the next iteration.

Considering $X_0 = \{x_{0,ij}\}$ as matrix of point coordinates of the desired shape (input mesh in the initial step) and $X_{in,k} = \{x_{in,k,ij}\}$, $X_{out,k} = \{x_{out,k,ij}\}$ as the matrices of input and output meshes in k-th iteration, all of them $n \times 3$ size where n is the number of nodes, the above statement can be written as:

$$X_{in,1} = X_0 \quad (8.3)$$

$$X_{in,n+1} = X_{in,n} - (X_{out,n} - X_0) \quad (8.4)$$

The procedure is repeated until the desired simulation output (in this case deformed shape) is found within given tolerance. Assuming acceptable average and maximum distances between nodes of 0.05 mm and 0.1mm respectively, the convergence criteria can be expressed as follows:

$$\frac{1}{n} \sum_{i=1}^n \sqrt{\sum_{j=1}^3 (x_{out,k,ij} - x_{0,ij})^2} \leq 0.05mm \quad (8.5)$$

$$\max \left(\left\{ \sqrt{\sum_{j=1}^3 (x_{out,k,ij} - x_{0,ij})^2} : i = 1 \dots n \right\} \right) \leq 0.1mm \quad (8.6)$$

Presented method is rather slow yet simple and robust for given problem category, therefore its application is justifiable. However, investigation of more efficient formfinding techniques is recommended as a direction in future development.

8.2.2 Limitations

The approach described in this chapter is restrained by a number of limitations. Firstly, this report covers the use of technology exclusively for small scale moulds filled with concrete. Only instantaneous response of the 3D printed components in controlled environment is taken into account, which means that no conclusions about the design against creep or long-term behaviour should be made. Also other infill or load types could not be directly applicable to the presented solution - such changes would need additional verification. In case of using the material under increased temperatures, it is recommended to investigate the issue in more detail to confirm the findings discussed in this document with sufficient confidence. Finally, behaviour of the material at lowered temperatures remains unknown.

One serious restraint is contained within given scope of use. Moulds with predefined negative deformation are sensitive to loss of stability through limitation of equilibrium. This phenomenon is characteristic for shallow curved structures, which transfer the loads due to compressive axial forces. Concave walls under concrete pressure belong to this category. Limitation of equilibrium leads to snap-through buckling, i.e. a situation, in which a compressively loaded elastic system passes from one equilibrium state directly to another, non adjacent one. This is visually explained in Figure 8-2: at the point B, the force/deflection graph of a shallow arch with fixed ends reaches its local maximum P_{cr} , at which it drops to continue further growth, this time under tension. At point C it again reaches P_{cr} - distance between the two mentioned points represents the moment of switch of the system from one deformed state to another.

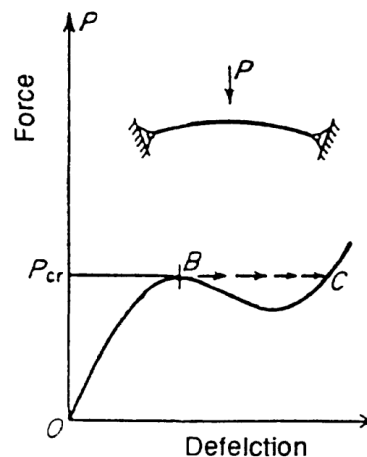


Figure 8-2: Snapthrough effect (Farshad, 1992).

Practical meaning of this issue for the design of 3D printed moulds refers to limitation of the wall span to thickness ratio. Since the occurrence of snap-through is dependent on a number of parameters, no universal, quantifiable solution can be proposed. An educated hint can be given by the predeformation algorithm: in case of exceeding the allowable theoretical domain, the solution will not converge, switching between convex and concave resultant shapes. However, it needs to be noted that no imperfections are taken into account in the numerical simulation, therefore it is likely not to be a reliable source of information about buckling. During preparation stage of the covered research project the above issue was solved by limiting the ratio between normal and flexural stresses within a wall to 0.1 (almost pure bending),

which practically eliminated the risk. However, further investigation is suggested in this field in order to allow more efficient use of the method.

Another danger related to the computational part of the design process is reliability of the code in case of complex shapes. Although it is meant to be universal for given category of problems, such risks as incorrect meshing or lack of convergence for certain problems cannot be completely ruled out without a large batch of tests.

In the end it needs to be highlighted that the method presented in this section assumes no need for remeshing and element reorientation. This indeed is true for investigated strain range and expected accuracy, but should be re-evaluated in case of applying to different problem category.

8.3 Software implementation

The design approach presented in previous sections has been implemented into the graphic software Rhinoceros (v.5.0), specifically into its parametric plugin Grasshopper. It has been done mainly with the use of Grasshopper's ghpythonlib component library, incidentally supplemented by RhinoCommon and RhinoScript (for particular classes and data migration between Rhino and Grasshopper). Whole deployment consists of the following components (numbers in brackets refer to the groups in Figure 8-3):

Model directories (1)

Definition of working directories.

Input: model directory, cross-sectional data directory, CalculiX executable path, Gmsh executable path.

Mesher input preparation (2)

Translation of the geometry located on Rhino canvas into an input for Gmsh meshing package. Done by the means of a Python script running a sequence of Rhino commands (splitting intersecting surfaces and edges, removing redundant objects), exporting the geometry to .stp file and generating a .geo input file for mesher. The latter includes meshing parameters and topology of the geometry. Full code is not presented in this report as it contains several hundred lines handling the exceptions that have been encountered. In usual cases, however, simple splitting of the intersecting surfaces and export of the geometry is sufficient. Rules for generation of a Gmsh input can be found in Geuzaine and Remacle (2016).

Input: model geometry, meshing parameters.

Output: .stp file with geometry definition, .geo input file for mesh generator (Gmsh).

Meshing (3)

Trigger mesh generation in external application (Gmsh) based on prepared .geo file. Done with a single-line shell command.

Input: .stp file with geometry definition, .geo input file for mesh generator.

Output: mesh file in .inp format.

Input model definition (4)

Generation of a base model file for further analysis and predeformation based on the mesh in .inp format. The code can be found in Appendix K.

- Input: mesh file in .inp format, geometrical tolerance, information about printer's coordinate system, infill in the form of Rhino volume objects, infill pressure function, support type, analysis type.
- Output: global .inp model including mesh topology, local coordinate system of each element, load specification, boundary conditions and parameters of the analysis.

Generation of cross-sectional data (5)

Calculation of stress-strain relationships within a shell with given thickness based on the experimental force-displacement data. The code is presented in Appendix L.

- Input: material data directory, material age, wall thickness in the FE model.
- Output: four .dat files with stress-strain data for the use in structural analysis.

Predeformation (6)

Invocation of the predeformation routine described in Section 8.2.1 for given model and convergence criteria. The code can be found in Appendix M.

- Input: base .inp model, cross section class, modelled wall thickness, relevant stress-strain data, convergence criteria.
- Output: mesh of a predeformed mould in .inp format, .dat file with stress, strain and displacement output of the final iteration.

Result viewer (7)

Translation of the output data into visual form. Code for reading the mesh topology and calculating the resultant internal forces is presented in Appendix N.

- Input: initial and predeformed mesh definition in .inp format, .dat file with stress and strain results in integration points.
- Output: visual representation of the results (initial and deformed mesh, internal force maps) on Rhino canvas.

Schematic representation of the functionality provided by the above set of components is presented in a form of flowchart in Figure 8-4. Additionally, points (5) and (6), are schematized in more detail in Figure 6-8.

With its modular structure, the implementation is meant to be easily accessible and adjustable. It also provides clarity to regular user and allows straightforward development and addition of new components. Furthermore, most of the components can be accessed from outside Rhino environment, which can be helpful for advanced users with programming skills. Besides practical reasons, such structure has been strongly driven by the technical limitations related to the obsolete architecture of Grasshopper plugin itself. Firstly, the software reruns whole component each time when anything changes on its execution path, which means that switching any input parameter results in execution of the whole code embedded in given component, even if it is unnecessary for certain parts of it. Moreover, as a single-threaded process,

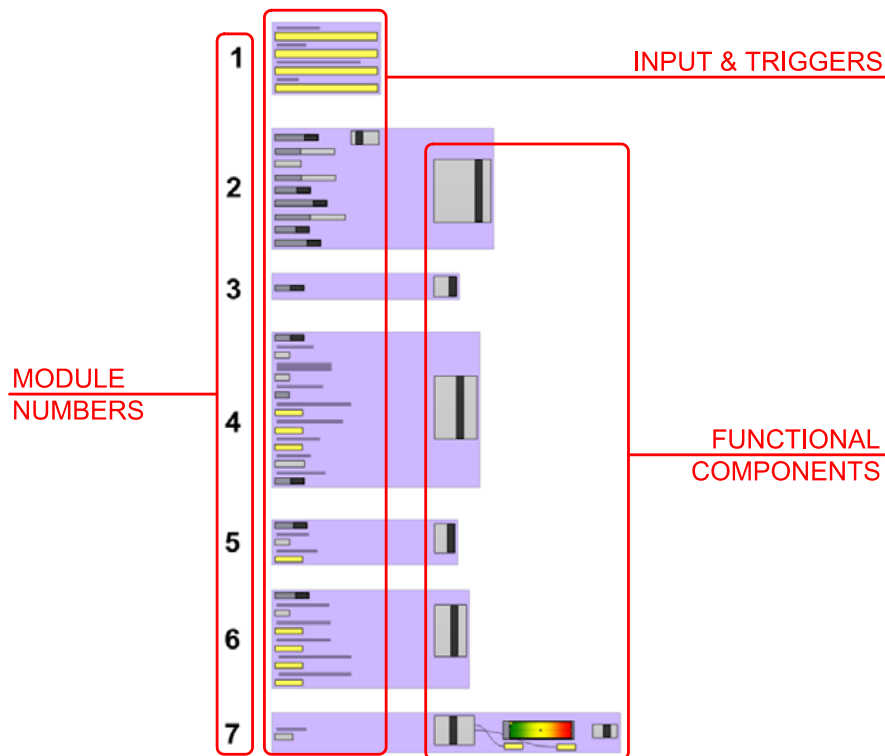


Figure 8-3: Grasshopper implementation - canvas layout.

Grasshopper is practically not capable of using more than one processor, and it freezes for the time when anything is executed on the canvas, including invocation of the external threads (e.g. Gmsh or CalculiX). Besides that, Python programming language interpreter used by Rhino (IronPython) contains a fault, which causes unavoidable, gradual slowdown of execution of the code as it is being rerun. The only known solution for that issue is restart of the whole package. Finally, IronPython is not compatible with many useful Python libraries like numpy or scipy, which limits its usability even further.

Unfortunately, as of the beginning of 2016 hardly any reasonable alternatives to Rhino and Grasshopper could be found on the market. Therefore, it has been used despite all its drawbacks. What is important, all routines written for the purpose of this research project run as expected, while the slowdown is caused by the environment. This gives the perspectives for successful use of the method in combination with other software.

For now, a temporary improvement could be achieved by rewriting the code to other language supported by Rhino, e.g. C#. This could at least eliminate the issues related to IronPython. Another, potentially best option would be migration outside Rhino/Grasshopper environment and creation of a separate in-house tool for all stages after the export of initial geometry. This should generally be feasible since most of Rhino/Grasshopper libraries used in this project could be reproduced relatively easily (operations on nodes and elements are predominant). However, several bottlenecks would need to be solved, e.g. operations on freeform volumes done during definition of the infills.

To sum up, the current implementation is usable, but its performance would be vastly improved by changing the software strategy. This issue is recommended to be solved by future researchers with passion in programming.

8.4 Summary

Chapter 8 contains the proposed design approach for 3D printed moulds. In the first part the design criteria are discussed, including quantitative definition of the strength requirements, followed by the guidelines addressing temperature influence and practical aspects. Noticeably, no recommendations concerning stiffness are given, which is the result of specific technique used to achieve the desired shape of the final product.

This method, elaborated in the second section, is based on the concept of fabricating pre-deformed forms, which deflect into desired shapes after casting the infill. Besides stepwise explanation of the general design procedure, basic mathematical background is provided and limitations listed out. The latter are numerous, starting from scope of application (small scale moulds filled with concrete in controlled environment) through software-related risks up to theoretical issues regarding stability of shallow shell structures. The last aspect is pointed out as a field of potential improvement in the future.

The last part of the chapter presents the in-house set of design components, which allow to go through the whole design process from one Grasshopper (Rhino plugin) canvas. Structure of the package is briefly explained with its advantages and drawbacks. Strong emphasis is put to the limitations imposed by Grasshopper environment. These are, among others, obsolete single-thread architecture and faulty interpreter of Python programming language. In the end, possible solutions to these issues are given, with a conclusion that the best option would be creating a fully customized design package.

It needs to be highlighted that the content of this chapter (and the whole report) covers only one category of potential application of the product. This means that the methodology presented here should not be directly copied to other cases, especially when they include such phenomena as long-term and cyclic loading or temperatures significantly different than 20°C. On the other hand, the provided information and tools can be considered as a base for development of more specialized techniques and items.

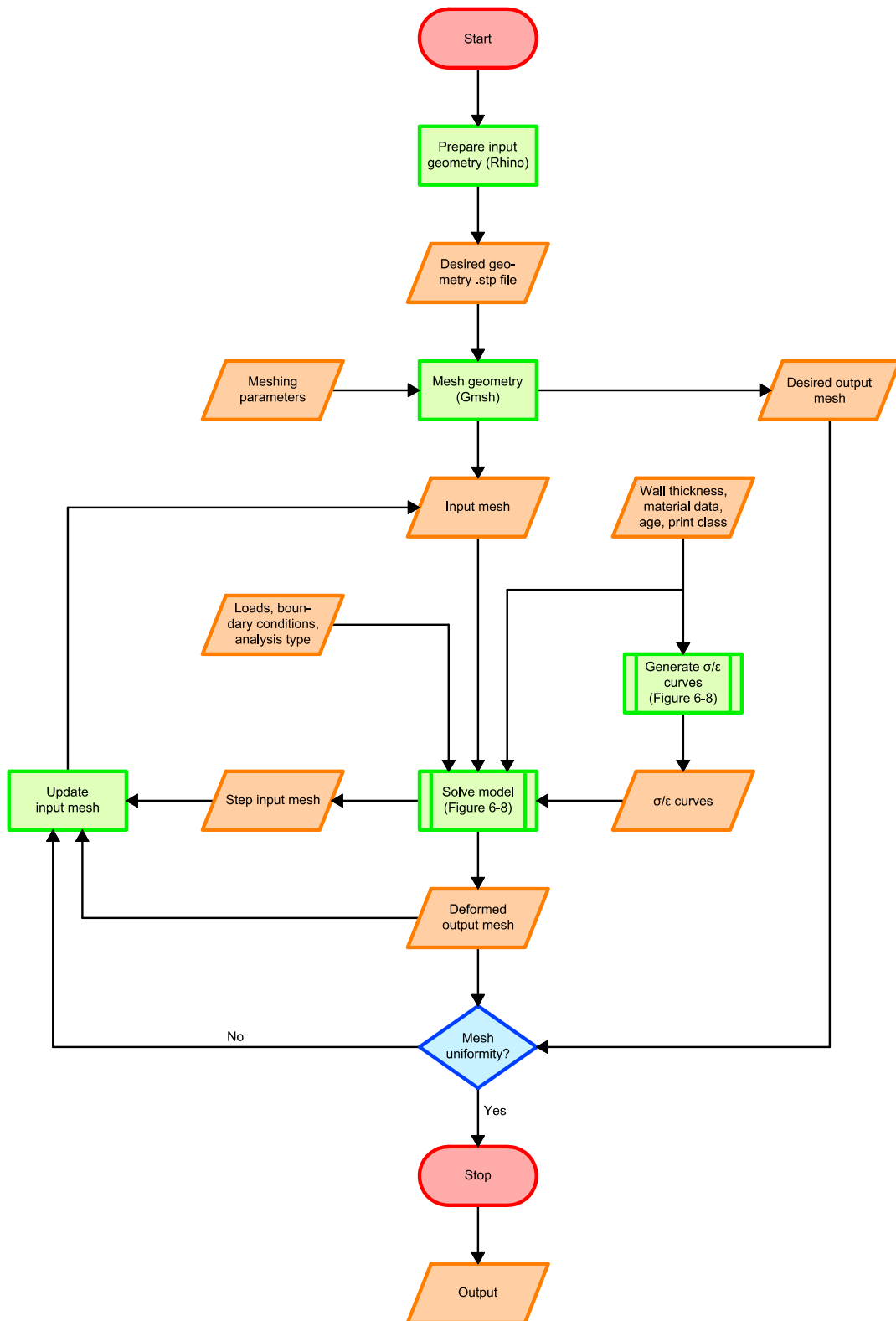


Figure 8-4: Mould predeformation flowchart.

Chapter 9

Case study

Practical applicability of proposed numerical and design approach has been evaluated with a case study. Three different mould shapes have been prepared for fabrication according to Chapters 6 and 8, then printed and cast with concrete. Section 9.2 covers the first geometry, which is a $200 \times 200 \times 500$ mm block similar to the typical test specimens used in previous stage of the research and modified by applying negative deformation prior to casting. Next, the possibilities created by the technology are presented on two more distinct examples, namely a column section with a parametric pattern and a complex-shaped wall segment.

9.1 Evaluation method

The results shown in this chapter are defined as distances between each individual photogrammetry point and reference surface (planar or curved in case of the wall panel back side) fitted into the whole point cloud with the least square method (see Appendix I). Deviation (+) means excessive deflection compared to the desired shape, deviation (-) - the opposite. In case of a curved surface only the absolute deviation values are given due to lack of single reference plane.

9.2 Prismatic blocks

Three pieces of mould for $200 \times 200 \times 500$ mm prism have been manufactured and filled with normal weight concrete after 14 days from printing. The amount of negative deformation necessary to achieve desired shape after casting (default convergence criteria met after 4 predeformation iterations) is shown in Figure 9-4. The results presented in Table 9-1 represent the average values obtained for the measurement series consisting of 9 walls (3 per each block). It can be seen that the final shape is almost perfectly prismatic, with only minor concave deviation in the bottom part. This setback is the effect of excessive deformation predicted by the numerical model, which has also been observed in 7.3. The cause of that effect can only be speculated, with support conditions being the most probable reason.

9.3 Parametric column

A 750 mm tall single layer mould for a column section with a parametric pattern on its surface (shown in Figure 9-1) has been cast with concrete after 14 days from fabrication. The predeformed shape presented in Figure 9-5 has been found after only two iterations, which is the consequence of particularly small expected deflections of less than a millimeter. Table 9-2 contains the average of two sets of results collected from two opposite faces.

The extreme deviations are of the same order of magnitude as applied negative deformation, almost reaching one millimeter. Again, a concave pattern is visible in the support zone, which stands in line with previous observations. Distribution of deviations over the rest of specimen is rather random and is most likely caused by the concrete imperfections and measurement errors.

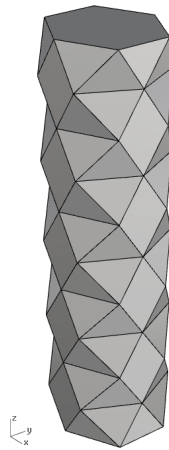


Figure 9-1: Case study: expected shape of a parametric column.

9.4 Wall panel

A complex-shaped wall panel visible in Figure 9-2 has been cast in a 14-days old, 750 mm tall mould built of double layer, cross-linked walls. Front and back side of the predeformed form (found in the 4th iteration) are shown in Figures 9-6 and 9-7 respectively. Tables 9-3 and 9-4 provide the deviation analysis of both sides.

Extreme deviations from the designed plane of the frontal face lie within range of ± 2 mm, with no visible pattern and values over ± 1 mm being reached only incidentally. In case of back side of the panel, most of the absolute deviations are below 1 mm. Locally appearing maximum (a visible bulge on the surface) reaches almost 3 mm.

Figure 9-3 highlights the practical issue that has been observed after demoulding of the piece. As can be seen, horizontal edges of the pattern are sharp, while diagonals - rounded. This effect is caused by the print plane coincident with global XY plane: sharpening of the edges happens as a consequence of filling the spaces between adjacent layers, which is possible only in printer's base plane.

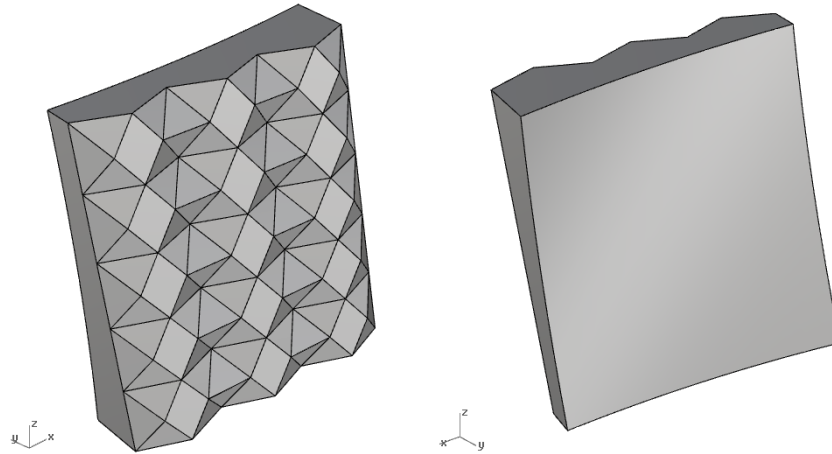


Figure 9-2: Case study: expected shape of a wall panel. Left: front. Right: back.

9.5 Summary

The comparison of designed and actual measured shapes of all case study blocks yields satisfying results. Standard specimens as well as parametric column show very low deviations (below 1 mm). In case of a wall panel this value locally grows to 3 mm, although the standard deviation remains below 1 mm. A pattern of concaves can be distinguished in the bottom zone of both tested column types - it is correspondent to the phenomenon observed in chapter 7 and is most likely related to the way in which the support conditions are modelled.

The column investigated in Section 9.3 highlights the influence of global geometry and local pattern on the wall deflection. Although it has almost the same cross-sectional area and is 50% taller than the standard $200 \times 200 \times 500$ mm specimens, it exhibits around 10% of their expected deformation. Definition of the extent and practical boundaries of this effect is suggested as a subject for the further research.



Figure 9-3: Case study: wall panel after demoulding.

In the end two practical aspects need to be mentioned. As can be seen in Figure 9-3, shapes of the edges differ depending on plane in which they lie: in printing plane they are sharp, while in all others - round. Furthermore, demoulding of the hardened blocks can differ from fast and simple (e.g. prismatic, single layer moulds) to very laborous and even potentially damaging to the product (most complex, multi-layered systems).

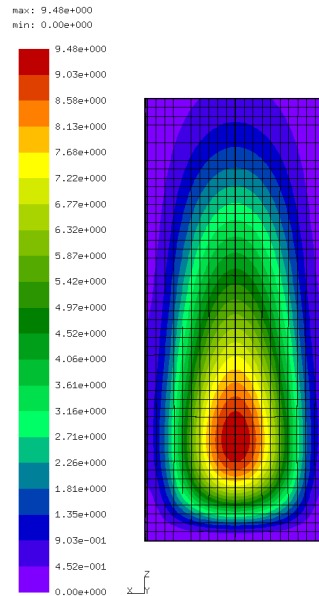


Figure 9-4: Case study: numerical model of a single layer mould. Predeformation.

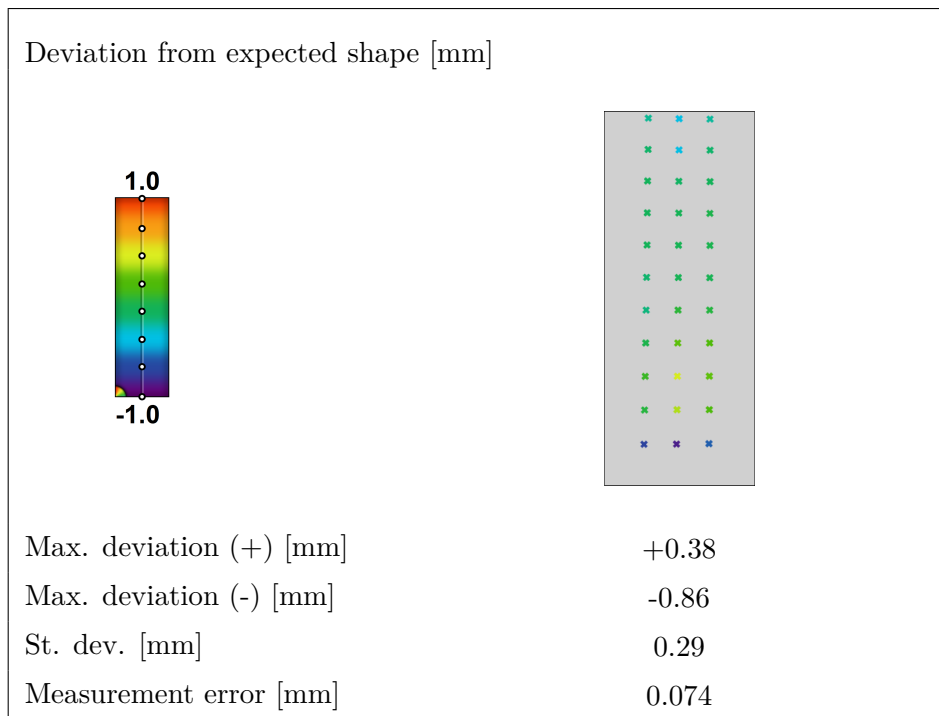


Table 9-1: Case study results: Single layer mould.

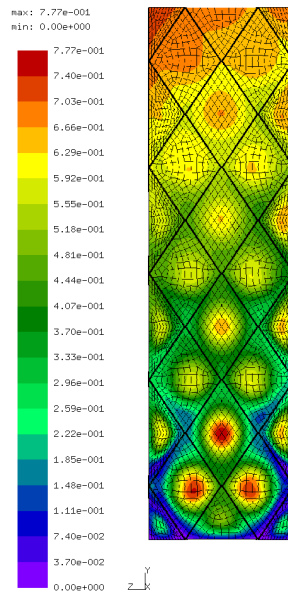


Figure 9-5: Case study: numerical model of a parametric column. Predeformation.

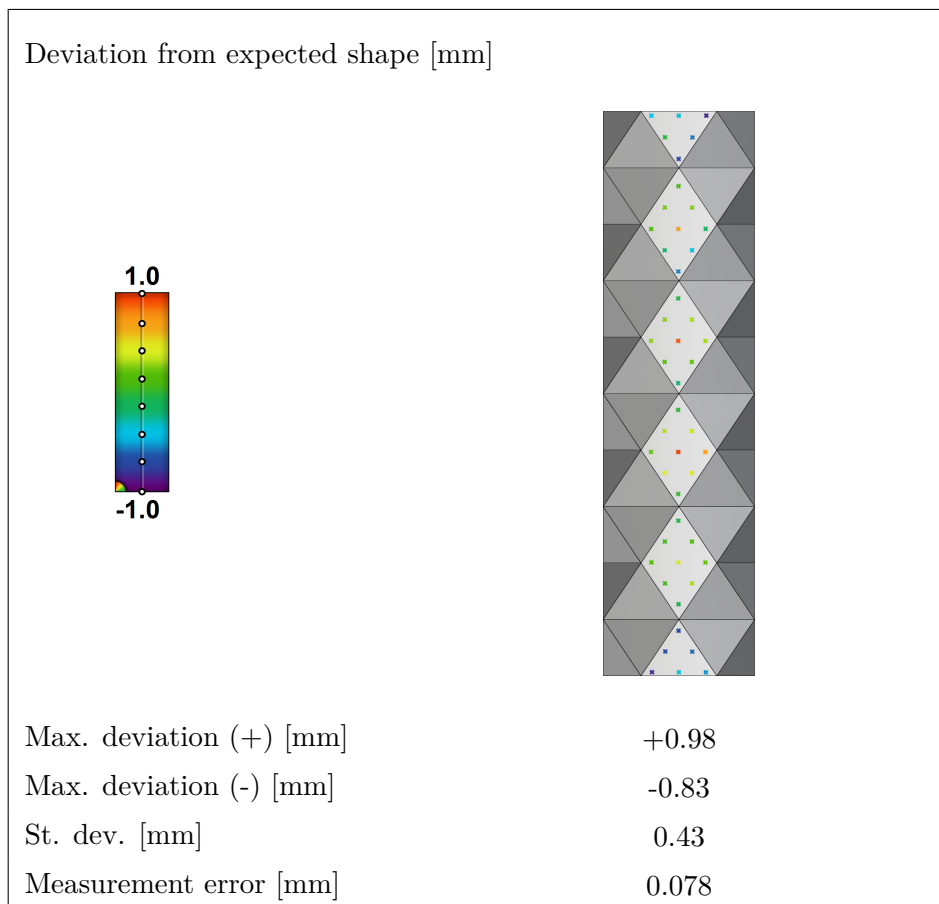


Table 9-2: Case study results: Parametric column.

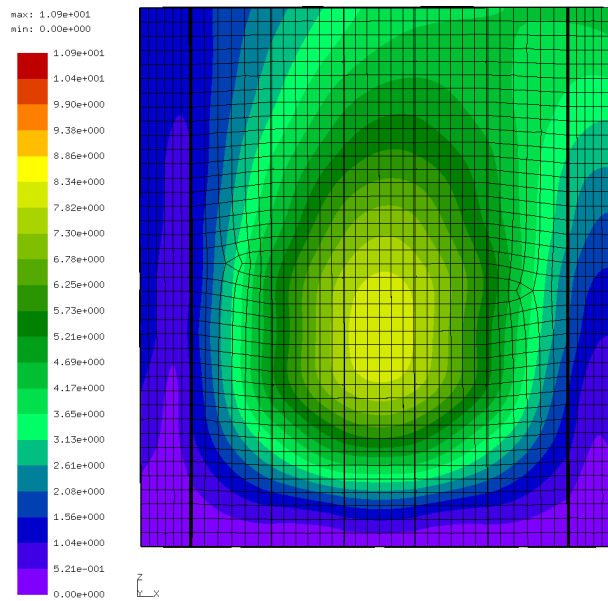


Figure 9-6: Case study: numerical model of panel front. Predeformation.

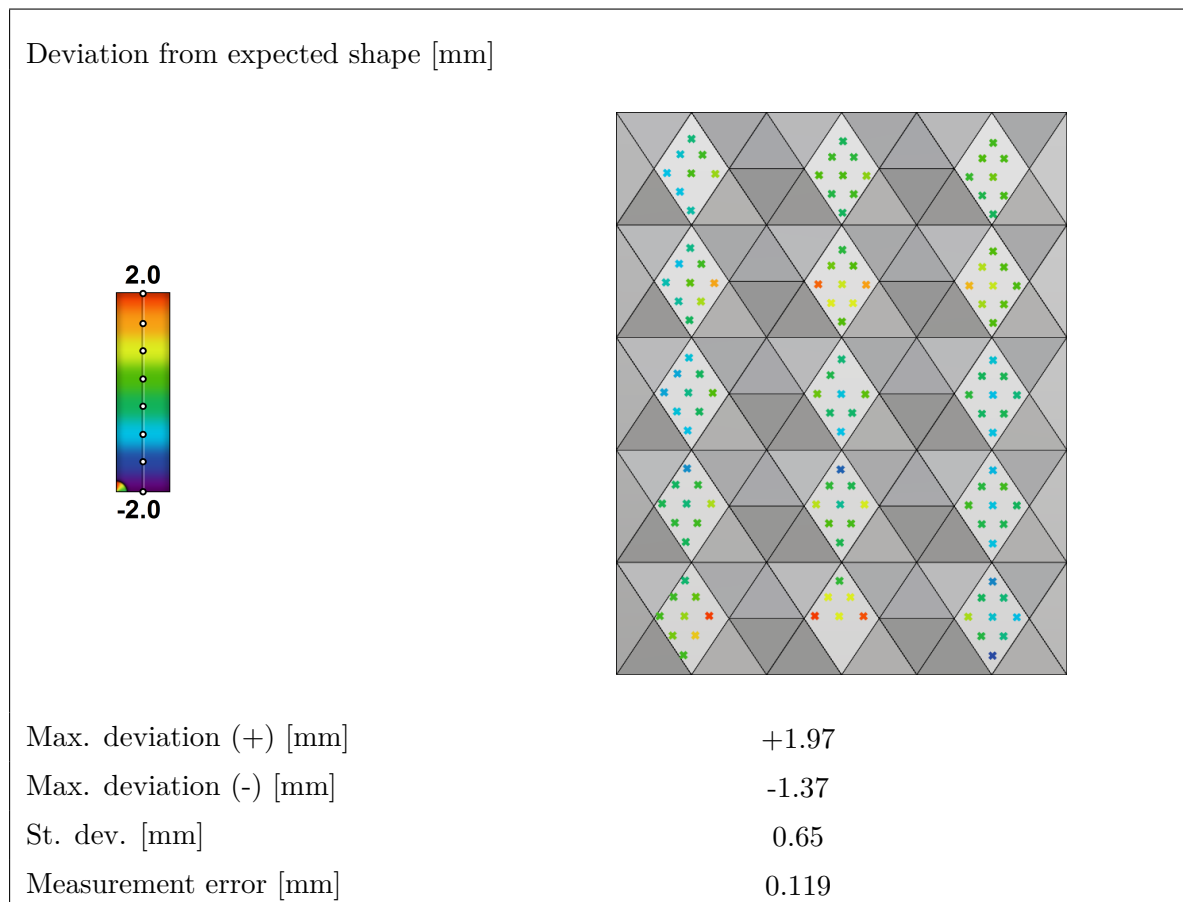


Table 9-3: Case study results: Panel front.

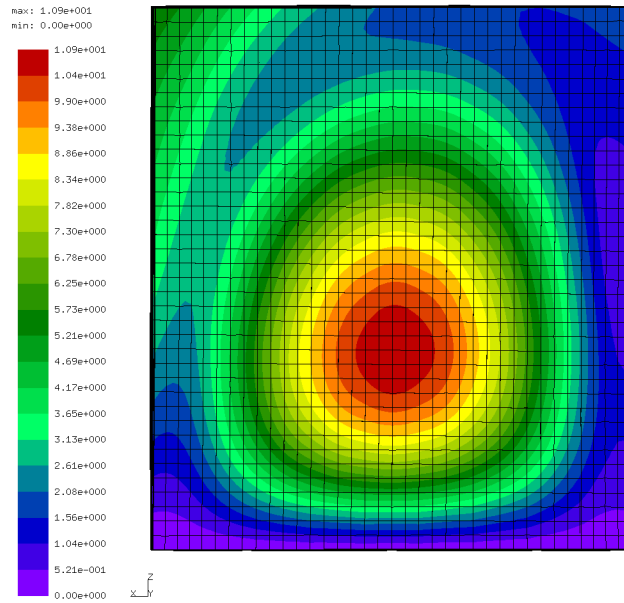


Figure 9-7: Case study: numerical model of panel back. Predeformation.

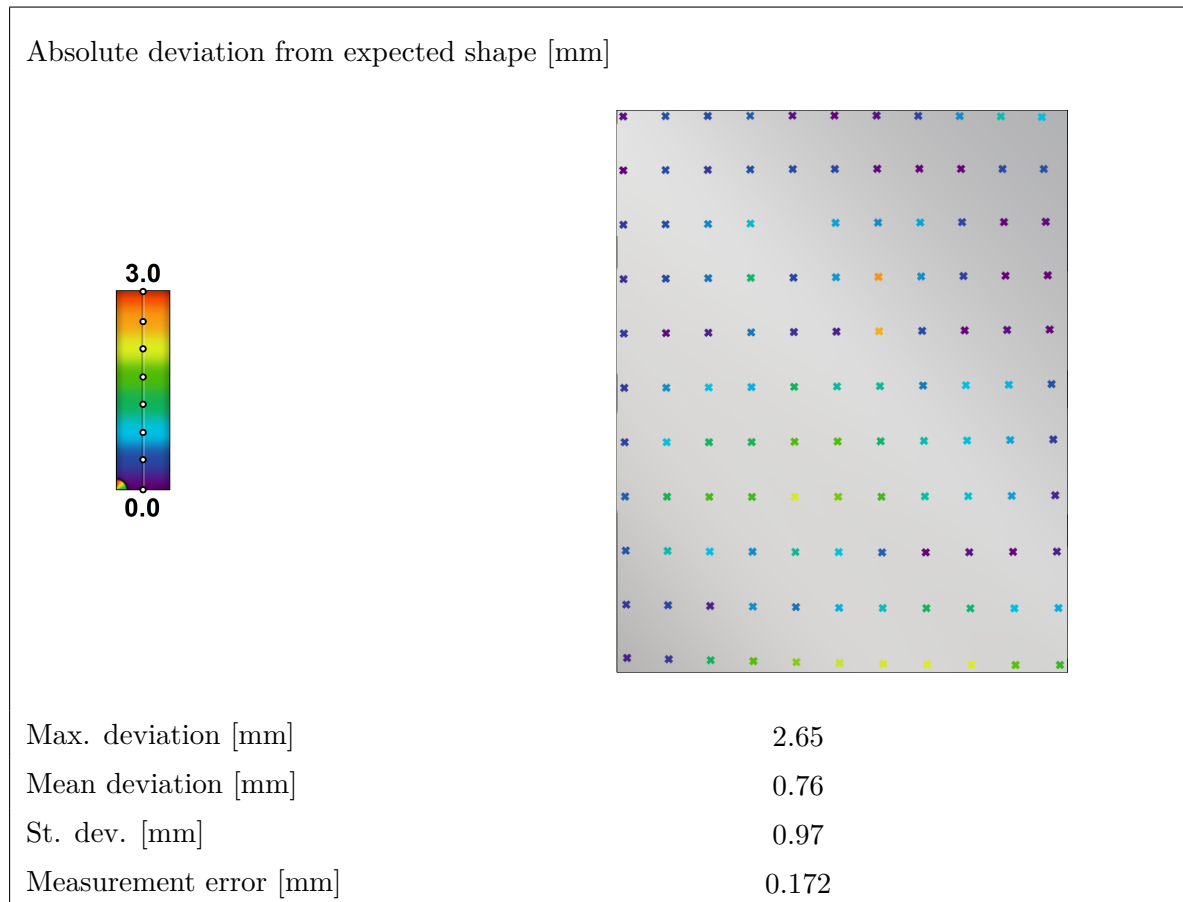


Table 9-4: Case study results: Panel back.

Chapter 10

Discussion

In the following chapter the content of this report is discussed and critically evaluated. Besides summarizing the most important aspects of the research, suggestions and speculations are made concerning possible alterations and improvements. Three factual categories have been distinguished within the presented work: Section 10.1 focuses on the laboratory tests and their results, 10.2 covers the proposed numerical approach and 10.3 treats the design methodology together with software implementation.

10.1 Testing and mechanical properties

The test stage was generally successful: vast majority of planned experiments has been performed and documented. However, collected results clearly show that at the current stage of development industrial application of the 3D printed components as formwork elements or moulds would not be easy. The material, being viscoelastic and highly sensitive to temperature, provides limited applicability and perhaps almost no scalability. What is more, even behaviour of a predefined, well designed component is not fully predictable due product's inherent orthotropy combined with random print quality. Currently, the latter can be resolved to certain extent by introducing the quality classification, but this should be considered as provisional only. The ultimate product would preferably incorporate uniform, predictable print quality as well as material formulation/treatment lowering its thermal sensitivity.

Regarding the material testing framework itself, it provided enough data for relatively precise prediction of deformations within the given scope. On the other hand, multiple assumptions and simplifications could be eliminated by running further experiments, such as Poisson's ratio, biaxial tension and bending or cyclic loading tests. This would allow to build a coherent cross-sectional definition of the product as well as to develop more customized mechanical model for the numerical analysis.

Several observations have also been made on the concrete test stage. Firstly, during hardening of the mixture the moulds exhibit small yet noticeable inverse creep, which source has not been clearly identified. What is more, the measurements made on the cured specimens

prior to and after demoulding provide slightly different results, which causes confusion over the choice of representative values. Solution for these issues should be sought with the use of a single, efficient and precise measurement method instead of the two utilized in this research (strain gauges, which turned to be unreliable and laborous photogrammetry). A suggested technique would be 3D scanning.

Concrete property tests were performed on a limited number of samples, which was caused by lack of facilities - cutting of the large blocks into $150 \times 150 \times 150$ mm cubes proved to be particularly challenging. Despite that fact, collected results laid within the expected ranges, so the issue has been considered as provisionally covered. Although such generalized approach can be acceptable on the current stage, it would definitely need to be improved in case of introduction of the technology into the industry.

Finally, significant improvement in terms of applicability could be achieved by optimization of the concrete infill. Firstly, light-weight mixtures would provide an immediate benefit of lower pressure on the walls. Likewise, self-compacting concrete proves to exert lower loads thanks to lack of vibration. The latter can also be an advantage in case of limited access for compaction tools, e.g. freeform shapes or reconstruction works. Furthermore, compounds with low heat of hydration could be especially favourable for large-volume elements.

10.2 Numerical analysis

For the given scope, the proposed numerical approach yields accurate results, which is confirmed in Chapters 7 and 9. However, it is rather straightforward and could potentially be improved in several ways. Firstly, application of Hooke's law to large strains yields risk of inaccuracies and error propagation in case of misuse, e.g. for combination of large bending and tensile strains. Suitable formulation based on energy function (e.g. hyperelastic) would be more universal and safe in use, although Section 6.2.2 shows that it might exhibit stability issues that are hard or impossible to eliminate.

What is more, the use of a triple-layer shell element is not optimal. A huge improvement would be achieved with development of a special-purpose formulation, which would allow merger of three separate stiffness matrices of three layers into a single one representing a set of four non-uniform, nonlinear parameters EA_x, EA_y, EI_x, EI_y described in Section 6.2.1. However, it needs to be stressed that development of a custom model is in general a labourous task requiring programming skills and understanding of both mechanics and mathematics. Considering complexity of the above definition, the expected amount of work (including formulation, calibration and validation) could cover a separate Master's project. Consequently, such effort has not been made here and a simple, proven solution was used instead.

Also implementation of the material nonlinearity presented in this report is computationally inefficient - it provides correct output, but works *on top* of the existing FEM code, updating stiffness after full solution of a nonlinear problem. It would be more efficient to implement the strain-dependent refactoring of element's stiffness matrix after each step of a single nonlinear solution, which would result in only one solution with one set of convergence criteria instead of running multiple simulations of the same model with gradually updated stiffness.

10.3 Design and implementation

Even with all drawbacks related to the mechanical characteristics that are mentioned in Section 10.1, the 3D printed moulds can become an interesting fabrication method for freeform or hardly accessible concrete elements. It has been proven that for certain applications the proposed predeformation-based design approach combined with print quality classification gives satisfying results both in terms of dimensional correctness as well as quality of the final product.

With basic issues successfully covered, content of this report can be used as a cornerstone and reference for development of a state of the art design method. There are numerous potential improvements concerning such aspects as geometrical and structural optimization of the moulds or software performance (mentioned in more detail in Section 12.3).

Finally, as the technology becomes more mature, further upgrades are expected in the field of both material properties and manufacturing process. Shifting to more efficient thermoplastics, stabilization of the print quality or scaling up the fabrication capabilities are only few of them. This means that one should not only focus on what *is* now, but also look into what *will be* in the predictable future.

Chapter 11

Conclusions

Thanks to elimination of the human factor and freedom of form that it provides, large-scale 3D printing is considered a promising innovation in the construction industry. The reported graduation project is focused on testing, modelling and design of moulds fabricated with the use of this technology. The general objective of transferring these processes from conceptual to practical level has been contained in three research questions presented in Chapter 1. In following sections an answer to each issue is given in form of a short summary, which highlights the most important findings that have been discovered.

11.1 Testing and mechanical properties

From the analysis of the experimental data it can be concluded that:

- Proposed basic property test scheme provides information that is sufficient to model mechanical behaviour of the 3D printed cross sections with chosen numerical approach.
- After printing, the investigated material exhibits highly orthotropic characteristics with nonuniform EI , EA parameters in each direction and low stiffness to strength ratio. As a thermoplastic, it is also viscoelastic and highly sensitive to temperature.
- Uncontrolled, locally variable print quality severely influences structural characteristics of the 3D printed cross sections. This has been resolved to certain extent by introducing quality classification.
- Mould tests proved general applicability of the technology, with only minor issues (limited bottom leakage, potentially laborous demoulding) being observed.
- Problems with cutting led to numerous flaws on concrete property test specimens, which caused particularly high spread in the results. Nevertheless, mean values of both strength and porosity lie within the acceptable range.

11.2 Numerical analysis

Investigation of possible strategies for numerical modelling led to following findings:

- In case of small scale moulds cast in controlled environment, the temperature influence can be neglected, while material viscoelasticity and eventual plasticity can be replaced with strain-based, nonlinear Hookean elasticity.
- The experimental output can be implemented into the numerical model by translating relevant force-displacement data into stress-strain curves using thickness of the shell element.
- The lack of uniformity in cross-sectional characteristics EI , EA in both directions can be accurately modelled with the use of a composite, triple-layer 2nd order quad shell element with reduced integration.
- Implementation of the above method in CalculiX FE package with the use of in-built material and element formulations is precise (relative deviations from the expected deformations under normal and flexural loads below 10%) and workable yet non-optimal as it requires overhead computations.

11.3 Design and implementation

Research on the influence of various design aspects with subsequent development and implementation of the proposed design method resulted in a range of conclusions, which can be summarized as follows:

- 3D printed components should be designed with particular focus on deflection due to material's low stiffness to strength ratio.
- Method involving application of the numerically determined negative deformation provides desired shape of the final concrete product with precision only incidentally exceeding 1 mm.
- Material integrity can be provided by the means of minimum strength requirements that are then used to evaluate the simulation output.
- Choice of Rhino/Grasshopper software package results in performance limitations due to its single-thread architecture combined with faulty interpreter of Python programming language. Consequently, the method cannot be used at its full speed until the implementation is migrated to other (preferably special-purpose) program or Grasshopper plugin is substantially modified.

Finally, it needs to be emphasized that the proposed solutions are directly applicable only to the components where creep and temperature can indeed be neglected (in other cases it should be revised and validated again).

Chapter 12

Recommendations

The 3D printed mould system, being a new and scarcely studied technology provides a multitude of opportunities and development paths, which cannot be all covered with a single Master's project. Observant reader of this report can find a number of paragraphs pointing out potential improvements, which have not been investigated due to time and volume limitations. In this chapter most important of them are pooled and summarized to create a reference for the researchers working on the subject in the future.

12.1 Testing and mechanical properties

In case of sufficient resources it would be desirable to supplement the mechanical property test framework with more complex tests (e.g. biaxial bending and tension, cyclic loading, compression and buckling). Additionally, an in-depth investigation of the material long-term behaviour and hardening process is recommended in order to collect the information about the evolution of its characteristics over time and under different conditions. All this would allow building models that will be more universal in terms of scale and scope of application. Subjects for further research can also be pointed in the field of practical experiments. Seemingly minor issues such as bottom leakage and negative creep of the forms should be systematized in order to evaluate their potential influence on the final product. Search for improvements is also recommended within the domain of concrete mixture. Besides that, more experimental work could be done on larger moulds and formwork systems as well as other structural applications in general.

12.2 Numerical analysis

Main suggestion with regard to numerical modelling of the 3D printed thermoplastics is development of a custom material model that would allow more efficient analysis compared to the current hybrid implementation utilizing triple-layered shell elements.

The custom solution should aim for at least following two objectives:

- definition of an adequate (preferably strain energy-based) material formulation capable of handling nonlinear stress-strain relationships
- introduction of a custom stiffness matrix representing nonuniform cross-sectional characteristics EI and EA in both directions

For the time being, extension of the solution presented in this report by adding e.g. creep and temperature components could also be considered.

12.3 Design and implementation

The proposed design approach utilizing negative deformation method contains a number of uncertainties and unknowns of various importance. The most serious matter that needs to be covered is behaviour of the cross-section under combination of bending and normal force. In-depth investigation of the stability issues affecting predeformed walls (Section 8.2.2) is also recommended in order to take the method to the limits of applicability. Additionally, the simplified way of taking the temperature into account should be verified in practice in case of predicted substantial deviations from 20°C.

Large benefits are expected to be obtained with the improvement in performance of the created design tool. Besides the alterations explained in the previous section, this could be achieved by migrating the code out of the Rhino package, which is currently causing slow-down due to its inherent limitations.

What is more, the material use could possibly be optimized by e.g. educated use of mould's wall pattern to increase its stiffness, implementation of a structural optimization algorithm for complex geometries or introduction of three-dimensional support conditions using temporary structures made of other materials.

Finally, the given methodology could be modified in order to adjust it to other size or scope. This refers mainly to the research on general scalability as well as behaviour of the components under other types as well as longer duration of loads. Development of worker-friendly connection and demoulding techniques is also suggested, especially in case of using moulds made of built-up walls.

12.4 Other aspects

Significant advances are seen in improvement of the manufacturing process. Current capabilities of the 3D printer do not provide adequate quality control, which results in large variability of cross-sectional properties and subsequent, unavoidable introduction of the correction factors. This serious drawback could be eliminated with the increase in predictability of the extrusion process.

Another field, which is strongly recommended to be explored, is recycling, reuse and modularity. Introduction of a multi-use system would be a development that could redefine perspectives for the whole concept of 3D printing in the architecture.

Appendix A

Mechanical properties of thermoplastics - overview

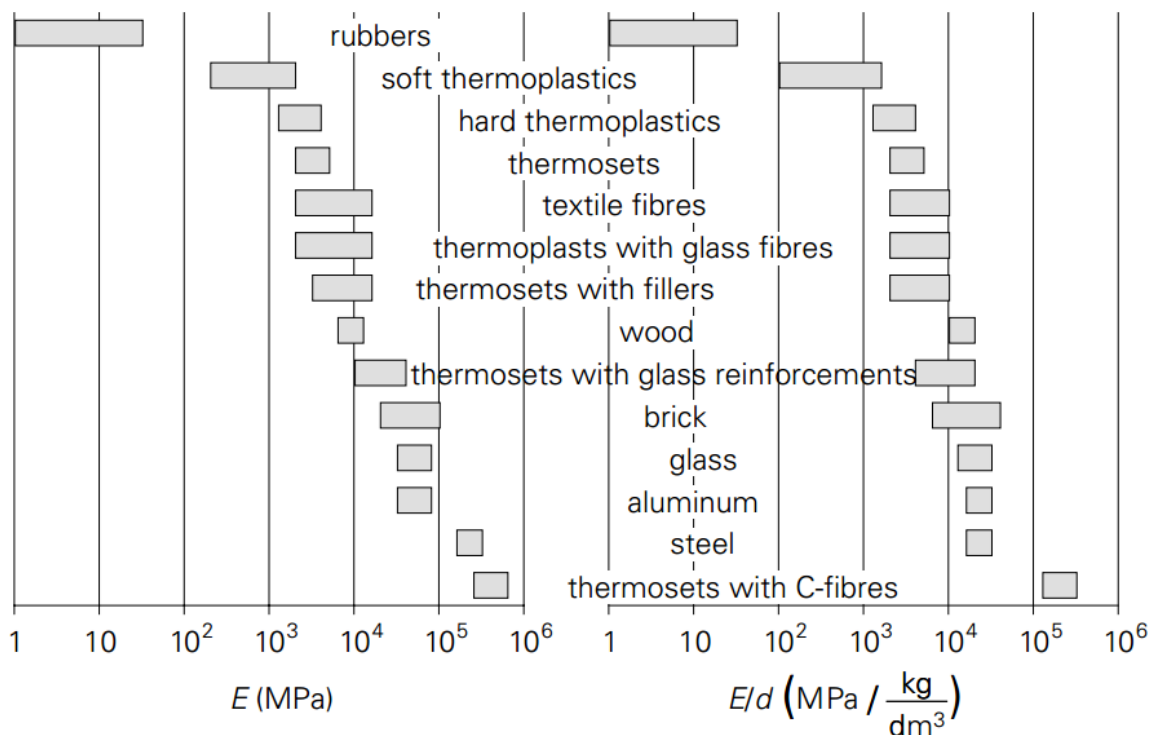
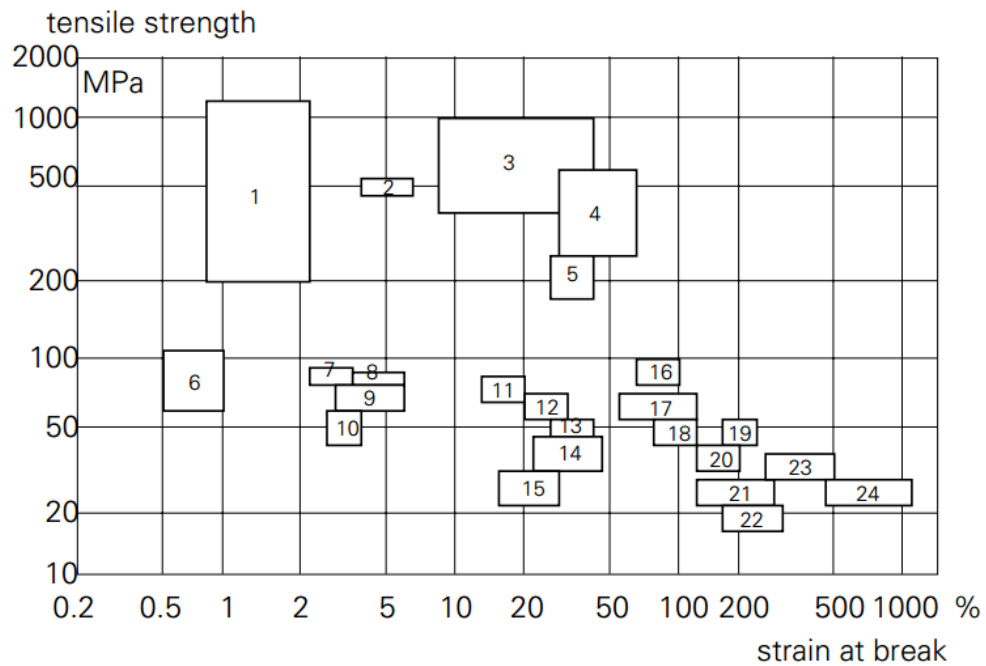


Figure A-1: Stiffness of thermoplastics: overview (van der Vegt, 2006).



- | | | |
|--------------------------|---------|-------------|
| 1. reinforced thermosets | 9. PMMA | 17. PPO, PC |
| 2. cotton | 10. PS | 18. PETP |
| 3. construction metals | 11. PVC | 19. PA |
| 4. synthetic fibres | 12. POM | 20. PP |
| 5. wool | 13. CA | 21. HDPE |
| 6. thermosets | 14. TPS | 22. LDPE |
| 7. PPS | 15. ABS | 23. PTFE |
| 8. SAN | 16. PSU | 24. rubber |

Figure A-2: Tensile strength of thermoplastics: overview (van der Vegt, 2006).

Appendix B

Raw material test output

In this appendix raw output of the material property tests is provided. It has not been altered by any means, neither the classes have been applied.

B.1 Tensile longitudinal tests

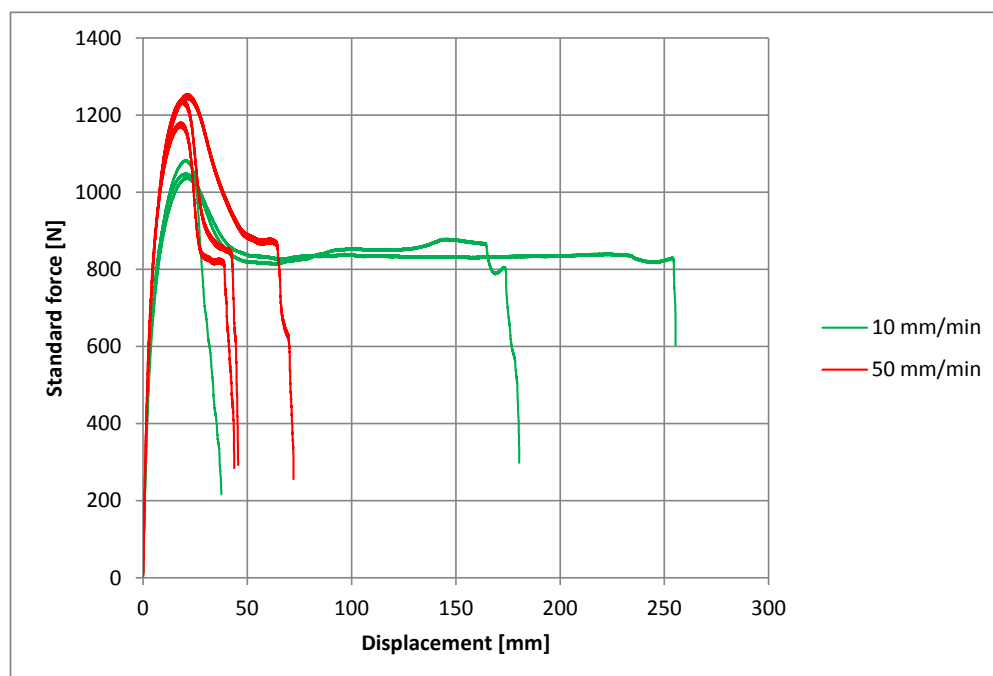


Figure B-1: Raw results: tensile longitudinal test, 1 day.

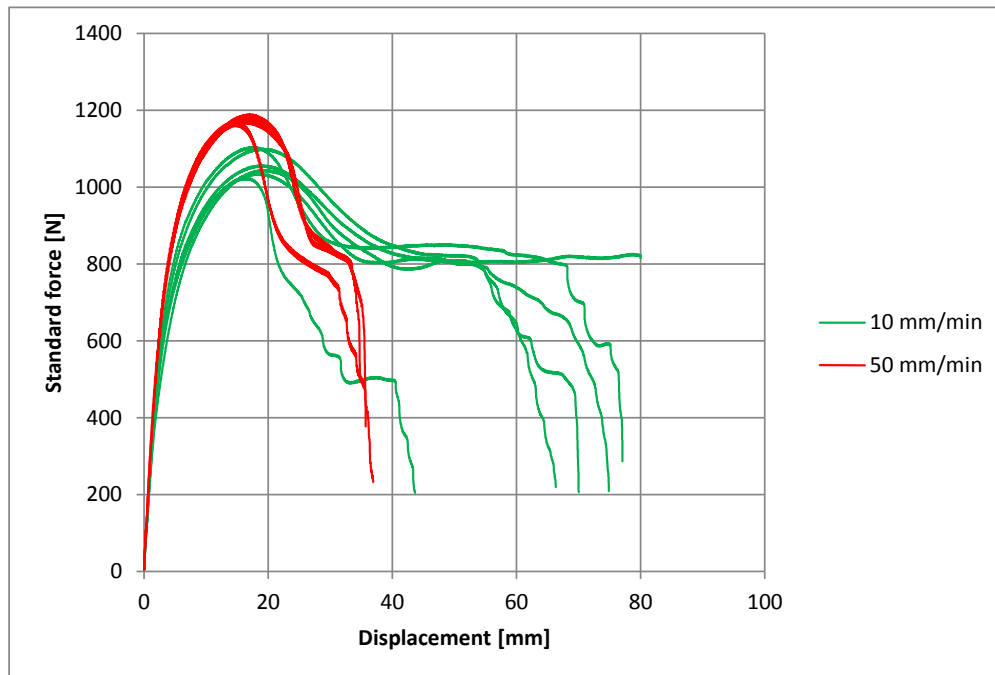


Figure B-2: Raw results: tensile longitudinal test, 3 days.

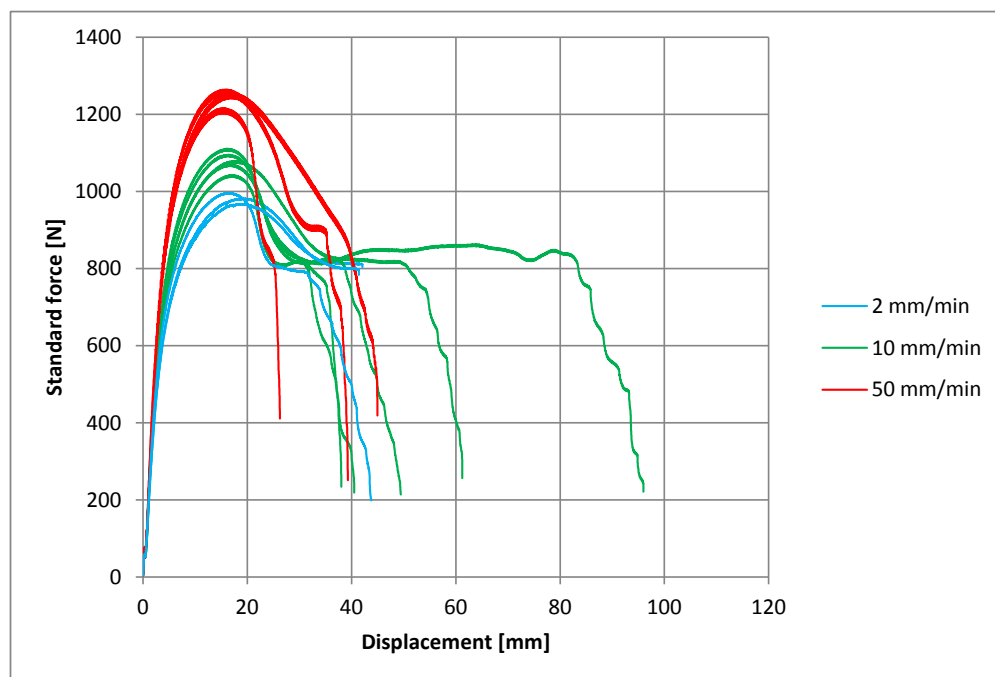


Figure B-3: Raw results: tensile longitudinal test, 7 days.

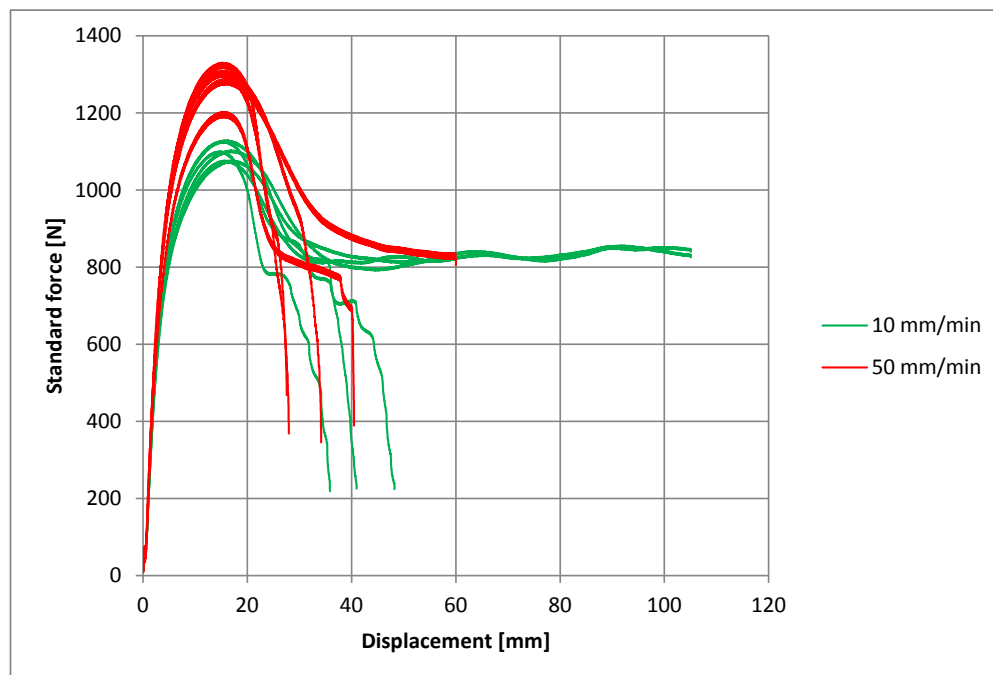


Figure B-4: Raw results: tensile longitudinal test, 14 days.

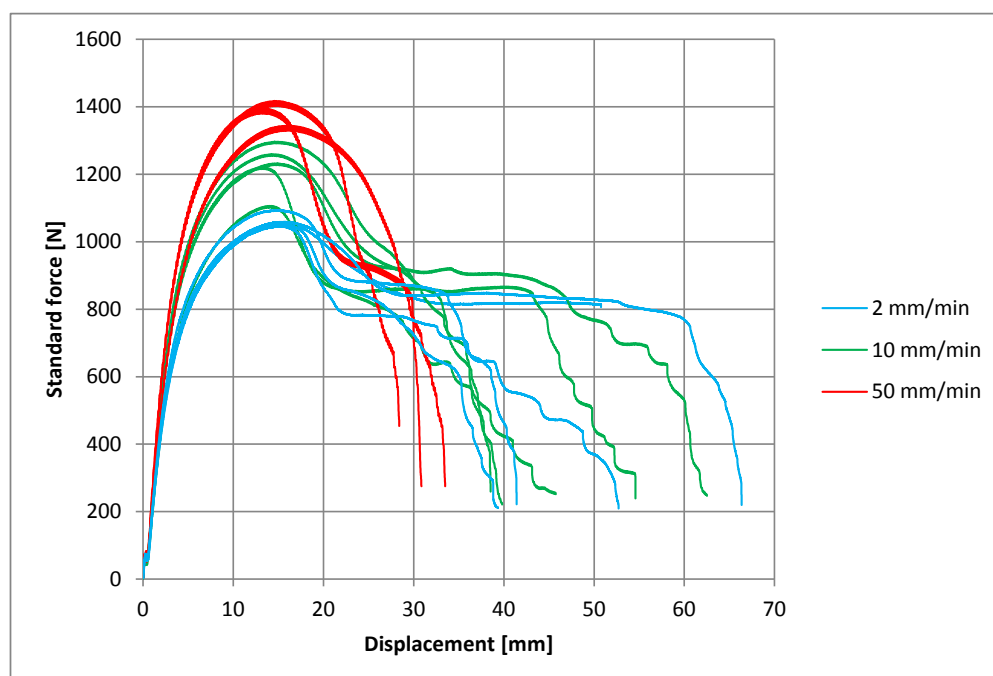


Figure B-5: Raw results: tensile longitudinal test, 28 days.

B.2 Tensile transverse tests

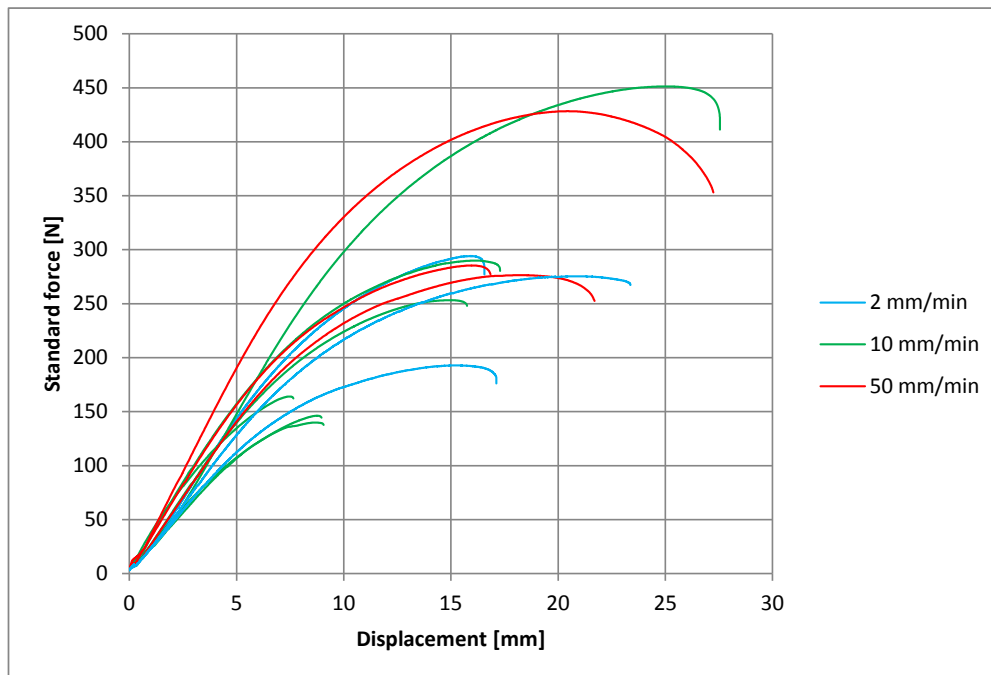


Figure B-6: Raw results: tensile transverse test, 1 day.

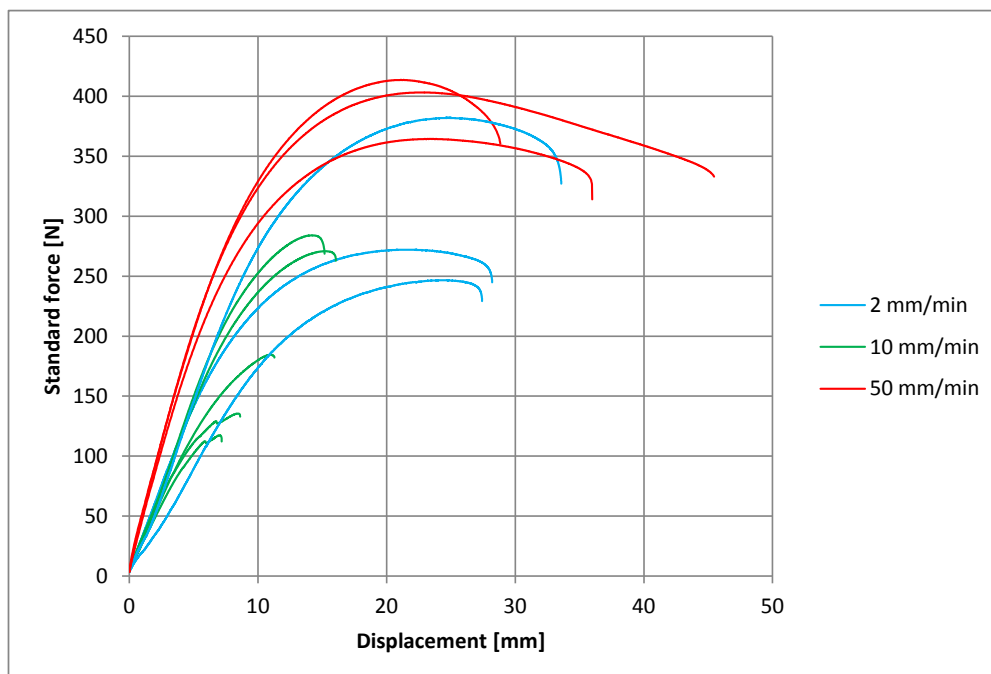


Figure B-7: Raw results: tensile transverse test, 3 days.

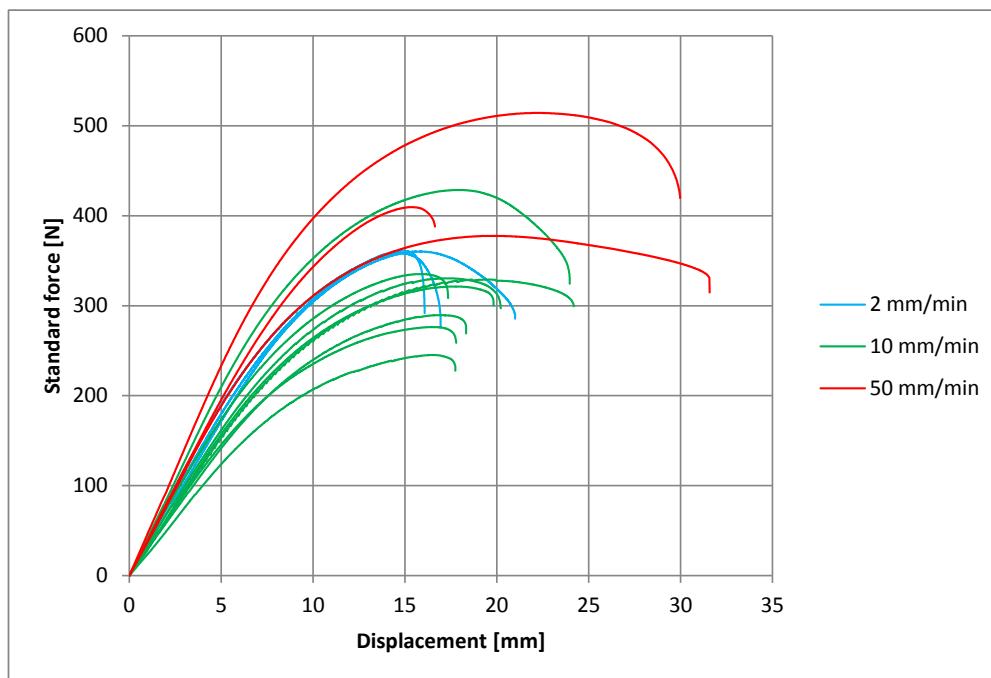


Figure B-8: Raw results: tensile transverse test, 7 days.

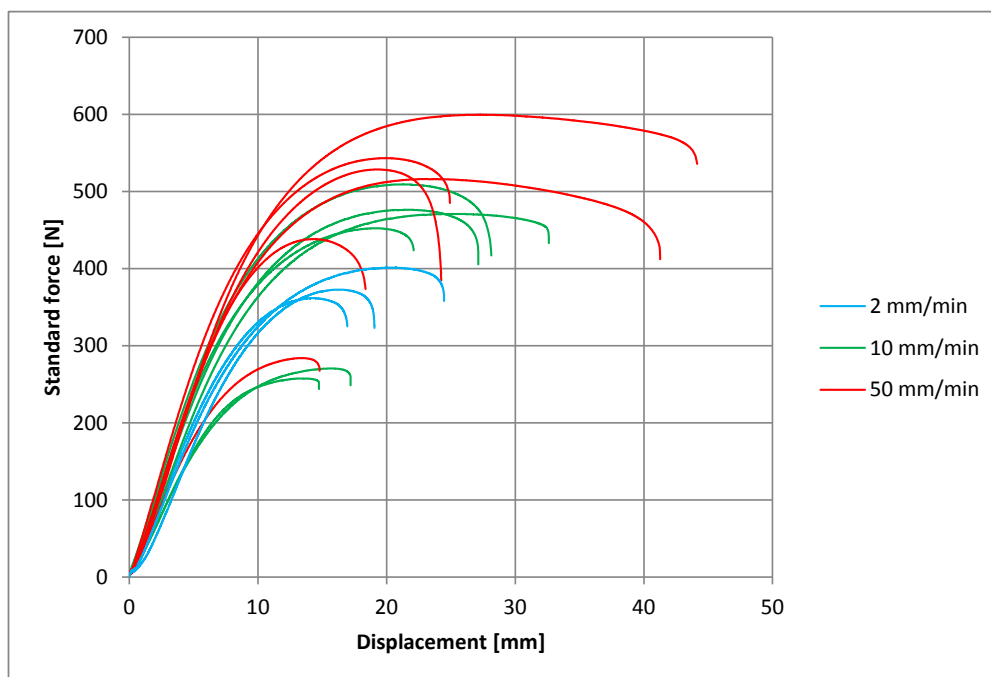


Figure B-9: Raw results: tensile transverse test, 14 days.

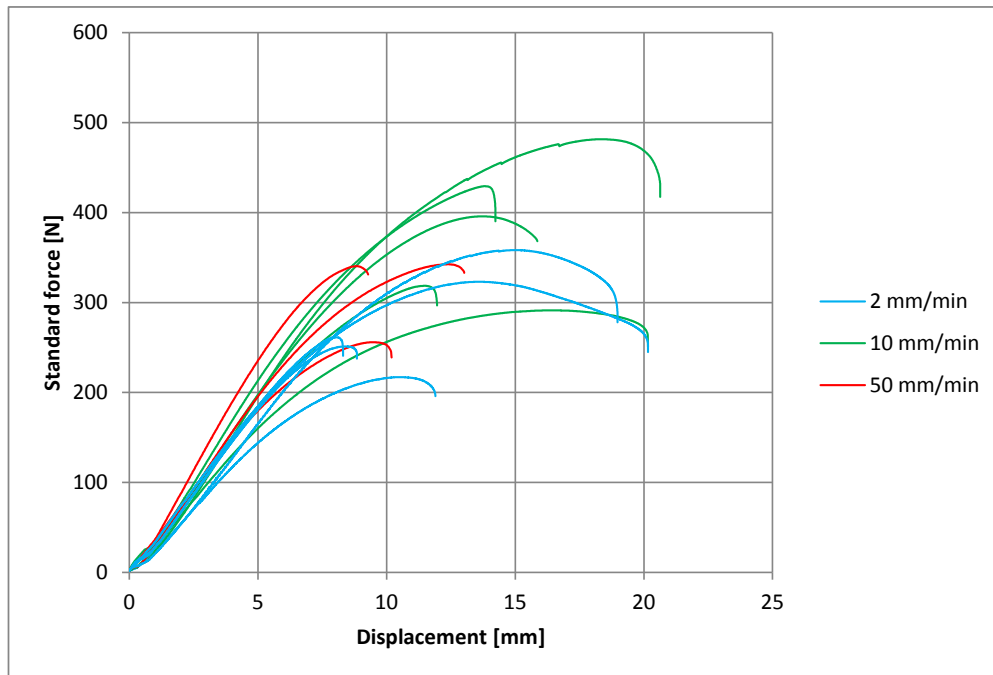


Figure B-10: Raw results: tensile transverse test, 28 days.

B.3 Flexural longitudinal tests

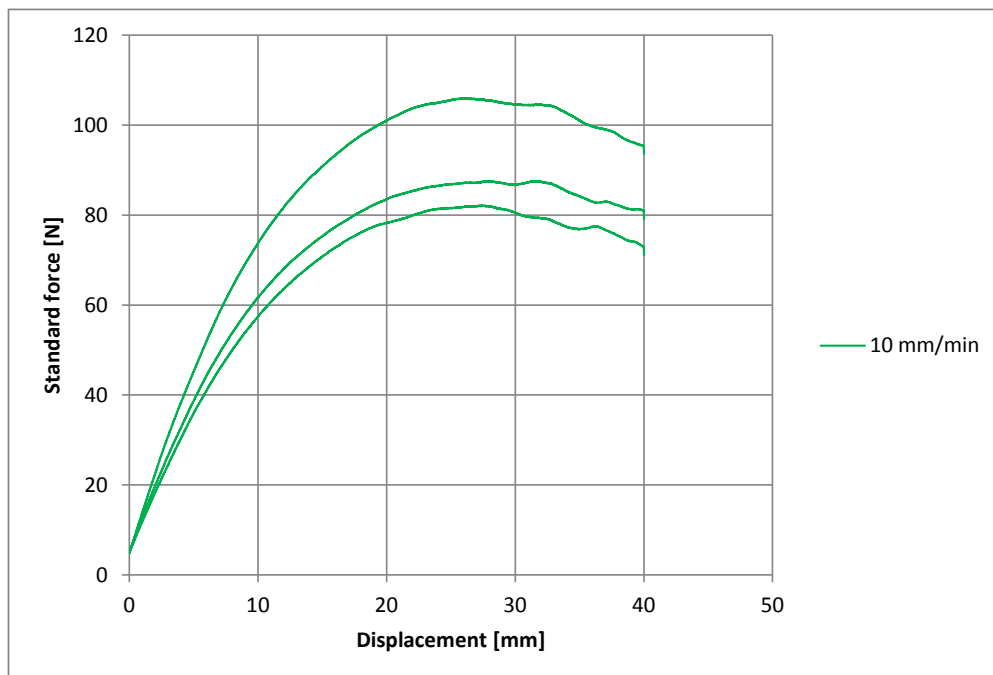


Figure B-11: Raw results: flexural longitudinal test, 1 day.

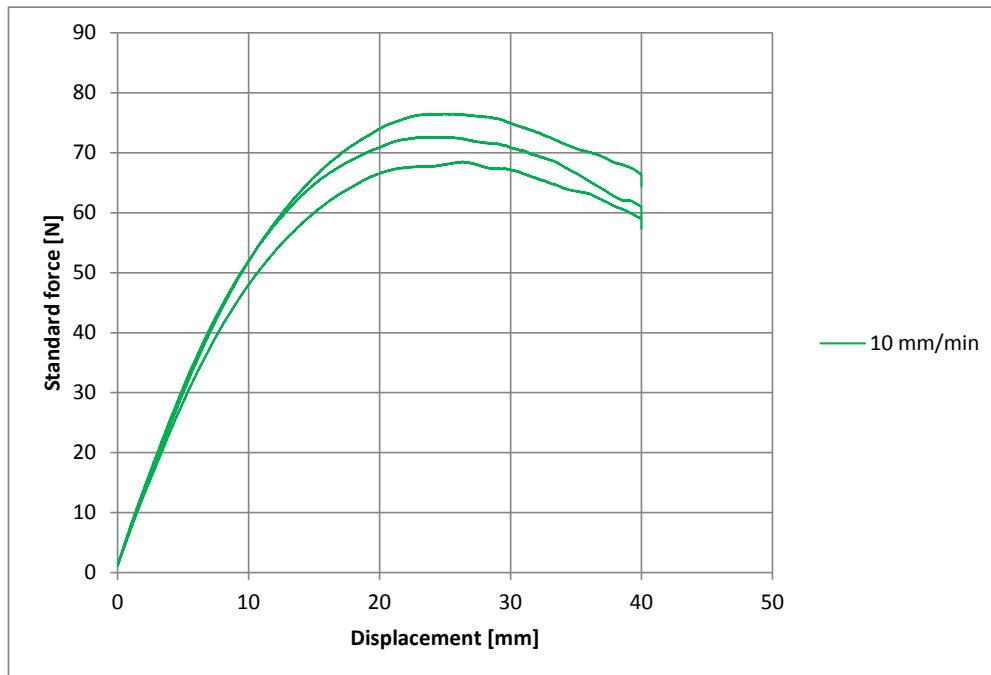


Figure B-12: Raw results: flexural longitudinal test, 3 days.

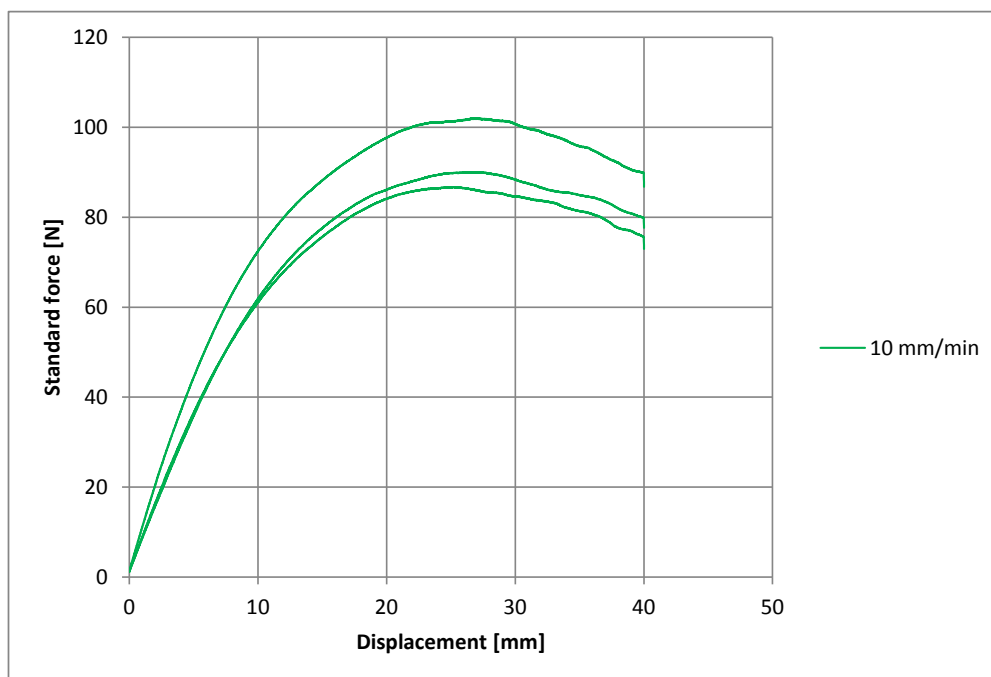


Figure B-13: Raw results: flexural longitudinal test, 7 days.

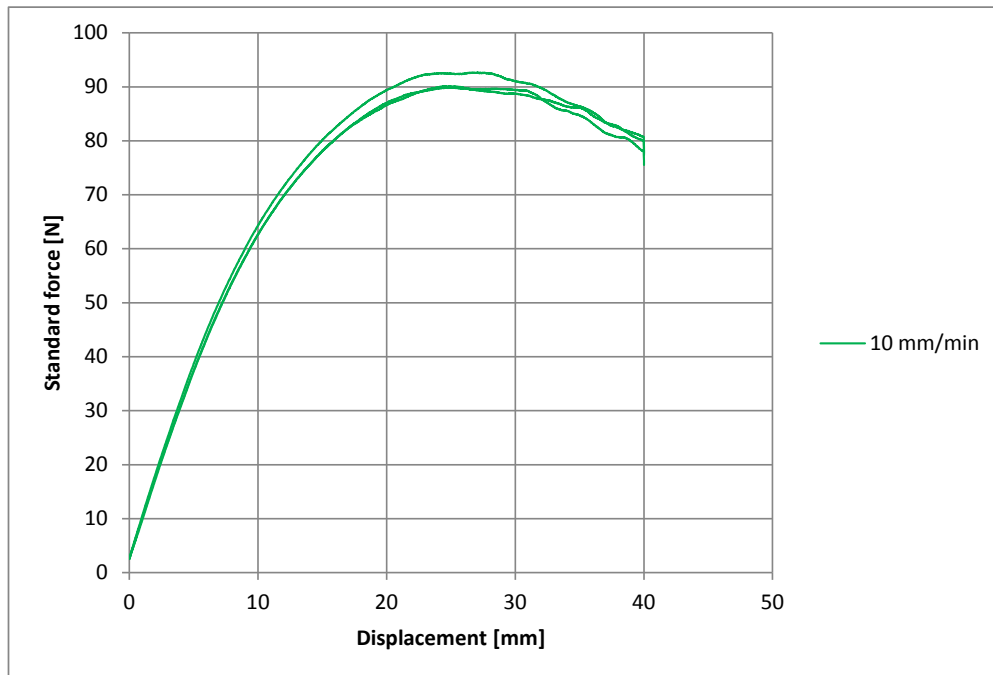


Figure B-14: Raw results: flexural longitudinal test, 14 days.

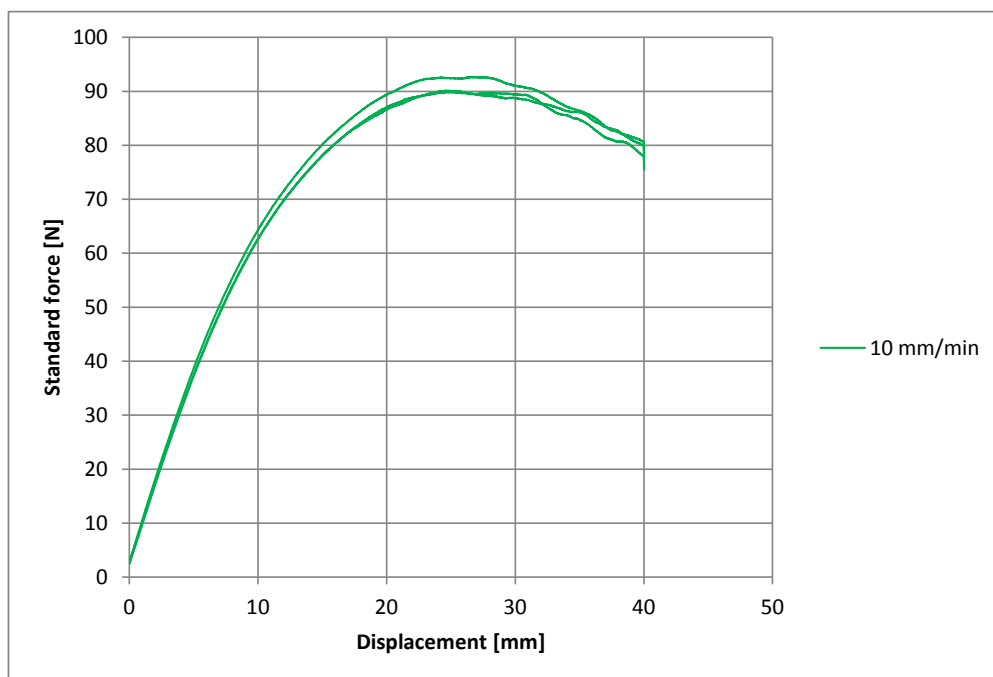


Figure B-15: Raw results: flexural longitudinal test, 28 days.

B.4 Flexural transverse tests

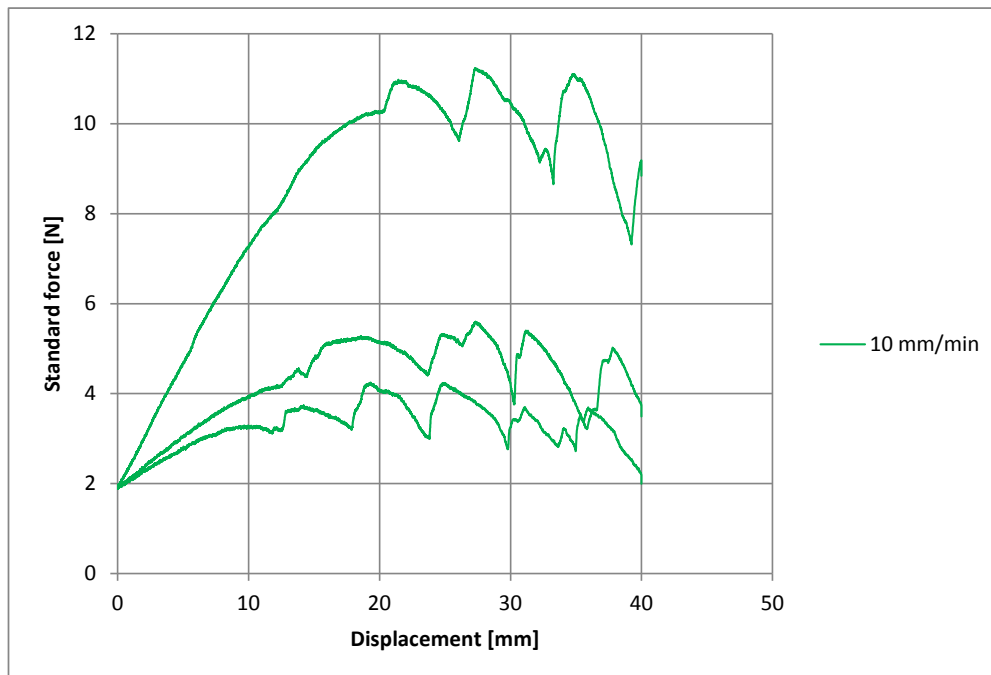


Figure B-16: Raw results: flexural transverse test, 1 day.

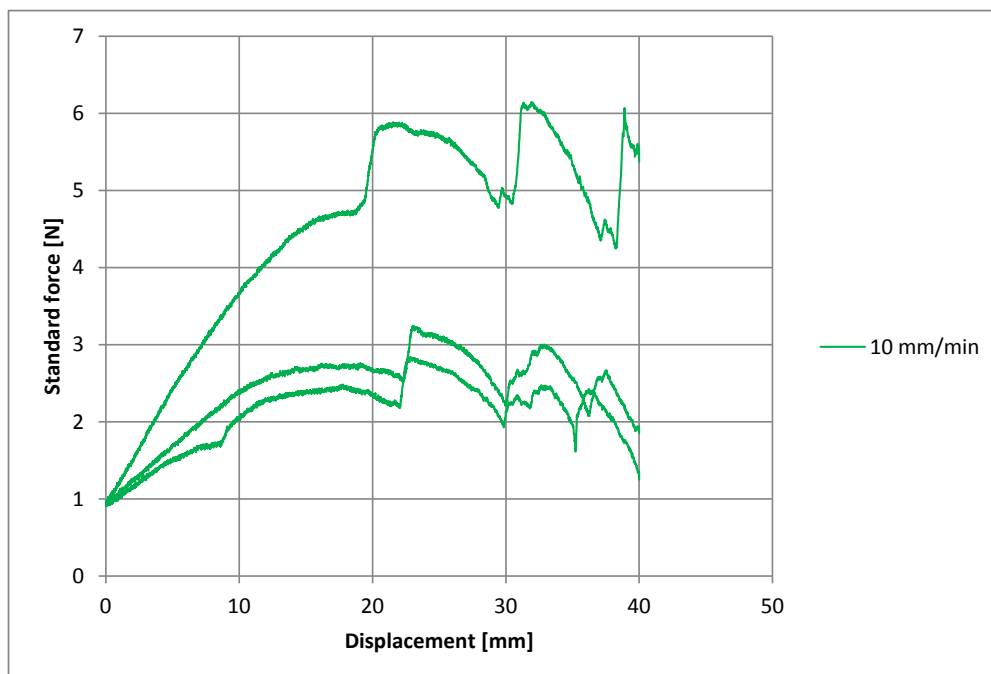


Figure B-17: Raw results: flexural transverse test, 3 days.

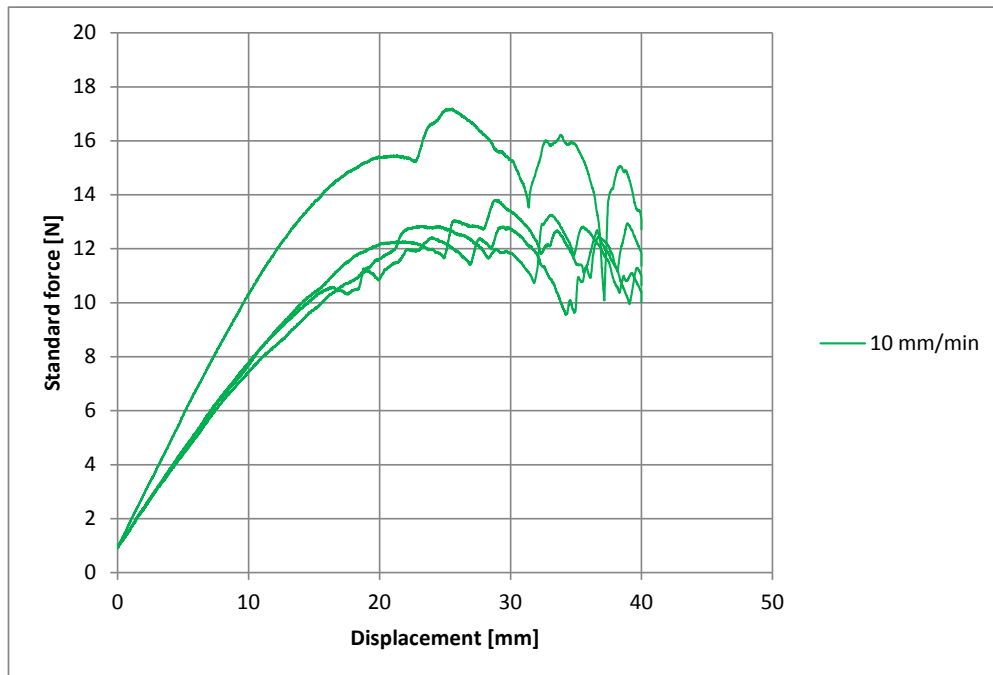


Figure B-18: Raw results: flexural transverse test, 7 days.

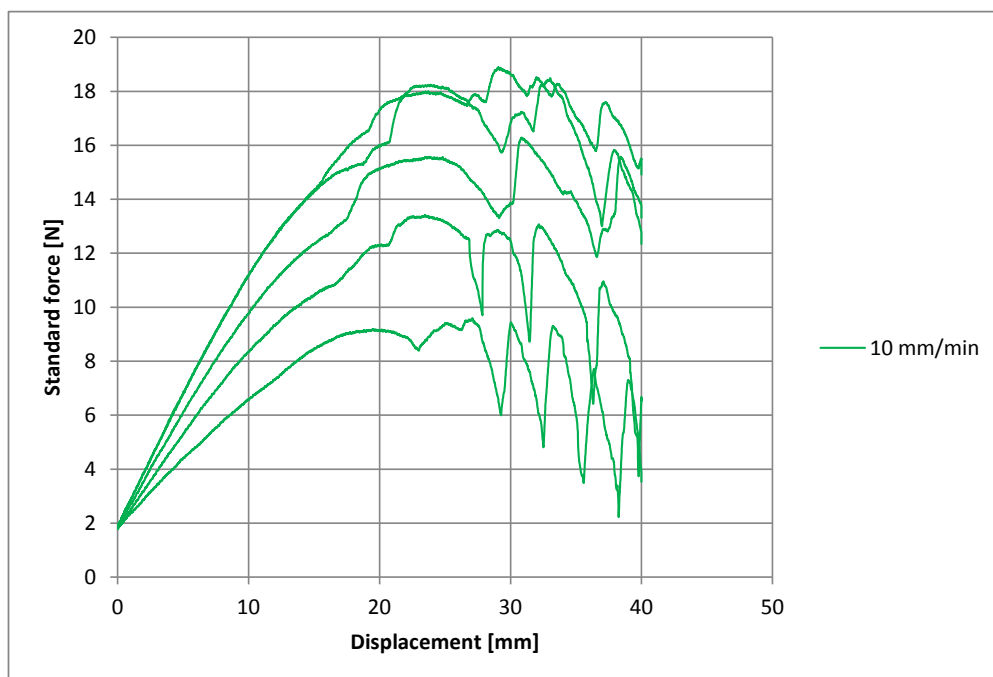


Figure B-19: Raw results: flexural transverse test, 14 days.

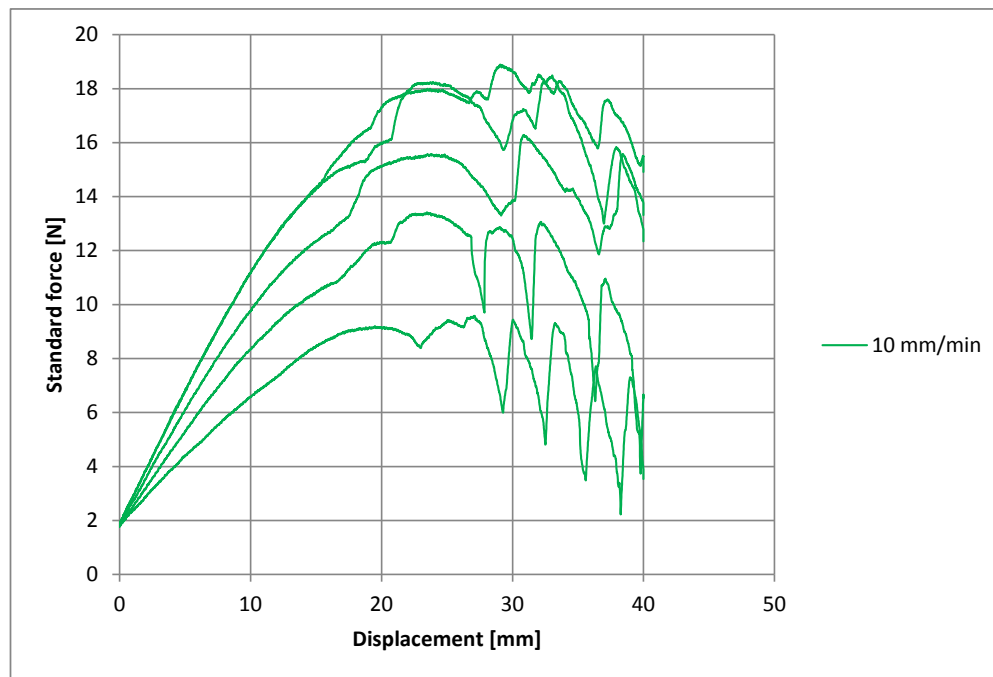


Figure B-20: Raw results: flexural transverse test, 28 days.

B.5 Shear along the print

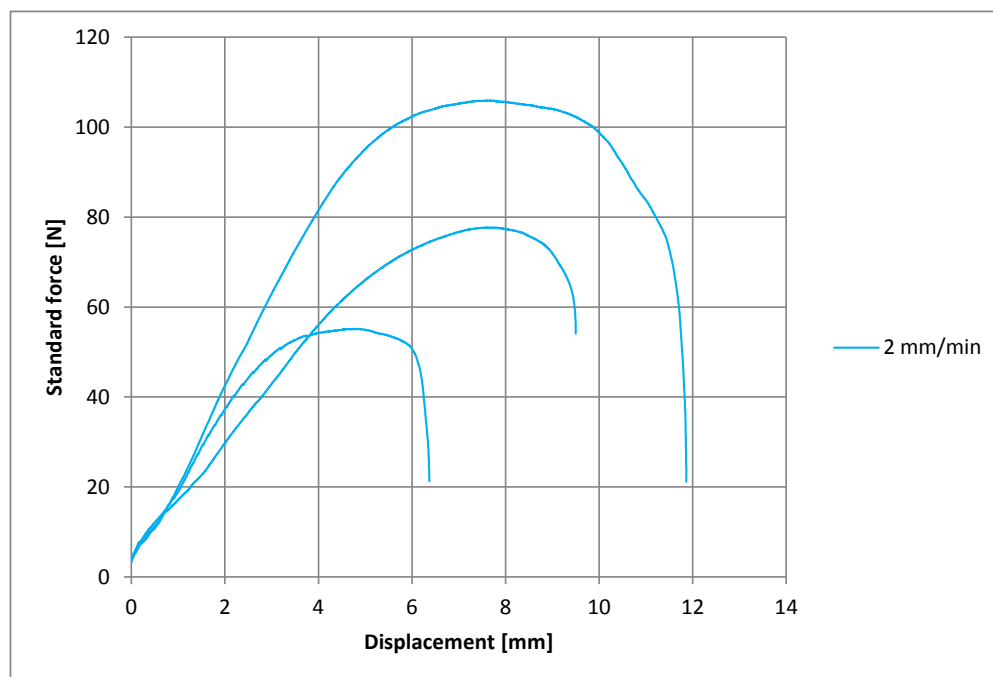


Figure B-21: Raw results: shear along the print test, 1 day.

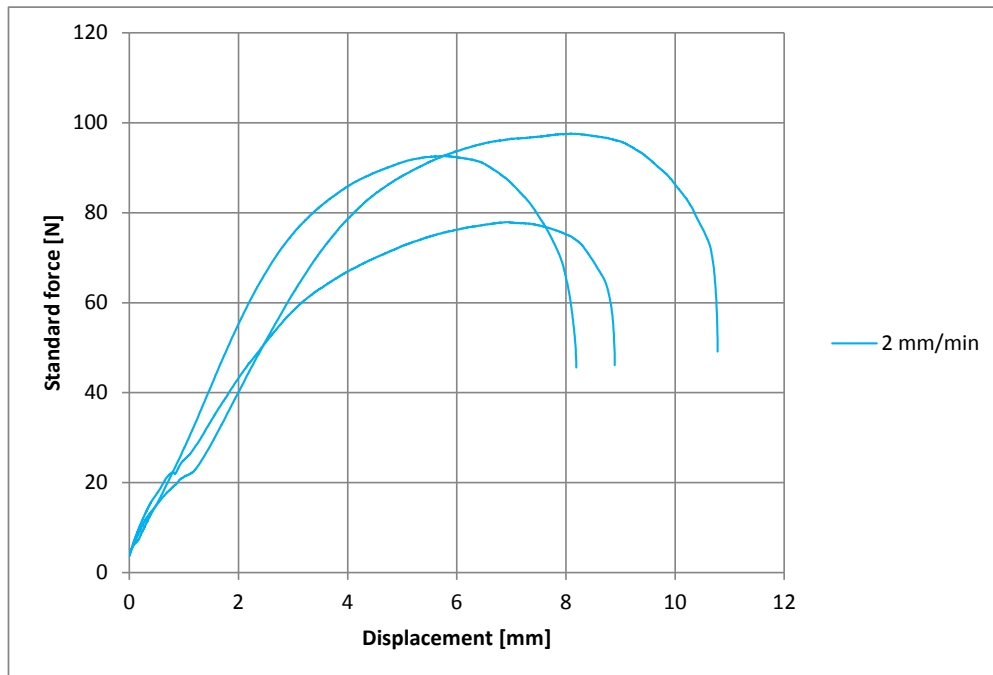


Figure B-22: Raw results: shear along the print test, 3 days.

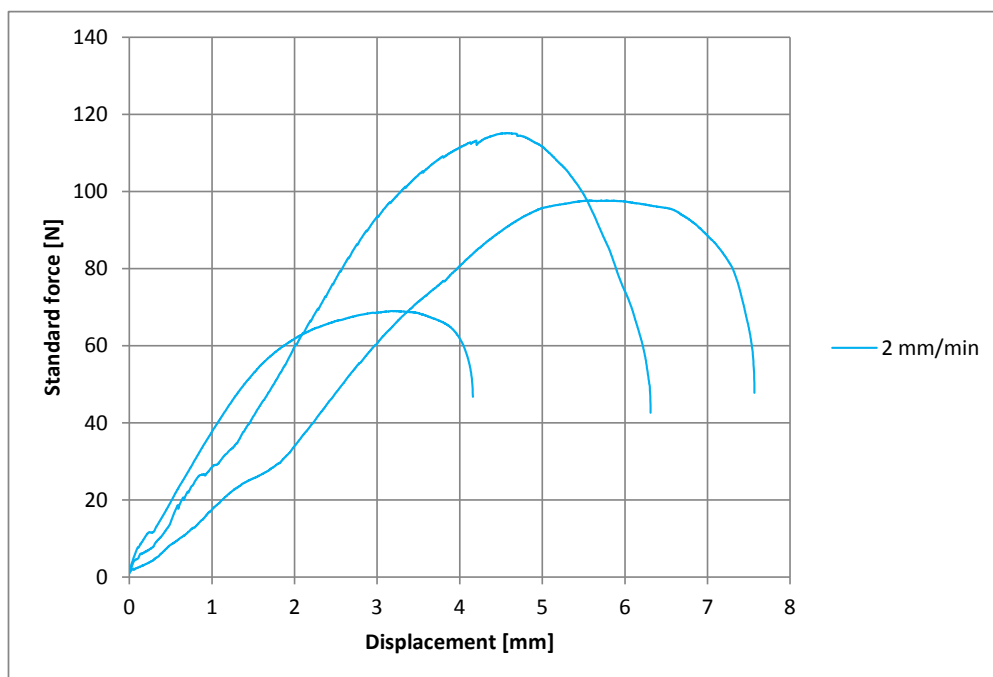


Figure B-23: Raw results: shear along the print test, 7 days.

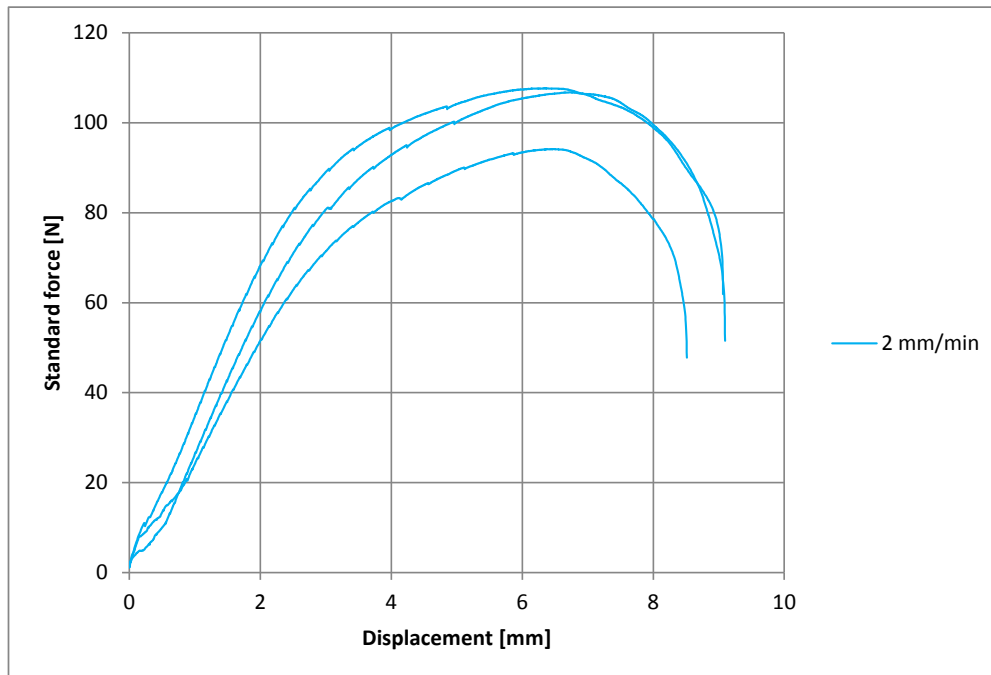


Figure B-24: Raw results: shear along the print test, 14 days.

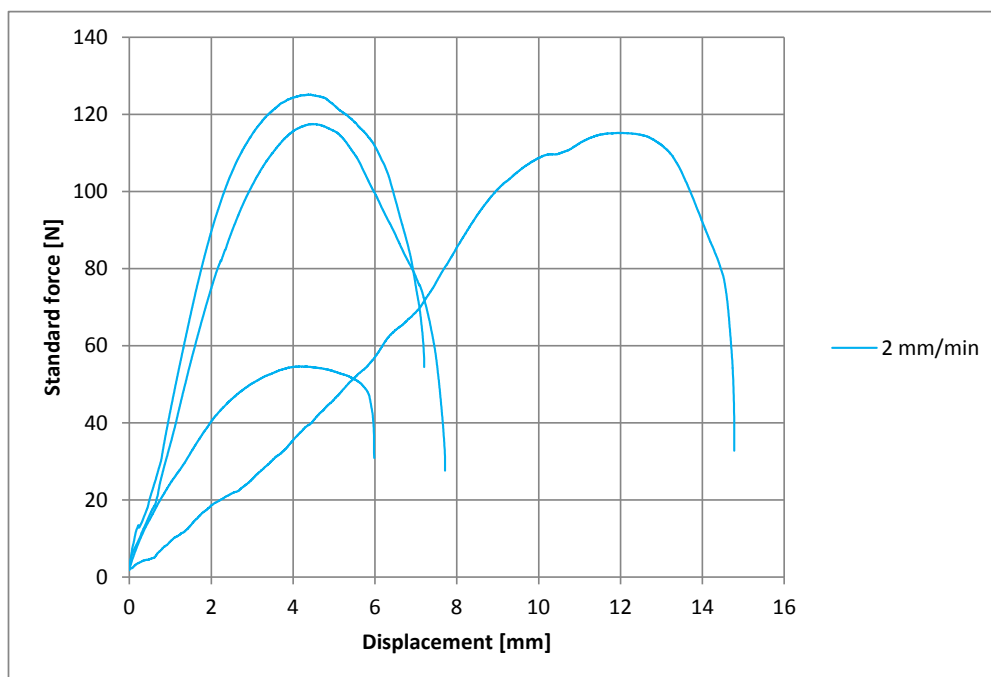


Figure B-25: Raw results: shear along the print test, 28 days.

B.6 Shear across the print

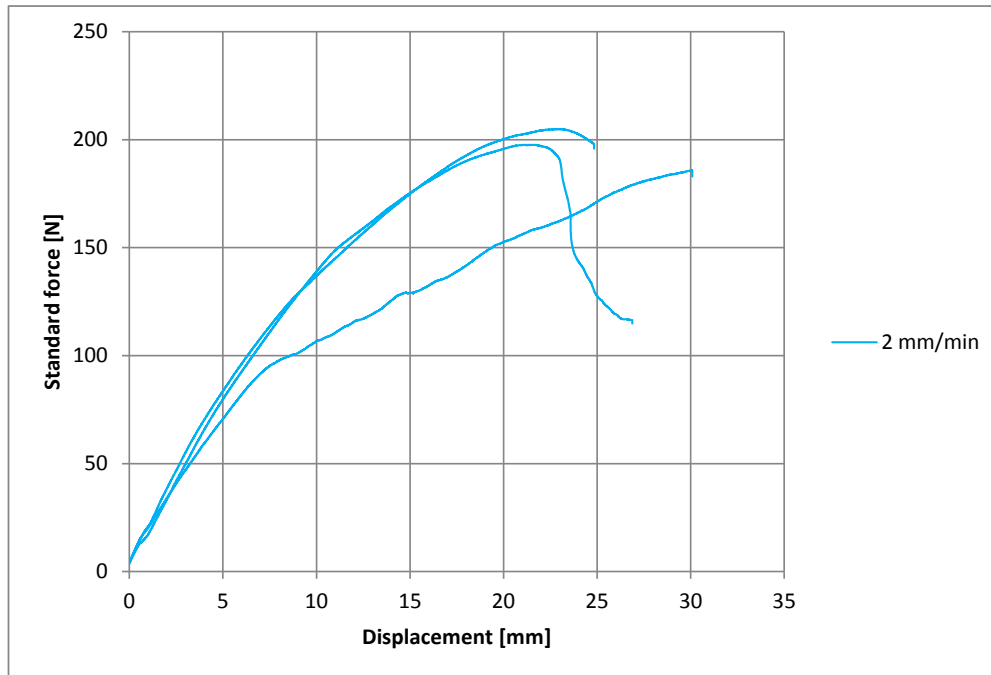


Figure B-26: Raw results: shear across the print test, 1 day.

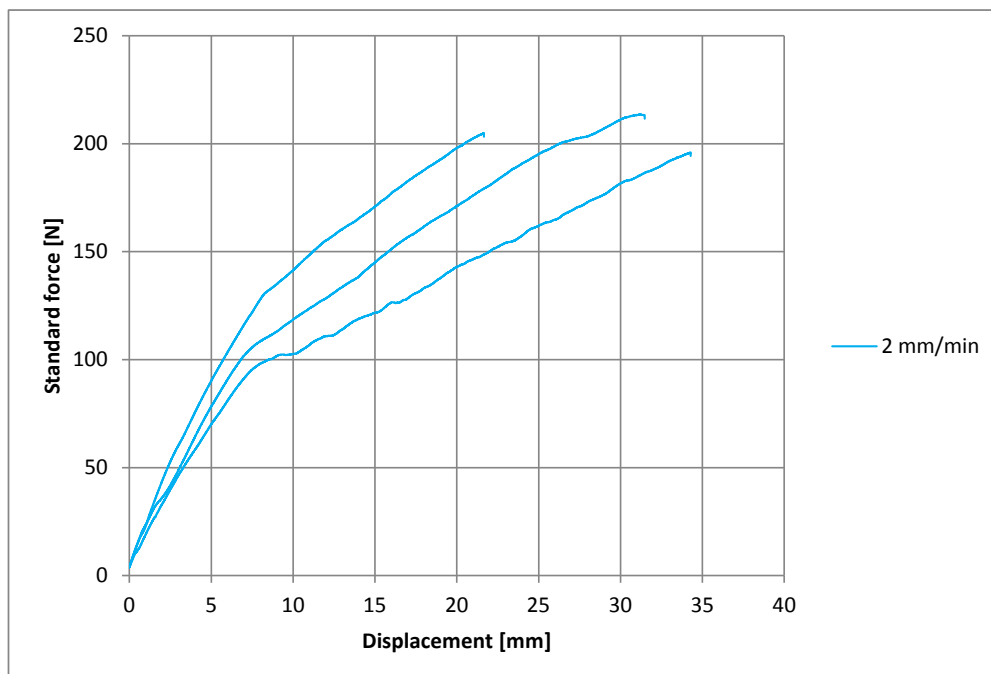


Figure B-27: Raw results: shear across the print test, 3 days.

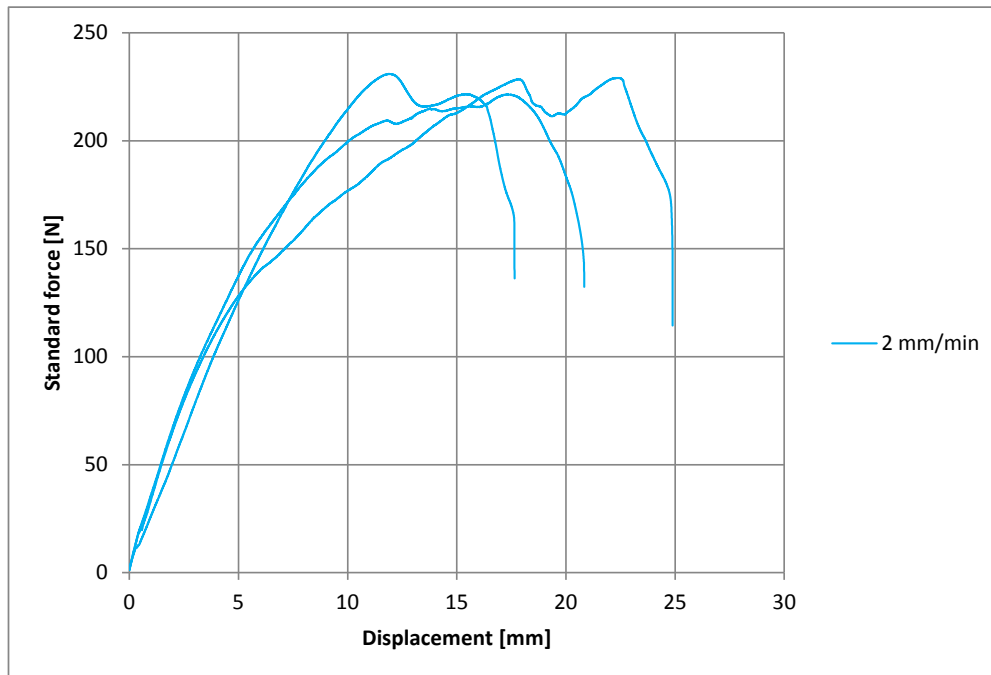


Figure B-28: Raw results: shear across the print test, 7 days.

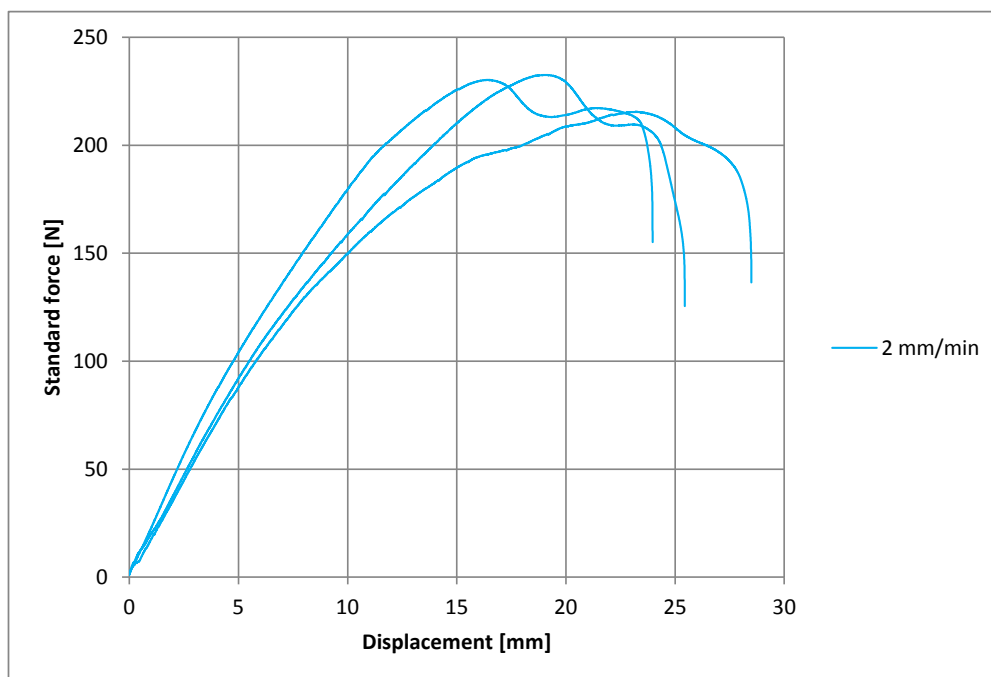


Figure B-29: Raw results: shear across the print test, 14 days.

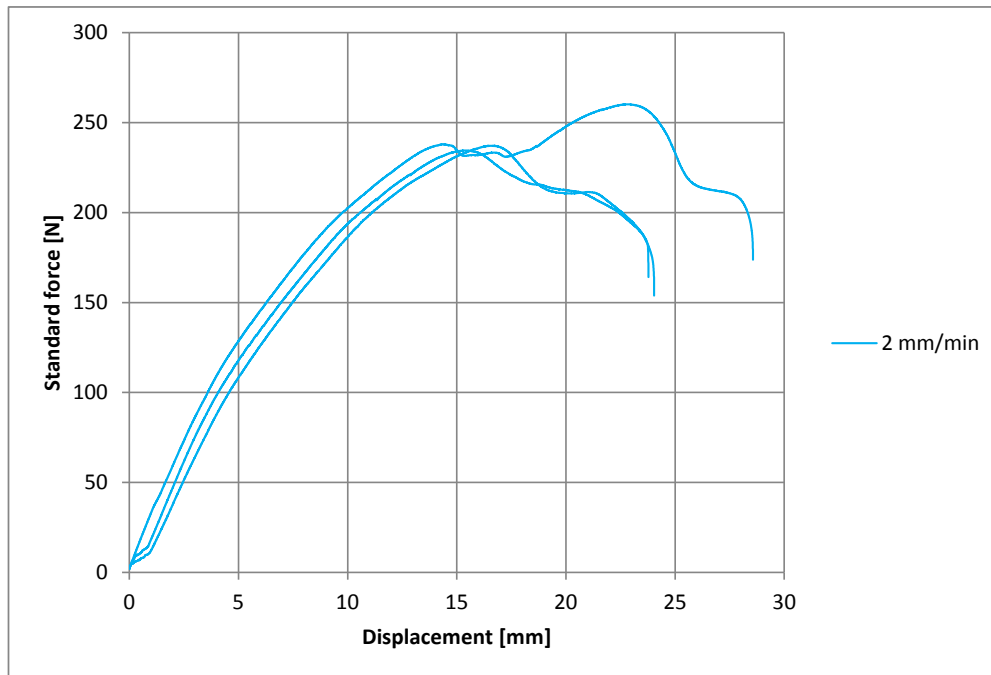


Figure B-30: Raw results: shear across the print test, 28 days.

B.7 Creep test

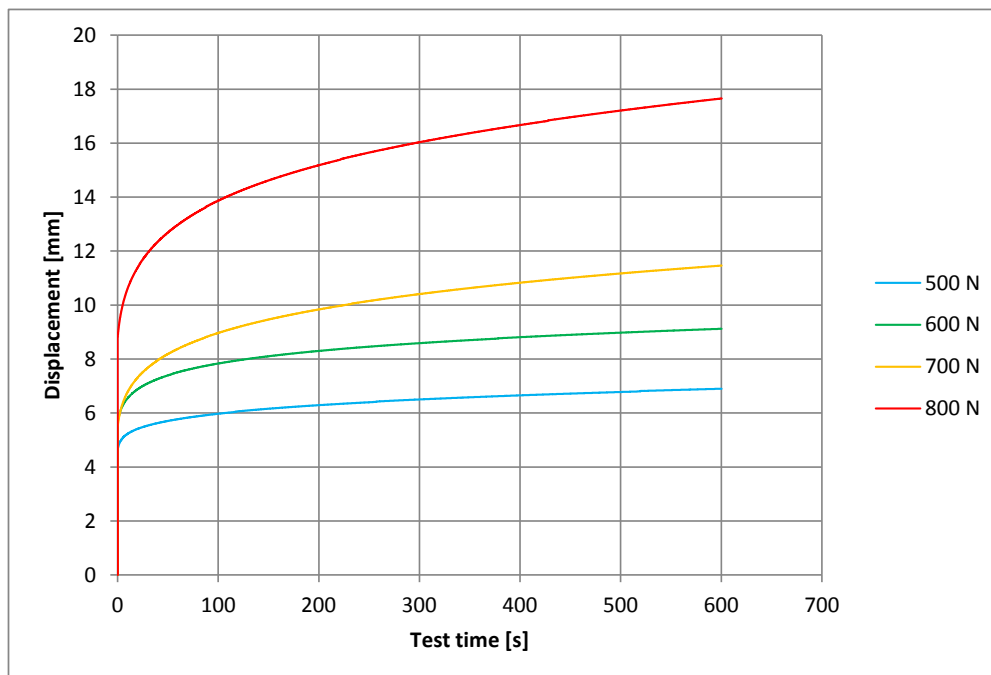


Figure B-31: Raw results: creep test, 1 day.

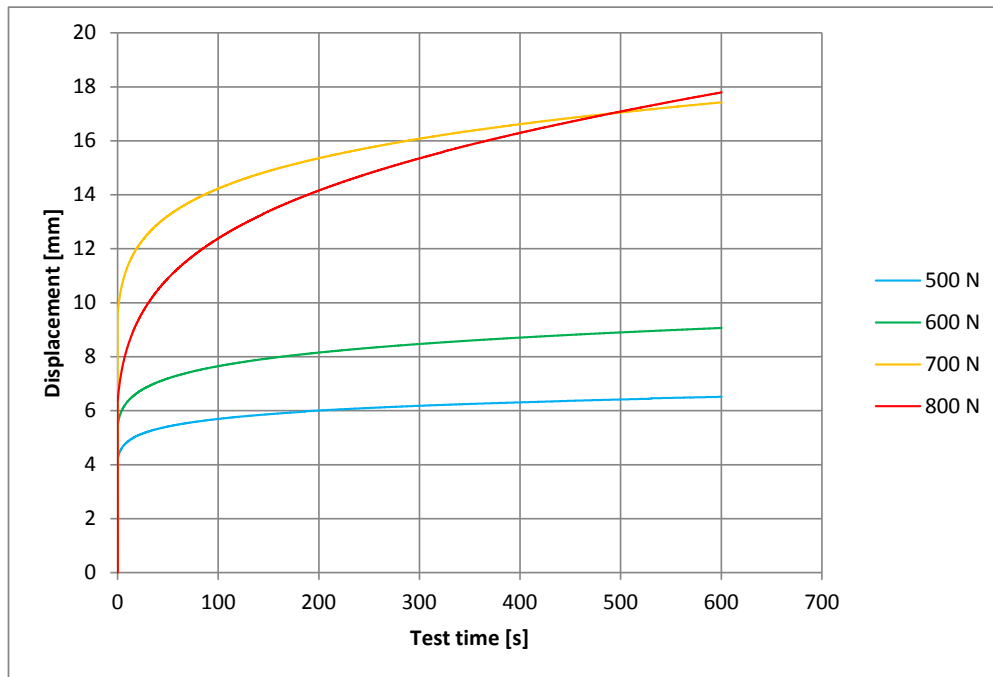


Figure B-32: Raw results: creep test, 3 days.

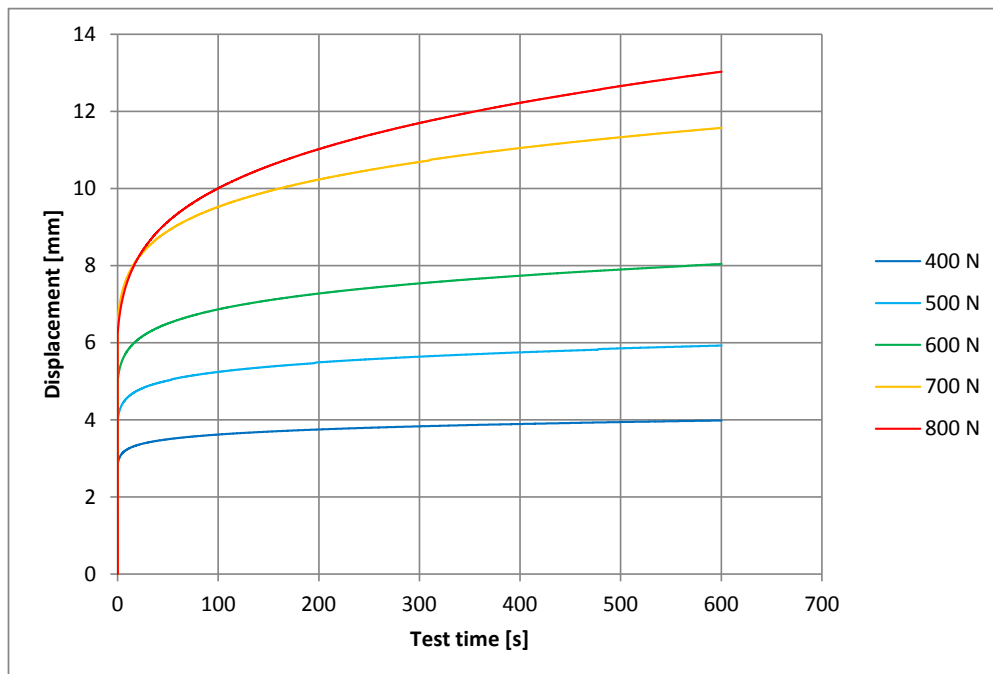


Figure B-33: Raw results: creep test, 7 days.

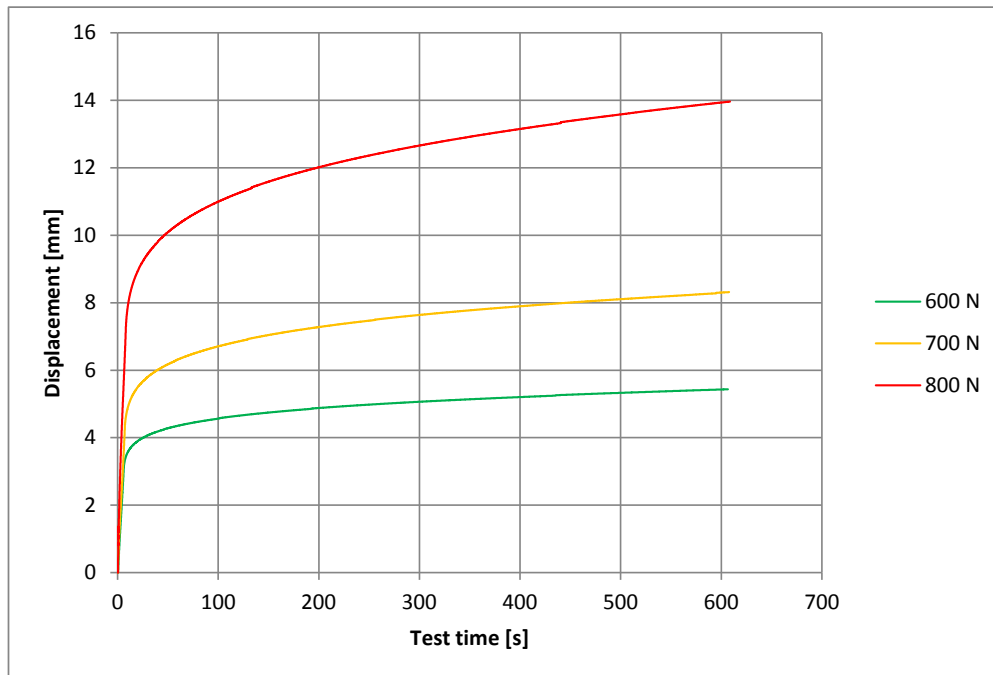


Figure B-34: Raw results: creep test, 14 days.

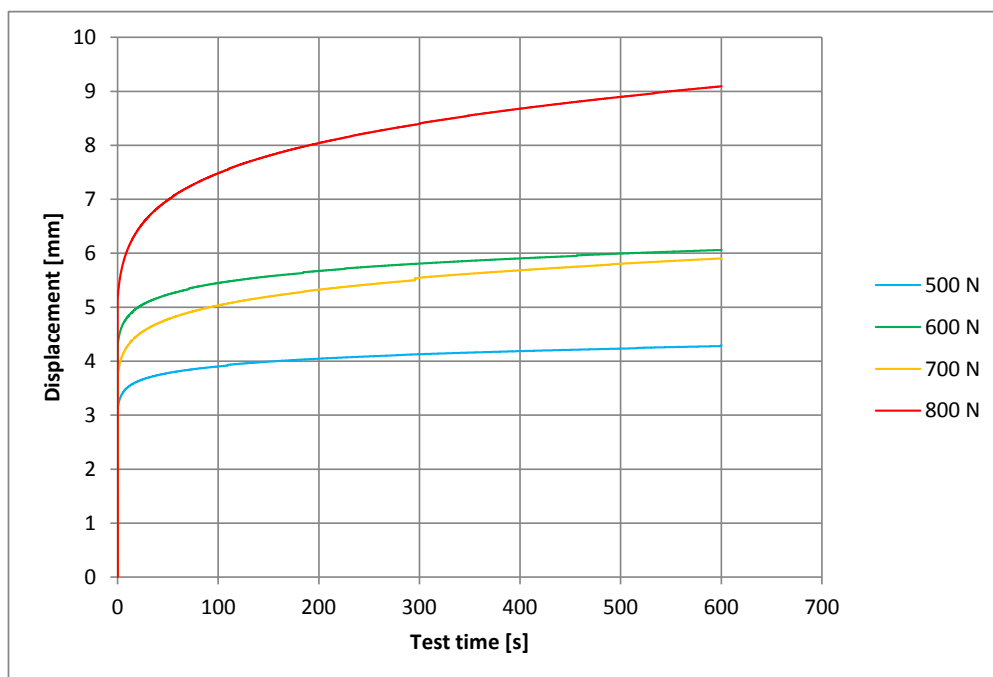


Figure B-35: Raw results: creep test, 28 days.

B.8 Thermal sensitivity tests

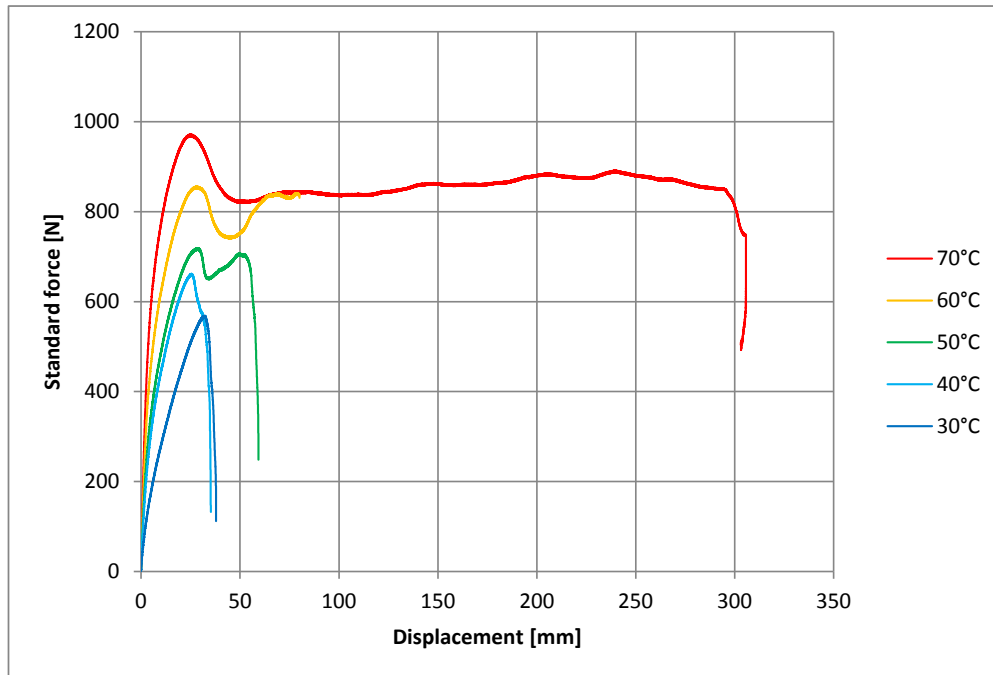


Figure B-36: Raw results: thermal sensitivity test, 1 day.

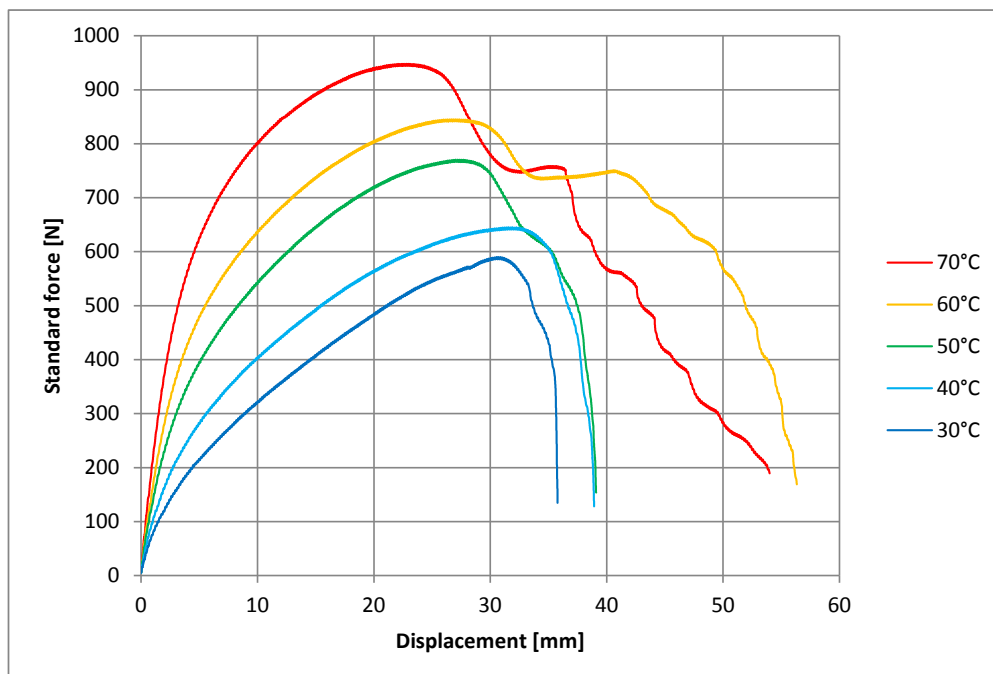


Figure B-37: Raw results: thermal sensitivity test, 3 days.

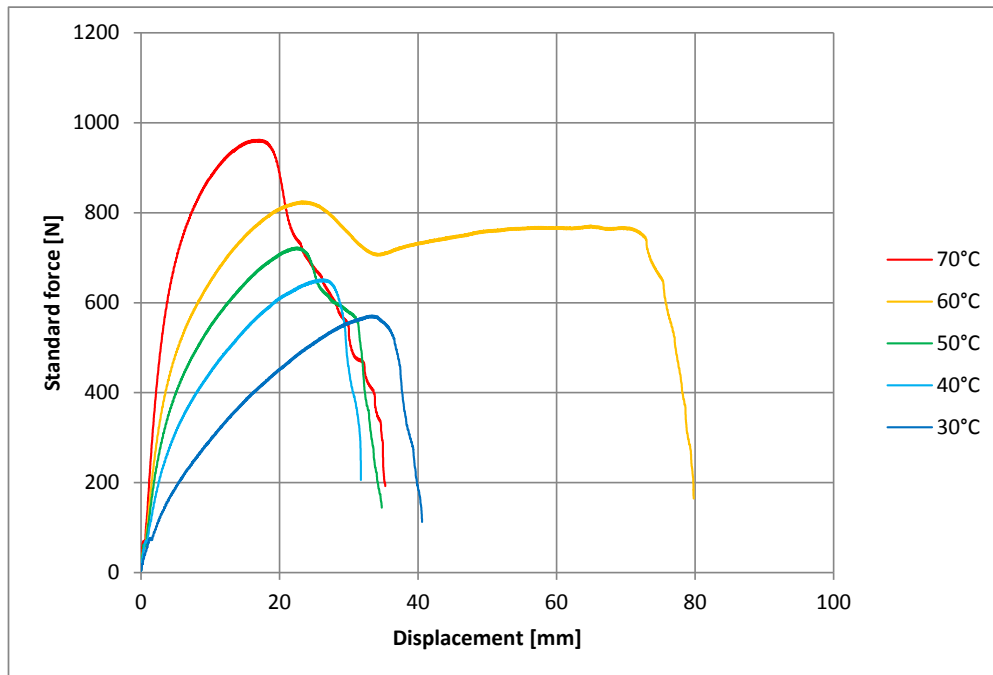


Figure B-38: Raw results: thermal sensitivity test, 7 days.

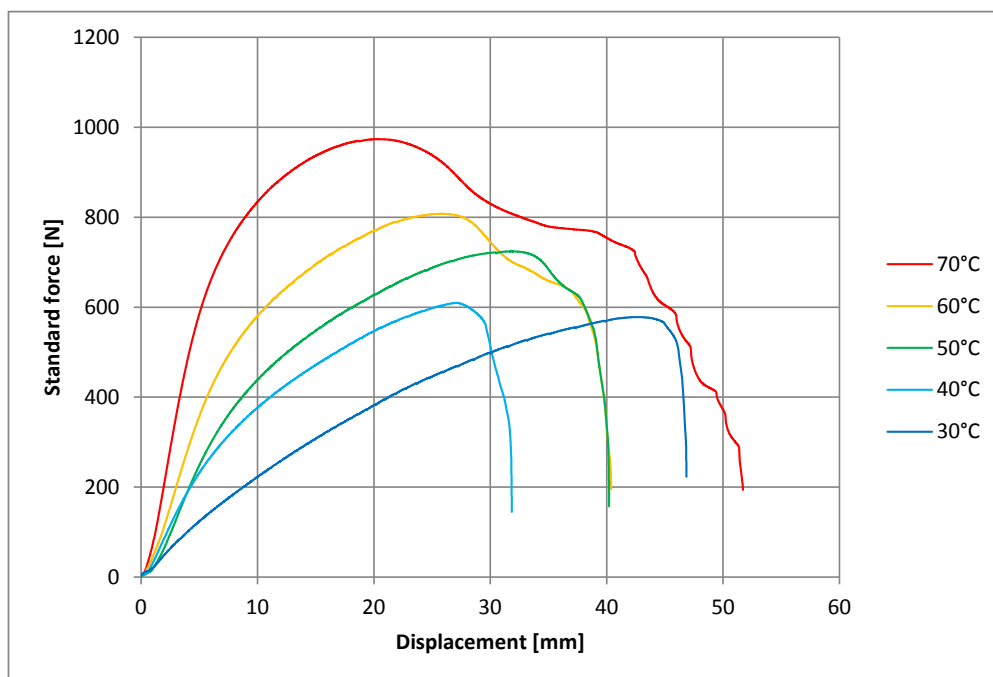


Figure B-39: Raw results: thermal sensitivity test, 14 days.

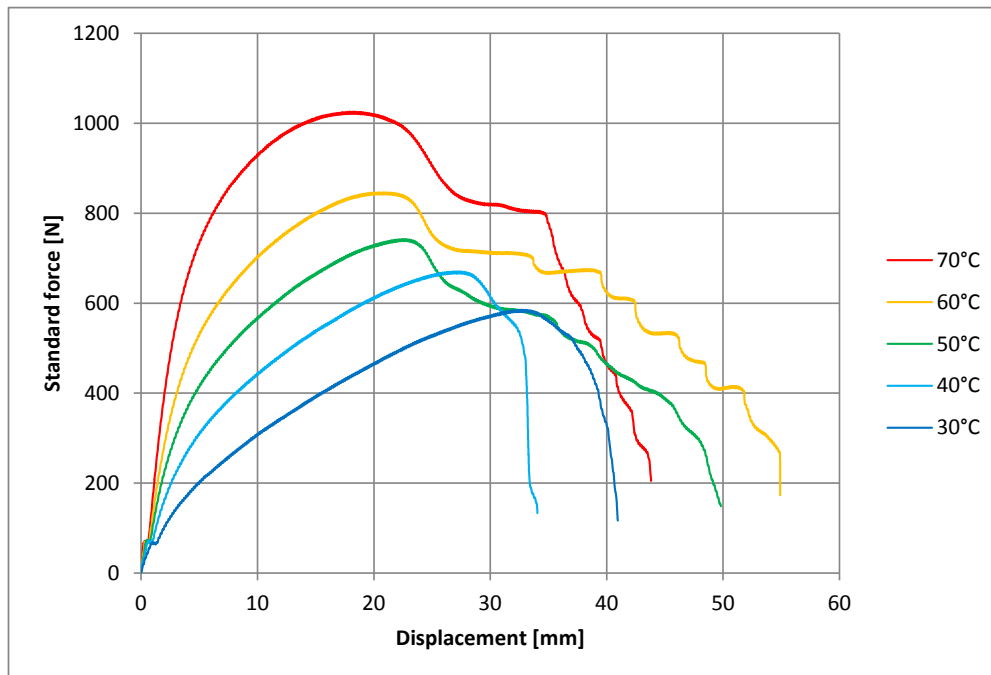


Figure B-40: Raw results: thermal sensitivity test, 28 days.

Multiplication factors for cross section reclassification

This appendix contains information about multiplication factors recommended for use in transition between different classes. In the beginning of each section formulas defining the relationships are given, followed by matrices consisting of resultant values. Relevant numbers provided in Table 5-1 are used in calculations.

C.1 Stiffness

Longitudinal tensile stiffness:

$$EA_{L,n} = \frac{A_{L,n}}{A_{L,m}} EA_{L,m} \quad (\text{C.1})$$

Since A_L is the same for all classes, a global value of 1.00 is used.

Transverse tensile stiffness:

$$EA_{T,n} = \frac{t_{min,n}}{t_{min,m}} EA_{T,m} \quad (\text{C.2})$$

Longitudinal flexural stiffness:

$$EI_{L,n} = \frac{I_{L,n}}{I_{L,m}} EI_{L,m} \quad (\text{C.3})$$

Transverse flexural stiffness:

$$EI_{T,n} = \frac{t_{min,n}^3}{t_{min,m}^3} EI_{T,m} \quad (\text{C.4})$$

From To	1	2	3	4	5
1	1.00	1.11	1.24	1.41	1.64
2	0.90	1.00	1.12	1.28	1.48
3	0.80	0.89	1.00	1.14	1.32
4	0.71	0.78	0.88	1.00	1.16
5	0.61	0.68	0.76	0.86	1.00

Table C-1: Reclassification: transverse tensile stiffness.

From To	1	2	3	4	5
1	1.00	1.12	1.09	0.88	0.79
2	0.89	1.00	0.97	0.79	0.70
3	0.92	1.03	1.00	0.81	0.72
4	1.14	1.27	1.24	1.00	0.90
5	1.27	1.42	1.38	1.12	1.00

Table C-2: Reclassification: longitudinal flexural stiffness.

From To	1	2	3	4	5
1	1.00	1.36	1.92	2.83	4.41
2	0.73	1.00	1.41	2.08	3.24
3	0.52	0.71	1.00	1.47	2.30
4	0.35	0.48	0.68	1.00	1.56
5	0.23	0.31	0.43	0.64	1.00

Table C-3: Reclassification: transverse flexural stiffness.

C.2 Strength

Longitudinal tensile strength:

$$F_{L,n} = \frac{A_{L,n}}{A_{L,m}} F_{L,m} \quad (\text{C.5})$$

Since A_L is the same for all classes, a global value of 1.00 is used.

Transverse tensile strength:

$$F_{T,n} = \frac{t_{min,n}}{t_{min,m}} F_{T,m} \quad (\text{C.6})$$

Longitudinal flexural strength:

$$M_{L,n} = \frac{I_{L,n} h_{max,m}}{I_{L,m} h_{max,n}} M_{L,m} \quad (C.7)$$

Transverse flexural strength:

$$M_{T,n} = \frac{t_{min,n}^2}{t_{min,m}^2} M_{T,m} \quad (C.8)$$

Shear strength across the print:

$$S_{x,n} = \frac{A_{L,n}}{A_{L,m}} S_{x,m} \quad (C.9)$$

Since A_L is the same for all classes, a global value of 1.00 is used.

Shear strength along the print:

$$S_{y,n} = \frac{t_{min,n}}{t_{min,m}} S_{y,m} \quad (C.10)$$

From To	1	2	3	4	5
1	1.00	1.11	1.24	1.41	1.64
2	0.90	1.00	1.12	1.28	1.48
3	0.80	0.89	1.00	1.14	1.32
4	0.71	0.78	0.88	1.00	1.16
5	0.61	0.68	0.76	0.86	1.00

Table C-4: Reclassification: transverse tensile strength.

From To	1	2	3	4	5
1	1.00	1.19	1.22	1.04	0.98
2	0.84	1.00	1.03	0.87	0.82
3	0.82	0.97	1.00	0.85	0.80
4	0.96	1.14	1.17	1.00	0.94
5	1.02	1.21	1.25	1.06	1.00

Table C-5: Reclassification: longitudinal flexural strength.

From To	1	2	3	4	5
1	1.00	1.23	1.54	2.00	2.69
2	0.81	1.00	1.26	1.63	2.19
3	0.65	0.80	1.00	1.29	1.74
4	0.50	0.61	0.77	1.00	1.35
5	0.37	0.46	0.57	0.74	1.00

Table C-6: Reclassification: transverse flexural strength.

From To	1	2	3	4	5
1	1.00	1.11	1.24	1.41	1.64
2	0.90	1.00	1.12	1.28	1.48
3	0.80	0.89	1.00	1.14	1.32
4	0.71	0.78	0.88	1.00	1.16
5	0.61	0.68	0.76	0.86	1.00

Table C-7: Reclassification: shear strength along the print.

Appendix D

Effective length of the tensile specimen

The original geometry of the tensile specimen is shown in Figure D-1. Tested zone between the clamps is marked with a double hatch. Considering given coordinate system, curvature of the rounded part of its perimeter can be expressed with a circle equation:

$$x^2 + (y - 25)^2 = 25^2 \quad (\text{D.1})$$

Which for the relevant section of the circle can be translated into:

$$y = 25 - \sqrt{625 - x^2} \quad (\text{D.2})$$

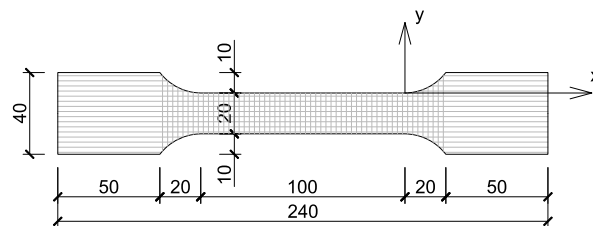


Figure D-1: Tensile specimen: original dimensions.

Knowing the function describing the edge of a specimen, cross-sectional area function can be derived for an arbitrary thickness t :

$$A(x) = \begin{cases} t(20 + 50 - 2\sqrt{625 - (20 - x)^2}) & 0 \leq x \leq 20 \\ 20t & 20 < x < 120 \\ t(20 + 50 - 2\sqrt{625 - (x - 120)^2}) & 120 \leq x \leq 140 \end{cases} \quad (\text{D.3})$$

Assuming linear theory, following force/displacement relationship is fulfilled for a member with variable cross-sectional area $A(x)$, length L , force F and Young's modulus E :

$$\delta = \int_0^L \frac{N dx}{EA(x)} \quad (\text{D.4})$$

For given geometry this can be written as:

$$\delta_{eq} = \frac{N}{Et} \int_0^{20} \frac{dx}{70 - 2\sqrt{625 - (20 - x)^2}} + \int_{20}^{120} \frac{dx}{20} + \int_{120}^{140} \frac{dx}{70 - 2\sqrt{625 - (x - 120)^2}} \quad (\text{D.5})$$

Which yields $\delta_{eq} = \frac{6.60N}{Et}$. Comparing this result with a 140 mm long section with uniform cross section of $20t$:

$$\delta_{140} = \frac{N}{Et} \int_0^{140} \frac{dx}{140} \quad (\text{D.6})$$

Equal to $\delta_{140} = \frac{7.00N}{Et}$, the equivalent specimen length can be found:

$$L_{eq} = 140 \frac{\delta_{eq}}{\delta_{140}} \quad (\text{D.7})$$

This finally gives $L_{eq} = 132$ mm.

Appendix E

Statistical foundations for derivation of the material characteristics

The *corrected sample standard deviation* used for processing of population samples is given by the formula:

$$s = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2} \quad (\text{E.1})$$

Where:

N - sample length

\bar{x} - mean value of x

Coefficient of variation, represented in this report with CV is calculated from the following expression:

$$CV = \frac{s}{\bar{x}} \quad (\text{E.2})$$

According to NEN-EN 1990 (Nederlands Normalisatie-instituut, 2008), the design value of a material property X_d is derived from:

$$X_d = \frac{\eta_d}{\gamma_m} m_X \{1 - k_n V_X\} \quad (\text{E.3})$$

Where $\frac{\eta_d}{\gamma_m}$ is the component linking characteristic and design values X_k and X_d :

$$X_d = \frac{\eta_d}{\gamma_m} X_k \quad (\text{E.4})$$

Consequently, the characteristic value formula can be derived:

$$X_k = m_X \{1 - k_n V_X\} \quad (\text{E.5})$$

Where:

m_X - mean value of a property ($m_X = \bar{x}$)

k_n - value dependent on sample length according to Table E-1

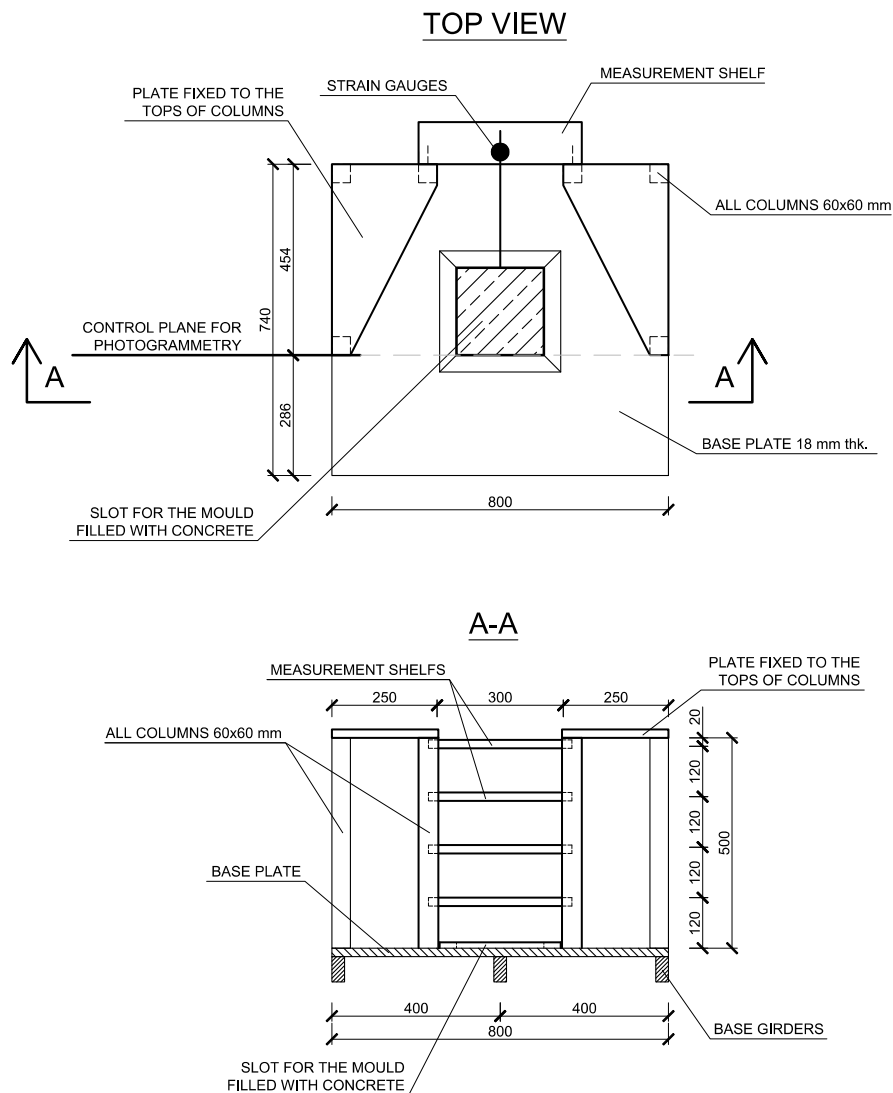
V_X - coefficient of variation ($V_X = CV$)

N	1	2	3	4	5	6	8	10	20	30	∞
k_n	2.31	2.01	1.89	1.83	1.80	1.77	1.74	1.72	1.68	1.67	1.64

Table E-1: Values of k_n for the 5% characteristic value. Extract from Table D1 in NEN-EN 1990 (Nederlands Normalisatie-instituut, 2008).

Appendix F

Single layer mould test setup



Appendix G

Standard concrete mixture specification

Receptgegevens

Code: C25C3S316J
Naam: C25C3S316J
Sterkte klasse: C20/25
Statistische milieu klasse: XC3 Matige vochtigheid
Normenset: NEN-EN 206/NEN 8005
Betonfamilie:
Zeeflijnen set: 0/16 (A-C)
Receptberekingsmethode: Niet lineair, minimale kosten

Resultaten

Water-bindmiddel factor:	0,550	Totaal massa toeslag (droog):	1755,5kg
Water-cement factor:	0,564	Totaal massa grind (droog):	988,9kg
Max. wbf door sterkte:	0,725	Totaal massa zand (droog):	766,5kg
Gevraagde sterkte:	31,6N/mm ²	Zand (t.o.v. toeslag):	43,48% V/V
Berekende sterkte:	37,5N/mm ²	Fijnheidmodulus:	4,871
Gebaseerd op BQS:	Geen	Totaal hulpstof:	0,0L
Bindmiddelgehalte:	332kg (Bindmiddel uit vulstof: 8kg)	Chloridegehalte:	0,12% M/M
Cementgehalte:	324kg	Alkaligehalte:	4,63kg
Waterbehoefte:	182,7L	Luchtgehalte:	15,0L
Aanmaakwater:	187,0L	Totaal fijn:	164,0L
Eff. aanmaakwater (zonder slib):	182,7L	Slib (t.o.v. toeslagmassa):	10,59kg / 0,60% M/M
Geabsorbeerd water	28,4L	Totaal volume:	1000L
		CO ₂ -equivalenten	0,114ton

Cement	Droge massa	Te doseren massa	Te doseren volume	Volume fijn	VochtAbsorptie		Totale fractie V/V
CEM III/B 42.5 N LH HS ENCI IJMUIDEN	324kg	324kg	110L	109,9L			10,99%
Totaal Cement	324kg	324kg	110L	109,9L			10,99%
Zand	Droge massa	Te doseren massa	Te doseren volume	Volume fijn	VochtAbsorptie		Totale fractie V/V
Zeezand 0/4 mm	767kg	773kg	291L	32,1L	0,80% M/M	0,80% M/M	29,15%
Totaal Zand	767kg	773kg	291L	32,1L			29,15%
Grind	Droge massa	Te doseren massa	Te doseren volume	Volume fijn	VochtAbsorptie		Totale fractie V/V
Zeegrind 4/16 mm	989kg	1011kg	379L	0,0L	2,25% M/M	2,25% M/M	37,89%
Totaal Grind	989kg	1011kg	379L	0,0L			37,89%
Vulstof	Droge massa	Te doseren massa	Te doseren volume	Volume fijn	VochtAbsorptie		Totale fractie V/V
Poederkoolvliegass (met K-factor)	40kg	40kg	18L	17,8L			1,78%
Totaal Vulstof	40kg	40kg	18L	17,8L			1,78%
Water en slib	Massa bij droge materialen	Te doseren massa	Te doseren volume	Volume fijn	VochtAbsorptie		Totale fractie V/V
Slibwater	193kg	193kg	187L	4,2L			18,70%
Totaal Water en slib	193kg	193kg	187L	4,2L			18,70%
Lucht			15L	0,0L			1,50%
Totaal (5 grondstoffen):		2341kg	1000L	164,0L			100%

Appendix H

Self-compacting concrete mixture specification

Receptgegevens

Code:	2BOGLNZLAW
Naam:	Bogl - N-Z lijn
Omschrijving:	Mix AW: B35 mkl 2 Flowcrete 7/5 16 CEMIII/B 42,5 N LH
Artikelcode:	2BOGLNZLAW
Opmerkingen op bon:	2653
Verwerkingsduur:	
Opmerkingen mengmeester:	
Sterkte klasse:	B35
Statistische milieu klasse:	Milieuklasse 2 (vochtig)
Normenset:	VBT 95
Betonfamilie:	
Zeeffijnen set:	-
Receptberekeningsmethode:	Rekenmethode absoluut

Categorieën

Milieuklasse	Milieuklasse 2 (vochtig)
Gradering	Gradering AB
Maximale korrelgrootte	DMax = 16
Chloride klasse	Chlorideklasse Cl 0,40
Consistentie	Vloeimaatklasse F7

Resultaten

Water-bindmiddel factor:	0,518	Totaal massa toeslag (droog):	1548,0kg
Water-cement factor:	0,555	Totaal massa grind (droog):	717,0kg
Max. wbf door sterkte:	0,543	Totaal massa zand (droog):	831,0kg
Gevraagde sterkte:	41,6N/mm ²	Zand (t.o.v. toeslag):	52,91% V/V
Berekende sterkte:	43,1N/mm ²	Fijnheidmodulus:	4,642
Gebaseerd op BQS:	Geen	Totaal hulpstof:	4,0L
Bindmiddelgehalte:	325kg (Bindmiddel uit vulstof: 22kg)	Chloridegehalte:	0,19% M/M
Cementgehalte:	303kg	Alkaligehalte:	13,51kg
Waterbehoefte:	168,3L	Luchtgehalte:	15,0L
Aanmaakwater:	130,0L	Totaal fijn:	242,9L
Eff. aanmaakwater (zonder slib):	130,0L	Slib (t.o.v. toeslagmassa):	0kg
Geabsorbeerd water	22,4L	Totaal volume:	1000L

Cement

	Droge massa	Te doseren massa	Te doseren volume	Volume fijn		Totale fractie V/V
CEM I 52.5 R ENCI MAASTRICHT	30kg	30kg	10L	9,5L		0,95%
CEM III/B 42.5 N LH HS ENCI IJMUIDEN	273kg	273kg	93L	92,5L		9,25%
Totaal Cement	303kg	303kg	102L	102,1L		10,20%

Zand

	Droge massa	Te doseren massa	Te doseren volume	Volume fijn	VochtAbsorptie M/M	VochtAbsorptie M/M	Totale fractie V/V
Rivierzand 0/4 mm	831kg	861kg	344L	28,7L	3,60%	0,50%	31,82%
Totaal Zand	831kg	861kg	344L	28,7L			31,82%

Grind

	Droge massa	Te doseren massa	Te doseren volume	Volume fijn	VochtAbsorptie M/M	VochtAbsorptie M/M	Totale fractie V/V
Zeegrind 4/16 mm	717kg	748kg	296L	2,9L	4,30%	2,55%	28,33%
Totaal Grind	717kg	748kg	296L	2,9L			28,33%

Vulstof

	Droge massa	Te doseren massa	Te doseren volume	Volume fijn		Totale fractie V/V
Poederkoolvliegias (met K-factor)	246kg	246kg	109L	109,3L		10,93%
Totaal Vulstof	246kg	246kg	109L	109,3L		10,93%

Hulpstof	Droge massa	Te doseren massa	Te doseren volume	Volume fijn bindmiddel	T.o.v. M/M	T.o.v. cement M/M	Totale fractie V/V
Cugla HR Superplastificeerder con. 20%	1,03kg	1,03kg	0,98L	0,0L	0,32%	0,34%	0,10%
Cugla LR Superplastificeerder con. 20%	3,12kg	3,12kg	2,97L	0,0L	0,96%	1,03%	0,30%
Totaal Hulpstof	4,15kg	4,15kg	3,95L	0,0L			0,40%
Water en slib	Massa bij droge materialen	Te doseren massa	Te doseren volume	Volume fijn			Totale fractie V/V
Bronwater	168kg	130kg	130L	0,0L			16,82%
Totaal Water en slib	168kg	130kg	130L	0,0L			16,82%
Lucht			15L	0,0L			1,50%
Totaal (8 grondstoffen):		2292kg	1000L	242,9L			100%

Appendix I

Photogrammetric method for measuring surfaces

The method used for measurement of the surfaces of deformed moulds and hardened concrete pieces is based on the principle of photogrammetry. The adaptation of this broad concept presented here is an unpublished author's solution developed by dr.ir. B. Gorte (Department of Geoscience & Remote Sensing at the Faculty of Civil Engineering and Geosciences, TU Delft). It is founded on the *Adaptive Least Squares Correlation* method introduced by Gruen (1985), which is a powerful data matching technique widely used in the field of image processing. B. Gorte's custom algorithm embraces it to reproduce the cloud of target points based on four pictures (two taken from each side of the setup, angle around 60°) and four control points with known coordinates.

I.1 Point cloud reconstruction

The measurement and data processing procedure has been done in following steps:

1. A point grid was placed on the measured surface with the use of either stickers (in case of moulded components, Figure I-1a) or a beamer (blocks after demoulding, Figure I-1b). Four pictures of the setup consisting of a specimen and reference points were taken with a consumer camera Samsung WB30F.
2. Data was preprocessed: each .jpg picture was converted to .pgm format and supplemented with a file containing approximate point coordinates.
3. Script developed by B. Gorte was executed, reconstructed point cloud was saved.

Besides processing all four pictures at one time, the used script repeats the procedure on cross-linked couples of pictures (L1-R1, L1-R2, L2-R1, L2-R2). The output of the latter is used for validation purposes: each of four resultant point clouds is compared with the final

result. Accuracy of the measurement is expressed by the means of an average of root mean square errors within each compared dataset. This value, referred in this report as *method error* proved not to exceed 0.2 mm in any case, which has been considered accurate.

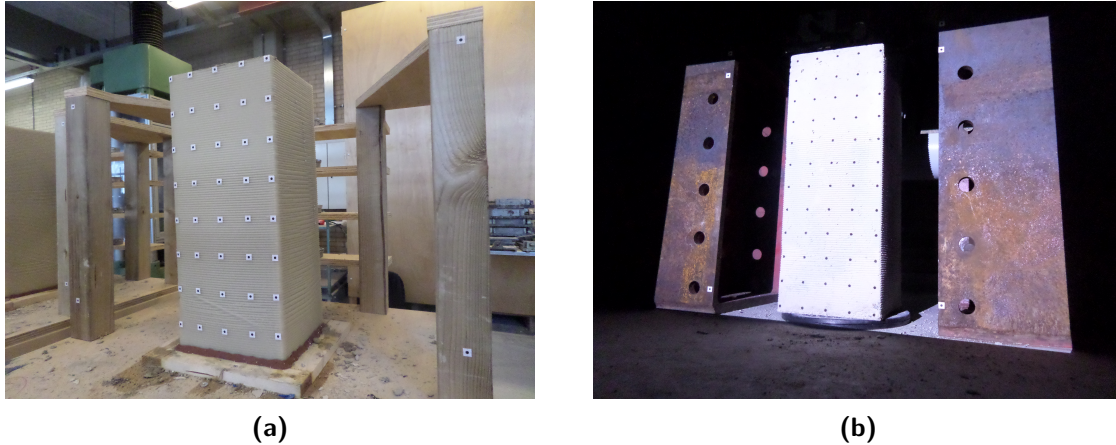


Figure I-1: Photogrammetry setups. (a) before demoulding, glued dots (b) after demoulding, projected dots.

1.2 Result processing

The geometrical interdependence between measured object and resultant point cloud embedded in an externally defined coordinate system is not known *a priori*. Two techniques were used to establish the missing relationship:

- a) In case of deflections not being predicted prior to casting, the reference plane was fitted into the group of points that were expected not to move out of plane during the experiment. For prismatic blocks, these are the ones located on corner edges (Figure I-2). In order to avoid distortions caused by local imperfections, only points located no further than 0.5 mm from the fitted plane were taken into account. This was achieved by iterative removal of nonconforming points and refitting the plane until the criterion was met for the whole pool.
- b) In case of deflections being predicted prior to casting (e.g. predeformed moulds presented in Chapter 9), the plane (or surface, in case of curved objects) was fitted directly into the output point cloud.

In both cases least square method was used for fitting. Finally, the distance between each point and reference plane/surface in the direction normal to the latter was calculated. All operations mentioned in this chapter were (semi)automatized with the use of Python programming language and geometrical transformation libraries available in Rhino/Grasshopper software.

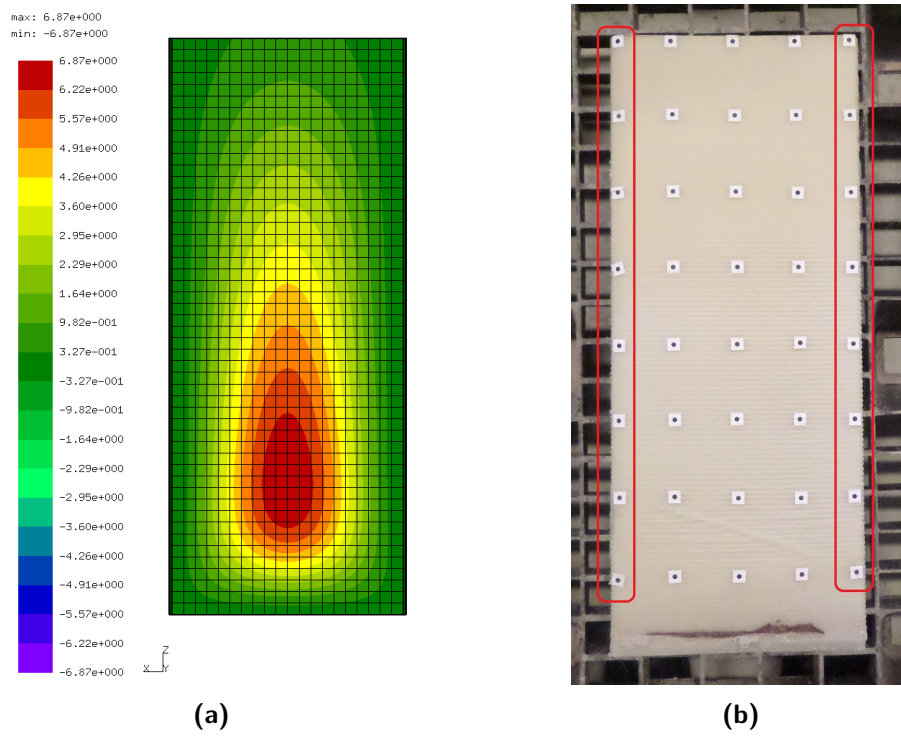


Figure I-2: Reference plane definition. (a) predicted out of plane deflection (b) points used to fit the reference plane.

Determination of layer thicknesses t_1 , t_2 for modelling of element's stiffness

A plane symmetric, triple layer element used for modelling of the 3D printed walls is shown in Figure 6-4. Considering equations 6.17, 6.18 and taking into account the fact that E_{normal} , $E_{flexural}$, t_1 , t_2 , E_1 , E_2 need to be non-negative, the cross-sectional stiffness characteristics of an element in each direction can be expressed with the following set of relationships:

$$\begin{cases} E_{normal}t = E_1t_1 + 2 \cdot E_2t_2 \\ E_{flexural} \frac{t^3}{12} = E_1 \frac{t_1^3}{12} + 2 \cdot E_2 \left(\frac{t_2^3}{12} + \left(\frac{t_1 + t_2}{2} \right)^2 \cdot t_2 \right) \\ t = t_1 + 2t_2 \\ E_{normal}, E_{flexural} \geq 0 \\ E_1, E_2 \geq 0 \\ t_1, t_2 \geq 0 \end{cases} \quad (J.1)$$

With the use of trivial mathematics the above system can be transformed into two 2nd order expressions describing the allowable range of t_2 (and consequently t_1) for which (respectively) E_1 and E_2 are non-negative:

$$\begin{cases} -4t_2^2E_{normal} + 4tt_2E_{normal} + t^2(E_{flexural} - E_{normal}) \geq 0 \\ -4t_2^2E_{normal} + 6tt_2E_{normal} + t^2(E_{flexural} - 3E_{normal}) \leq 0 \end{cases} \quad (J.2)$$

Taking into account physical boundaries:

$$t_2 \leq \frac{t}{2} \quad (J.3)$$

$$t_2 \geq 0 \quad (J.4)$$

Boundaries of allowable t_2 domain can be defined for each direction:

$$t_{2,min} = \max \left(0, \frac{4t^2 E_{normal} - \sqrt{16t^4 (E_{normal}(E_{flexural} - E_{normal}) + 4E_{normal}^2)}}{8t E_{normal}} \right) \quad (J.5)$$

$$t_{2,max} = \min \left(\frac{t}{2}, \frac{3t E_{normal} - \sqrt{-t^2 E_{normal}(3E_{normal} - 4E_{flexural})}}{4E_{normal}} \right) \quad (J.6)$$

In case of $-t^2 E_{normal}(3E_{normal} - 4E_{flexural}) < 0$, the second component of Equation J.6 should be neglected.

Since the non-negativity criteria need to be fulfilled for both directions at the same time, the ultimate values of $t_{2,min}$ and $t_{2,max}$ are defined as:

$$t_{2,min} = \max(t_{2,min,x}, t_{2,min,y}) \quad (J.7)$$

$$t_{2,max} = \min(t_{2,max,x}, t_{2,max,y}) \quad (J.8)$$

The values of t_1 and t_2 used for modelling of each element are derived from average of $t_{2,min}$ and $t_{2,max}$:

$$t_2 = \frac{t_{2,min} + t_{2,max}}{2} \quad (J.9)$$

$$t_1 = t - 2t_2 \quad (J.10)$$

It needs to be mentioned that there is a theoretical possibility of reaching an empty allowable domain in case of large difference between normal and flexural stiffness in one or two directions. Following relationship can be then found based on Equations J.7 and J.8:

$$t_{2,min} > t_{2,max} \quad (J.11)$$

However, such situation has not occurred in any simulation presented in this report.

Appendix K

Source code: generator of global input model

```
1 import ghpythonlib.components as ghcomp
2 import math
3
4 #prepare directory path
5 dir=dir+"\\\"
6
7 #define tolerance
8 tol=float(tolerance)
9
10 #printing direction for horizontal surfaces
11 if horizontal_dir=='X':
12     hordir=ghcomp.UnitX()
13 elif horizontal_dir=='Y':
14     hordir=ghcomp.UnitY()
15 elif horizontal_dir=='Z':
16     hordir=ghcomp.UnitZ()
17
18 #printing plane
19 def printplane(point,plane):
20     if plane=='XY':
21         horplane=ghcomp.XYPlane(point)
22     elif plane=='XZ':
23         horplane=ghcomp.XZPlane(point)
24     elif plane=='YZ':
25         horplane=ghcomp.YZPlane(point)
26     return horplane
27
28 #basic classes
29 class element(object): #element
30     def __init__(self,eltype,id,nodes):
31         if eltype=="CPS8": #check element type
```

```

32         self.ncount=4
33     else:
34         raise RuntimeError('Unknown element type!')
35     self.id=id
36     self.nodes=nodes
37     self.normal=None
38     self.orientation=None
39     self.vertices=None
40     self.orset=None
41
42     def flip(self): #flip element normal
43         flippednormal=ghcomp.Reverse(self.normal[1])
44         self.normal=(self.normal[0],flippednormal)
45         stnodes=self.nodes[:self.ncount][::-1]
46         ndnodes=self.nodes[self.ncount:]
47         ndnodes=(ndnodes[-1:]+ndnodes[0:-1])[::-1]
48         self.nodes=stnodes+ndnodes
49
50 class node(object): #node
51     def __init__(self,id,point):
52         self.id=id
53         self.point=point
54
55 class infill(object): #infill
56     def __init__(self,id,vol):
57         self.id=id
58         self.vol=vol
59         self.slices=[]
60
61 class slice(object): #load slice
62     def __init__(self,name,zmin,zmax,zdomain):
63         self.name=name
64         self.zmin=zmin
65         self.zmax=zmax
66         self.elids=[]
67         zextremes=ghcomp.DeconstructDomain(zdomain)
68         z=math.fabs(zextremes[0]-zextremes[1])
69         g=float(infill_dens)
70         x=zextremes[1]-(zmin+zmax)/2
71         self.lvalue=eval(infill_function)/(-10**6)
72
73 class support(object): #support
74     def __init__(self):
75         self.type=support_type
76         self.nodes=[]
77
78     def addnode(self,id):
79         self.nodes.append(id)
80
81 class OR(object): #element orientation
82     def __init__(self,id,orvalue,elids):
83         self.id=id
84         self.orvalue=orvalue

```

```
85         self.elids=elids
86
87 #find substring between two strings
88 def find_between( s, first, last ):
89     try:
90         start = s.index( first ) + len( first )
91         end = s.index( last, start )
92         return s[start:end]
93     except ValueError:
94         return ""
95
96 #chunk a list into n-sized sublists
97 def chunks(l, n):
98     for i in range(0, len(l), n):
99         yield l[i:i+n]
100
101 if generate:
102     #read .inp file
103     inpfile=open(dir + "topology.inp" , 'r')
104     rlines=inpfile.readlines()
105     inpfile.close()
106
107 #generate nodes & elements from file
108 nodes ,elements = [], []
109 nid=0
110 elid=0
111 nlines=False
112 ellines=False
113 eltype=""
114 for line in rlines:
115     #generate nodes from file
116     if "*NODE" in line:
117         nlines=True
118         continue
119     elif nlines and "*" in line:
120         nlines=False
121     elif nlines:
122         nodeinf=line.replace("\n","").split(", ")
123         nid=int(nodeinf[0])
124         x=float(nodeinf[1])
125         y=float(nodeinf[2])
126         z=float(nodeinf[3])
127         if len(nodes)<nid: #fix node numbering if needed
128             for i in range(nid-len(nodes)):
129                 nodes.append(None)
130         nodes.insert(nid+1,node(nid,ghcomp.ConstructPoint(x,y,z)))
131
132 #generate elements from file
133 elif "*ELEMENT" in line and "Surface" in line:
134     eltype=find_between(line,"type=","")
135     ellines=True
136     continue
137 elif ellines and "*" in line:
```

```

138         ellines=False
139     elif ellines:
140         elid+=1
141         elinf=line.replace("\n","").split(", ")
142         elnodes=elinf[1:]
143         elements.append(element(eltype,elid,elnodes))
144
145     #determine element vertices and normals, orientate the x-axis
146     #horizontally
147     zeropoint=ghcomp.ConstructPoint(0,0,0)
148     for element in elements:
149         #get vertices
150         elverts=[]
151         for i in range(element.ncount):
152             nid=int(element.nodes[i])
153             elverts.append(nodes[nid].point)
154         element.vertices=elverts
155
156         #define normal
157         elface=ghcomp.ConstructMesh(elverts)
158         element.normal=ghcomp.FaceNormals(elface)
159
160         #orientate local coordinate system (see CalculiX manual)
161         horplane=printplane(element.normal[0],print_plane)
162         elplane=ghcomp.PlaneNormal(element.normal[0],element.normal[1])
163         xdir=ghcomp.PlaneXPlane(horplane,elplane)
164         if not xdir: #element in horizontal plane
165             xvect=hordir
166         else:
167             xvect=ghcomp.EvaluateCurve(xdir,0)[1]
168             zeronormal=ghcomp.LineSDL(zeropoint,element.normal[1],1)
169             yvect=ghcomp.DeconstructVector(ghcomp.RotateAxis(xvect,math.pi/2,
170                 zeronormal)[0])
171             xvect=ghcomp.DeconstructVector(xvect)
172             ordef=[]
173             for num in xvect:
174                 ordef.append(str(round(num,6)))
175             for num in yvect:
176                 ordef.append(str(round(num,6)))
177             element.orientation=",".join(ordef)
178
179         #define orientation sets
180         orients=[]
181         for element in elements:
182             orients.append(element.orientation)
183         values = set(map(lambda x:x, orients))
184         ORs=[]
185         k=1
186         for value in values:
187             sorted_or=[]
188             for element in elements:
189                 if element.orientation==value:
190                     sorted_or.append(element.id)

```

```

189         element.orset=k
190     ORs.append(OR(k,value,sorted_or))
191     k+=1
192
193     #generate bottom supports
194     supports=support()
195     for node in nodes:
196         if node!=None:
197             nodealt=ghcomp.Deconstruct(node.point)[2]
198             if -tol < nodealt < tol:
199                 supports.addnode(node.id)
200     if support_type=="botphys":
201         suppnodes=[]
202         for element in elements:
203             for node in supports.nodes:
204                 for id in element.nodes:
205                     if node==int(id):
206                         suppnodes.extend(element.nodes[:4])
207                         break
208
209         supports=support()
210         for node in set(map(lambda x:x, suppnodes)):
211             supports.addnode(node)
212
213     #merge infill volumes if applicable
214     jointinfills=ghcomp.SolidUnion(infill_vols)
215     mergedinfills=ghcomp.MergeFaces(jointinfills)[0]
216     if not isinstance(mergedinfills,list):
217         mergedinfills=[mergedinfills]
218
219     #generate infills
220     k=1
221     infills=[]
222     if not mergedinfills==[None]:
223         for volume in mergedinfills:
224             infills.append(infill(k,volume))
225             k+=1
226
227     #define infill elements
228     if len(infills)>0:
229         for infill in infills:
230             zdomain=ghcomp.DeconstructBox(infill.vol)[3]
231             zdomains=ghcomp.DivideDomain(zdomain,slice_count)
232             zmins,zmaxs=[],[]
233             for i in range(int(slice_count)):
234                 zmin=ghcomp.DeconstructDomain(zdomains[i])[0]
235                 zmax=ghcomp.DeconstructDomain(zdomains[i])[1]
236                 zmins.append(zmin)
237                 zmaxs.append(zmax)
238                 infill.slices.append(slice("Infill_"+str(infill.id)+"
                _Slice_"+str(i+1),zmin,zmax,zdomain))
239         for element in elements:
240             for vert in element.vertices:

```

```

241         #check if the element (not) encloses the infill
           volume
242         if ghcomp.PullPoint(vert, infill.vol)[1]>tol:
243             break
244     else:
245         #check the normal orientation & flip if necessary
246         for nn in range(1000):
247             if ghcomp.PointInBrep(infill.vol, ghcomp.Move(
                element.normal[0], nn*element.normal[1])[0]):
248                 if not ghcomp.PointInBrep(infill.vol, ghcomp.
                    Move(element.normal[0], -nn*element.normal
                        [1])[0]):
249                     break
250                 else:
251                     continue
252             elif ghcomp.PointInBrep(infill.vol, ghcomp.Move(
                element.normal[0], -nn*element.normal[1])[0]):
253                 element.flip()
254                 break
255
256         #refer the elements to respective slices
257         elaltid=ghcomp.Deconstruct(element.normal[0])[2]
258         for i in range(int(slice_count)):
259             if zmins[i] <= elaltid < zmaxs[i]:
260                 infill.slices[i].elids.append(element.id)
261
262     #define slices and loads
263     sliceloads, slicelines = [], []
264     for infill in infills:
265         for slice in infill.slices:
266             if len(slice.elids)>0:
267                 slicelines.append("*ELSET, ELSET="+slice.name)
268                 for chunk in chunks(slice.elids, 16):
269                     chunk=map(str, chunk)
270                     slicelines.append(", ".join(chunk))
271                 sliceloads.append(slice.name+" ,P, "+str(slice.lvalue))
272
273     #generate output file
274     outputlines=[]
275     outputlines.append("*ELCOUNT="+str(len(elements)))
276     outputlines.append("*NODE, NSET=Nall")
277     for node in nodes:
278         if node!=None:
279             outputlines.append(str(node.id)+", "+str(node.point))
280     outputlines.append("*ELEMENT, type=S8R, ELSET=Eall")
281     for element in elements:
282         eldescr=list(element.nodes)
283         eldescr.insert(0, str(element.id))
284         outputlines.append(", ".join(eldescr))
285     if len(supports.nodes)>0:
286         outputlines.append("*NSET, NSET=SUPPORT")
287         for chunk in chunks(supports.nodes, 16):
288             chunk=map(str, chunk)

```

```
289         outputlines.append(", ".join(chunk))
290     outputlines.append("*BOUNDARY")
291     if support_type=="botphys":
292         outputlines.append("SUPPORT,1,3")
293     else:
294         outputlines.append("SUPPORT,"+support_type)
295     for line in slicelines:
296         outputlines.append(line)
297     for orr in ORs:
298         outputlines.append("*ORIENTATION,NAME=OR"+str(orr.id))
299         outputlines.append(orr.orvalue)
300     outputlines.append("*MATERIAL INFO")
301     outputlines.append("*ELEMENT ORIENTATIONS")
302     for element in elements:
303         outputlines.append("OR" + str(element.orset))
304     if nlinear:
305         outputlines.append("*STEP,NLGEOM")
306     else:
307         outputlines.append("*STEP")
308         outputlines.append("*STATIC")
309         outputlines.append("*DLOAD")
310         for load in sliceloads:
311             outputlines.append(load)
312         outputlines.append("*EL PRINT,ELSET=Eall,GLOBAL=NO")
313         outputlines.append("S,E")
314         outputlines.append("*EL FILE")
315         outputlines.append("S,E")
316         outputlines.append("*NODE PRINT,NSET=Nall,GLOBAL=YES")
317         outputlines.append("U")
318         outputlines.append("*NODE FILE")
319         outputlines.append("U")
320         outputlines.append("*END STEP")
321
322     writefile=open(dir + "ccxinput.inp", 'w')
323     for line in outputlines:
324         line=line+"\n"
325         writefile.write(line)
326     writefile.close()
```

Appendix L

Source code: stress-strain data generator

```
1 #prepare directories
2 dir=dir+"\\"
3 datadir=datadir+"\\"
4
5 types=["tensL", "tensT", "bendL", "bendT"]
6
7 #read cross-sectional property multiplier data
8 csdatafile=open(datadir+"CSfactors.csv", 'r')
9 csdatalines=csdatafile.readlines()
10 csdatafile.close()
11
12 def gensscurve(dir, datadir, type, age):
13     datafile=open(datadir+type+str(age)+"d.csv", 'r')
14     datalines=datafile.readlines()
15     datafile.close()
16
17     #retrieve cross-sectional property multipliers
18     classes=False
19     csmultipliers=[]
20     classcount=0
21     for line in csdatalines:
22         if type in line:
23             classes=True
24             continue
25         elif classes:
26             try:
27                 int(line.split(";")[0])
28                 csmultipliers.append(float(line.split(";")[1]))
29                 classcount+=1
30             except ValueError:
31                 break
```

```

32     if classcount==0:
33         classcount=1
34
35     #translate data file into data array
36     dataarr,specdata=[],[]
37     dataflag=False
38     for line in datalines:
39         #retrieve specimen parameters
40         if "LENGTH" in line or "SPAN" in line:
41             l=float(line.split(";")[1])
42         elif "WIDTH" in line:
43             w=float(line.split(";")[1])
44
45         #retrieve specimen class, start collecting data
46         elif "CLASS" in line:
47             if classes:
48                 specclass=int(line.split(";")[1])
49                 specmultiplier=csmultipliers[specclass-1]
50             else:
51                 specmultiplier=1.0
52             dataflag=True
53             if len(specdata)>0:
54                 dataarr.append(specdata)
55                 specdata=[]
56             continue
57
58         #translate engineering stress and strain into true stress and
59         #Lagrangian strain
60         elif dataflag==True and not "SPECIMEN" in line:
61             data=line.split(";")
62             fdata=[]
63             if "bend" in type:
64                 engstrain=6*float(data[0])*t/1**2
65                 engstress=6*float(data[1])*1/4/(w*t**2)*specmultiplier
66             else:
67                 engstrain=float(data[0])/1
68                 engstress=float(data[1])/(w*t)*specmultiplier
69
70         #calculate strain and stress data for class 1
71         fdata.append(engstrain+0.5*engstrain**2)
72         fdata.append(engstress*(1+engstrain)*(1-engstrain)**2)
73         specdata.append(fdata)
74     dataarr.append(specdata)
75
76     #find shortest stress/strain curve of all specimens
77     maxstrains=[]
78     for spec in dataarr:
79         maxstrains.append(float(spec[-1][0]))
80     maxstrain=min(maxstrains)
81
82     #define length of a single step of output curve
83     step=maxstrain/200

```

```

84     #determine average stress values from all specimens at each point of
      the curve per each class
85     crvnodes=[[0]*(classcount+1)]
86     for i in range(1,201):
87         specvalues=[]
88         for spec in dataarr:
89             for j in range(len(spec)):
90                 if spec[j][0]>=i*step:
91                     specvalues.append((spec[j][1]-(spec[j][1]-spec[j]
92                                     -1)[1])*(spec[j][0]-i*step)/(spec[j][0]-spec[j]
93                                     -1)[0]))
94                     break
95         stressinterp=float(sum(specvalues))/len(specvalues)
96
97     #calculate the values per each class
98     if classes:
99         nodeinfo=[step*i]
100        for multiplier in csmultipliers:
101            nodeinfo.append(stressinterp/multiplier)
102        crvnodes.append(nodeinfo)
103    else:
104        crvnodes.append([step*i, stressinterp])
105
106    #write output curve to a new file
107    writefile=open(dir+type+".dat", 'w')
108    for node in crvnodes:
109        line=[]
110        for ndata in node:
111            line.append(str(ndata))
112        line=",".join(line)+"\n"
113        writefile.write(line)
114    writefile.close()
115
116    if run:
117        for type in types:
118            gensscurve(dir, datadir, type, age)

```

Appendix M

Source code: predeformation component

```
1 from os import system
2
3 #prepare directory
4 dir=dir+"\\\"
5
6 #MATERIAL CONSTANTS:
7 #vol fraction of fibers
8 vf=0.052
9 #stiffness of fibers
10 Gf=30000.0
11 #Poisson's ratios
12 v12=0.42
13 v13=v12
14 v23=0.43
15
16 #declare global variables
17 global optfile,initfile
18 optfile="predeformation"
19 initfile="ccxinput"
20
21 #read stress-strain data
22 def readSS(type):
23     datafile=open(dir+type+".dat", 'r')
24     datalines=datafile.readlines()
25     datafile.close()
26     ssdata=[]
27     for line in datalines:
28         content=line.split(",")
29         crvstrain=float(content[0])
30         if len(content)>2:
31             crvstress=float(content[printclass])
```

```

32         else:
33             crvstress=float(content[1])
34             ssdata.append([crvstrain, crvstress])
35     return ssdata
36
37 #initiate the computations: read global variables
38 def initiate():
39     #read stress/strain curves
40     global tensL, tensT, bendL, bendT
41     tensL=readSS("tensL")
42     tensT=readSS("tensT")
43     bendL=readSS("bendL")
44     bendT=readSS("bendT")
45
46     #read initial model data
47     global initlines, initcoords, orients, elcount, nodestart, matline,
48         restline
49     initpath=dir + initfile + ".inp"
50     initdata=open(initpath, 'r')
51     initlines=initdata.readlines()
52     initdata.close()
53
54     #find number of elements
55     for line in initlines:
56         if "*ELCOUNT" in line:
57             elcount=int(line.split("=")[1])
58             break
59     else:
60         raise Exception('Oooops, element count not found!')
61
62     #find node definition start, initial coordinates of the nodes and
63     #material info section
64     nodes=False
65     orient=False
66     initcoords=[]
67     orients=[]
68     for linid, line in enumerate(initlines):
69         if "*NODE," in line:
70             nodestart=linid
71             nodes=True
72             continue
73         elif "*" in line:
74             nodes=False
75         elif nodes:
76             nodeinf=line.split(",")
77             initcoords.append([float(nodeinf[1]), float(nodeinf[2]), float(
78                 nodeinf[3])])
79             continue
80         if "*MATERIAL INFO" in line:
81             matline=linid
82             continue
83         elif "*ELEMENT ORIENTATIONS" in line:
84             orient=True

```

```

82         continue
83     elif orient and not "*" in line:
84         orients.append(line)
85     elif orient and "*" in line:
86         restline=linid
87         break
88
89 #derive the value of secant modulus based on stress/strain curve
90 def secantE(e,curve):
91     if e<curve[1][0]:
92         E=curve[1][1]/curve[1][0]
93     else:
94         for i in range(len(curve)):
95             if curve[i][0]>=e:
96                 #interpolate stress value
97                 s=curve[i][1]-(curve[i][1]-curve[i-1][1])*(curve[i][0]-e)
98                   /((curve[i][0]-curve[i-1][0]))
99                 #find secant modulus
100                E=s/e
101                break
102            else:
103                raise Exception('Strain beyond the range provided in test
104                                data. Calculations terminated. Decrease strains or provide
105                                more data.')
```

```

103     return E
104
105 #generate layer thicknesses t1 and t2
106 def thkgen(t,EtensL,EbendL,EtensT,EbendT):
107     #based on Equations J.5, J.6 for both directions:
108     t2mins=[0.0]
109     t2maxs=[t/2.0]
110     t2mins.append(((4.0*t**2.0*EtensL-(16.0*t**4.0*EtensL*(EbendL-EtensL)
111                   +16.0*t**4.0*EtensL**2)**0.5)/(8.0*t*EtensL))
112     t2mins.append(((4.0*t**2.0*EtensT-(16.0*t**4.0*EtensT*(EbendT-EtensT)
113                   +16.0*t**4.0*EtensT**2)**0.5)/(8.0*t*EtensT))
114     if -EtensL*t**2.0*(3.0*EtensL-4.0*EbendL)>=0: #check if delta>0
115         t2maxs.append(((3.0*EtensL*t-(-EtensL*t**2.0*(3.0*EtensL-4.0*
116                   EbendL)**0.5)/(4.0*EtensL))
117     if -EtensT*t**2.0*(3.0*EtensT-4.0*EbendT)>=0: #check if delta>0
118         t2maxs.append(((3.0*EtensT*t-(-EtensT*t**2.0*(3.0*EtensT-4.0*
119                   EbendT)**0.5)/(4.0*EtensT))
120
121 #from Equations J.7-J.11:
122 t2min=max(t2mins)
123 t2max=min(t2maxs)
124 if t2min>t2max:
125     raise Exception("Cannot find a combination of positive layer
126                     thicknesses and Young's moduli. Change wall thickness.")
127 else:
128     t2=(t2min+t2max)/2.0
129     t1=t-2.0*t2
130     return t1,t2

```

```

127 #determine E1 and E2 multipliers of E based on layer thickness, bending
    and tensile stiffness
128 def secfactors(t1,t2,Etens,Ebend):
129     L1fctr=(2.0*t2+t1)/t1-(Ebend/Etens*(t1+2.0*t2)**2.0-t1**2.0)/(2.0*t1
        *(t1+t2))
130     L2fctr=(Ebend/Etens*(t1+2.0*t2)**2.0-t1**2.0)/(4.0*t2*(t1+t2))
131     return L1fctr,L2fctr
132
133 #cross-sectional properties based on element strain state and thickness
134 def propgen(strainset):
135     exxT,exxF,eyyT,eyyF=strainset[:]
136     EtensL=secantE(exxT,tensL)
137     EbendL=secantE(exxF,bendL)
138     EtensT=secantE(eyyT,tensT)
139     EbendT=secantE(eyyF,bendT)
140
141     #update t1, t2 thicknesses
142     t1,t2=thkgen(t,EtensL,EbendL,EtensT,EbendT)
143
144     #calculate stiffness parameters
145     L1Lfactor,L2Lfactor=secfactors(t1,t2,EtensL,EbendL)
146     L1Tfactor,L2Tfactor=secfactors(t1,t2,EtensT,EbendT)
147     E1L1=EtensL*L1Lfactor
148     E1L2=EtensL*L2Lfactor
149     E2L1=E3L1=EtensT*L1Tfactor
150     E2L2=E3L2=EtensT*L2Tfactor
151     G23L1=E2L1/(2.0*(1.0+v23))
152     G23L2=E2L2/(2.0*(1.0+v23))
153     migL1=(Gf/G23L1-1.0)/(Gf/G23L1+1.0)
154     migL2=(Gf/G23L2-1.0)/(Gf/G23L2+1.0)
155     G12L1=G13L1=(1.0+vf*migL1)/(1.0-vf*migL1)*G23L1
156     G12L2=G13L2=(1.0+vf*migL2)/(1.0-vf*migL2)*G23L2
157
158     #return thicknesses and stiffness parameters
159     elprops=[t1,t2,E1L1,E1L2,E2L1,E2L2,E3L1,E3L2,G23L1,G23L2,G12L1,G12L2,
        G13L1,G13L2]
160     return elprops
161
162 #read avg. strain in outer faces of the element
163 def readstrain(resfile,elprops):
164     #read ccx result file
165     resFile = open(dir + resfile + ".dat", 'r')
166     reslines=resFile.readlines()
167     resFile.close()
168     for rlinid,line in enumerate(reslines[:: -1]):
169         if "displacements" in line:
170             strainend=len(reslines)-rlinid-2
171             continue
172         elif "strains" in line:
173             if float(" ".join(line.split()).split(" ")[-1])!=1:
174                 raise Exception('Oooops, previous iteration did not
                    converge!')
175         else:

```



```

176             strainstart=len(reslines)-rlinid+1
177             break
178     strainlines=reslines[strainstart:strainend]
179
180     #initialize calculation
181     bottomstrainxx,bottomstrainyy,midstrainxx,midstrainyy,topstrainxx,
182     topstrainyy,elstrain = [], [], [], [], [], [], []
183     n=0
184     k=0
185     strainresults=[]
186     gfac=1.0+(1.0/3.0)**0.5 #factor for reading strains (see Section
187     6.3.3)
188
189     #calculate strains
190     for line in strainlines:
191         content=" ".join(line.split()).split(" ")
192         if n<4:
193             bottomstrainxx.append(float(content[2]))
194             bottomstrainyy.append(float(content[3]))
195         elif n>7 and n<16:
196             midstrainxx.append(float(content[2]))
197             midstrainyy.append(float(content[3]))
198         elif n>19:
199             topstrainxx.append(float(content[2]))
200             topstrainyy.append(float(content[3]))
201         n+=1
202         if n==24:
203             #calculate and append values, reset arrays, go to next step
204             t1,t2=elprops[k][:2]
205             elstrain.append(sum(midstrainxx)/8.0)
206             elstrain.append(((abs(float(sum(bottomstrainxx))-float(sum(
207                 topstrainxx))))/8.0)*(t1+2.0*t2)/(t1+gfac*t2))
208             elstrain.append(sum(midstrainyy)/8.0)
209             elstrain.append(((abs(float(sum(bottomstrainyy))-float(sum(
210                 topstrainyy))))/8.0)*(t1+2.0*t2)/(t1+gfac*t2))
211             strainresults.append(elstrain)
212             bottomstrainxx,bottomstrainyy,midstrainxx,midstrainyy,
213             topstrainxx,topstrainyy,elstrain = [], [], [], [], [], [], []
214             n=0
215             k+=1
216     return strainresults
217
218 #build input file
219 def buildinput(elprops,initlines,nodestart,matline,restline,orients,
220 outputfile):
221     #append topology
222     outputlines=[]
223     for i in range(nodestart,matline):
224         outputlines.append(initlines[i])
225
226     #append material info
227     for i in range(elcount):

```

```

222     E1L1 , E1L2 , E2L1 , E2L2 , E3L1 , E3L2 , G23L1 , G23L2 , G12L1 , G12L2 , G13L1 , G13L2
        =elprops [ i ] [ 2 : ]
223     outputlines . append ( "*MATERIAL , NAME=EL "+str ( i + 1 ) + " LAYER1 \n " )
224     outputlines . append ( "*ELASTIC , TYPE=ENGINEERING CONSTANTS \n " )
225     line1L1 = " , " . join ( map ( str , [ E1L1 , E2L1 , E3L1 , v12 , v13 , v23 , G12L1 , G13L1
        ] ) ) + " \n "
226     line2L1 = str ( G23L1 ) + " \n "
227     outputlines . append ( line1L1 )
228     outputlines . append ( line2L1 )
229     outputlines . append ( "*MATERIAL , NAME=EL "+str ( i + 1 ) + " LAYER2 \n " )
230     outputlines . append ( "*ELASTIC , TYPE=ENGINEERING CONSTANTS \n " )
231     line1L2 = " , " . join ( map ( str , [ E1L2 , E2L2 , E3L2 , v12 , v13 , v23 , G12L2 , G13L2
        ] ) ) + " \n "
232     line2L2 = str ( G23L2 ) + " \n "
233     outputlines . append ( line1L2 )
234     outputlines . append ( line2L2 )
235
236     #append element info
237     for i in range ( len ( orients ) ) :
238         OR = orients [ i ]
239         t1 , t2 = elprops [ i ] [ : 2 ]
240         outputlines . append ( "*ELSET , ELSET=EL " + str ( i + 1 ) + " \n " )
241         outputlines . append ( str ( i + 1 ) + " \n " )
242         outputlines . append ( "*SHELL SECTION , ELSET=EL " + str ( i + 1 ) + " ,
            COMPOSITE \n " )
243         outputlines . append ( str ( t2 ) + " , , EL " + str ( i + 1 ) + " LAYER2 , " + OR )
244         outputlines . append ( str ( t1 ) + " , , EL " + str ( i + 1 ) + " LAYER1 , " + OR )
245         outputlines . append ( str ( t2 ) + " , , EL " + str ( i + 1 ) + " LAYER2 , " + OR )
246
247     #add remaining stuff
248     for i in range ( restline , len ( initlines ) ) :
249         outputlines . append ( initlines [ i ] )
250
251     #write output file
252     writefile = open ( dir + outputfile + ".inp" , 'w' )
253     for line in outputlines :
254         writefile . write ( line )
255     writefile . close ( )
256
257     #iteratively update the stiffness of each element based on strains state ,
        until the difference between 2 substeps lies within tolerances
258     def iterateE ( avgstrainstol , maxstrainstol ) :
259         #set initial strains to 0.0
260         strains = []
261         for i in range ( elcount ) :
262             strains . append ( [ 0.0 , 0.0 , 0.0 , 0.0 ] )
263
264         #start values >> tolerances
265         avgstrain = avgstrainstol + 1.0
266         maxstrain = maxstrainstol + 1.0
267
268         #reset substep
269         substep = 0

```

```

270
271     #iteratively update stiffness parameters of each element as in
        Chapter 6
272     while avgstrain>avgstrainstol or maxstrain>maxstrainstol:
273         strainconvcheck=[]
274
275         #generate element properties, build FE input file
276         props=[]
277         for i in range(elcount):
278             props.append(propgen(strains[i]))
279         buildinput(props,initlines,nodestart,matline,restline,orients,
            optfile)
280
281         #substep zero solution
282         command = "set OMP_NUM_THREADS=" + nthreads + " & pushd " + dir +
            ' & "' + calculix + "' -i ' + optfile
283         system(command)
284
285         #read strain results and compare with previous iteration
286         newstrains=readstrain(optfile,props)
287         xxTdiff,xxFdiff,yyTdiff,yyFdiff=[],[],[],[]
288         for i in range(len(strains)):
289             xxTdiff.append(abs(strains[i][0]-newstrains[i][0]))
290             xxFdiff.append(abs(strains[i][1]-newstrains[i][1]))
291             yyTdiff.append(abs(strains[i][2]-newstrains[i][2]))
292             yyFdiff.append(abs(strains[i][3]-newstrains[i][3]))
293         avgstrainsxxT=float(sum(xxTdiff))/len(xxTdiff)
294         avgstrainsxxF=float(sum(xxFdiff))/len(xxFdiff)
295         avgstrainsyyT=float(sum(yyTdiff))/len(yyTdiff)
296         avgstrainsyyF=float(sum(yyFdiff))/len(yyFdiff)
297         maxstrainsxxT=max(xxTdiff)
298         maxstrainsxxF=max(xxFdiff)
299         maxstrainsyyT=max(yyTdiff)
300         maxstrainsyyF=max(yyFdiff)
301
302         #write strain results to log
303         optstrainsFile = open(dir + optfile + "strains.opt", 'a')
304         if substep==0:
305             optstrainsFile.write("\nStrain convergence in step " + str(
                int(step)) + ":\n")
306         optstrainsFile.write('%i,%f,%f,%f,%f,%f,%f,%f,%f\n' % (substep,
            avgstrainsxxT,maxstrainsxxT,avgstrainsxxF,maxstrainsxxF,
            avgstrainsyyT,maxstrainsyyT,avgstrainsyyF,maxstrainsyyF))
307         optstrainsFile.close()
308
309         #check convergence criteria
310         avgstrain=max(avgstrainsxxT,avgstrainsxxF,avgstrainsyyT,
            avgstrainsyyF)
311         maxstrain=max(maxstrainsxxT,maxstrainsxxF,maxstrainsyyT,
            maxstrainsyyF)
312         if avgstrain>avgstrainstol or maxstrain>maxstrainstol:
313             if Eupd:
314                 nstrains=[]

```

```

315         for i in range(len(strains)):
316             istrains=[]
317             for j in range(4):
318                 istrains.append(0.5*(strains[i][j]+newstrains[i][
                    j]))
319             nstrains.append(istrains)
320             strains=nstrains
321         else:
322             strains=newstrains
323             substep+=1
324     else:
325         print 'Yay! Result has been found within given tolerance!'
326
327 #predeform the shape acc. to the method shown in Chapter 8
328 def predeform(avgdeftol,maxdeftol,avgstrainstol,maxstrainstol):
329     #set step to zero, read initial info
330     global step
331     step=0
332     initiate()
333
334     #clear strain convergence and optimization logs
335     optstrainsFile = open(dir + optfile + "strains.opt", 'w')
336     optstrainsFile.close()
337     optresFile = open(dir + optfile + ".opt", 'w')
338     optresFile.close()
339
340     #step zero solution
341     iterateE(avgstrainstol,maxstrainstol)
342
343     #initial values >> tolerances
344     avgdef=avgdeftol+1.0
345     maxdef=maxdeftol+1.0
346
347     #iteratively search for predeformed shape, which after deformation
        complies to desired geometry
348     while avgdef>avgdeftol or maxdef>maxdeftol:
349         #find displacements section in result file
350         resFile = open(dir + optfile + ".dat", 'r')
351         reslines=resFile.readlines()
352         resFile.close()
353         for linid,line in enumerate(reslines[::-1]):
354             if "displacements" in line:
355                 if float(" ".join(line.split()).split(" ")[-1])!=1.0:
356                     raise Exception('Oooops, previous iteration did not
                        converge!')
357             else:
358                 resid=len(reslines)-linid+1
359                 break
360
361         #retrieve coordinates of deformed mesh
362         defcoords=[]
363         for i in range(len(initcoords)):
364             newnode=[]

```

```

365         defnode=[]
366         procline=i+nodestart+1
367         for j in range(4):
368             if j==0:
369                 newnode.append(initlines[procline].split(",")[0])
370             else:
371                 defcoord=float(initlines[procline].split(",")[j])+
                    float(" ".join(reslines[resid+i].split()).split("
                    ")[j])
372                 defnode.append(defcoord)
373                 newnode.append(str(float(initlines[procline].split(",
                    ")[j])+initcoords[i][j-1]-defcoord))
374
375         defcoords.append(defnode)
376         initlines[procline]=",".join(newnode)+"\n"
377
378         #compute distances between nodes of deformed and initial shapes
379         reldefs=[]
380         for i in range(len(initcoords)):
381             reldefs.append(((initcoords[i][0]-defcoords[i][0])**2+(
                    initcoords[i][1]-defcoords[i][1])**2+(initcoords[i][2]-
                    defcoords[i][2])**2)**0.5)
382         avgdef=sum(reldefs)/float(len(reldefs))
383         maxdef=max(reldefs)
384
385         #write results to log
386         optresFile = open(dir + optfile + ".opt", 'a')
387         if step==0:
388             optresFile.write("Deformation convergence:\n")
389             optresFile.write('%i,%f,%f\n' % (step, avgdef, maxdef))
390         optresFile.close()
391
392         step+=1
393         #check convergence criteria
394         if avgdef>avgdeftol or maxdef>maxdeftol:
395             iterateE(avgstrainstol, maxstrainstol)
396         else:
397             print 'Yay! Predeformed shape has been found within given
                    tolerance!'
398
399     if run:
400         predeform(avgd, maxd, avgE, maxE)

```

Appendix N

Source code: post-processing

```
1 import Rhino.Geometry as rg
2 import ghpythonlib.components as ghcomp
3
4 #prepare directory
5 dir=dir+"\"
6
7 #read stresses from result file
8 def readstress(thks):
9     #read ccx result file
10    resFile = open(dir + "predeformation.dat", 'r')
11    reslines=resFile.readlines()
12    resFile.close()
13    for rlinid,line in enumerate(reslines[:: -1]):
14        if "strains" in line:
15            stressend=len(reslines)-rlinid-2
16            continue
17        elif "stresses" in line:
18            if float(" ".join(line.split()).split(" ")[-1])!=1:
19                raise Exception('Oooops, the solution did not converge!')
20            else:
21                stressstart=len(reslines)-rlinid+1
22                break
23    stresslines=reslines[stressstart:stressend]
24
25    #initialize reading
26    elresults=[]
27    stressLxx=[[[] for i in range(6)]]
28    stressLyy=[[[] for i in range(6)]]
29    stressGxy=[[[] for i in range(4)]]
30    stressGxz=[[[] for i in range(4)]]
31    stressGyz=[[[] for i in range(4)]]
32    Lcount=0
33    n=0
```

```

34     k=0
35
36     #read stresses and translate them into internal forces acc. to
37     Section 6.3.7
38     for line in stresslines:
39         #read stresses
40         content=" ".join(line.split()).split(" ")
41         if n%4==0 and n>0:
42             Lcount+=1
43             stressLxx[Lcount].append(float(content[2]))
44             stressLyy[Lcount].append(float(content[3]))
45             stressGxy[n%4].append(float(content[5]))
46             stressGxz[n%4].append(float(content[6]))
47             stressGyz[n%4].append(float(content[7]))
48             n+=1
49
50     #calculate internal forces from 24 integration points, reset
51     reader
52     gfac=(3.0*0.5-1)*0.5 #factor for reading strains (see Section
53     6.3.3)
54     if n==24:
55         t1=thks[k][0]
56         t2=thks[k][1]
57
58     #translate stresses at integration points to stresses at
59     layer faces
60     for l in range(6):
61         stressLxx[l]=sum(stressLxx[l])*0.25
62         stressLyy[l]=sum(stressLyy[l])*0.25
63         stress1xx=stressLxx[0]+(stressLxx[0]-stressLxx[1])*gfac
64         stress1yy=stressLyy[0]+(stressLyy[0]-stressLyy[1])*gfac
65         stress2xx=(stressLxx[0]+stressLxx[1])*0.5
66         stress2yy=(stressLyy[0]+stressLyy[1])*0.5
67         stress3xx=stressLxx[1]+(stressLxx[1]-stressLxx[0])*gfac
68         stress3yy=stressLyy[1]+(stressLyy[1]-stressLyy[0])*gfac
69         stress4xx=stressLxx[2]+(stressLxx[2]-stressLxx[3])*gfac
70         stress4yy=stressLyy[2]+(stressLyy[2]-stressLyy[3])*gfac
71         stress5xx=(stressLxx[2]+stressLxx[3])*0.5
72         stress5yy=(stressLyy[2]+stressLyy[3])*0.5
73         stress6xx=stressLxx[3]+(stressLxx[3]-stressLxx[2])*gfac
74         stress6yy=stressLyy[3]+(stressLyy[3]-stressLyy[2])*gfac
75         stress7xx=stressLxx[4]+(stressLxx[4]-stressLxx[5])*gfac
76         stress7yy=stressLyy[4]+(stressLyy[4]-stressLyy[5])*gfac
77         stress8xx=(stressLxx[4]+stressLxx[5])*0.5
78         stress8yy=(stressLyy[4]+stressLyy[5])*0.5
79         stress9xx=stressLxx[5]+(stressLxx[5]-stressLxx[4])*gfac
80         stress9yy=stressLyy[5]+(stressLyy[5]-stressLyy[4])*gfac
81
82     #translate stresses at layer faces into internal forces
83     Sxys ,Sxzs ,Syzs = [] ,[] ,[]
84     for Gxy in stressGxy:
85         Sxys.append(abs((Gxy[0]+Gxy[1]+Gxy[4]+Gxy[5])*t2+(Gxy[2]+
86             Gxy[3])*t1)*0.5)

```



```

82     Sxy=max(Sxys)
83     for Gxz in stressGxz:
84         Sxzs.append(abs((Gxz[0]+Gxz[1]+Gxz[4]+Gxz[5])*t2+(Gxz[2]+
85             Gxz[3])*t1)*0.5)
86     Sxz=max(Sxzs)
87     for Gyz in stressGyz:
88         Syzs.append(abs((Gyz[0]+Gyz[1]+Gyz[4]+Gyz[5])*t2+(Gyz[2]+
89             Gyz[3])*t1)*0.5)
90     Syz=max(Syzs)
91     Nxx=abs((stress2xx+stress8xx)*t2+stress5xx*t1)
92     Nyy=abs((stress2yy+stress8yy)*t2+stress5yy*t1)
93     Mxx=abs((stress2yy-stress8yy)*t2*(t1+t2)*0.5+t2*(stress1yy-
94         stress3yy-stress9yy+stress7yy)*0.5*(0.5*t1+2.0/3.0*t2)+t1
95         *0.25*(stress4yy-stress6yy)/3.0*t1)
96     Myy=abs((stress2xx-stress8xx)*t2*(t1+t2)*0.5+t2*(stress1xx-
97         stress3xx-stress9xx+stress7xx)*0.5*(0.5*t1+2.0/3.0*t2)+t1
98         *0.25*(stress4xx-stress6xx)/3.0*t1)
99
100     #write results, reset reader
101     elresults.append([Nxx,Nyy,Mxx,Myy,Sxy,Sxz,Syz])
102     stressLxx=[[[] for i in range(6)]]
103     stressLyy=[[[] for i in range(6)]]
104     stressGxy=[[[] for i in range(4)]]
105     stressGxz=[[[] for i in range(4)]]
106     stressGyz=[[[] for i in range(4)]]
107     Lcount=0
108     n=0
109     k+=1
110     return elresults
111
112 #read mesh topology
113 def toporead(meshfile):
114     #read .inp file
115     infile=open(dir+meshfile+".inp", 'r')
116     rlines=infile.readlines()
117     infile.close()
118     meshels,nodes,elements=[[[]],[[]],[[]]]
119     nid=0
120     elid=0
121     nlines=False
122     ellines=False
123     for num,line in enumerate(rlines):
124         if "*NODE," in line or "*NODE\n" in line:
125             nlines=True
126             continue
127         elif nlines and "*" in line:
128             nlines=False
129         elif nlines:
130             nodeinf=line.replace("\n","").split(",")
131             nid=int(nodeinf[0])
132             if len(nodes)<nid:
133                 for i in range(nid-len(nodes)):
134                     nodes.append(None)

```

```

129         x=float(nodeinf[1])
130         y=float(nodeinf[2])
131         z=float(nodeinf[3])
132         nodes.insert(nid+1,ghcomp.ConstructPoint(x,y,z))
133     if "*ELEMENT" in line:
134         if "Surface" in line and not "CPS8" in line:
135             raise Exception('Unknown element type! Mesh is probably
136                             invalid!')
137         elif "CPS8" in line or "S8R" in line:
138             ellines=True
139             continue
140         elif ellines and "*" in line:
141             ellines=False
142             break
143         elif ellines:
144             elinf=line.split(",")
145             elnodes=elinf[1:5]
146             elverts=[]
147             for id in elnodes:
148                 elverts.append(nodes[int(id)])
149             meshels.append(ghcomp.ConstructMesh(elverts))
150     return meshels
151
152 #read layer thicknesses
153 def thkread(modelfile):
154     #read file
155     inpf=open(dir+modelfile+".inp", 'r')
156     mlines=inpf.readlines()
157     inpf.close()
158
159     #read thicknesses
160     thks=[]
161     for lid,line in enumerate(mlines):
162         if "*SHELL SECTION" in line and "COMPOSITE" in line:
163             t2=float(mlines[lid+1].split(",")[0])
164             t1=float(mlines[lid+2].split(",")[0])
165             thks.append([t1,t2])
166     return thks
167
168 #visualize internal forces
169 def visuforces(type):
170     global gradts,gradL0,gradL1
171     gradts=[]
172     gradL0=min(type)
173     gradL1=max(type)
174     for force in type:
175         gradts.append(force)
176
177 #read predeformed mesh
178 if retype=="predef":
179     meshels=toporead("predeformation")
180     mesh=ghcomp.MeshJoin(meshels)
181     meshedges=ghcomp.MeshEdges(mesh)

```

```
181     preview=meshedges[0]+meshedges[1]
182
183 #read results
184 else:
185     #read mesh
186     meshels=toporead("topology")
187     mesh=ghcomp.MeshJoin(meshels)
188     meshedges=ghcomp.MeshEdges(mesh)
189     preview=meshedges[0]+meshedges[1]
190
191 #generate internal force maps
192 if not restype=="init":
193     gradmesh=meshels
194
195     #read layer thicknesses
196     thks=thkread("predeformation")
197
198     #read avg. stress in outer faces of the element
199     results=readstress(thks)
200     Nxxs,Nyys,Mxxs,Myys,Sxys,Sxzs,Syys = [], [], [], [], [], [], []
201     for result in results:
202         Nxxs.append(result[0])
203         Nyys.append(result[1])
204         Mxxs.append(result[2])
205         Myys.append(result[3])
206         Sxys.append(result[4])
207         Sxzs.append(result[5])
208         Syys.append(result[6])
209
210     #calculate representative shear forces
211     Sxs,Sys = [], []
212     for i in range(len(Sxys)):
213         Sxs.append(max(Sxys[i],Sxzs[i]))
214         Sys.append(max(Syys[i],Syys[i]))
215
216     #visualize internal forces
217     if restype=="Nx":
218         visuforces(Nxxs)
219     elif restype=="Ny":
220         visuforces(Nyys)
221     elif restype=="Mx":
222         visuforces(Mxxs)
223     elif restype=="My":
224         visuforces(Myys)
225     elif restype=="Sx":
226         visuforces(Sxs)
227     elif restype=="Sy":
228         visuforces(Sys)
```

Bibliography

- American Heritage Dictionary (2011). *American Heritage Dictionary of the English Language*, 5th edn, Houghton Mifflin Harcourt Publishing Company. Polymer.
- Arup (2015). *3D Makeover for Hyper-efficient Metalwork*, http://www.arup.com/news/2015_05_may/11_may_3d_makeover_for_hyper-efficient_metalwork/. [Accessed: 21 July 2016].
- Assaad, J. and Khayat, K. H. (2006). Effect of viscosity-enhancing admixtures on formwork pressure and thixotropy of self-consolidating concrete, *ACI Materials Journal* **103**(4): 280–287.
- Bamforth, P., Chisholm, D., Gibbs, J. and Harrison, T. (2008). *Properties of Concrete for Use in Eurocode 2: How to Optimise the Engineering Properties of Concrete in Design to Eurocode 2*, The Concrete Centre, Camberley.
- Bao, Y. and Wierzbicki, T. (2004). On fracture locus in the equivalent strain and stress triaxiality space, *International Journal of Mechanical Sciences* **46**(1): 81–98.
- Belgisch Instituut voor Normalisatie (1989). *NBN 15-215:1989. Proeven op beton – Wateropslorping door onderdompeling*, Belgisch Instituut voor Normalisatie, Brussels.
- Bergstrom, J. S. (2015). *Mechanics of Solid Polymers*, 1st edn, William Andrew, San Diego-London-Waltham.
- Bower, A. F. (2010). *Applied Mechanics of Solids*, 1st edn, CRC Press, Boca Raton.
- Bártolo, P. J. and Gibson, I. (2011). History of stereolithographic processes, in P. J. Bártolo (ed.), *Stereolithography: Materials, Processes and Applications*, Springer, New York-Dordrecht-Heidelberg-London, pp. 37–56.
- Chua, C., Leong, K. and Lim, C. (2003). *Rapid Prototyping: Principles and Applications*, 2nd edn, World Scientific, New Jersey-London-Singapore-Hongkong.
- D-Shape (2014). *D-Shape Printers*, <http://d-shape.com/d-shape-printers/>. [Accessed: 20 July 2016].

- Dassault Systèmes (2010). *Abaqus 6.10. Benchmarks Manual*.
- Dassault Systèmes (2011). *Abaqus 6.11. Theory Manual*.
- Dhondt, G. (2004). *The Finite Element Method for Three-dimensional Thermomechanical Applications*, John Wiley & Sons, Chichester.
- Dhondt, G. (2015). Calculix crunchix user's manual version 2.8p2.
- DiStasio, C. (2016). *Dubai debuts world's first fully 3D-printed building*, <http://inhabitat.com/dubai-debuts-worlds-first-fully-3d-printed-building/>. [Accessed: 20 July 2016].
- DUS Architects (2014). *3DPRINTCANALHOUSE by DUS Architects*, <http://3dprintcanalhouse.com/>. [Accessed: 21 March 2016].
- DUS Architects (2016). *Europe Building*, <http://www.dusarchitects.com/projects.php?categorieid=publicbuildings&projectid=europebuilding/>. [Accessed: 22 July 2016].
- Farshad, M. (1992). *Design and Analysis of Shell Structures*, Springer-Science+Business Media, Dordrecht.
- Flory, P. J. (1953). *Principles of Polymer Chemistry*, Cornell University Press, Ithaca.
- Fu, S.-Y., Lauke, B. and Mai, Y.-W. (2009). *Science and Engineering of Short Fibre Reinforced Polymer Composites*, Woodhead Publishing, [s.l.].
- Galjaard, S., Hofman, S. and Ren, S. (2015). New opportunities to optimize structural designs in metal by using additive manufacturing, in P. Block, J. Knippers, N. Mitra and W. Wang (eds), *Advances in Architectural Geometry 2014*, Springer, Cham-Heidelberg-New York-Dordrecht-London, pp. 79–93.
- Gardiner, J. B. and Janssen, S. R. (2014). Freefab: Development of a construction-scale robotic formwork 3d printer, in W. McGee and M. P. de Leon (eds), *Robotic Fabrication in Architecture, Art and Design 2014*, Springer International Publishing, [s.l.], pp. 131–146.
- Geuzaine, C. and Remacle, J.-F. (2016). *Gmsh 2.13*, <http://gmsh.info/doc/texinfo/gmsh.html>. [Accessed: 5 August 2016].
- Gosselin, C., Duballet, R., Roux, P. and Morel, P. (2016). Large-scale 3d printing of ultra-high performance concrete – a new processing route for architects and builders, *Materials & Design* **100**(15): 102–109.
- Gramazio, F., Kohler, M. and Willmann, J. (2014). Authoring robotic processes, in F. Gramazio and M. Kohler (eds), *Made by Robots: Challenging Architecture at the Large Scale*, John Wiley & Sons, London, pp. 14–21.
- Gruen, A. (1985). Adaptive least squares correlation: a powerful image matching technique, *South African Journal of Photogrammetry, Remote Sensing and Cartography* **14**(3): 175–187.

- Gutiérrez, E., Dimova, S. and Pinto, A. (2007). Purpose and justification for new design standards regarding the use of fibre-reinforced polymer composites in civil engineering, *Technical report*, European Commission Joint Research Centre, Luxembourg.
- Hack, N. and Lauer, W. (2014). Mesh-mould: Robotically fabricated spatial meshes as reinforced concrete formwork, in F. Gramazio and M. Kohler (eds), *Made by Robots: Challenging Architecture at the Large Scale*, John Wiley & Sons, London, pp. 44–53.
- Harper, C. A. (2002). *Handbook of Plastics, Elastomers, and Composites*, 4th edn, McGraw-Hill, [s.l.].
- Holzapfel, G. A., Gasser, T. C. and Ogden, R. W. (2000). A new constitutive framework for arterial wall mechanics and a comparative study of material models, *Journal of elasticity and the physical science of solids* **61**(1): 1–48.
- Hull, C. W. (1986). *Apparatus for Production of Three-dimensional Objects by Stereolithography*, US4575330 A.
- Institute for Advanced Architecture of Catalonia (2015). *3D Printing with Soil and Natural Materials*, <http://pylos.iaac.net/main.html/>. [Accessed: 21 July 2016].
- Keating, S., Spielberg, N. A., Klein, J. and Oxman, N. (2014). A compound arm approach to digital construction: A mobile large-scale platform for on-site sensing, design, and digital fabrication, in W. McGee and M. P. de Leon (eds), *Robotic Fabrication in Architecture, Art and Design 2014*, Springer International Publishing, [s.l.], pp. 99–110.
- Le, T., Austin, S., Lim, S. and Thorpe, A. (2012). Hardened properties of high-performance printing concrete, *Cement and Concrete Research* **42**(3): 558–566.
- Malaeb, Z., Hachem, H., Tourbah, A., Maalouf, T., Zarwi, N. E. and Hamzeh, F. (2015). 3d concrete printing: Machine and mix design, *International Journal of Civil Engineering and Technology* **6**(6): 14–22.
- McGee, W. and de Leon, M. P. (2014). Preface, in W. McGee and M. P. de Leon (eds), *Robotic Fabrication in Architecture, Art and Design 2014*, Springer, Cham-Heidelberg-New York-Dordrecht-London, pp. VII–X.
- MX3D (2015). *MX3D Bridge*, <http://mx3d.com/projects/bridge/>. [Accessed: 21 July 2016].
- Nederlands Normalisatie-instituut (1997). *NEN-EN-ISO 527-4:1997 (en). Plastics - Determination of tensile properties - Part 4: Test conditions for isotropic and orthotropic fibre-reinforced plastic composites*, Nederlands Normalisatie-instituut, Delft.
- Nederlands Normalisatie-instituut (2003). *NEN-EN-ISO 899-1:2003 (en). Plastics - Determination of creep behaviour - Part 1: Tensile creep*, Nederlands Normalisatie-instituut, Delft.
- Nederlands Normalisatie-instituut (2008). *NEN-EN 1990 (en). Eurocode: Basis of structural design*, Nederlands Normalisatie-instituut, Delft.

- Nederlands Normalisatie-instituut (2009). *NEN-EN 12390-3:2009 (en). Testing hardened concrete - Part 3: Compressive strength of test specimens*, Nederlands Normalisatie-instituut, Delft.
- Nederlands Normalisatie-instituut (2010). *NEN-EN-ISO 178:2010 (en). Plastics - Determination of flexural properties*, Nederlands Normalisatie-instituut, Delft.
- Nederlands Normalisatie-instituut (2012). *NEN-EN-ISO 527-1:2012 (en). Plastics - Determination of tensile properties - Part 1: General principles*, Nederlands Normalisatie-instituut, Delft.
- Neville, A. M. (2012). *Properties of Concrete*, 5th edn, Pearson Education Limited, Harlow.
- Oak Ridge National Laboratory (2015). *AMIE Demonstration Project*, <http://web.ornl.gov/sci/eere/amie/>. [Accessed: 21 July 2016].
- Perrot, A., Rangeard, D. and Pierre, A. (2016). Structural built-up of cement-based materials used for 3d-printing extrusion techniques, *Materials and Structures* **49**(4): 1213–1220.
- Peters, B. P. (2015). *Formwork for architectural applications and methods*, US 2015/0336297.
- Peulen, S. (2015). *Vergroten knik gedrag van een stalen staaf door inzet van 3d geprint materiaal*, Bachelor's thesis, Delft University of Technology, Delft.
- Ployaert, C. (2009). *Duurzaam Beton door Beheersing van de Waterabsorptie*, Federatie van de Belgische Cementnijverheid, Brussels.
- Price, W. H. (1951). Factors influencing concrete strength, *Journal of American Concrete Institute* **22**(6): 417–432.
- Proske, T. and Graubner, C.-A. (2002). Self-compacting concrete - pressure on formwork and ability to deaerate, *Concrete Structures* **17**.
- Reitz, K. and Schlusser, T. (2016). *The Hitchhiker's Guide to Python: Best Practices for Development*, O'Reilly Media, Sebastopol.
- Rudenko, A. (2014). *3D Printed Concrete Castle is Complete*, <http://www.totalkustom.com/3d-castle-completed.html/>. [Accessed: 20 July 2016].
- Schaffer, J. P., Saxena, A., Antolovich, S. D., Sanders, T. H. and Warner, S. B. (1999). *The Science and Design of Engineering Materials*, 2nd edn, WCB/McGraw-Hill, [s.l.].
- Sevenson, B. (2015). *Shanghai-based WinSun 3D Prints 6-Story Apartment Building and an Incredible Home*, <https://3dprint.com/38144/3d-printed-apartment-building/>. [Accessed: 20 July 2016].
- Shah, V. (2007). *Handbook of Plastics Testing and Failure Analysis*, 3rd edn, John Wiley & Sons, Hoboken.
- Swallowe, G. M. (1999). Tensile and compressive testing, in G. M. Swallowe (ed.), *Mechanical Properties and Testing of Polymers: An A-Z Reference*, Kluwer Academic Publishers, Dordrecht-London, pp. 242–247.

- Technical information: MACROMELT 6900E* (2009). *Technical report*, Henkel Italia S.p.A., Casarile.
- van Baarsen, S. B., Schönwälder, J., Houtman, R., van der Veen, A., Vermeulen, H. and de Haan, S. (2015). The 3d printed canal house, *Proceedings of the International Association for Shell and Spatial Structures(IASS). Symposium 2015, Amsterdam*.
- van der Veen, A. (2014). *The structural feasibility of 3d-printing houses using printable polymers*, Master's thesis, Delft University of Technology, Delft.
- van der Vegt, A. K. (2006). *From Polymers to Plastics*, VSSD, Delft.
- Ward, I. M. and Sweeney, J. (2013). *Mechanical Properties of Solid Polymers*, 3rd edn, John Wiley & Sons, Chichester.
- World's Advanced Saving Project (2016). *SHAMBALLA – WASP'S technological village*, <http://www.wasproject.it/w/en/shamballa-wasps-technological-village/>. [Accessed: 21 July 2016].

List of Figures

1-1	3D Print Canal House.	1
1-2	Research method.	4
2-1	Buildings reported to be printed by WinSun.	5
2-2	D-Shape binder-based printer.	6
2-3	3D printing with steel.	7
2-4	Mesh-Mould technology, ETH Zurich.	7
2-5	Kamermaker 3D printer.	8
2-6	3D Print Canal House: application of concrete.	9
3-1	The structure of polyethylene.	11
3-2	Linear, branched and crosslinked polymer.	12
3-3	Load–elongation curves of a semi-crystalline thermoplastic.	13
3-4	Temperature/stiffness relationship of different types of materials.	14
3-5	Polymer extension modes.	14
3-6	Elastic response of a viscoelastic body.	15
3-7	Fiber distribution within the cross-section.	17
3-8	Varying print quality within a single element.	18
3-9	Expected maximum acceptable print span between the turns of the nozzle.	18
4-1	Local coordinate system of a triple-layer element.	19
4-2	Timeframe for the test phase.	21
4-3	Print piece consisting of 13 square plates with workable areas of 240 × 240 mm.	21
4-4	Optimized laser cut layout and processed plate.	27
4-5	Built-up mould geometry, axial dimensions.	28
5-1	Stress-strain curve illustrating the toe effect.	32

5-2	Effective cross section working against transverse tension and bending.	33
5-3	Results: longitudinal tensile test, 2 mm/min.	36
5-4	Results: longitudinal tensile test, 10 mm/min.	37
5-5	Results: longitudinal tensile test, 50 mm/min.	38
5-6	Tensile test: different failure modes.	39
5-7	Results: transverse tensile test, 2 mm/min.	40
5-8	Results: transverse tensile test, 10 mm/min.	41
5-9	Results: transverse tensile test, 50 mm/min.	42
5-10	Results: longitudinal flexural test.	44
5-11	Results: transverse flexural test.	46
5-12	Results: shear between the layers test. Part 1.	47
5-13	Results: shear between the layers test. Part 2.	47
5-14	Shear between the layers test: interpolation for class 3.	48
5-15	Results: shear across the print test.	49
5-16	Shear test: yield due to delamination.	49
5-17	Results: influence of the temperature on tensile stiffness.	52
5-18	Results: influence of the temperature on tensile strength.	52
5-19	Casting of a single layer form.	54
5-20	Casting of a built-up mould.	59
5-21	Concrete compressive strength test: crack pattern.	61
6-1	Unequal material curing throughout the cross section.	66
6-2	Expansion of a 2D 8-node element into a 3D brick element in CalculiX.	70
6-3	Distribution of integration points within C3D20R element.	70
6-4	Plane symmetric, triple layer element.	71
6-5	Modelling of the concrete pressure.	73
6-6	Distribution of integration points, strains and stresses over a triple layer shell.	74
6-7	Element scheme for derivation of shear forces.	76
6-8	Flowchart of an individual finite element simulation.	77
7-1	Numerical model of a tensile test.	79
7-2	Numerical model of a flexural test.	81
7-3	Specimen slippage during 3-point bending.	82
7-4	Numerical model of a single layer mould test.	83
7-5	Numerical model of a built-up mould test.	84
8-1	Predeformation-based design method principle.	87
8-2	Snapthrough effect.	92
8-3	Grasshopper implementation - canvas layout.	95
8-4	Mould predeformation flowchart.	97

9-1	Case study: expected shape of a parametric column.	100
9-2	Case study: expected shape of a wall panel.	101
9-3	Case study: wall panel after demoulding.	101
9-4	Case study: numerical model of a single layer mould.	102
9-5	Case study: numerical model of a parametric column.	103
9-6	Case study: numerical model of panel front.	104
9-7	Case study: numerical model of panel back.	105
A-1	Stiffness of thermoplastics: overview.	115
A-2	Tensile strength of thermoplastics: overview.	116
B-1	Raw results: tensile longitudinal test, 1 day.	117
B-2	Raw results: tensile longitudinal test, 3 days.	118
B-3	Raw results: tensile longitudinal test, 7 days.	118
B-4	Raw results: tensile longitudinal test, 14 days.	119
B-5	Raw results: tensile longitudinal test, 28 days.	119
B-6	Raw results: tensile transverse test, 1 day.	120
B-7	Raw results: tensile transverse test, 3 days.	120
B-8	Raw results: tensile transverse test, 7 days.	121
B-9	Raw results: tensile transverse test, 14 days.	121
B-10	Raw results: tensile transverse test, 28 days.	122
B-11	Raw results: flexural longitudinal test, 1 day.	122
B-12	Raw results: flexural longitudinal test, 3 days.	123
B-13	Raw results: flexural longitudinal test, 7 days.	123
B-14	Raw results: flexural longitudinal test, 14 days.	124
B-15	Raw results: flexural longitudinal test, 28 days.	124
B-16	Raw results: flexural transverse test, 1 day.	125
B-17	Raw results: flexural transverse test, 3 days.	125
B-18	Raw results: flexural transverse test, 7 days.	126
B-19	Raw results: flexural transverse test, 14 days.	126
B-20	Raw results: flexural transverse test, 28 days.	127
B-21	Raw results: shear along the print test, 1 day.	127
B-22	Raw results: shear along the print test, 3 days.	128
B-23	Raw results: shear along the print test, 7 days.	128
B-24	Raw results: shear along the print test, 14 days.	129
B-25	Raw results: shear along the print test, 28 days.	129
B-26	Raw results: shear across the print test, 1 day.	130
B-27	Raw results: shear across the print test, 3 days.	130
B-28	Raw results: shear across the print test, 7 days.	131

B-29 Raw results: shear across the print test, 14 days.	131
B-30 Raw results: shear across the print test, 28 days.	132
B-31 Raw results: creep test, 1 day.	132
B-32 Raw results: creep test, 3 days.	133
B-33 Raw results: creep test, 7 days.	133
B-34 Raw results: creep test, 14 days.	134
B-35 Raw results: creep test, 28 days.	134
B-36 Raw results: thermal sensitivity test, 1 day.	135
B-37 Raw results: thermal sensitivity test, 3 days.	135
B-38 Raw results: thermal sensitivity test, 7 days.	136
B-39 Raw results: thermal sensitivity test, 14 days.	136
B-40 Raw results: thermal sensitivity test, 28 days.	137
D-1 Tensile specimen: original dimensions.	143
F-1 Test platform design, overview drawing.	147
I-1 Photogrammetry setups.	156
I-2 Reference plane definition.	157

List of Tables

3-1	Composition of the material used in the project.	16
4-1	Specification: tensile test.	22
4-2	Specification: flexural test.	23
4-3	Specification: shear test.	24
4-4	Specification: tensile creep test.	24
4-5	Specification: thermal sensitivity test.	25
4-6	Specimen summary.	26
5-1	Dimensional properties of cross section classes.	33
5-2	Results: longitudinal tensile test, 2 mm/min.	36
5-3	Results: longitudinal tensile test, 10 mm/min.	37
5-4	Results: longitudinal tensile test, 50 mm/min.	38
5-5	Results: transverse tensile test, 2 mm/min.	40
5-6	Results: transverse tensile test, 10 mm/min.	41
5-7	Results: transverse tensile test, 50 mm/min.	42
5-8	Results: longitudinal flexural test.	44
5-9	Results: transverse flexural test.	45
5-10	Results: shear between the layers test.	48
5-11	Results: shear across the print test.	50
5-12	Results: creep test.	50
5-13	Results: deformations of single layer moulds, 1 day old.	55
5-14	Results: deformations of single layer moulds, 3 days old.	55
5-15	Results: deformations of single layer moulds, 7 days old.	56
5-16	Results: deformations of single layer moulds, 14 days old.	56
5-17	Results: deformations of single layer moulds, 28 days old.	57

5-18	Results: deformations of single layer moulds, 28 days old, SCC.	57
5-19	Results: deformations of a built-up mould, 28 days old.	58
5-20	Results: normal weight C20/25 concrete compressive strength.	60
5-21	Results: self-compacting C28/35 concrete compressive strength.	60
5-22	Results: concrete porosity.	61
6-1	Overview of material models used in modelling of polymers.	64
6-2	Results: longitudinal tensile test, influence of the strain rate.	73
7-1	Validation: longitudinal tensile test, classes 1-5.	80
7-2	Validation: transverse tensile test, class 1.	80
7-3	Validation: transverse tensile test, class 5.	80
7-4	Validation: longitudinal flexural test, class 1.	81
7-5	Validation: longitudinal flexural test, class 5.	81
7-6	Validation: transverse flexural test, class 1.	82
7-7	Validation: transverse flexural test, class 5.	82
7-8	Validation: deformations of single layer forms.	85
7-9	Validation: deformations of built-up mould.	86
8-1	Characteristic strength: tensile longitudinal $\frac{F_{x,k}}{b}$	88
8-2	Characteristic strength: tensile transverse $\frac{F_{y,k}}{b}$	88
8-3	Characteristic strength: flexural longitudinal $\frac{M_{y,k}}{b}$	89
8-4	Characteristic strength: shear across the layers $\frac{S_{x,k}}{b}$	89
8-5	Characteristic strength: shear between the layers $\frac{S_{y,k}}{b}$	89
9-1	Case study results: Single layer mould.	102
9-2	Case study results: Parametric column.	103
9-3	Case study results: Panel front.	104
9-4	Case study results: Panel back.	105
C-1	Reclassification: transverse tensile stiffness.	140
C-2	Reclassification: longitudinal flexural stiffness.	140
C-3	Reclassification: transverse flexural stiffness.	140
C-4	Reclassification: transverse tensile strength.	141
C-5	Reclassification: longitudinal flexural strength.	141
C-6	Reclassification: transverse flexural strength.	142
C-7	Reclassification: shear strength along the print.	142
E-1	Values of k_n for the 5% characteristic value.	146

