Feed velocity feedback for high speed fused deposition modelling machines

Ward Johannes Heij

Supervisors: ir. W. Wolfslag dr. ir. E. de Bruijn

Delft, October 31, 2016





Faculty of Mechanical, Maritime and Materials Engineering Department of Biomechanical Engineering

Feed velocity feedback for high speed fused deposition modelling machines

by

Ward Johannes Heij

Submitted to the Department of Biomechanical Engineering in partial fulfilment of the requirements for the degree of

Master of Science in Mechanical Engineering

 at

DELFT UNIVERSITY OF TECHNOLOGY Faculty of Mechanical, Maritime and Materials Engineering Department of Biomechanical Engineering to be defended publicly on Monday, October 31, 2016 at 10:00.

> Supervisors: ir. W. Wolfslag dr. ir. E. de Bruijn

| Committee: | ir. W. Wolfslag | TU Delft, BME |
|------------|-----------------------------|----------------|
| | prof. dr. ir. M. Wisse | TU Delft, BME |
| | dr. ir. T.J.J. van den Boom | TU Delft, DCSC |

Preface

Ever since my youth, I have had a passion for technology and engineering. The choice of going to a technical university was never hard, and at Delft University of Technology I finally learned what was necessary to upgrade my home-made cars - made from toothbrush motors and pieces of wood - into impressive machines. During my bachelor and master, I developed a fond appreciation for robotics, mechatronics and the entire design process around it: from hardware to kinematics to software and control.

My interest for 3D printing resulted from the fact that some (robotic) parts sometimes had exquisite shapes, and to manufacture those a non-traditional manufacturing process was required. Even though my academic encounter with 3D printing was short, it was enough to provoke my interest and a drive to learn more. The field of 3D printer is, to me, highly interesting since it is so new and unexplored.

During the conception of this thesis, I got the opportunity to work at the front lines of innovation and technology by means of a full-year internship at Leapfrog 3D Printers. I would foremost like to thank Leapfrog and her employees for the enjoyable work atmosphere and open attitude towards new ideas. It has been inspiring to be part of a young company that aims to venture into the unknown realm of 3D printing. This is also how I reinforced my fondness of innovative technologies.

I would like to express my personal thanks to Edo de Bruijn from Leapfrog 3D Printers, who was an excellent supervisor. Edo was open-minded, receptive to new ideas and - despite his extremely crowded schedule - always available for sparring and feedback. After a sparring session, I was always left with a head full of new ideas and considerations.

Additionally, I wish express great appreciation for the supervision of Wouter Wolfslag. Next to obtaining his PhD degree, he found the time to regularly meet and provide feedback, which was always very clear and to the point. I could not have written this thesis as it is now without his constructive though critical attitude.

W. Heij, July 2016

vi

Contents

| Pı | refac | e | \mathbf{v} |
|----|--------------|-------------------------------------------------------|-----------------|
| Su | ımm | ary | xi |
| N | omer | nclature | xiii |
| 1 | Intr | roduction | 1 |
| | $1.1 \\ 1.2$ | Problem statement Goals and outline of this thesis | $\frac{3}{6}$ |
| 2 | An | introduction to additive manufacturing | 7 |
| | 2.1 | AM technologies | 7 |
| | 2.2 | FDM in-depth | 10 |
| | 2.3 | Materials | 15 |
| 3 | Sys | tem identification and modelling | 17 |
| | 3.1 | System description | 19 |
| | 3.2 | Slip-free system dynamics | 25 |
| | | 3.2.1 Identification framework | 25 |
| | | 3.2.2 ARMAX implementation and system parameters | 27 |
| | 3.3 | Application of the Pacejka tyre slip model | 31 |
| | | 3.3.1 Theoretical framework and reworking the formula | 31 |
| | | 3.3.2 Implementation | 33 |
| | 3.4 | Combined model | 38 |
| 4 | Cor | ntroller synthesis | 41 |
| | 4.1 | Algorithm | 41 |
| | | 4.1.1 Hybrid control | 43 |
| | 4.2 | Off-line implementation using the model | 45 |
| | | 4.2.1 Tuning | 46 |
| | 4.3 | Adding noise to the model and controller | 51 |
| | | 4.3.1 Model | 51 |
| | | 4.3.2 Control with noise | 51 |
| | 4.4 | On-line implementation | 55 |
| | | 4.4.1 Experimental setup | $\overline{55}$ |

CONTENTS

| | | 4.4.2 | Control signal: getting the controller and printer to chat | 55 | |
|--------------------------|-------------------------------------------------------------|---------|------------------------------------------------------------|----|--|
| | | 4.4.3 | Performance metrics | 57 | |
| | | 4.4.4 | Tuning and results | 57 | |
| 5 Performance evaluation | | | | 61 | |
| | 5.1 | Metho | d for evaluation | 61 | |
| | 5.2 | Result | 8 | 61 | |
| | | 5.2.1 | Step input tracking | 62 | |
| | | 5.2.2 | Sawtooth input tracking | 66 | |
| | | 5.2.3 | On the effectiveness of on-line filtering | 70 | |
| 6 | Discussion 7: | | | 73 | |
| 7 | 7 Conclusions | | | 77 | |
| Bi | bliog | graphy | | 81 | |
| A | A G-Codes | | | 83 | |
| в | B Experiment Description 85 | | | 85 | |
| | B.1 | List of | Used Equipment | 85 | |
| | B.2 Connection of Hardware | | | 85 | |
| | B.3 | Visual | material | 87 | |
| С | C Feed-forward slip prevention compared to feedback control | | | 89 | |
| D | Dig | ital ap | D Digital appendix 9 | | |

List of Figures

| Schematic of AM | 2 |
|----------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Thermodynamic effects in Leapfrog hot end | 4 |
| Extrusion mechanism and slip | 5 |
| FDM part affected by insufficient material deposition | 5 |
| Overview of AM technologies | 8 |
| Selective laser sintering | 9 |
| Stereolithography | 9 |
| Inkjet 3D printing | 10 |
| Fused deposition modeling in detail | 10 |
| 3D printing workflow | 11 |
| STL file meshing | 12 |
| The slicing process | 13 |
| G-code explained | 13 |
| Technical workflow | 14 |
| Schematic of Leapfrog Creatr HS | 14 |
| Direct drive and Bowden tube systems compared | 18 |
| Leapfrog Xeed | 19 |
| Extruder pinch rollers | 20 |
| Velocity input profile showing slip behaviour of the extrusion mechanism | 21 |
| Velocity input profile showing slip behaviour of the extrusion mechanism, detail | 21 |
| Velocity input profile averaged over 10 experiments | 22 |
| Step inputs averaged over 10 experiments | 23 |
| Analog electrical circuit representing extruder dynamics. | 24 |
| IO data used for ARMAX | 27 |
| Determination of ARMAX model order | 28 |
| Simulated no-slip model step response | 29 |
| Simulated no-slip model response for higher feed velocities | 30 |
| Simulated no-slip model step response for input velocity sawtooth | 30 |
| Example tyre slip curve | 32 |
| Influence of Pacjeka model parameters B and C on achieved data fit | 34 |
| Influence of Pacjeka model parameters B and E on achieved data fit | 34 |
| Pacejka slip curve | 37 |
| Complete model block scheme | 38 |
| | Schematic of AM Thermodynamic effects in Leapfrog hot end Extrusion mechanism and slip FDM part affected by insufficient material deposition Overview of AM technologies Selective laser sintering Stereolithography Inkjet 3D printing The slicing process STL file meshing The slicing process G-code explained Technical workflow Schematic of Leapfrog Creatr HS Direct drive and Bowden tube systems compared Leapfrog Xeed Extruder pinch rollers Velocity input profile showing slip behaviour of the extrusion mechanism Velocity input profile showing slip behaviour of the extrusion mechanism, detail Velocity input profile showing slip behaviour of the extrusion mechanism, detail Velocity input profile showing slip behaviour of the extrusion mechanism, detail Velocity input profile showing slip behaviour of the extrusion mechanism, detail Velocity input profile showing slip behaviour of the extrusion mechanism, detail Velocity input profile showing slip behaviour of the extrusion mechanism, detail Velocity input profile showing slip behaviour of the extrusion mechanism, detail Velocity input profile showing slip behaviour of the extrusion mechanism, detail |

| 3.19 | Simulated response of combined model | 38 |
|-------------|------------------------------------------------------------|----------|
| 3.20 | Simulated response of combined model | 39 |
| | | |
| 4.1 | Theoretical relay loop of a Schmitt trigger | 44 |
| 4.2 | Full system block scheme | 45 |
| 4.3 | Implementation of ESC control | 45 |
| 4.4 | Implementation of ESC control | 46 |
| 4.5 | Tuning of Schmitt trigger switch boundary magnitude | 48 |
| 4.6 | Tuning of Schmitt trigger switching boundary amplitude | 49 |
| 4.7 | Tuning of Schmitt trigger up- and down ramp | 50 |
| 4.8 | Full system block scheme | 51 |
| 4.9 | Implementation of ESC control | 52 |
| 4.10 | Implementation of ESC control | 53 |
| 4.11 | Effects of noise model on optimal Schmitt trigger settings | 54 |
| 4.12 | Ring buffer schematic | 56 |
| 4.13 | Controller block scheme | 56 |
| 4.14 | Controller block scheme using multiplication | 56 |
| 4.15 | Step input controller tuning | 58 |
| 4.16 | Step input controller ramp tuning | 59 |
| F 1 | | co |
| 0.1 7 0 | Reference step input time trace | 02 C2 |
| 5.2 | Step input time trace with control | 63 |
| 5.3 | Step input time trace with control, detail | 64 67 |
| 5.4 | Performance metrics for step input | 65 |
| 5.5 | Reference sawtooth input time trace | 66 |
| 5.6 | Sawtooth input time trace with control | 67 |
| 5.7 | Sawtooth input time trace with control, detail | 68 |
| 5.8 | Performance metrics for sawtooth input | 69 |
| 5.9 | Performance metrics for sawtooth input using SMA | 71 |
| B .1 | Communication layout | 86 |
| B 2 | Wiring schematic | 87 |
| B 3 | Experimental setup | 88 |
| B.4 | Detail of print head, annotated | 88 |
| | • ••••,• • •••• | |
| C.1 | Performance metrics for step input, feed-forward optimized | 91 |

Summary

Additive manufacturing (AM) is a manufacturing method with two distinct advantages over traditional subtractive methods. First, it allows for much higher form freedom, allowing for shapes that are unobtainable by conventional techniques. Second, this method is also referred to 'rapid prototyping' given the fact that objects, and mainly prototype iterations, may be produced in a matter of hours instead of days. Fused deposition modelling (FDM) is the most ubiquitous and accessible AM technology, relying on the layer-by-layer deposition of molten thermoplastics. While FDM is very accessible and shows great potential as a future manufacturing technique, it is still inferior to traditional techniques due to the very low production capacity and limited speed. Print speed is mainly limited by volumetric flow rate through the nozzle, of which the counter pressure at high velocity causes filament slip in the extrusion mechanism. Therefore, the goals of this thesis are (1) to design an accurate extruder model to describe input-output behaviour of feed velocity. This may yield better understanding of FDM extrusion and form a basis for model-based design. (2) The synthesis of an anti-slip controller allowing for high print speed while retaining object quality. When a higher print speed could be attained, one of the largest disadvantages of FDM would be mitigated and a new step is made towards industrial application of this technology.

To make the model of the extruder, the only known velocity input-output model available was taken and verified. This black-box transfer function model accurately describes low-speed behaviour of the extruder. This model, however, did not incorporate slip phenomena, making it an inaccurate model at high volume flow. Hence, a model from the automotive industry, also known as the 'Magic Formula' was combined with this known transfer function, producing a new black-box model covering all feed velocities. This new model has opportunity to be used as a basis for model-based design of 3D printers, which are now generally designed based on past experience and trial-and-error. For the future, it would be highly interesting to find an analytical model rather than a transfer-based one in order to gain physical understanding in the influence of extruder parameters. Regarding the synthesis of an anti-slip controller, we implemented a type of extremum seeking hybrid control which essentially finds an feasible maximum velocity for a limited amount of slip. The logic of the controller was based on Schmitt trigger logic, defining three control parameters that together define the switching behaviour of the controller. After tuning, if was found that the controller improved extrusion errors as well as completion time compared to open-loop control, achieving the set goals. The successful application of such a controller could mitigate the low velocity disadvantages of FDM and facilitate the acceptance of this technology. In the future, it is recommended to further investigate the practical applications of the controller, find the limitations of the controller in real printing and explore different control architectures.

LIST OF FIGURES

Nomenclature

Symbols

| d(t) | Disturbance signal |
|-----------|---------------------------------------------|
| e_{RMS} | Root-mean-squared error |
| e(t) | Error |
| k | Ramp gain of ESC controller |
| r(t) | Reference signal |
| s | Laplace operator |
| w(t) | Zero-mean white noise |
| y(t) | Output signal |
| A, B, C | State space matrices |
| C(s) | Controller model |
| G(s) | Plant model |
| V_{IL} | Lower boundary threshold of Schmitt trigger |
| V_{IH} | Upper boundary threshold of Schmitt trigger |
| | |

Greek Symbols

- α Slip coefficient
- μ_k Friction coefficient
- ϕ Volumetric flow rate (mm³/s)

Abbreviations and Acronyms

| 3D | Three-dimensional |
|----------------|-----------------------------------------------------|
| AM | Additive manufacturing |
| ARMAX | Auto-Regressive Moving Average with eXogenous input |
| DOF | Degree Of Freedom |
| ESC | Extremum Seeking Control |
| FDM | Fused Deposition Modeling |
| \mathbf{FFF} | Fused Filament Fabrication |
| \mathbf{FFT} | Fast Fourier Transform |

LIST OF FIGURES

| I/O | input-output |
|--------|--------------------------------------|
| I^2C | Inter-Integrated Circuit |
| IVM | Instrument variable method |
| MLM | Maximum likelihood method |
| MIMO | Multiple Input, Multiple Output |
| P(I)D | Proportional, (Integral,) Derivative |
| PCB | Printed circuit board |
| PTFE | Polytetrafluorethylene |
| PWM | Pulse Width Modulation |
| RMSE | Root-mean-squared error |
| SID | Subspace-based system identification |
| SISO | Single Input, Single Output |
| TTC | Time-to-complete |
| | |

 xiv

Chapter 1

Introduction

In the 1930's, archaeologist Louis Leakey discovered a set of ancient, prehistoric stone tools in the Olduval Gorge in Tanzania. This range of stone tools would become known as the Oldowan, being the oldest fabricated tools known to man and dating back until 2.6 million years ago. Our ancestors from the Lower Paleolithic have come a long way since then, and showing these stone tool-wielding humans the current state of technology probably would have dropped a few jaws. During the last few millennia, humans learned to manipulate metal and create even stronger tools. In the last few centuries, the arrival of the steam machine catalysed the industrial revolution, bringing unprecedented manufacturing capability and capacity. After that, digital control of production was introduced, and computer-controlled milling, turning and casting machines further increased freedom of design. Where traditional casting and milling severely limited the possibility in shapes, computerized equipment and high-precision tools vastly increased the possibilities.

However, traditional production techniques still have their limitations. Many shapes, such as hollow objects or curved surfaces are hard to make or require expensive tools such as moulds. An additional problem to such items is that their fabrication process is very specific: a mould is made to generate one particular shape, and any tweak to the design requires a completely new one. Especially for engineers, designers and architects, a high amount of design freedom, as well as the ability to quickly apply changes to a design and realize it, is required. Therefore a relatively new production technique, allowing for virtually unlimited design freedom, is rapidly gaining popularity: additive manufacturing. Additive manufacturing is more widely known as 3D printing and concerns the layer-by-layer fabrication of computer generated models (fig. 1.1). Leapfrog 3D Printers is a Netherlands-based company producing 3D printers operating with fused deposition modelling (FDM), which melts and deposits thermoplastic polymers. FDM is the most common AM technology.

The core principle of this method is the addition of many small layers, eventually creating an object of highly complex geometry within a relatively small amount of time. The applications of this design freedom are numerous: architects can 3D print maquettes, engineers can quickly design and produce prototypes. If changes are made, a new prototype can be available within hours instead of weeks, often at a fraction of the costs of traditional manufacturing techniques.



Figure 1.1 : Schematic of fused deposition modelling (FDM, a type of AM) 3D printing. (Turner et al., 2014)

It may seem like 3D printing is an omnipotent technology that will make all other production techniques obsolete. So what has prevented 3D printing from conquering the globe thus far. If 3D printing really were so easy and convenient, wouldn't everyone buy a 3D printer and print everything they needed?

The answer, of course, is that 3D printing is not *at all* that convenient, and therefore has not - yet - conquered the world. A number of reasons exist for this, the first being the limitations in available materials. Most FDM printers are limited to plastics, so the production of metal or exceptionally strong parts is quite hard using FDM.

Another problem in FDM printing is build volume. For the creation of parts larger than ≈ 30 cm, one needs either an industrial (and thus rather expensive) printer, or another production method. This problem is partially solved by Leapfrog with their prototype Xeed XL, which is a very large FDM machine with a larger nozzle and high build volume, allowing for objects of up to two metres high to be printed.

However, this last solution brings to light the largest problem with FDM: speed. Small objects can easily take hours to print. For large objects, the time scale changes to *days*, making it painstakingly slow. This long print time is currently the main limitation to 3D printing, and one of the prime attention points in the development of new AM techniques in general (Bikas et al., 2015).

Hence, the acceptance and usefulness of FDM printing can be greatly increased when print speed is increased (Pham and Gault, 1998). If parts were made faster while retaining quality, 3D printing will become attractive for small scale production rather than prototyping, especially when printing large objects becomes possible. In such a scenario, the prime advantages of AM such as extremely high form freedom and short time from design to physical object would really be emphasized. Second, it will also become more likely to be accepted by the consumer market. Therefore,

decreasing the required time to print an object could yield huge advantages.

1.1 Problem statement

As previously mentioned, FDM speed prohibits the wide acceptance of this technology. Let us first review the two main bottlenecks currently limiting print speed:

- 1. Motion of the print head. Due to the feed-forward nature of the actuation system and the hardware used (stepper motors, finite-bandwidth stepper drivers) the translational velocity of the head is limited.
- 2. Attainable extrusion volumetric flow rate. Both the thermodynamical properties of the extruder, as well as the mechanical means for applying an extrusion force to the filament determine the maximum feed velocity.

In previous investigation, the attainable translational velocity of the head was increased with over a twofold, from 200 mm/s to 500 mm/s (Heij, 2015), and the attainable feed velocity was increased (Riemens, 2016). From this previous work, it became clear that extrusion feed velocity is the dominant bottleneck, as the translational head velocity far exceeds the attainable feed velocity. Currently, the maximum volumetric flow rate that can be attained is 15 mm³/s, corresponding to a head velocity of approximately 150 mm/s. Feed velocity is mainly limited by two factors, being the thermodynamic characteristics of the heated extruder, and the mechanical properties of the extrusion mechanism. We will shortly investigate both for potential to improve.

First, let us review thermodynamic properties of the extruder and how these impose a maximum feed velocity. Heating of the plastic occurs by direct wall contact between the plastic and the inside of the nozzle. The bulk temperature of the hot end is kept constant at a desired extrusion temperature by a heating cartridge and thermistor in a closed-loop PID temperature control system. Thermal effects limit the attainable extrusion velocity since the maximum heat flow between filament and wall may not be high enough to fully heat the filament to the temperature set point. This is depicted in the simulation shown in figure 1.2. Wall temperature is assumed constant at 220 °C. For this, we can see that the filament core temperature increases along the length of the hot end, but does not reach the desired set point. It's final value is 420 K or 147 °C, giving it an error of 73 °C. This large error is simulated for an extrusion volumetric flow rate of $\phi = 15 \text{ mm}^3/\text{s}$. The viscosity of the melt decreases with temperature according to the power-law model (Nikzad et al., 2009):

$$\eta = K\hat{\gamma}^{n-1} \tag{1.1}$$

where $\eta, K, \hat{\gamma}$ and *n* are the viscosity, consistency index, shear rate, and power law index which have to be determined empirically for each material. Given the large temperature error, viscosity of the melt is much lower at the extrusion point compared to fully molten plastic. Since a higher viscosity yields a higher pressure drop, much greater extrusion force is required, which cannot be delivered by the extrusion mechanism. This fails and introduces slip. As a secondary result of insufficient heating, the extruded filament does not stick to previous layers, resulting in bad part quality.



Figure 1.2 : Thermodynamic effects in Leapfrog hot end. (a) Computational fluid dynamics (CFD) simulation of Leapfrog hot-end. Wall temperature is assumed constant at 220 °C. (b) Core temperature of the extrusion melt assuming the power-law model, along the length dimension of the hot tube. The peak at $L \approx 0.05$ is a numerical artefact and may be disregarded.

Not only does the pressure drop increase due to insufficient heating, extrusion pressure increases as well due to higher feed rates. These two factors combined results in a higher reaction force on the extrusion mechanism (fig. 1.3). The reaction force from the pressure drop causes slip between the mechanism and the feed material as there is insufficient traction between the pinch roller and filament strand. Hence, it is currently impossible to print with speeds > 150 mm/s due to slip occurring, caused by too high reaction force from the extruder.

Any slip has a significant impact on part integrity. Since current FDM printers are feed-forward controlled, slip is not accounted for when it occurs and the machine does not adapt its speed to filament slippage. As a result, insufficient material is deposited and part integrity decreases (fig. 1.4). Hence, it is desired to implement a feedback controller that aims to minimize slip while maximizing feed velocity. The implementation of such a control loop will prevent filament slippage, thus ensuring constant material deposition, adequate heat transfer in the hot tube and good part integrity. Second, it will push any FDM machine to its speed limit without sacrificing part quality due to insufficient polymer deposition. If successful, this is one step closer to high-speed FDM and will increase this technology's applicability in production processes.



Figure 1.3 : (a) Schematic overview of the extrusion mechanism indicating the force required to extrude filament. The extrusion force is applied by a pinch roller, shown in (b).



(a) Detail of FDM-produced part with adequate extrusion. No gaps exist between layers and deposition of material is constant.



(b) Detail of FDM-produced part showing insufficient extrusion. Material deposition varies among and inside layers, creating holes.

Figure 1.4 : FDM part affected by insufficient material deposition.

1.2 Goals and outline of this thesis

In order to reduce the effects of slip, the purpose of this thesis will be:

- 1. Finding an accurate numerical model that describes extruder input-output behaviour and incorporates slip.
- 2. The synthesis of a multi-objective extrusion controller for a Leapfrog FDM machine extruder, which maximizes velocity while retaining high object quality.

The controller will aim to maximize feed velocity and minimize slip at the same time. These objectives will push the print speed to its maximum. Since the current maximum is approximately 150 mm/s, it is desired to significantly increase the attainable no-slip velocity to at least 200 mm/s.

Chapter 2 will provide the reader with the necessary terminology and background concerning 3D printing. In chapter 3, existing models for FDM processes will be explored, and applied to a Leapfrog extruder using system identification methods. Additionally, the model will be expanded to describe slip conditions in the extrusion mechanism. After identification of system parameters and acquiring an accurate model, this model will be used to synthesize a controller which will minimize slip while maximizing feed velocity in chapter 4. Chapter 5 will discuss the physical implementation of said controller and evaluate its performance. Chapter 6 and 7 will discuss the results and draw subsequent conclusions.

Chapter 2

An introduction to additive manufacturing

Before diving into the world of 3D printing, extrusion and slip minimization, it is necessary to develop a basic understanding of additive manufacturing concepts, available AM technologies and the process of making a 3D printed object. This background knowledge is essential to fully grasp the concepts dealt with in this thesis. For this reason, this section will introduce the reader to the world of additive manufacturing. First, additive manufacturing technologies in general will be discussed, after which we will zoom in on FDM.

2.1 AM technologies

Let us first review the general working principles of additive manufacturing, as well as the different AM technologies that are most widely used. Additive manufacturing relies upon the layer-by-layer construction of 3D objects based on 3D model data. Since material is added instead of removed (as in traditional manufacturing, subtractive manufacturing methods), a high degree of form freedom exists in AM, allowing for shapes that cannot be made subtractively. Second, additive manufacturing processes produce less waste material compared to subtractive processes. Third, costs and production time for small series and prototypes are much lower. The layer-by-layer fabrication can be done by different processes, ranging from laser-based processes to extrusion. Figure 2.1 presents an overview of the most ubiquitous AM technologies.

| CATEGORIES | TECHNOLOGIES | PRINTED "INK" | POWER SOURCE | STRENGTHS / DOWNSIDES | |
|-------------------------------|----------------------------------------------------------------------------|-------------------------------------------------------------------------|-----------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------|
| Material Extrusion | Fused Deposition Modeling (FDM) | Thermoplastics, Ceramic slurries, The Metal pastes | Thermal Energy | Inexpensive extrusion machine Multi-material printing | |
| | Contour Crafting | | | Construction Poor surface finish | |
| | Selective Laser Sintering (SLS) | Polyamides /Polymer | | High Accuracy and Details | |
| | Direct Metal Laser Sintering (DMLS) | Atomized metal powder (17-4 PH | Atomized metal powder (17-4 PH | High-powered Laser Beam | Fully dense parts High specific strength & stiffness |
| Powder Bed Fusion | Selective Laser Melting (SLM) | stainless steel, cobalt chromium, titanium Ti6Al- | | Powder handling & recycling Support and anchor structure | |
| | Electron Beam Melting (EBM) | 4V), ceramic powder | Electron Beam | Fully dense parts High specific strength and stiffness | |
| Vat Photopolymerization | Stereolithography (SLA) | Photopolymer, Ceramics (alumina, zirconia, PZT) | Ultraviolet Laser | High building speed Good part resolution Overcuring, scanned line shape High cost for supplies and materials | |
| Material Jetting | Polyjet / Inkjet Printing | Photopolymer, Wax | Thermal Energy / Photocuring | Multi-material printing High surface finish Low-strength material | |
| Binder Jetting | Indirect Inkjet Printing (Binder 3DP) | Polymer Powder (Plaster, Resin), Ceramic powder, Metal powder | Thermal Energy | Full-color objects printing Require infiltration during post- processing Wide material selection High porosites on finished parts | |
| Sheet Lamination | Laminated Object Manufacturing (LOM) | Plastic Film, Metallic Sheet, Ceramic Tape | Laser Beam | High surface finish Low material, machine, process cost Decubing issues | |
| Directed Energy Deposition | Laser Engineered Net Shaping (LENS) Electronic Beam Welding (EBW) | Molten metal powder | Laser Beam | Repair of damaged / worn parts Functionally graded material printing Require post-processing machine | |

Figure 2.1 : Overview of AM technologies (Gao et al., 2015).

The technologies mentioned above differ greatly from each other in working principle, cost, accuracy and materials. They will be shortly explained below.

Powder bed fusion relies upon the solidification of raw powder materials. In this process, a roller spreads out a very thin layer of powder, which is usually a metal. Then, a high intensity laser or electron beam is trained across this flat pellet surface to either sinter the pellets together (resulting in a porous structure) or fully melting and solidifying them (resulting in a fully dense, metal part). Then, the build platform is lowered, a new layer of powder is applied and the process is repeated. These technologies are generally very expensive to apply, though able to achieve high object quality and strength, as a result of both the possibility to print metals, as well as the high layer accuracy attainable.

An example of powder bed fusion is selective laser sintering or SLS (fig 2.2), in which a roller applies layer after layer of metal pellets. A high-intensity laser is then trained across the surface, fusing the pellets together. High strength objects can be obtained by this method, though through sintering the pellets a porous structure is obtained. Power bed fusion technologies are the most ubiquitous AM technique in high-quality prototyping.

2.1. AM TECHNOLOGIES



Figure 2.2 : Schematic representation of selective laser sintering.

Vat photopolymerization concerns the photo curing of resins and ceramics. In this application, a light source, which is usually an ultraviolet (UV) laser, is used to solidify a photosensitive liquid. This type of process is generally very fast, but also one of the most expensive ones. In a very recent discovery that has been given a large amount of media attention, the object is pulled out of a resin bath while using an oxygen-permeable membrane to project the curing light, obtaining print speeds of hundreds of millimetres per hour. This method is called Continuous Liquid Interface Production (CLIP) (Tumbleston et al., 2015).

The first AM technology ever used was a type of vat photopolymerization, being stereolithography. It was invented in 1986 (Hull, 1986) and works by curing layers from a resin bath onto a build platform that is lowered (fig. 2.3). Though high accuracy is achievable, this method limits the build material to one at a time.



Figure 2.3 : Schematic representation of stereolithography.

Inkjet 3D printing Inkjet 3D printing is a binder jetting technology very similar to, for instance, SLS. Tiny powder pellets are drawn from a feed system and spread evenly across a build surface. After depositing a new layer of pellets, an inkjet print head trains across the layer and selectively binds the powder using CYMK binder ink. This technology is used mainly for aesthetic purposes since part strength is lower compared to other techniques. Additionally, this technique allows for full color printing of parts due to the use of CYMK ink.



Figure 2.4 : Schematic representation of inkjet 3D printing.

Fused Deposition Modelling (FDM) or Fused Filament Fabrication (FFF) is the most common AM process, though many other AM processes exist. In this process, a plastic filament is molten in a heated liquefier and then deposited on a build stage, where the print head or build stage moves vertically after each layer. The plastic filaments used are non-toxic plastic which exist is a large number of varieties, of which poly-lactic acid (PLA) and acrylonitrile butadiene styrene (ABS) are the most common. These filaments are loaded onto spools, of which they are fed into the printing machine. This machine uses stepper motors as feed drives to feed the solid filament into the liquefier. In most desktop FDM machines, resistive electrical heating combined with a PID control loop is used to stabilize the liquefier temperature at a desired melting temperature. The extruder head is then moved as filament is extruded at a constant rate, depositing a uniform layer. A representation of FDM is shown in fig. 2.5.



Figure 2.5 : Extensive schematic representation of FDM process (Turner et al., 2014).

2.2 FDM in-depth

FDM is the most common technique in desktop, consumer and small business 3D printers due to its plug-and-play character. No hazardous materials are in play, print materials are switched easily and

2.2. FDM IN-DEPTH

relatively large prints can be made relatively cheap. The main disadvantages of FDM are surface finish due to the layered character of the technology, and the limitation of materials to thermoplastics (Pham and Gault, 1998). Due to the ubiquitousness of FDM, along with the fact that Leapfrog 3D Printers produce FDM printers only, the scope of this thesis will be limited to FDM printers.

Regarding mechatronics, the most common desktop FDM printers have feed-forward controlled motion systems. Stepper motors are used to accurately position the extruder head and move the build platform. Horizontal motion is realized using timing belts, while vertical motion occurs through spindle gears. Typical positioning accuracy is $\approx 20\mu$ m. The feed-forward character of the printers means no data is available to the machine regarding its actual current position, and the machines are thus sensitive to external disturbances and vibrations. Central to the machine is an Atmel-based micro controller board which controls all peripheral motors and sensors. Additionally, an additional on-board computer may be present to act as a user interface.

Thus far, we have discussed the hardware side of 3D printing. Software is equally important however, since 3D printed objects are all generated from a machine code file (G-code), which is produced by a computer. This process is known as *slicing* and concerns the conversion of a graphical 3D model to lines of machine input. The slicing process may be executed by a wide variety of commercial and free-ware software packages. Using this input file, a real-time controller in the printer grabs line after line of code and converts it into print actions. For FDM, this translates into moving the print head or extruding liquefied filament.

Leapfrog only uses FDM, since this technology is easiest to apply in desktop 3D printers. Let us now consider the process of making a 3D printed object.



Figure 2.6 : The general 3D printing workflow (image courtesy of Leapfrog).

The first step in any design process is the same: formulation of an idea. This idea can be any shape when it comes to 3D printing, often with a build size limited to a few dozen centimeters in each direction. Subsequently, this idea must be digitalized in a computed using CAD software. Making a 3D model of the part is critical, since every 3D printer needs this as the input for its routine. To create a 3D model, any CAD software can be used, from professional grade programs like SolidWorks or AutoDesk to free applications like Google SketchUp, which is gaining ground with 3D printing enthusiasts.

After creation of the 3D model, this must be converted to a .STL file. STL files are the standard input format for 3D printers, as well as CNC machines and laser cutters. It is from this file that later the machine code is generated to control the printer. STL files describe a 3D volume by a large

number of triangles, i.e. convert it to a set of faces instead of a single volume (meshing). A larger number of polygons can of course describe the volume in higher detail and is therefore desired for a high quality print (fig. 2.7). There is a drawback to very fine mesh settings however, as infinitely small lines tend to cause errors in the machine code calculations. Therefore, mesh size should be chosen to allow for the desired accuracy of a part, but not much finer than that.



Figure 2.7 : STL files using different mesh settings. The right model has the finest mesh, but can generate errors in the machine code. (image courtesy of Leapfrog)

After meshing the part, a material must be selected for the 3D printing process. In essence, this can be any thermoplastic material, or any photosensitive liquid, depending on the procedure used. An (far from complete) overview of materials can be found in section 2.3.

After selection of the material, the last and most important software operation takes place: *slicing* of the part. Slicing converts the STL file, containing the collection of polygons defining the part, to lines of machine language (G-code) which is the direct input for the 3D printer (fig. 2.8). G-code is a universal, open-source machine language used by virtually all 3D printers and CNC machines and dictates the exact path of the machine (refer to appendix A for an overview of G-codes applicable in FDM). The generation of G-code is a complicated process executed by advanced software such as Simplify3D or Materialise. From a user's perspective, this software can be used to tweak print settings like material configuration, layer height, support strategy, print speed, infill, and a host of other variables. These settings influence the way in which the G-code is eventually generated.

2.2. FDM IN-DEPTH



plify3D slicing software.



Figure 2.8 : The slicing process consists of a 3D preview, the ability to tweak a vast amount of print settings and finally a G-code output.



Figure 2.9 : Explanation of G-code syntax. This line moves (G1) the print head to Cartesian coordinates x, y, z while extruding 3 mm of filament at a velocity of 3000 mm/min.

When the G-code has been generated, everything is ready to produce a 3D-printed part. The G-code can be uploaded into the 3D printer, which will execute the G-code and print the part. Until now, we have only discussed the software side of the 3D print, but the machine converting the digital bits into physical plastic is just as important. For this purpose, we will take Leapfrog's prime printer, the Creatr HS, to illustrate the functionality of 3D printers in general. The workflow of a 3D printer is depicted in figure 2.10, where the printer and it's components are shown in fig. 2.11.



Figure 2.10 : 3D printing technical workflow (image courtesy of Leapfrog).



Figure 2.11 : Schematic of 3D printer. Source: Leapfrog Creatr HS manual.

The printer processes the uploaded G-code by storing it on it's internal G-code server, which is in this case an Olimex board. This board or PC stores the G-code, and runs the Linux system with which a user interfaces. The Linux system sends the G-code line commands to the controller board. In case of Leapfrog printers, this board is called the Leapfrog Motor Controller or LMC, and is a custom board based on the Arduino Mega. It holds an ATMega 2560 chip and converts the G-code line commands into signals that are sent to the motors and other actuators. This LMC also contains the path planner, which optimizes motor speed profiles to obtain an optimal print. The LMC sends commands to the printer's sensing and actuation hardware, making the print.

In terms of hardware, thermal control of the hot end is performed by resistive heating elements combined with thermistors which serve as temperature sensors. The printer head and filament feed

2.3. MATERIALS

is actuated by stepper motors. Motion in the XY-plane is obtained by belt transmission, while Z-movement is achieved by spindle drives. Inherently to the usage of stepper motors, most FDM printers are controlled feed-forward, so no feedback information on the position of the print head is available to the controller.

2.3 Materials

In the previous section, FDM printing was explained in detail, stating that a wide range of thermoplastic polymers is suitable for FDM. Throughout the industry, six types of filament are widely used, which are discussed below to give a complete picture of the FDM process.

Polyactic Acid (PLA) PLA is the most common printing material, known for its smooth finish. PLA's most important characteristic is its relatively small shrinkage when cooling down. Due to this, it is insensitive to temperature gradients across a part and can be used to print large parts without the layers separating. Second, it is soluble in sodium hydroxide, which makes it a viable support material for ABS prints. Third, its relatively low melting temperature allows for fast printing and high accuracy in small details.

Acrylonitrile butadiene styrene (ABS) ABS is more sensitive to temperature changes than PLA, making it less suited for large parts. It is, however, a stronger material, making it perfect for fine, small parts. Its higher strength also allows for after-printing surface treatment procedures such as sanding. It is soluble in acetone, meaning after treatment using acetone vapor is possible.

Compound materials Regular PLA and ABS filaments may be compounded with a variety of exotic material fibres to create hybrid properties. For instance, PLA can be compounded with wood, glass or carbon fibres of mimic the structure of these materials and obtain some of its properties. For instance, wood-compounded materials appear like actual wood and can be sanded after printing, where carbon compounded filaments are very light and strong. For electrical conductivity or high specific mass of the object, the material may be compounded with metal grains.

Polyamide (nylon) Nylon is mainly used due to its cost-effectiveness, selling for around half the price of other mainstream filaments. Second, it is less brittle compared to PLA or ABS, making more durable parts due to their reduces hardness and increased flexibility. A main issue with nylon is the tendency to absorb moisture from the surrounding air, meaning the print material degrades in quality over time. Hence, it must be heated for a prolonged time to evaporate the water before usage.

Polyvinyl alcohol (PVA) PVA is a water-soluble plastic, which makes it an ideal support material. Breaking away PLA or ABS supports can damage the surface, or sometimes support material is printed in hard-to-reach areas. In such a situation, soluble support materials can be used for easy support removal leaving no surface marks.

High Impact Polystyrene (HIPS) HIPS is mostly used as a support material for ABS printing since it has about the same physical properties but works with different solvents. It can be solved in d-limonene, making it easy to remove from a print without leaving any surface traces on the final object.

Chapter 3

System identification and modelling

This chapter discusses modelling and identification of the extrusion system. Understanding the mechanics of the extrusion mechanism is critical to successful problem analysis and controller development. Therefore, the first section will explore the mechanics behind filament extrusion and identify the main principles involved. It will also explore existing models describing the FDM process, which currently only describe a very specific part of the extrusion process (Bikas et al., 2015) or disregard the slip phenomenon. Next, a model will be developed to describe the extrusion process in an FDM machine. This model will be the basis for designing an anti-slip controller in the next chapter.

Let us first introduce the printer which we will use to conduct our analysis. As a printer model, the Leapfrog Xeed has been selected. This is Leapfrog's most high-end model intended for professional applications. Hence, it is also the most likely candidate for a feedback system to be applied in which was a minor motivation for selecting this printer. Selection of the Xeed as the experiment setup for this investigation is caused mainly by the drive system used for this type of 3D printer. 3D printers utilize two types of drive systems (Griffey, 2014) for feeding filament into the extruder:

1. Bowden tube systems have a filament drive unit outside the print head, attached to a stationary part of the machine frame. Here, a stepper motor latches onto the filament using a pinch wheel and applies pressure to the filament. In order to get the filament to the extruder, a polymer low-friction tube guides the filament to the moving print head. Thus, no actuation occurs inside the head itself, which is only used for heating the build material and depositing it in the correct location. Moving the actuation system of the extruder outside of the head greatly reduces head mass, since the required stepper motors and gearing are heavy in order to produce enough (hold) torque. Reduced head mass allows for much greater head motion and acceleration using the same actuation system, and thus faster printing. As a major disadvantage, the filament strand between extrusion mechanism and nozzle acts as a viscous element with significant delay in the control chain. Each control action on the drive unit is delayed once it arrives at the extruder, making the use of feedback-based control systems in slip control inadvisable. Second, Bowden tube systems are not compatible with all build materials, since highly elastic or brittle materials tend to fracture or curl up somewhere between the extruder and the drive mechanism due to the counter pressure generated in the extruder.

2. Direct drive systems incorporate the actuation system directly into the print head. Hence, the print head is heavier compared to Bowden systems since motors and gearing have to be actuated as well. This means that direct drive machines are capable of lower speeds and accelerations compared to Bowden system machines. Placement of the drive system inside the print head has quite some advantages, the main one being the fact that extrusion occurs virtually without delay. Given the short distance between drive and nozzle, viscous effects have less influence on the behaviour of the extruder. Thus, slip control is more viable in direct drive systems. Second, the lower distance between drive and extruder leads to a wider material range, since brittle or elastic materials can no longer show unwanted behaviour.



Figure 3.1 : Comparison of direct drive systems (left) versus Bowden tube systems (right). In direct drive systems, the filament is enclosed rigidly between the last motor and the extruder. For Bowden tubes, a flexible PTFE tube guides the filament to the head. The print head consists of a tube, a nozzle, and a heating block used for transferring heat from a resistive heater, as well as holding a thermistor for temperature control.

Due to the large delays occurring in Bowden tube systems, accurate slip control becomes virtually impossible using feedback mechanisms. The Leapfrog Xeed was thus selected for this investigation since it incorporates a direct drive system, which allows for slip control more easily.



Figure 3.2 : Two representations of the Leapfrog Xeed, real (a) and schematic containing the most important components (b).

| Property | Value |
|--------------------------|-----------------------|
| Build volume (L x W x H) | 220 x 250 x 230 mm |
| Nozzle diameter | 0.35 mm |
| Layer thickness | 0.01-0.35 mm |
| XY positioning motors | NEMA 34 stepper motor |
| Z positioning motors | NEMA 17 stepper motor |
| Extrusion motors | NEMA 17 stepper motor |
| XY positioning accuracy | $18 \ \mu \mathrm{m}$ |
| Z positioning accuracy | $10 \ \mu \mathrm{m}$ |

Table 3.1 : Leapfrog Xeed main properties.

3.1 System description

In FDM machines, a pinch roller mechanism is used to push thermoplastic filament into the extruder. The basic premise of the mechanism is that it aims to deliver a constant volumetric flow rate Q, since this yields an equal material distribution across one build layer. Stepper motors are mainly used to impose a constant angular velocity on a toothed pinch wheel. The system for consideration is a 1-DOF filament extrusion system as depicted in fig. 3.3 (recall also fig. 2.5). An actuated pinch wheel is pressed against a passive wheel to which a normal force F_{pinch} is applied, usually by pre-tensioning a spring using a set screw. The feed material is thus clamped between the actuated and passive wheels. The filament clamping force can then be expressed as

$$F_{pinch} = (l_0 - x) \cdot k \tag{3.1}$$

in which l_0 is the free length, x the actual length, and k the stiffness of the spring.

Assuming no slip, the feed velocity can be expressed as

$$v = \omega R \tag{3.2}$$

where ω is the angular velocity and R is the radius of the pinch roller. As mentioned, this feed velocity v must be kept as constant as possible to ensure a uniform volumetric flow rate, expressed by

$$\phi = v \frac{\pi}{4} D_{nozzle}^2 \tag{3.3}$$

where D_{nozzle} depicts the nozzle diameter of the extruder. If slip occurs, the required extrusion force exceeds the maximally attainable friction force (Agarwala et al., 1996), computed by

$$F_f = \mu_k F_{pinch} \tag{3.4}$$

Where μ_k is the friction coefficient between pinch roller and filament strand, determined by material properties and manufacturing method of the pinch roller. Given the large influence of pinch wheel parameters on extrusion performance, all experiments later in this thesis were conducted with the same pinch roller.



Figure 3.3 : Schematic of pinch wheels extruding filaments.

The slip between pinch wheel and filament is defined as

$$\alpha = \frac{\omega r - v}{\omega r} \tag{3.5}$$

This slip only occurs when driving the filament at high feed velocity. This results directly from the pressure drop ΔP across the extruder, which increases with feed velocity. As ΔP increases, so does the required extrusion force. As this force exceeds the maximum friction of the extruder wheels, slip occurs and the filament stops as the wheels continue turning. This process can be illustrated if we slowly increase the speed of an FDM extrusion process and measure the output velocity of the filament strand. Fig. 3.4 depicts a velocity input executing a sawtooth with a maximum velocity of 24 mm³/s. It can be seen that for low velocities, no slip can be assumed and the system behaves as a lower-order transfer function. For high speeds, non-linear stick-slip phenomena obtain a prominent role in the system response. A detail of this behaviour is shown in figure 3.5. It is this process that we must map before being able to synthesize an adequate anti-slip controller.

20



Figure 3.4 : Velocity input profile showing slip behaviour of the extrusion mechanism in one experiment run. Note how for low speeds, slip may be disregarded. At speeds > $15 \text{ mm}^3/\text{s}$, significant slip occurs.



Figure 3.5 : Detail of velocity input profile showing slip behaviour of the extrusion mechanism in one experiment run, for one single up- and down ramp. Note how for low speeds, slip may be disregarded. At speeds > $15 \text{ mm}^3/\text{s}$, significant slip occurs.

To better understand system behaviour, we must repeat our observations multiple times. The output of both a velocity sawtooth and a succession of step inputs was observed to distil the average system response. As an added advantage, averaging data not only makes our model fit more

representative, but also filters out noise. Fig. 3.6 depicts the average and 95% confidence interval for 10 sawtooth inputs, while figure 3.7 shows the same for 10 increasing step inputs. The 95% confidence interval was computed assuming normally distributed data, which was verified using Lilliefors testing (Yap, 2011).



Figure 3.6: Velocity input profile showing slip behaviour of the extrusion mechanism for 10 repetitions. The 95% confidence interval is shown in black. We observe comparable behaviour between multiple data sets.


Averaged test results for 10 runs

Figure 3.7: Steps input profile showing slip behaviour of the extrusion mechanism for 10 repetitions. The 95% confidence interval is shown in black. We observe comparable behaviour between multiple data sets.

From both figures, we can conclude that the output is consistent in terms of magnitude and that slip occurs at roughly the same point every repetition.

Using the averaged data obtained from these experiments, we can start developing a model fit for the IO data. Let us first review existing models regarding the dynamics of the extruder. Exploring scientific literature, we conclude that literature on modelling FDM extrusion is very scarce. Some models exist that treat specifically the mechanical properties of created parts or focus on heat transfer in the hot end. In fact, only one paper describes the input-output behaviour of an FDM extruder in-depth (Bikas et al., 2015). It is this model that will form the basis for the extended model in this thesis.

Bellini et al. (2015) investigated the feasibility of developing a model describing the relation between input feed velocity and deposited material velocity. They first developed an analytical model, in which the individual physical processes comprising the entire extrusion process were considered. Though this type of model provides clear insight in the mechanics of extrusion, Bellini et al. found that its implementation for development and controller synthesis is not recommendable. Instead of attempting an analytical approach, a transfer function approach to characterize extrusion dynamics proved to be much more successful:

"The good prediction of the real flow, obtained when applying the transfer function approach, suggests that the integration of this function as part of an overall control strategy will provide an effective flow control and consequently will improve part integrity. [...] The mathematical model, which gives a physical explanation to the process taking place in the liquefier, demonstrated nevertheless to be of difficult implementation for further development. "

(Bellini and Bertoldi, 2015)

To build this generalized transfer function, an analog electric circuit was proposed to represent the dynamics of the extruder (fig. 3.8). The reason for selecting an analog circuit to represent the extruder is to obtain better insight in the dynamics, and to be able to propose a transfer function since we know the dynamics of an analog circuit, as opposed to an FDM liquefier. Comparing filament flow to current, the feed flow rate can be represented as the applied current i at A, where R represents the (constant) slip condition between the rollers and the filament. Material properties and heat flux in the extruder are represented by the inductance L. Inductance is a good parallel since it can be seen as an opposition to sudden changes in the flow rate, which is exactly the same as an electrical inductance.



Figure 3.8 : Analog electrical circuit representing extruder dynamics. (Bellini and Bertoldi, 2015)

The (first-order) response of the extruder, with I the observed output flow rate (or current, in this case) may then be written as

$$\begin{cases} L\frac{di(t-\tau)}{dt} + Ri(t-\tau) &= u(t-\tau) \\ y(t) &= I(t) \\ i(t-\tau) &= I(t) \end{cases}$$
(3.6)

After some mathematics, this yields the transfer function between input and output to be

$$G(s) = \frac{Y(s)}{U(s)} = \frac{\mu}{Ts+1}e^{-\tau s}$$
(3.7)

24

which is a first-order transfer function with time constant τ , describing the delay between changes in reference compared to change in the observed output. Other research, not considering thermal extrusion but ceramic extrusion, found a first-order model of the same form (Mason et al., 2008). This modelling as a first-order equation limits its usefulness since higher-order effects are lost. This thesis will address the transfer function model in two ways:

- 1. Since most physical meaning is lost in the transfer function modelling approach, this thesis will further expand this method by considering the extruder purely as a black box system: it's physics will be disregarded, but using system identification procedures an accurate model describing input-output behaviour will be obtained. For this, input-output data with low velocity will be used to minimize slip. This will be sufficient to model the behaviour in fig. 3.4 up to $\sim 15 \text{ mm}^3/\text{s}.$
- 2. Bellini and Bertoldi assumed a constant slip between filament and rollers, thereby disregarding the highly non-linear stick-slip behaviour that we observe in the high-velocity regions of fig 3.4. Hence, the identified black-box model will be expanded with a friction model to include stick-slip behaviour at high velocities.

3.2 Slip-free system dynamics

We wish to expand the transfer model of eq. (3.7) and apply it to a Leapfrog extruder. When this is achieved, a model is obtained that adequately describes the dynamics of the extruder when no slip is considered. After this, slip conditions will be implemented as well. The first step is choosing a system identification framework to obtain the no-slip transfer.

3.2.1 Identification framework

System identification in general is the art of distilling a mathematical model from input-output data from a dynamical system. A first step in selection of an identification algorithm is to distinguish in the amount of knowledge prior to modelling. This amount of prior knowledge may be 'color-coded' to describe the type of model obtained from the identification. *White-box models* can be completely constructed from physical insight and previous observations. *Grey-box models* require partial determination of parameters while others can be obtained through physical insight, whereas *black-box models* use no physical insight, but utilize a flexible model structure that fits the IO data well (Sjöberg et al., 1995). The transfer function approach of this thesis originates from a black-box model, which means we will limit our analysis to black-box identification models. Black-box identification methods can be subdivided into two main categories, being maximum likelihood (ML) methods and instrumental variable methods (IVM). ML methods s are essentially independent of system representation. For both classes, a number of algorithms has been shown to perform well on both simulated and real data. Extension of IVMs to state space or transfer function form is much less obvious (Vibergt, 1995), whereas ML methods are easily adapted to state-space models. Hence, an ML is the preferred method for system identification.

As an alternative category in identification methods, one may consider subspace-based system identification (SID) methods such as 4SID, CVA and MOESP. SID methods have smaller computational complexity compared to ML, and are also known for higher numerical stability

(Vibergt, 1995). The previous statements are true mostly for system orders > 10. Subspace identification is mostly advantageous for MIMO systems and, as a downside, is known to have less accuracy compared to ML methods in general (Qin, 2006). Since 3D printer system identification is expected to have a low order, the system is SISO and we require a high accuracy, the advantages of SID do no outweigh the advantages of ML.

Prediction error (PE) methods are closely related to ML methods and offer expanded functionality. Common examples for PE methods are ARX, ARMAX, Box-Jenkins and output-error (OE) methods (Verhaegen and Verdult, 2007). An ARMAX (Autoregressive Moving Average Model with eXogenous inputs) is the most ubiquitous prediction error model, which is very closely related to ML (Aström, 1980), and was used to obtain a black-box transfer function of the extruder. The choice for this system identification methods follows from the notions that ARMAX can handle time domain input-output data, which is much more convenient compared to using frequency domain data, and the possibility for filtering out certain types of noise in the IO data. ARMAX attempts to fit a model to empirically determined input-output data, assuming the model can be written as

$$y(k) = \frac{b_1 q^{-1} + \dots + b_n q^{-n}}{1 + a_1 q^{-1} + \dots + a_n q^{-n}} u(k) + \frac{c_1 q^{-1} + \dots + c_n q^{-n}}{1 + a_1 q^{-1} + \dots + a_n q^{-n}} e(k)$$
(3.8)

Which, using the shift operator q, can be written more compactly as

$$y(k) = \frac{B(q)}{A(q)}u(k) + \frac{C(q)}{A(q)}e(k)$$
(3.9)

Here, e(k) is zero-mean white noise independent from the input u(k) and the numerator and denominator coefficients a_i , b_i and c_i . n denotes the order of the polynomials A(q), B(q) and C(q). Once this order is known, an estimation problem can be defined to estimate the polynomial coefficients in such a way that the fitted model corresponds to a finite set of input-output measurements.

Using said input-output data, the fitted polynomials can be used to convert the ARMAX model to a SISO state-space system, in order to further analyse the system dynamics. Writing (3.8) in a state-space form, we obtain

$$x(k+1) = \begin{bmatrix} -a_1 & 1 & 0 & \cdots & 0\\ -a_2 & 0 & 1 & \cdots & 0\\ \vdots & & \ddots & \vdots\\ a_{n-1} & 0 & 1 & \cdots & 0\\ a_n & 0 & 1 & \cdots & 0 \end{bmatrix} x(k) + \begin{bmatrix} b_1\\ b_2 \vdots\\ b_{n-1} \end{bmatrix} u(k) + \begin{bmatrix} c_1 - a_1\\ c_2 - a_2\\ \vdots\\ c_{n-1} - a_{n-1}\\ c_n - a_n \end{bmatrix} e(k)$$
(3.10)

$$y(k) = \begin{vmatrix} 1 & 0 & 0 & \cdots & 0 \end{vmatrix} x(k) + e(k)$$
(3.11)

These equations form our identified system model without slip occurring between the rollers and the filament. The next section will discuss how this ARMAX algorithm was applied to the extruder system.

3.2.2 ARMAX implementation and system parameters

To execute an accurate system identification and determine parameters of the extruder system, an input signal should be selected that contains a broad spectrum of frequencies, and ideally all of them. A Heaviside step function contains all input frequencies and will thus be selected for our purpose using a feed velocity of 5.4 mm/s. The choice for this low feed velocity originates from the fact that at this low speed, slip may be neglected and we will purely obtain the transfer function from velocity input to velocity output assuming no slip. The step input will be applied to the stepper motor to execute a constant angular velocity at t = 1 s. Assuming ideal synchronism in the stepper motor (which is justified as the required extrusion torque at this speed is well below peak torque, the input-output relation may be determined by observing the output in the filament feed velocity. This velocity is recorded using an incremental optical encoder with a resolution of 2000 pulses per revolution, sampled with a frequency of 350 Hz. When the system is driven with above input, the open-loop velocity output has been recorded and shown in fig. 3.9.



Figure 3.9 : IO data used for ARMAX system identification. Blue line: set point. Red line: measured output.

A few preliminary observations can be made from the open-loop extruder response, mainly the notions that the response is very quick (which is to be expected) and very noisy (which is inherent to differentiating discrete encoder signals). This noise is compensated for by our ARMAX algorithm by modelling an extra noise term e(k) under the assumption that our noise is zero-mean Gaussian.

Having obtained our input-output data set, it was attempted to synthesize an ARMAX model relating the two. In application of the model, in important parameter to first determine is the order of the model. A higher-order model will better grasp the higher-order dynamics of the system, but also requires more effort to fit. Second, a higher order means that the model will also try to fit on the noise in the output signal, which is undesired. An adequate method for determining model order is by observing the value of the prediction error cost function (Verhaegen and Verdult, 2007):

$$J_N(\theta) = \frac{1}{N-m} \sum_{k=m}^{N-1} (y(k) - \hat{y}(k,\theta))^2$$
(3.12)

with N the number of samples, m the model order, y(k) our observed output and $\hat{y}(k,\theta)$. By plotting the value of $J_N(\theta_{ARX})$, we may observe the cost function approaching an optimal value, which corresponds to the standard deviation of e(k). Theoretically, the value of this cost function is monotonically decreasing and gives an impression of the model order to choose: once the value as approached the optimum, further increasing the model order will only yield higher computational effort and noise fitting. Given our input-output data, the value of $J_N(\theta_{ARX})$ is plotted against model order in fig. 3.10.



Figure 3.10 : Determination of ARMAX model order.

The course of $J_N(\theta_{ARX})$ shows at first that the prediction error cost function is *not* monotonically decreasing as theoretically implied, but fluctuates. The trend however, is decreasing. Given the small scale on the *y*-axis, we can observe that increasing the model order from 2 to 18 only reduces the cost function value by ~ 10%. To ensure low model complexity and avoid noise fitting, a model of order 1 is selected, as this visually yielded the best fit to the IO data and also corresponds to the recommendations of Bellini and Bertoldi.

The obtained first-order discrete model was put into state-space form and used to simulate a step response using our input signal. The resulting discrete transfer function is

$$G(z) = \frac{0.004649z^{-1}}{1 - 0.9952z^{-1}}$$
(3.13)

The simulated model output is depicted in fig. 3.11. The model seems to have a very small time delay and a small steady-state error, which both correspond to the properties τ and R which Bellini and Bertoldi have found in their first-order model. Second, we can observe that no noise fitting

3.2. SLIP-FREE SYSTEM DYNAMICS

occurs and that the model output nicely matches the physical system output. Hence, we have successfully developed a model for low-speed, no-slip extrusion using the ARMAX framework.



Figure 3.11 : Simulated model step response with fitted ARMAX model to experimental output data. Though the time constant appears to be large for this particular data set, bear in mind that the model was obtained based on a best fit for the *average* response of the extruder.

To illustrate that the validity of this model is limited to low-velocity extrusion, it has been applied to a series of increasing step inputs as well as the velocity sawtooth we used earlier. The measured output feed rate was compared to that of the model, as depicted in figures. 3.12 and 3.13.



Figure 3.12 : Simulated no-slip model step response for higher feed velocities, applied to single data set.



Figure 3.13 : Simulated no-slip model step response for input velocity sawtooth, applied to single data set.

The model is visibly inaccurate for high feed velocities. At high feed velocities, we observe the same

phenomenon as in a velocity input sawtooth: higher velocity yields a (non-linearly) higher slip, which is not grasped but he current first-order model. Especially for rates > 15 mm³/s, slip becomes significant and requires a model expansion. Hence, the next task will be expanding the model to include the non-linear slip phenomena occurring in fig. 3.4 for feed velocities > 15 mm³/s.

3.3 Application of the Pacejka tyre slip model

The greatest challenge in modelling extrusion dynamics is adding slip conditions to the already expanded Bellini model. To achieve this, we must first determine the main characteristics of the slip behaviour.

3.3.1 Theoretical framework and reworking the formula

Recalling the slip as observed in fig. 1.3, this phenomenon firstly occurs in a stick-slip fashion. As soon as a threshold feed rate is reached, the filament feed velocity stops with a very high deceleration. As the velocity has dropped, the filament pinch wheels regain their grip on the filament strand and the velocity increases with very high acceleration. This process repeats, creating slip in a stick-slip pattern. The modelling of stick-slip behaviour is a highly complex field of study which itself poses a wide field of research (Kammer et al., 2015). Second, and more importantly, we observe that the output velocity drops at a certain input feed rate, and after that stick-slip occurs. To prevent filament slip and ensure good part quality, we wish to prevent slip altogether and never end up in the stick-slip domain as we prevent slip from happening and are controlling in a relatively low-slip area. Thus, due to the fact that describing the stick slip is not a necessity for developing a good anti-slip controller, implementation of stick-slip is considered to be beyond the scope of this thesis. Therefore, the slip will be considered to be continuous for modelling purposes. The second observation to be made regarding slip, is the fact that the degree of slip depends on the imposed feed velocity in a highly non-linear fashion. Based on this notion, a slip model will be selected since this

Hence, we need a suitable model to describe the non-linear slip occurring in the extrusion mechanism. A particularly suitable one for this case is the slip model as described by Pacejka. His "Magic Formula" (Pacejka, 1987) originates from the automotive industry. Following a back-box modelling approach, hardly based on an accurate physical interpretation, Pacejka developed a parametrized expression that could be made to fit a wide range of tyres and other rolling contacts. Due to the versatility of the formula, and the fact that it may be tuned to fit a large range of friction scenarios, this formula is suitable for our situation. The equation writes

$$Y = D\sin\left(C\tan^{-1}\left[B(1-E)X + \frac{E}{B}\tan^{-1}(BX)\right]\right)$$
 (3.14)

In this equation, the parameters B, C, D, E must be chosen as such that the formula matches the data at hand. Pacejka denotes the slip angle or slip percentage α in his model by X, so these symbols are equivalent. Y represents the longitudinal force on the tyre. The slip percentage α can be interpreted as a measure for how much the set velocity and actual velocity actually differ by $v_{act} = v_{set}(1 - \alpha)$. An example of a tyre slip curve as obtained by (3.14) is shown in fig. 3.14.



Figure 3.14 : Example type slip curve determined using the Pacejka slip formula. Parameters B, C, D, E determine the shape of various curves of the graph.

This figure shows us that as the force exceeds a certain threshold, it drops as the slip rapidly increases. As the slip has increased sufficiently, the force has diminished to such an extent that friction is regained and slip diminished. This hysteresis curve repeats and repeats. The next step in our process is to find a means to apply this slip formula to a filament extruder system. The original slip model is force-dependent, and since $F = m\ddot{x}$, mainly dependent on the acceleration of the tyre. However, in the extrusion mechanism, accelerations of the filament strand are not the main source of reaction forces on the strand causing slip. Since we observe that slip increases with feed velocity, we may state that the pressure drop across the extruder causes a reaction force on the filament with eventually introduces slip as the feed velocity (and thus extrusion pressure) becomes too large. According to Bellini (2015), who developed an analytical expression for the pressure drop across the extruder, the pressure drop relates to feed velocity. Second, the extrusion force has a strong linear correlation with extrusion mass flow and thus feed velocity (Mason et al., 2008):

$$\Delta P \propto v \tag{3.15}$$

Given this, and the fact that the extrusion force required to push the filament through the nozzle is given by $F_e = \Delta P \cdot A_{nozzle}$, we may state that

$$F \propto v$$
 (3.16)

So, we can conclude that in an extrusion system, the main determinant of force that induces slip is not acceleration, but extrusion speed. Therefore, we may rewrite (3.14) into

$$v = D\sin\left(C\tan^{-1}\left[B(1-E)\alpha + \frac{E}{B}\tan^{-1}(B\alpha)\right]\right)$$
(3.17)

which gives a direct non-linear relation between imposed feed velocity v and slip between pinch rollers and filament α . Next, we must find appropriate parameters for this expression in order to use the Pacejka formula in our extruder model.

3.3.2 Implementation

Determination of parameters for (3.17) is a non-linear optimization problem. In order to find parameters for our model, we must find an optimization framework that identifies optimal values for B, C, D, E in such a way the error between the model fit and the experimental data of fig. 3.4. For a data set of N samples, this root-mean-squared error can be computed by (Chai and Draxler, 2014):

$$e_{RMS} = \sqrt{\frac{\sum_{k=1}^{N} (y(k) - \hat{y}(k))^2}{N}}$$
(3.18)

To identify parameters that minimize the RMSE, must thus use a non-linear, unconstrained optimization routine that finds the global minimum in e_{RMS} as a function of B, C, D, E. For optimization routines, we have considered brute force methods, steepest descent, sequential programming (either linear or quadratic) and second-order methods. The selection of optimization routine now depends on the shape of the objective function e_{RMS} . Since four variables imply an objective function in five-dimensional space, we must explore the objective function in two or three dimensions in order to determine its properties.

Two explorations will be performed, one exploring the behaviour of parameters B and C, and one of parameters B and E. To investigate the behaviour of the RMSE, its value was investigated by simulating the model for small variations in the two parameters and plotting the RMSE between experimental data and model output. The model consists of an input signal, which passes through the adapted Pacejka formula and thus produces an output velocity. Since the applied model utilizes a one-dimensional lookup operation to determine the slip coefficient appropriate for a certain input velocity, a model is only valid if the extrusion force is monotonically increasing for α :

$$\nabla F_e(\alpha) > 0 \qquad \qquad \forall \Delta \alpha > 0 \qquad (3.19)$$

If this constraint is violated, the model cannot perform the lookup. Therefore, a very high RMSE is automatically assigned to parameter combinations that yield non-monotonic slip functions. Figures 3.15 and 3.16 show the two-dimensional exploration of the objective function for two sets of parameters.



Figure 3.15 : Influence of Pacjeka model parameters B and C on achieved data fit. An arbitrary error of 2 was assigned when the parameters yielded an unusable model.



Figure 3.16 : Influence of Pacjeka model parameters B and E on achieved data fit. An arbitrary error of 2 was assigned when the parameters yielded an unusable model.

Figures 3.15 and 3.16 plot the value of the RMSE for various combinations of the parameters B, C, E. The observations that may be made from these two exploratory plots are the following:

- The objective function contains discontinuities and is therefore non-linear, i.e. for some x, y we have $a(x+y) \neq ax + ay$. The discontinuities are caused by the artificial assignment of high RMSE values to infeasible models.
- The objective function is highly non-convex, meaning that there may be feasible non-convex domains that have multiple local boundary optima for some parameters. Second, the Hessian is not positive (semi-)definite.
- The objective function is not monotonic. Hence, we may not assume a single global minimum.
- Generating all values of the objective function for all parameter combinations is highly computationally intensive. This rules out brute force search, and also means that no information regarding the Hessian of the objective function is available.

Given the observations given above, we may not use any methods that are sensitive to local optima, cannot use Hessian or steepest descent information and our behaviour to catch is very complex. Hence, the most promising approaches are genetic or swarm intelligence algorithms. Neural nets require many data sets for training, and swarm intelligence methods are sensitive to local optima. Given the high complexity of our problem, the unknown amount of optima, a genetic algorithm will be used since this can theoretically achieve Pareto optimality.

The Augmented Lagrangian Genetic Algorithm (ALGA) was chosen for this application, for multiple reasons This approach has been shown to provide solutions closer to the true optimum, reach this optimum more efficiently (Mahdavi and Ebrahim, 2015), and handle constraints more efficiently (Deb, 2000). Second, ALGAs have theoretical convergence properties and minimally distort the objective function (Deb and Srivastava, 2012). Globally, the ALGA functions as follows:

1. The algorithm begins by creating a random initial population. Each individual is in this case a combination of the parameters B, C, D, E. Coupled to the individual is a fitness value which is determined by the RMSE e_{RMS} belonging to that specific combination of parameters, along with a penalty function which penalizes violation of constraints. The fitness function F(x)used by the algorithm is given by

$$F(x) = e_{RMS} + \sum_{j=1}^{J} R_j |g_j(x)|^2$$
(3.20)

where R_j is the penalty parameter belonging to the j^{th} inequality constraint.

- 2. The algorithm creates a new population of individuals based on the current individuals. Just as in Darwinian evolutionism, the individuals with the highest fitness values are selected as 'parents' to facilitate the next generation. The next generation of individuals, or 'children', consists of three types of children:
 - Crossover children, where the properties of a pair of parents are combined. This makes the population converge towards the optimum.
 - Elite children, which are individuals with the highest fitness values. These individuals survive to the next generation.

• Mutation, where the properties of a single parent change randomly and pass on to an individual in the next generation.

This process is repeated until the fitness of a generation converges.

3. The algorithm terminates as the fitness function of a generation does not change significantly with respect to that of the previous generation. In such a case, the algorithm has converged.

Convergence of a genetic algorithm is determined by the balance in the three types of children. Increasing the degree of crossover generally increases the rate of convergence. Random mutations may move generations away from the current minimum, either finding a new (global) optimum or delaying convergence. The number of elite individuals may increase convergence speed, but a too high elite count introduces too few change in each generation, slowing it again. Hence, an optimum exists for each of the three types of children.

The standard MATLAB genetic algorithm ga was run to find a global minimum in the four parameters in the adapted Pacejka formula. After running this algorithm, an optimum was found as shown in table 3.2.

| Pacejka parameter | Value [-] |
|-------------------|-----------|
| В | 42.0250 |
| \mathbf{C} | 240.7860 |
| D | 27.9944 |
| Ε | 70.19403 |

Table 3.2 : Pacejka parameters determined using GA.

When these parameters are implemented in (3.17), a slip curve as in fig. 3.17 is obtained. It is this slip curve that will be built into the combined model capturing the full dynamics of the extrusion mechanism.



Figure 3.17: Slip curve for the feed rollers using the Magic Formula.

3.4 Combined model

In this section, the identified extruder and slip dynamics will be combined to form one comprehensive model that describes the output of the extruder. To achieve this, the models will be put in series. First, the slip model will take the input velocity v_{set} and determine the amount of slip appropriate for this velocity. The actual velocity at which the filament enters the extruder v_{act} is then used by the extruder dynamics transfer function to determine the output feed velocity v_{out} . This process is drawn in fig. 3.18. The simulated model response is shown in figs. 3.19 and 3.20.



Figure 3.18 : Complete model block scheme including both slip behaviour and extrusion dynamics.



Figure 3.19 : Simulated response of the combined model for a sawtooth velocity input, fit to 10 averaged data sets.



Figure 3.20 : Simulated response of the combined model for a sequence of step velocity inputs, fit to 10 averaged data sets. Note that the effect of slip is somewhat overestimated at lower velocities, which is inherent to the shape of the Magic Formula curve. Over the entire velocity range, this is an optimal fit.

When observing the model fit, we can see that good conformity between model and measurement data was obtained: even though the stick-slip phenomenon is ignored, the Pacejka formula describes the magnitude of the slip quite well. In low-slip conditions, the transfer function of the extruder dynamics accurately tracks the input and has very low error when compared to the measured data. Overall, the RMSE between model and data is 0.62.

In this section, we synthesized a complete model to describe the dynamics of an FDM extruder. By combining a transfer function and tyre slip model, an accurate representation of the measurement data was obtained. Using this model, an anti-slip controller will be designed in the next chapter.

Chapter 4

Controller synthesis

This chapter will describe the synthesis of a feedback controller for an FDM extrusion mechanism, using the black-box extruder model obtained in the previous chapter.

4.1 Algorithm

The controller for our purpose must fulfil two control objectives:

- 1. Minimize filament slip α ;
- 2. Maximize the filament volumetric flow rate ϕ .

In essence, we have one input (the reference velocity) which produces a single output (output velocity). From this output, we compute another state, being the velocity error and from this we compute the slip percentage. This poses a multiple-input multiple-output (MIMO) control problem: we wish to maximize one state while minimizing the other. Since both control objectives are directly contradictory, some means of adding a certain reward or cost to both objectives is required. By adequate determination of the rewards of costs assigned to the satisfaction of each objective, both objectives may be satisfied optimally. We will first shortly explore the control possibilities at hand, and select the most promising control approach.

Proportional-(integral-)derivative control The ubiquitous P(I)D control scheme is the primary choice for any control problem and is often good enough to achieve the control objective. It attempts to minimize the error between input and output e(t) by the feedback law for the input u(t):

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt}$$
(4.1)

Though used widely, P(I)D is not suitable for application in extruder slip control. PID is a SISO system control method, whereas our application poses a MIMO problem. In order to apply PID control to our system, one would get into trouble while determining an appropriate set point. Since we are controlling velocity, when measuring a larger output error, the controller would take more action and push the filament harder, which will only have counter productive effects. Hence, we have

to define not a velocity reference, but a desired slip reference and then we would need to find a translation from measured slip to velocity set point.

Full state feedback Full state feedback (FSF) works similarly to PID control, and is a control method in which all system states \boldsymbol{x} are used to determine the input signal u(t):

$$u(t) = -\mathbf{K}\mathbf{x} \tag{4.2}$$

FSF places the poles of the closed-loop plant at desired locations, hence having a large degree of control over the closed-loop system dynamics. However, next to the restrictions we have with PID control, FSF also requires a full state-space representation of the system in order to find the feedback matrix K. This is problematic for a part of the model we've previously created. Let us recall the definition of the Laplace transform, which we need to convert any dynamical system to state-space representation. For any time domain function f(t), the Laplace transform is defined by

$$\mathcal{L}(s) = \int_0^\infty e^{-st} f(t) dt \tag{4.3}$$

Since (3.14) contains tangent functions, we obtain a Laplace transform of (4.3) with $f(t) = \tan(t)$. This integral does not converge but goes to infinity, meaning that no Laplace transform exists for the Magic Formula slip curve. Hance, we cannot - theoretically - place poles to synthesize an FSF controller, making it unusable for our application. When empirically fitting a state space representation to data, we might obtain a state-space model of much higher order than our first-order no-slip model. Thus, we would need to obtain a higher-order fit and secondly, find a very large feedback matrix. Even though these steps may be taken successfully, FSF is only for SISO systems, rendering it inapplicable.

Linear-quadratic regulator Linear quadratic control is a type of state feedback, where gain matrix K assigns weighting factors to the various states. By use of optimization, an LQR automatically finds the optimal gain matrix to achieve the control objectives by the control law u = -Kx. The cost function used in optimization for K is

$$J(u) = \int_0^\infty (x^T Q x + u^T R u + 2x^T N u) dt$$

$$\tag{4.4}$$

Due to the optimization involved, LQR control is, in fact, suitable for MIMO problems. However, and LQR requires a fully defined state-sapce as with FSF. This state-space is currently unavailable, and when determined empirically, will be of very high order rendering LQR control hard.

Extremum seeking control (ESC) The most common control approach in slip control is using the same control architectures used in ABS (anti-lock braking systems). For this, it is common to use hybrid control: a controller which uses state logic in order to determine the controller output. It is, in its most basic form, implemented as sliding mode ESC (extremum seeking control). This type of control is very popular in slip control systems (Calli et al., 2012). It may be defined by a logic that can be described as follows: for a given error ϵ in a dynamical system, the output of the ESC algorithm is given by

$$u = k \cdot sign(\sin(\pi\epsilon/\alpha)) \tag{4.5}$$

4.1. ALGORITHM

With α a positive constant, and k is the ramp gain which determines how fast the system converges. Being in a sense a slightly more elaborate variation on bang-bang control, this algorithm functions in such a way that if the system state (which is in our case the velocity error) goes in the wrong direction, this will trigger a sign change in the algorithm output and hence the control signal, so controlling the input to steer the output in the desired direction. When the optimum is reached (i.e. the slip is minimized adequately), the system will oscillate within the allowed error bound. The error bound that we allow can be tuned by altering parameter α : a smaller value of α yields stricter control and faster but less grave oscillation. A disadvantage of this type of control is that it will yield high-frequency chattering around the reference. The advantage of this algorithm is that no MIMO control scheme is necessary and that, when applied in FDM machines, we can simply use our velocity set points as imposed by the G-code. The extent to which our controller interferes with this velocity set point is determined by k. Given the fact that ESC is common in slip control applications and the fact that it may be applied to an FDM extruder, this control approach will be selected for this research.

4.1.1 Hybrid control

To be more flexible in choosing our error bounds, we expand the functionality of sliding mode ESC by Schmitt trigger or relay behaviour. Now, we no longer have one definition of the switch boundary, but define an upper and lower limit between which we would like our error to be (i.e. make it non-symmetrical). The output is switched between two states depending on the input of the controller. This logic is thus defined by two points: the switch on point V_{IH} and the switch off point V_{IL} instead of the error boundary α . Note that using Schmitt trigger logic has the same functionality as the official ESC notation using sine inside a signum function, however, defining it in this way allows us to use asymmetrical error bounds and is more insightful.

The controller switches to a high output state V_{OH} once the input has exceeded the switch on point. The controller input may then keep increasing or drop below the switch on point, but the output will remain high. Only as the input drops below the switch off point, then the controller state switches to the low state V_{OL} . This logic is visualized in fig. (4.1).



(a) Theoretical relay loop of a Schmitt trigger, explaining the controller logic.

(b) Example of Schmitt trigger applied to a sine input (top). The discretized output is shown in the lower graph.

Figure 4.1 : Representation and example of the Schmitt trigger logic. When applied to a sine wave, the circuit output goes to the high state V_{OH} as the input $u(t) \ge V_{IH}$. This high state is retained until the input $u(t) \le V_{IH}$. Images courtesy of The Mathworks.

The next step is translating the high and low states into a control signal that achieves the control objectives (maximize feed velocity while minimizing slip). In order to do so, the slip will be taken as the system state to dictate the control output. For slip lower than the high input threshold V_{IH} , the state remains low and the controller ramps up the input signal with a gain k_{ramp} . This ramp gain is a tunable parameter which defines the performance of the controller. As soon as the system state exceeds V_{IH} , the controller state goes high and the sign of k_{ramp} flips, now ramping down the input velocity and thus decreasing the slip. This process repeats as the system state drops below V_{IL} . Effectively, we obtain a hybrid controller which stabilizes the velocity in such a way that the slip level is at a predetermined value. The logic of said hybrid controller is based on Schmitt trigger logic.

Performance of the controller is thus defined by the switching boundaries V_{IL} , V_{IH} and the up- or down ramp gain k_{ramp} . Defining, for instance, a very low slip threshold will yield a more strict controller which restricts the feed velocity further to attain a low slip level. Increasing the ramp gain will yield a faster controller with higher performance, but both measures are more sensitive to noise and less robust. Hence, this controller design is a trade-off between performance and stability. Summarizing, this yields a controller structure as in fig. 4.2.



Figure 4.2: Full system block scheme depicting the interactions of various plant and controller subsystems. The system state is used to trigger the Schmitt triggering hybrid controller, which then ramps the controller output up and down, creating an oscillating control signal.

4.2 Off-line implementation using the model

The control structure of fig. 4.2 was implemented into MATLAB/Simulink and solved using a discrete-time solver. The Schmitt trigger was implemented and an arbitrary reference input of 10 mm³/s was given since this is known to yield high slip. The controller was engaged at $t_1 = 3$ s in order to see the difference in output with and without control. An example output with $V_{IL} = 0.15$ and $V_{IH} = 0.30$ is shown in fig. 4.3.



Figure 4.3 : Implementation of ESC control on model using Schmitt trigger logic. Visibly, the controller performance behaves as expected; an oscillating control signal is produced by the controller resulting in an oscillating output. Arbitrary switch boundaries for the Schmitt trigger logic have been selected.

Bear in mind that no control parameters have been tuned in this situation, this figure serves purely as to analyze the behaviour of the controller in stabilizing the input velocity around a given desired slip percentage. Let us first look at the volumetric flow rate. At t = 0, we impose a feed rate of 10 mm³/s, which yields visibly high slip. At the moment that our controller engages, the Schmitt trigger detects a slip $\alpha \geq V_{IH}$. Hence, the state triggers to ramp down the input. This can be observed in the sudden drop in input feed velocity at t_1 . The velocity keeps dropping until V_{IL} is triggered, the Schmitt trigger state slips, and the velocity is ramped up again. We can observe a ramped bang-bang behaviour that stabilizes the input and output velocity.



Figure 4.4 : Implementation of ESC control on model using Schmitt trigger logic, comparing multiple control parameters. Higher allowed slip yields a greater slip percentage in the output and more violent switching.

4.2.1 Tuning

The next step in finding a suitable controller for our purpose is tuning the control parameters V_{IL}, V_{IH} and k_{ramp} . In order to do so, the model was run for multiple parameter combinations and the results were observed to select the best performance. No repeated measurement was required due to the nature of a numerical model. For performance evaluation, we plot the percentual error or slip agains time and observe it's behaviour. First, we tune the magnitude of V_{IL} and V_{IH} with a fixed spacong of 0.1 between them to observe the best combination. These results are plotted in fig. 4.5.

Model behaviour is as expected. We observe oscillation of the error around the predefined set point. As we increase the magnitude of the boundaries, we can see that the observed error increases. Since this is a quantity we wish to minimize, it is best to select the lowest possible magnitude for V_{IL} and V_{IH} . Note that, in theory, an infinitely small magnitude for the boundaries is optimal since this would imply zero error. However, a zero-error state is impossible and thus a small amount of slip must be allowed in order to achieve extrusion at all. Second, addition of noise (discussed in the next paragraph) will yield very low robustness to noise for very low boundary magnitude since in such a

case, the controller will switch to noise instead of the actual system state. Hence, we must select a small, but not too small switching boundary magnitude. Since the lowest magnitudes still yield best performance and are robust to noise (as shown later), we select $V_{IL} = 0.10$ and $V_{IH} = 0.20$. A second motivation for this is the fact that we observe increased oscillation frequency in the error, which is an unwanted phenomenon for smooth operation. Hence, we wish to minimize the magnitude of the switch boundaries. Regarding the value of the ramp k, we plot the model response in fig. 4.7. We wish to have a ramp value for which we obtain adequately fast settling, but not heavy oscillations. Hence, k = 10 fits our needs best and this is the parameter we select.



Figure 4.5 : Tuning of Schmitt trigger switch boundary magnitude. For choosing our error bounds higher, we see that (expectedly), the slip around which the output oscillates increases. Since we desire this to be as low as possible, we which to minimize the error boundaries.



Figure 4.6 : Tuning of Schmitt trigger switching boundary amplitude.For choosing wider error bounds, we see that (expectedly), the slip around which the output oscillates increases and that it oscillates less frequently since it takes longer to settle to the now further away error bound. Since we desire low-amplitude, high-frequency oscillation, we wish to keep the error bounds together as closely as possible.



Figure 4.7 : Tuning of Schmitt trigger up- and down ramp. A higher value of k yields faster settling and also graver oscillation. Since we mainly desire the error to settle quickly, we select k > 10.

4.3 Adding noise to the model and controller

Up to now, we have only modelled system behaviour in a noise-free situation. However, various noise sources are present in our physical system and must be accounted for in order to do accurate modelling and control design. Various types of noise exist, and of each type we will discuss the necessity of implementing them as well as how to implement them into the model.

4.3.1 Model

Measurement noise is noise introduced by sensing equipment. From preliminary investigation, we known that using an encoder yields high levels of measurement noise on our signal. Thus, this cannot be disregarded. System noise and input noise are much harder to quantify from output measurements only, but since the steppers generating the input are very accurate and few external noise sources are present, measurement noise is the only noise type that is incorporated. The addition of this noise w(t) source is depicted in fig. 4.8. The noise is modelled as Gaussian zero-mean white noise having a power equal to the power of noise measured in our physical test setup.



Figure 4.8 : Full system block scheme depicting the interactions of various plant and controller subsystems. The system state is used to trigger the Schmitt trigger, which then ramps the controller output up and down. At the system output, a zero-mean white noise source is added to emulate measurement noise.

4.3.2 Control with noise

In the previous paragraph, we described how noise was added to the model. Now, we again evaluate model response, with noise included. In general, noise will yield higher slip variance. As a second influence, noise tends to destabilize the model quicker since the observed slip may become very large due to noise spikes. Thus, the controller will have to be tuned less strictly and the slip cannot be reduced to the same extent as with the slip-free control. The extent to which this occurs is investigated in this paragraph, and the model is explored for noise conditions in general.

Let us first observe, again, an arbitrary model output to observe it's behaviour. We take the same Schmitt trigger logic $V_{IL} = 0.15$ and $V_{IH} = 0.30$. Model response for a step disturbance and

controller engage after 3 seconds is shown in fig. 4.9. We may observe model and controller behaviour quite similar to the noise-free model. Velocity stabilizes and the noise as controlled to a level that holds an approximate mean between V_{IL} and V_{IH} . The main influence of noise is, as expected, a larger variance in the output and hence a larger spread on the slip. Extreme values occur more often and maxima lie further from the mean.



Figure 4.9 : Implementation of ESC on model using Schmitt trigger logic with simulated noise to check controllability of the system. This model verifies that noise levels are acceptable for slip control without destabilizing the system. Performance is limited when incorporating noise, as smaller slip boundaries will destabilize the system.

52



Figure 4.10: Implementation of ESC on model using Schmitt trigger logic for various switching boundaries. As with before, we observe controllability for both scenarios. Added noise decreases performance, since the strict switching boundaries will attain 20% slip at best, compared to 10% for the no noise simulation.

To investigate further the effects of noise, figure 4.12 shows the influence of noise power on the model and output RMSE. In the figure, warmer colour represents a higher RMSE. We see that for no noise, we are rather free in our choices of control parameters and RMSE is low for a wide range. As the noise power is increased, we observe a higher RMSE for all combinations of control parameters. This makes sense, since the controller will trigger on noise rather than real inputs, switching state even when the actual optimum has not yet been reached. The plots are constructed in such a way that the average between V_{IL} and V_{IH} is shown on the y-axis, whereas the switching boundaries (i.e. V_{IL} and V_{IH} equal this average \pm the switch boundary). Our observations match the expectations that control will be less accurate when noise is implemented, but still possible. Since noise introduces inaccuracies, we are forced to loosen our control boundaries which introduces an additional error.



Figure 4.11 : Effects of noise model on optimal Schmitt trigger settings. The warmer the colour, the higher RMSE between data and model. Visibly, adding noise obviously increases RMSE but does not render the system uncontrollable.

4.4 On-line implementation

In the previous section, we discussed synthesis of the controller in a Simulink model. Physical implementation of said controller is discussed in this section. We will give a general overview of the setup used and discuss some practical considerations regarding the experiment. For a full description of the setup, operation instructions and visual material, please refer to appendix B.

4.4.1 Experimental setup

In short, the experimental setup for this thesis consisted of a Xeed 3D printer with an additional incremental encoder, analog input channel and parallel serial communication with a host PC running MATLAB. MATLAB communicates with the printer and sends G-code commands. Control is done by a dedicated Arduino board which reads the encoder, computes the control signal and communicates this to both the printer and the PC for data analysis. Refer to appendix B for the full experiment description.

4.4.2 Control signal: getting the controller and printer to chat

As a practical consideration, we must find a method to communicate the control signal that throttles the speed to the 3D printer firmware. To ensure continuous operation of the machine, the printer firmware has a 16-slot ring buffer in which it buffers G-code commands retrieved from the G-code server. As the firmware runs, it grabs commands from the beginning of the buffer (called the *head*) and executes them. As the execution of commands, especially for long moves, is naturally much slower than the communication of new commands, new commands are stored at the back of the buffer (the *tail*) as soon as an empty slot in the buffer arises. At execution of a block, a stepper interrupt routine fires repeatedly at a high frequency, grabs a block from this buffer and pulses the stepper driver pins appropriately. The frequency at which the interrupt routine fires determines the pulse frequency and thereby the motor speed.

This method for controlling the motors has a few implications. Foremost, we cannot use serial communication (which is the primary communication protocol for allowing control between for instance a PC and controller board) for our control signal or change the speed in individual G-code blocks. For real-time control, we must be able to change the speed in the middle of a G-code block. When controlling the G-code commands, there will be a 16-line delay since we can only alter the blocks on the tail of the buffer, and we cannot adapt a block that is currently executed. This means that we must somehow change the speed at which the current block is executed from within the interrupt. Second, the interrupt service routine running on the Atmega chip that powers the controller is pushing against its hardware limits. Hence, implementing a serial communication read within this interrupt routine (which is very intensive, computationally) slows it down too much resulting in non-smooth motion. The most efficient means of speed control is real-time control of the ring buffer.



Figure 4.12 : Ring buffer schematic depicting the firmware construction implemented commonly in FDM machines. The machine firmware reads g-code commands from the head as long as buffered commands are available. An external g-code server fills the buffer from the tail.

Controlling the interrupt fire frequency is done by scaling a timer that dictates when the interrupt fires. In other words, when the timer is decreased by a factor of two, the interrupt will fire twice as fast and the printer will print twice as fast. This means that wile our control signals is usually applied as in fig. 4.13, we now require the feedback controller to produce a scaling factor that we multiply with the reference, depicted in fig. 4.14.



Figure 4.13 : Controller block scheme.



Figure 4.14 : Controller block scheme using multiplication.

To implement our control, we must thus translate the control signal which we usually add to our reference into a multiplication factor. This is easily done by realising that for a reference r and control signal c, our true input signal is normally

$$u = r + c \tag{4.6}$$

4.4. ON-LINE IMPLEMENTATION

Our desired control input is calculated by

$$u = kr \tag{4.7}$$

Since eqs. (4.6) and (4.7) must be equal, we can compute the scaling factor k to be

$$k = 1 + \frac{c}{r} \tag{4.8}$$

4.4.3 Performance metrics

In order to quantify the performance of our anti-slip controller, we selected a number of performance metrics. Five quantitative metrics were selected for this research to describe the performance of the controller:

- Root-mean-squared error (RMSE) shows superior performance quantification properties compared to other mean error methods (Chai and Draxler, 2014). RMSE was discussed previously in section 3.3.2 and is computed by (3.18). RMSE is expected to improve when the controller is applied;
- Mean slip was selected since this quantity yields higher interpretability compared to RMSE and is computed simply by $\alpha_m = \frac{\sum \alpha_n}{n}$. It is expected to see a lower mean slip;
- Time-to-complete (TTC) describes the time required for the 3D printer to properly finish all imposed G-code commands and extrude a cumulative amount of filament. TTC is expected to decrease with respect to open-loop control since the controller optimizes feed velocity at low reference velocities. Time-to-complete is measured as the time to extrude either 1000 or 2000 mm of filament, depending on the length of the experiment;
- Cumulative extrusion error describes the amount of under extrusion that occurred during print and is a good measure for surface quality as low extrusion error corresponds to adequate material deposition. It is defined as the integral of the measured volumetric flow rate. Sometimes, there may be a discrepancy between cumulative error and RMSE where a low cumulative error may exist along with high RMSE. This is less contradictory than it seems: when the input velocity suddenly drops, the output velocity drops with a delay (recall the extruder transfer function) leading to the output sometimes being larger than the input. This means that extra filament is extruded, which is beneficial to a low cumulative error, but is considered erroneous by an RMSE calculation. Therefore, RMSE is a more compelling metric since this better represents how well the output tracks the input compared to cumulative error.

4.4.4 Tuning and results

This section describes tuning of the on-line ESC control implementation. Before evaluation of controller performance can be done, we require appropriate values for control parameters V_{IL} , V_{IH} and k. In order to determine these, two approaches may be considered: analytical tuning, based on modelling and equations, or empirical tuning which relies on our experiments. Previously, the controller was tuned empirically using our numerical model. Given the black-box nature of our model and the fact that we do not have analytical expressions that fully describe extruder behaviour, controller tuning can only be performed empirically. To do so, a step input was controlled using multiple controller settings, and the best settings were selected based on

performance metrics and qualitative assessment of the control behaviour. Figure 4.15 shows a detail of the slip under step tracking.



Performance of controller for VIL and VIH in tracking step input

Figure 4.15 : Step input controller tuning. Time trace detail for various control parameter settings. When increasing the error bounds, we can see that generally the combination of all performance metrics deteriorates. Since the system can be controlled with the lowest error bounds and this yields best performance, those bounds are selected.

First evaluating the Schmitt trigger threshold boundaries, we increase them in increments of 0.05. The bounds are kept symmetrical to keep the number of controller variations limited and because neither too much nor too little slip is desired, is does not make sense to take asymmetrical bounds.
4.4. ON-LINE IMPLEMENTATION

Theoretically, the lowest bounds should yield the lowest error and the best control performance, but a very strict boundary also yields high noise sensitivity. Based on these results, the lowest thresholds yield the best performance: mean slip and RMSE are both minimal. TTC does not differ between settings, and extrusion error is somewhat higher than with the other settings, but still far below that of the open-loop condition. Second, cumulative error is less compelling than RMSE, as mentioned before. Qualitatively, we can see that the lowest error shows the most constant oscillations under these control conditions, which is beneficial to part quality.

If we plot the control performance for various ramp gains, we obtain figure 4.16, similar conclusions may be drawn. Again, TTC does not seem to differ much with k, For k = 10, RMSE and mean slip are second-lowest while the error is lowest. Hence, k = 10 shows most promise for good control. We therefore select $V_{IL} = 0.05$, $V_{IH} = 0.10$ and k = 10 as control parameters. Note that this tuning shows good correspondence to the model, except for the fact that we selected our boundaries slightly stricter.

Performance of controller for ramp gain k in tracking step input



Figure 4.16 : Step input controller ramp tuning. Time trace detail for various control parameter settings. Note that the average slip and RMSE, and cumulative error are all second best, making k = 10 the best choice.

Chapter 5

Performance evaluation

In this chapter, we cover the performance evaluation of the controller which we synthesized in the previous chapter. It is desired to compare the performance of the physically implemented controller to that of our theoretical model, as well as validate that the proposed controller brings performance improvements with respect to the original open-loop system

5.1 Method for evaluation

For performance validation of the controller proposed in the previous chapter, the performance of the controller has been tested for two input signals that together describe a large bandwidth of actuation. The following input signals were used, as in the modelling section for convenient comparison:

- Step input tracking, to check high bandwidth behaviour including high accelerations.
- Sawtooth and ramp input tracking to verify reference signal tracking performance.

These are the same reference signals that were used priorly for modelling and system identification, which allows for good comparison. G-code commands were sent and received by a host PC running MATLAB. Refer to appendix B for the full description of the set-up.

Regarding quantification of the performance, the same four performance metrics have been used as with tuning of the controller.

5.2 Results

In order to do a proper performance comparison, the behaviour of the system must be observed with and without our application of hybrid control. The reference condition will be referred to as the "control" condition or group, whereas the condition with controller enabled with be referred to as "Schmitt trigger" or "hybrid". 10 data sets were obtained for each condition to obtain sufficient power in the data. To analyse our data, statistical tests were performed on all test results. To determine the type of test required, we must consider the type of data we are using. The performance metrics selected are assigned to two groups: experiment (with controller) and reference condition. Since both conditions were not performed by persons, no degree of pairing between conditions exists. Hence, we have unpaired groups of ratio data, to which a number of tests may be applied depending on the normality of the data. Lilliefors testing (Yap, 2011), used for testing the null hypothesis that a data set comes from a normal distribution, showed non-normally distributed data in some of the data sets. Due to this non-normality, the data must be compared using a Mann-Whitney U test (Charan, 2011). This test validates the null hypothesis that the data in two arbitrary data sets x and y have equal medians, against the alternative that they do not.

5.2.1 Step input tracking

As an input signal, we first selected a continuous velocity input step with a set point velocity higher than the critical velocity at which we know significant slip to occur. As such, we may assume slip will occur enough to enable the slip controller and we will achieve good material for comparison (i.e. trying to print faster than the current capabilities will test the controller properly). As a first test, the results was executed for the control condition, of which a single data set is displayed in figure 5.1.



Figure 5.1 : Time trace of high-velocity step input (blue) and output velocity (red) without control. We observe continuous, high-magnitude stick slip behaviour indicating that this set velocity is unattainable by the printer. Slip percentages are high.

We observe - naturally - the same slip behaviour as in earlier observations. At high velocities, the pinch rollers lose traction on the filament strand, and the output velocity drops. As the velocity drops, the pressure drop across the hot end decreases, extrusion pressure decreases and the rollers regain traction, increasing the output velocity once more. Note that at the beginning of the experiment, the output velocity approaches the reference unusually well, and does not show slip. This is a thermal effect: under steady-state extrusion, heat transfer between the nozzle wall and

5.2. RESULTS

filament is insufficient to fully heat the plastic. When no extrusion occurs, the filament inside the nozzle is completely heated. Hence, during the first few seconds, the viscosity of the filament is significantly lower that under steady-state extrusion, yielding less pressure and thus less slip. Since this steady-state under-heating of filament is common in FDM, the behaviour under steady-state operation is more representative.

As a next step, we applied the controller as tuned in the previous chapter. Again, 10 data sets were collected containing the I/O behaviour of the extruder. The time trace of a single data set is shown in figure 5.2. We observe that when possible, the controller increases the velocity reference up to approximately 50 mm³/s, and also reduces it to slightly below 15 mm³/s, at which we know the largest slip to first occur. The most interesting notion to be made is in the subplot showing α , in which we observe the slip to be 1) quickly oscillating, which we would expect our hybrid controller to do, but 2) also become negative , which basically corresponds to the situation where more filament is extruded than desired. This negative slip value originates from the dynamics of the filament, we can see better if we zoom in. Figure 5.3 shows a detail of the data.



Figure 5.2 : Time trace of high-velocity step input (blue) and output velocity (red) with hybrid control enabled. As high slip occurs, the controller state flips and the input velocity oscillates in such a way that slip remains at an acceptable level. Noise artefacts and extruder dynamics result in negative slip values. Visually, slip is lower compared to open-loop control.

In this figure, we can closely observe the behaviour of the controller. Note that the reference input and observed output are coupled, and that they both respond to each other. For low slip, the controller increases the reference velocity and vice versa. In here lies the negative slip phenomenon, which we can observe for instance at t = 10.75 s. After a small input peak at 10.5 s, the output reacts as a damping element with a delay (recall the first-order transfer function for the extruder) and increases. This ramps up the input as well as it detects lower slip, since the output is now higher than the desired input, resulting from a delayed reaction on an earlier input signal. As the input is ramped up, slip increases and the controller ramps the speed down again. Due to the viscous effects in the filament, the output has a small delay causing output velocity to drop slower than input velocity. As such, the output becomes higher than the input and negative slip occurs. This additional extrusion is equally unwanted as insufficient extrusion since the printer does not receive any sensor feedback and does not know what it is doing. Especially if, at a later stage, head motion would also be coupled to this controller, the response of the head is much more immediate compared to that of the filament. Hence, no deviation from the desired input is good, regardless of the direction. Hence, negative and positive slip will be treated equally in computing performance metrics. To only observe the magnitude of the slip, we disregard the sign in computing average slip and only use absolute values.



Figure 5.3 : Detail time trace of high-velocity step input (blue) and output velocity (red) with ESC enabled. As high slip occurs, the controller state flips and the input velocity oscillates in such a way that slip remains at an acceptable level. Noise artefacts and extruder dynamics result in negative slip values. Visually, slip is lower compared to open-loop control.

After qualitative analysis of control performance, quantitative results are shown in fig. 5.4. For each performance metric we plot the raw data points, means and 95% confidence interval for the control and Schmitt trigger condition.



Performance metrics for 10 data sets - step input

Figure 5.4 : Performance metrics for a step input with statistical data. The blue dot indicates the mean value for each metric with error bars indicating the 95% confidence interval of the data. The CIs are generally very small, indicating good repeatability. All metrics improve significantly.

Performance increases significantly for all metrics at the p < 0.01 level according to Mann-Whitney U testing. Average slip is reduced most by 58.6 %. Note that while the open-loop control data sets have some spread, closed-loop data points show great overlap, which corresponds to the high probability values observed. The improvement of all metrics matches the prior expectations since an anti-slip controller should of course aim to minimize errors. The improved time-to-complete

indicates that the increase of input velocity for low-slip inputs is also successful. All results have been tabulated in table 5.1.

Table 5.1 : Performance metrics for step input with statistical data.

| Metric | Control condition | Schmitt trigger | Improvement | |
|--------------------------|-------------------|-----------------|-----------------------|--|
| Mean slip | 0.50 | 0.21 | 58.6% ($p < 0.001$) | |
| $RMSE (mm^3/s)$ | 9.62 | 5.9 | 38.5% (p < 0.001) | |
| TTC (s) | 147.8 | 69.9 | 52.7% $(p < 0.001)$ | |
| Extrusion error (mm^3) | 1779 | 454.6 | 74.4% $(p < 0.001)$ | |

5.2.2 Sawtooth input tracking

After applying a step input, we now apply a sawtooth reference signal to see how the controller performs under tracking a more complicated reference signal, which corresponds more to the circumstances of a real 3D print. Figure 5.5 shows the time trace of one control experiment. Recalling the system behaviour we described earlier, we observe slip at high velocities which occurs roughly from $\phi > 15 \text{ mm}^3/\text{s}$.



Figure 5.5 : Time trace of high-velocity step input (blue) and output velocity (red) without control. We observe continuous, high-magnitude stick slip behaviour indicating that this set velocity is unattainable by the printer. Slip percentages are high.

Enabling our hybrid controller, we obtain data sets as shown in figure 5.6. Now, velocity reference peaks up to approximately 35 mm³/s, and reduces to lower velocity than in step tracking, which is desired by the original reference signal. Again, in the subplot showing α , we observe the slip to be

5.2. RESULTS

highly oscillating and negative. In general, the controller ramps velocity up and down while still tracking the imposed velocity shape. Figure 5.7 shows a detail of the time data.



Figure 5.6 : Time trace of high-velocity step input (blue) and output velocity (red) with hybrid control enabled. As high slip occurs, the controller state flips and the input velocity oscillates in such a way that slip remains at an acceptable level. Noise artefacts and extruder dynamics result in negative slip values. Visually, slip is lower compared to open-loop control.



Figure 5.7 : Detail time trace of high-velocity step input (blue) and output velocity (red) with hybrid control enabled. As high slip occurs, the controller state flips and the input velocity oscillates in such a way that slip remains at an acceptable level. Noise artefacts and viscous extruder dynamics result in negative slip values. Visually, slip is lower compared to open-loop control.

In this detail plot, we observe the same behaviour as earlier. The velocity is increased as soon as the controller notices that a small amount of slip occurs. Due to the dynamics of the extruder (f.i. viscosity), output responds with a small delay to any change in reference, causing at time negative slip values. Overall, we observe good tracking of the reference signal and velocity optimization. Note that as with step tracking, the low and high thresholds of the Schmitt trigger are set to $\alpha_{IL} = 0.05$ and $\alpha_{IH} = 0.10$, respectively. Yet, it is not uncommon for the slip to approach values of 0.50. This, too, may be explained by the viscous behaviour of the filament: when the controller responds to the set slip thresholds, the actually attained slip will be different. Secondly, sampling is at 100 Hz, which also introduces delay. Combined with the input noise in the analog read routine of the printer, this means that slip is controlled less strictly than desired.

Quantitatively, the results are summarized in figure 5.8. Similar results are obtained as with step tracking, but the differences have become much smaller. This can be explained by the more complex dynamics in tracking a more complex reference signal. The controller is less capable of keeping up with these dynamics, resulting in less performance. Secondly, no longer all metrics improve. Mean slip actually increases, due to the more complex tracking. Furthermore, spread between the data sets is larger than with step tracking.



Performance metrics for 10 data sets - sawtooth input

Figure 5.8 : Performance metrics for a step input with statistical data. The blue dot indicates the mean value for each metric with error bars indicating the 95% confidence interval of the data. The CIs are generally very small, indicating good repeatability. All metrics improve albeit less compared to simple step tracking.

Lastly, all results have been tabulated in table 5.2. Only small improvements are made for RMSE and TTC, but overall extrusion error is reduced by 25 %.

| Metric | Control condition | Schmitt trigger | Improvement | |
|--------------------------|-------------------|-----------------|-------------|-------------|
| Mean slip | 0.135 | 0.24 | -77.5 % | (p < 0.001) |
| $RMSE (mm^3/s)$ | 3.8 | 3.56 | 4.6~% | (p < 0.05) |
| TTC (s) | 192.9 | 183.8 | 4.7~% | (p < 0.05) |
| Extrusion error (mm^3) | 400 | 290 | 25.4~% | (p < 0.001) |

Table 5.2 : Performance metrics for sawtooth input with statistical data.

5.2.3 On the effectiveness of on-line filtering

Given the reduced performance of the controller under complex reference signal tracking, it is interesting to investigate the effects of filtering and to see whether this will positively impact performance. As already discussed and showed in previous sections, the system used is subject to high measurement noise, i.e. the sensor used is highly sensitive to artefacts resulting from sampling frequency and discontinuities in the data. This results in control actions being taken based on noise instead of actual, accurate state information. Hence, it is desired to investigate whether on-line filtering of the sensor data will yield performance improvement. This subsection investigates the effectiveness of filtering in the controller.

A countless number of filtering options exists. One of the simplest is a Simple Moving Average (SMA), which is computationally very non-intensive, keeping controller performance and sampling rate up to par. Hence, we investigated the effectiveness of multiple SMAs. Since any moving average filtering has low-pass filter characteristics (with accompanying delays), any filtering will negatively impact control bandwidth. The central question is whether the noise reduction gained introduces enough performance improvement to compensate for the reduced bandwidth.

For any signal data x(k) containing *n* samples, the simple moving average is computed by (Makridakis and Wheelwright, 2010):

$$\hat{x}_k = \frac{x_k + x_{k-1} + \ldots + x_{k-(n-1)}}{n} \tag{5.1}$$

with n the number of samples in the filter; more samples corresponds to stronger smoothing behaviour and larger delays. To investigate the effectiveness of application of a SMA filter, it was investigated whether the performance of the controller changed for a varying delay. The system performance is shown in figure 5.9 under three different filters.



Performance metrics for 10 data sets - effectiveness of SMA

Figure 5.9 : Performance metrics for a step input with statistical data using SMA filtering. CIs are very large for high numbers of samples, and the computation of metrics is not reliable due to the instability of the system.

In this experiment, all conditions have the Schmitt trigger controller enabled, which is referred to as the "no filter" condition. For a two-sample SMA, performance stays approximately the same but both RMSE and extrusion error increase. Since a two-sample SMA has the least amount of samples possible, it is safe to say that this type of filtering will not yield further improvements since with higher numbers of samples, performance will only decrease. Taking a two-sample SMA effectively halves the sampling rate to 50 Hz, which can track less quick dynamics compared to 100 Hz sampling. The performance metrics for higher-sample SMAs can be somewhat misleading: for higher numbers of samples the system destabilizes and behaves highly unexpectedly. Therefore, while the 10-sample slip, TTC and error may look not too bad, in reality the tracking of reference signals is completely gone. Data sets have extreme spread, as indicated by the confidence intervals. RMSE could not be calculated for the 10-sample SMA due to high numbers of NaN or inf in the output, indicating high numerical instability in SMA filtering for this application. Observing above results, it is safe to state that moving average filtering is not a viable approach for performance enhancement of the hybrid controller.

Chapter 6

Discussion

Over the course of this thesis, an accurate model describing the input-output behaviour of an FDM extruder was produced, and an effective anti-slip feedback controller was synthesized to increase the quality of FDM printed parts, while at the same time reducing the time needed to create them. This chapter will discuss the major implications, limitations and recommendations of our findings.

Firstly, we will discuss the implications of the extruder model presented in chapter 3. It was found at the start of this thesis - that no model exists to accurately describe the combined input-output slip behaviour of and FDM extruder. By considering the extruder as a black box model and combining a known transfer function model with one from the automotive industry, we succeeded in numerically simulating the magnitude of extruder slip behaviour. Models such as this increase our insights in FDM extrusion processes and, more importantly, allow us to approach FDM printer design from a modelling angle rather than one based on physical prototyping and empirical measurement. The current design cycle of 3D printer hardware and software involves virtually no usage of models. Hence, large improvement in the efficiency of the design process can be achieved by model-based design.

A point of interest in our modelling approach is it's back-box nature, meaning that most physical sense is lost and model parameters do not translate directly to physical parameters. Therefore, though this model facilitates design of a machine around an extruder model, it does not facilitate the model-based design of the extruder itself. To enable this, a good analytical model is required that describes the slip behaviour of the extruder in physical quantities. For the scope of this thesis, the development of an analytical model was purposefully avoided due to the downsides of this approach mentioned in literature. However, the development of such a model will aid greatly in the extruder design cycle. Hence, this is greatly recommended as a topic for future research.

Secondly, considering the implications of the controller synthesis, we may say that the development of an anti-slip controller was successful. Using ESC control based on Schmitt trigger logic, the controller appropriately adapted the extrusion velocity of a Xeed 3D printer. Therefore, we achieved the goal of high-speed FDM printing while retaining object quality. Error-related performance metrics significantly improved over the currently implemented open-loop approach. This confirms that our hybrid controller does find a *feasible* maximum regarding extrusion velocity, thereby printing at high speed relative to the achieved object quality and low error. It must be noted, however, that this investigation only quantified part quality in terms of cumulative extrusion error, assuming that a lower error in the extrudate will imply less part deficiencies. The actual judgement of the quality of 3D printed parts involves a much greater number of variables, such as surface finish and roughness, layer height, ringing, and a host of thermal effects in heating and cooling the build materials. Especially since our control approach inherently introduces high-frequency chattering, the proposed method will be less applicable to real-world application than the results suggest since chatter in the output of filament will also lead to chatter in the motion of the print head. Second, high-speed extrusion requires a full revision of the thermal handling of the printed object. Printing at higher speeds will introduce much more residual heat in a 3D printed object which has a large influence on part quality. Hence, the ability to sufficiently cool an object is equally important as being able to print it quickly.

The successful application of this novel control method for FDM presents significant opportunity for the near future. FDM is becoming increasingly popular in industrial sectors with respect to do-it-yourself enthusiasts, who have long (and still do) form the biggest market for FDM. Industrial printers are characterized by very large build volumes and, inherent to the technology, very long print times. The results of this work could benefit the AM industry in 2 ways: (1) the anti-slip controller actively seeks the highest feasible speed, ensuring high speed combined with part integrity; (2) considering the feed-forward nature of the FDM process, critical slip halfway a very long print job results in unacceptable amounts of material and time loss, as the entire object has to be re-printed. Slip prevention mitigates this danger. Hence, this research opens a door to new possibilities regarding large-scale FDM. However, given the fundamental nature of this research, limited time and limited scope, a number of refinements and extensions to the work presented in this thesis are recommended.

As a first recommendation, other means of anti-slip control should be compared to the one described in this work. When considering other ways of reducing slip, one may consider a very simple option: simply lower the open-loop control speed to the point where we know that virtually no slip occurs under normal circumstances. This mitigates the need for any feedback control and is sufficient under normal conditions where no optimization of print speed is required. The controller has two benefits over this open-loop approach for situations where we wish to increase speed at the same time. First, by seeking an optimum equilibrium at a predetermined slip state, we overrule any conservative open-loop reference velocity and thus attain a lower time-to-complete at a reasonable error. Second, open-loop control is not robust to external disturbances such as changes in filament diameter. For low velocities, such disturbances are not of importance. When a high open-loop velocity is selected, sensitivity to small disturbances leads to critical slip more easily, justifying the usage of a complicated feedback mechanism. There are, though, a large number of feedback-based control architectures available for slip reduction, especially multiple variations on the simplified extremum seeking control (Calli et al., 2012) that we use in this thesis. The control applied may, for instance, show far too great oscillation to be of use in an actual print. Hence, the practical limitations of the controller should be investigated. More sophisticated control approaches (such as perturbation based ESC) may find an optimum more efficiently and quickly compared to the crude hybrid control implemented now, or show less grave oscillation which increases practical applicability. In particular, we recommend the exploration of control methods that are inherently more robust to noise, such as

Kalman filtering or the implementation of a state observer/estimator. Improving the control method will likely further improve performance and more importantly smooth the system response, resulting in higher part quality.

Many of these (possibly superior) control schemes were omitted from this research due to mathematical limitations in converting our slip model into transfer function or state space form. Even though the exact analytical expression for the Pacejka curve may not be transformed to the Laplace domain, there are multiple workarounds for this. As a first solution, we may opt to use a black-box system identification method as we used for low-slip conditions in order to obtain a state space or transfer function. A second alternative would be to find a function that approximates the shape of the slip curve applied in this work. Given the shape of the Pacejka curve with the identified parameters we found, it will likely be possible to find a good regression by fitting a low-order polynomial or sigmoid function to the slip data, yielding a function for which the Laplace transform does exist. After finding a suitable state-space in either of ways described above, we may apply more sophisticated control methods involving pole placement or linear-quadratic regularization. These control methods also open up new methods of tuning. Whereas the tuning used in this work was of purely empirical nature, having a full system state space allows for theoretical optimization of the controller parameters.

A second recommendation is based on the bottlenecks leading to filament slip: thermal behaviour of the extruder. As mentioned, the temperature of the hot end is set to a desired extrusion temperature. However, since heat transfer takes time, this does not necessarily imply adequate filament heating, especially at high feed velocities. Even with effective slip control applied, there will be a maximum to the extrusion since insufficient heat can be transferred between the tube wall and the filament. No control law yet exists that relates hot end temperature to extrusion. By finding even a simple open-loop control scheme that incorporates a feed-forward scaling of extrusion temperature based on implied feed velocity, one may be able to ensure constant and adequate viscosity of the melt, decreasing pressure drop significantly. It is highly recommended to investigate the potential of such a feed-forward temperature control.

The last recommendation for the improvement of this work relates to the experimental setup used. The feedback control used in this investigation was of limited quality, which was also inherent to the resource-constrained hardware used for sensing and computation. A 500 pulses-per-revolution resolution in the incremental encoder implies limited accuracy, especially combined with the low quality controller chip which has a rather unreliable and unsteady clock frequency of approximately 100 Hz. Last, using a 10-bit DAC limited the resolution of the control signal, which was also read in a noisy fashion by the low-quality printer hardware and firmware. Hence, improvement in hardware quality of components will yield higher control performance and thus higher applicability.

After follow-up of the above recommendations, significant steps towards the effective application of high-speed FDM will have been made, and we will be one step closer to the large-scale application of this technology.

Chapter 7

Conclusions

This work had two goals at the start. The first goal was to find an accurate numerical model that describes extruder input-output behaviour and incorporates slip. Concerning this, we conclude:

- We successfully created an extruder model for both low-slip and high-slip regions of operation. This model was achieved by combining a first-order no-slip model and the "Magic Formula" (Pacejka, 1987);
- The model accurately describes the magnitude of slip in an FDM extruder;
- Tuning of model parameters is a highly involved process requiring system identification and genetic algorithm optimization;
- The model found can prove of high usefulness in model-based design of FDM machines;
- The black-box model found should be converted to an analytical model at a later stage to allow for more physical interpretability.

Based on this model, the second goal was to synthesize of a multi-objective extrusion controller for a Leapfrog FDM machine extruder, which maximizes velocity while retaining high object quality.

- The obtained hybrid controller succeeded in significantly increasing performance: while increasing part quality, time-to-complete of print jobs decreased by over 50 % in step tracking and 25 % in tracking of a complex reference signal (sawtooth);
- The attained performance increase shows good promise for small-scale industrial use of FDM printing: a 25% time-to-complete decrease means 25% faster production. Whether this improvement is sufficient to compete with subtractive methods, depends on object geometry.
- High levels of noise and low quality hardware negatively impact controller performance;
- It is recommended to explore alternative control architectures incorporating noise suppression, alternative means of filtering, or to find more exact methods for controller synthesis.

Bibliography

- M. K. Agarwala, V. R. Jamalabad, N. a. Langrana, A. Safari, P. J. Whalen, and S. C. Danforth. Structural quality of parts processed by fused deposition. *Rapid Prototyping Journal*, 2(4):4–19, 1996. ISSN 1355-2546. doi: 10.1108/13552549610732034.
- K. J. Aström. Maximum Likelihood and Prediction Error Methods *. Automatica, 16, 1980.
- A. Bellini and M. Bertoldi. Liquefier Dynamics in Fused Deposition Modeling. 126(May 2004), 2015. doi: 10.1115/1.1688377.
- H. Bikas, P. Stavropoulos, and G. Chryssolouris. Additive manufacturing methods and modelling approaches: a critical review. *The International Journal of Advanced Manufacturing Technology*, pages 1–17, 2015. ISSN 0268-3768. doi: 10.1007/s00170-015-7576-2. URL http://link.springer.com/10.1007/s00170-015-7576-2.
- B. Calli, W. Caarls, P. Jonker, and M. Wisse. Comparison of Extremum Seeking Control Algorithms for Robotic Applications. *International Conference on Intelligent Robots and Systems (IROS)*, 2012.
- T. Chai and R. R. Draxler. Root mean square error (RMSE) or mean absolute error (MAE)? -Arguments against avoiding RMSE in the literature. *Geoscientific Model Development*, 7(3): 1247–1250, 2014. ISSN 19919603. doi: 10.5194/gmd-7-1247-2014.
- J. Charan. How to Select An Appropriate Statistical Tool? Journal of Pharmaceutical Negative Results, 1(2):2–4, 2011.
- K. Deb. An efficient constraint handling method for genetic algorithms. 186:311–338, 2000.
- K. Deb and S. Srivastava. A genetic algorithm based augmented Lagrangian method for constrained optimization. pages 869–902, 2012. doi: 10.1007/s10589-012-9468-9.
- W. Gao, Y. Zhang, D. Ramanujan, K. Ramani, Y. Chen, C. B. Williams, C. C. Wang, Y. C. Shin, S. Zhang, and P. D. Zavattieri. The status, challenges, and future of additive manufacturing in engineering. *Computer-Aided Design*, 69:65-89, 2015. ISSN 00104485. doi: 10.1016/j.cad.2015.04.001. URL http://www.sciencedirect.com/science/article/pii/S0010448515000469.
- J. Griffey. The Types of 3-D Printing. Library Technology Reports, pages 8–12, 2014.

- W. J. Heij. The need for speed: towards extreme high speed fused deposition modeling. Internship thesis, Delft University of Technology, 2015.
- C. W. Hull. Apparatus for production of three-dimensional objects by stereolithography, 1986.
- D. S. Kammer, M. Radiguet, J. P. Ampuero, and J. F. Molinari. Linear elastic fracture mechanics predicts the propagation distance of frictional slip. *Tribology Letters*, 57(3), 2015. ISSN 10238883. doi: 10.1007/s11249-014-0451-8.
- A. Mahdavi and M. Ebrahim. An augmented Lagrangian ant colony based method for constrained optimization. pages 263–276, 2015. doi: 10.1007/s10589-014-9664-x.
- S. Makridakis and S. C. Wheelwright. Adaptive Filtering : An Integrated Autoregressive / Moving Average Filter for Time Series Forecasting Author (s): Spyros Makridakis and Steven C. Wheelwright Published by : Palgrave Macmillan Journals on behalf of the Operational Research Society Stabl. Operational Research Quarterly, 28(2):425–437, 2010.
- M. S. Mason, T. Huang, R. G. Landers, M. C. Leu, and G. E. Hilmas. Aqueous-based extrusion of high solids loading ceramic pastes : Process modeling and control. 9:2946–2957, 2008. doi: 10.1016/j.jmatprotec.2008.07.004.
- M. Nikzad, S. H. Massod, I. Sbarski, and A. Groth. A Study of Melt Flow Analysis of Polycarbonate (PC) in Fused Deposition Modelling Process. *International Conference and Exhibition on Sustainable Energy and Advanced Materials*, 14(June):29–37, 2009.
- H. Pacejka. Tyre Modelling for Use in Vehicle Dynamics Studies. 1987.
- D. Pham and R. Gault. A comparison of rapid prototyping technologies. International Journal of Machine Tools and Manufacture, 38(10-11):1257-1287, 1998. ISSN 08906955. doi: 10.1016/S0890-6955(97)00137-5. URL http://www.sciencedirect.com/science/article/pii/S0890695597001375.
- S. J. Qin. An overview of subspace identification. 30(May):1502–1513, 2006. doi: 10.1016/j.compchemeng.2006.05.045.
- V. Riemens. Is een verbetering van de hot-end de sleutel tot een productiviteitsverhoging ? Technical report, Hogeschool van Amsterdam, 2016.
- J. Sjöberg, Q. Zhang, L. Ljung, A. Benveniste, and B. Delyon. Nonlinear Black-box Modeling in System Identification : Unified Overview *. 31(12):1691–1724, 1995.
- J. R. Tumbleston, D. Shirvanyants, N. Ermoshkin, R. Janusziewicz, A. R. Johnson, D. Kelly, K. Chen, R. Pinschmidt, J. P. Rolland, A. Ermoshkin, E. T. Samulski, and J. M. Desimone. Continuous liquid interface production of 3D objects. *Science*, 347:1349–1352, 2015.
- B. N. Turner, R. Strong, and S. A. Gold. A review of melt extrusion additive manufacturing processes : I. Process design and modeling. *Rapid Prototyping Journal*, 3(January 2013):192–204, 2014. doi: 10.1108/RPJ-01-2013-0012.

- M. Verhaegen and V. Verdult. *Filtering and System Identification: A Least Squares Approach*. Cambridge University Press, first pape edition, 2007.
- M. Vibergt. Subspace-based Methods for the Identification Time-invariant Systems * of Linear. 31 (12), 1995.
- B. W. Yap. Power Comparisons of Shapiro-Wilk , Kolmogorov-Smirnov , Lilliefors and Anderson-Darling Tests. (January), 2011.

BIBLIOGRAPHY

Appendix A

G-Codes

This appendix contains a list of possible G-codes, included as a convenient reference to the reader.

G Codes

- GO $\operatorname{See}\,\operatorname{G1}$
- G1 Coordinated Movement X Y Z E
- G4 Dwell S (seconds) or P(milliseconds)
- G10 Retract filament according to settings of M207 $\,$
- G11 Retract recover filament according to settings of M208 $\,$
- G28 Home all Axis
- G29 Test if Zprobe solenoid is working
- ${\rm G32}$ Level printing bed
- G40 Print amount of steps missed since last reset
- G90 Use Absolute Coordinates
- G91 Use Relative Coordinates
- G92 Set current position to coordinates given

M Codes

- M104 Set extruder target temperature
- M105 Read current temperature
- ${\rm M106}$ Fan on
- M107 Fan off
- $\rm M109$ Wait for extruder current temperature to reach target temperature.
- M140 Set bed target temperature
- M190 Wait for bed current temperature to reach target temperature.
- M114 Display current position

Custom M Codes

- M17 Enable/Power all stepper motors
- M18 Disable all stepper motors; same as M84 $\,$
- M42 Change pin status via gcode
- M50 Set Extruder 2 Offset. Does not reset with firmware reset M502.

M80 - Turn on Power Supply

M81 - Turn off Power Supply

 $\rm M82$ - Set E codes absolute (default)

M83 - Set E codes relative while in Absolute Coordinates (G90) mode

M84 - Disable steppers until next move, or use S (seconds) to specify an inactivity timeout, after which the steppers will be disabled. S0 to disable the timeout.

M85 - Set inactivity shutdown timer with parameter S (seconds). To disable set zero (default)

M92 - Set axis_steps_per_unit - same syntax as G92

M114 - Output current position to serial port

M119 - Output Endstop status to serial port

M200 - Set filament diameter

M201 - Set max acceleration in units/s² for print moves (M201 X1000 Y1000)

M203 - Set maximum feedrate that your machine can sustain (M203 X200 Y200 Z300 E10000) in mm/sec

M204 - Set default acceleration: S normal moves T filament only moves (M204 S3000 T7000) im mm/sec^2 also sets minimum segment time in ms (B20000) to prevent buffer underruns and M20 minimum feedrate

M205 - advanced settings: minimum travel speed S=while printing T=travel only, B=minimum segment time X= maximum xy jerk, Z=maximum Z jerk, E=maximum E jerk

M206 - set additional homing offset

M220 - S (factor in percent) set speed factor override percentage (M220 S110)

M221 - S(factor in percent) set extrude factor override percentage (M221 S75)

M301 - Set PID parameters P I and D

 $\rm M302$ - Allow cold extrudes

 $\rm M303$ - $\rm PID$ relay autotune S (temperature) sets the target temperature. (default target

temperature erature = 150 C)

M400 - Finish all moves

M500 - stores paramters in EEPROM

M501 - reads parameters from EEPROM (if you need reset them after you changed them temporarily).

M502 - reverts to the default "factory settings". You still need to store them in EEPROM afterwards if you want to.

M503 - print the current settings (from memory not from eeprom)

M601 - Set calibration offsets (X, Y, Z)

M999 - Restart after being stopped by error

Appendix B

Experiment Description

This appendix describes the components and steps required to hook up the real-time anti-slip controller and run an experiment.

B.1 List of Used Equipment

This subsection describes briefly a list of used equipment. The following hardware components are required in order to execute the experiments:

- Leapfrog Xeed. This 3D printer receives the analog control signal from the DAC, reads it from an analog pin, and scales the speed appropriately.
- Omron E6B2-CWZ6C incremental encoder, 500 ppr. This is the sensor that measures filament output position and velocity. Output data is transferred to the controller.
- Adafruit MCP4725 10-bit DAC (digital-analog converter) with I²C. This DAC converts the controller's desired analog control signal to a true analog output (whereas the controller itself is normally only capable of generating an effective voltage using PWM. The DAC enables accurate analog output up to 10 bits, or 1024 steps.
- Arduino Mega 2560 microcontroller. This microcontroller board uses an Atmel 2560 chip at 16 MHz and is the primary controller in this setup. It receives G-code from the host PC (input data) and output data from the encoder. It then computes the slip and generates an analog control signal outputted by the DAC.
- HP Envy ae046-nd. This is the host PC which sends G-code to the printer and controller, and receives the data for analysis.
- Creatr HS 3D printer. This printer was incorporated into the setup for its extra I/O slots on it's PCB.

B.2 Connection of Hardware

All hardware is connected as shown in fig. B.1. A detailed functionality of the setup is as follows: central to the entire setup is the PC, which acts as a server and data analysis machine. The PC's

main purpose is that of a G-code server run by MATLAB R2016a. The server is continuously communicating with the motor control PCB of the Leapfrog Xeed, sending G-code lines and receiving confirmation data via USB serial. The printer PCB runs on a company-adapted version of Marlin 1.0.0 firmware.

Next to sending the G-code to the printer PCB, MATLAB also extracts the velocity set point and sends this to the Arduino for slip control purposes, also via USB serial protocol. The Arduino sychronously reads the encoder and thereby has full I/O data available. The dedicated control firmware running on the Arduino (refer to appendix D for an overview of software and firmware used, as well as code excerpts) uses the I/O data to compute position, velocity and slip, and sends this data back to the PC for logging. The Arduino computes a control signal which has to be communicated to the motor control board. The Mega 2560 does not have any true analog output channels but PWM outputs, whereas we need a true analog signal for good quality analog feedback. To circumvent this, the MCP4725 DAC communicates to the Arduino over I²C protocol (figure B.2), sending a 10-bit analog value to the printer PCB.

The printer PCB then interprets the analog control signal and determines a scaling factor for the fire frequency of the Marlin stepper pulse interrupt as discussed in chapter 4. This scaling factor is also computed in the Arduino-based controller, in order to properly determine the set velocity for the printer and to compute the slip correctly. Extrusion occurs at a temperature of 200 $^{\circ}$ C.



Xeed actuation system

Figure B.1 : Communication layout of the setup.

86



Figure B.2 : Wiring schematic of the controller setup.

B.3 Visual material

This section contains some visual materials for the reader to familiarize him- or herself with the setup, and gain a sense of meaning with the diagrams already provided in this appendix. Figure B.3 shows the inside of the Leapfrog Xeed along with the print head used to conduct the experiments with. The printer was stripped bare of components not required for this research. Also, the left side of the head was not used.

Figure B.3 shows the inside of the Xeed 3D printer containing the modified print head. Figure B.4 shows a detailed view of the print head with annotations.



Figure B.3 : Full experimental setup, side view. This figure shows only the printer interior, head and sensor. Wiring, printer PCB, connections and the controller are not shown.



Figure B.4 : Annotated detail of the print head. Unmodified, the print head is symmetrical allowing for dual extrusion. Only the right side is used for this work and was modified with an encoder.

Appendix C

Feed-forward slip prevention compared to feedback control

The anti-slip controller as proposed in this thesis finds at which feed velocity the filament strand slips, and then oscillates the velocity around this point in order to maximize velocity while minimizing slip. Recalling the first figures describing system behaviour, we can roughly state, for the particular hot end used, that slip occurs from $\phi > 15 \text{ mm}^3/\text{s}$.

The critical reader may then ask: why not simply take an open-loop approach, set the velocity to this maximally feasible velocity, and thus gain maximum volumetric flow rate. In this case, the $\phi > 15 \text{ mm}^3$ bound was determined empirically, but could also be obtained by defining a maximally allowed slip percentage, and then make the controller learn the feasible maximum feed velocity. In order to test the effectiveness of such an open-loop approach, it was evaluated in the same fashion as the closed-loop control. The results are shown in figure C.1. In general, a trade-off between speed and accuracy may be observed. ESC offers lower TTC but generally has a larger error. Open-loop control is slower but will yield less error.

A number of drawbacks come with using open-loop control:

- We loose the ability of input tracking
- Less robustness to external disturbance such as filament impurities.
- No portability unless quick learning is possible

90APPENDIX C. FEED-FORWARD SLIP PREVENTION COMPARED TO FEEDBACK CONTROL

| Metric | Optimized open- | Schmitt trigger | Improvement | |
|--------------------------|-----------------|-----------------|-------------|-------------|
| | loop | | | |
| Mean slip | 0.13 | 0.21 | -56.2 % | (p < 0.001) |
| $RMSE (mm^3/s)$ | 2.48 | 5.9 | -138% | (p < 0.001) |
| TTC (s) | 73.2 | 69.6 | 4.6~% | (p < 0.001) |
| Extrusion error (mm^3) | 133.9 | 454.6 | -239~% | (p < 0.001) |

Table C.1 : Performance metrics for sawtooth input with statistical data.





Figure C.1

92APPENDIX C. FEED-FORWARD SLIP PREVENTION COMPARED TO FEEDBACK CONTROL

Appendix D

Digital appendix

This thesis is accompanied by a digital appendix. In general, the digital appendix contains

- 1. The Xeed printer firmware which runs the 3D printer,
- 2. All MATLAB code used for modelling, control and data analysis
- 3. All data files, along with analysis scripts for data interpretation and plotting
- 4. Firmware for the controller

The digital appendix may be found at this URL: http://bit.ly/2emfDTh