**Tool Fabrication for ABB IRB-6640 Robot**

Design, fabrication and development of 3D robotic printing tooling



**1.1 Fabrication Methods**

**1.1.1 Background**

Fused Deposition Methods has demonstrated its usability as a technology for small scale fabrication. Variations of normal fused deposition methods that consists in a layer by layer approach are those consisting in extruding filaments in the space, in which the plastic is molten and solidified in the air connected normally by nodes. This approach is an alternative that potentially permits the building of lighter structures at a reduced cost and time. Examples of this approach have been primarily studied at the ETH, Gramazio Kohler Research or at The Bartlett School of Architecture, UCL.

This thesis explores the fabrication through this method, using different type of materials, such as PLA, PLA +, PETG or ABS. Thermoplastics are commonly used in this method, as they require minimal heating energy in order to melt. (FIGURE XX TABLE OF TEMPERATURES)

ABS (230-250C)

ABS+ (220 - 260C)

PLA (180-250C)

PLA+ (205-260C)

PETG (230-250C)

PVA (180-210C)

**1.1.2 Spatial extrusion**

Although some large scale length tests have been tested, for this project, short rectilinear segments are chosen in order to design and fabricate. The process starts by setting the temperature of the nozzle at a temperature that melts the filament. The robotic arm moves the nozzle following a given toolpath. While this technique allows a high degree of freedom of movement, there are some constrains. For instance, printing vertical from top to bottom is not possible as the nozzle would collide with the filament printed. This would be different when printing upside down over existing infrastructures. The direction and taper of angles are directly constrained by the angle of the nozzle. For the agent Conductor algorithm, the toolpath follows a 2D version of the octet-truss patented by Buckminsterfuller in 1961 (Ashby, Deshpande and Fleck, 2001). This is a triangle-pattern that forms layers of trusses. In order to add a higher structural strength and continuity at the nodes, the triangle is broken at its vertices to short horizontal lines that provide with more support for upper layers to connect with. The logic of the pattern would be the following: lower horizontal segment – diagonally upwards segment – horizontal higher segment – diagonally downwards segment.

**1.1.3 Temperature**

One of the critical factors that intercede in a consistent extrusion is the control of temperature. For this, a Proportional – integral – derivative controller (or three term controller) is used to get a desired temperature. This is a widely used mechanism used in industrial control systems. A PID controller continuously calculates an error value e(t) as the difference between a desired setpoint (SP) and a measured process variable (PV) and applies a correction based on proportional, integral, and derivative terms.

This research has explored two different methods that are used as PID controllers. The first one is a standardized commercial PID, bought in Amazon. Normally, this type of controllers uses a thermocouple sensor (type K) to measure the temperature of the nozzle. The second and most used approach in this research takes an Arduino Uno with a PID algorithm coded in it. Instead of a thermocouple, this approach works better with Thermistors.

The signal that the Arduino outputs is converted through a metal-oxide-semiconductor field-effect transistor (MOSFET), which is a type of field-effect transistor (FET). It has an insulated gate, whose voltage determines the conductivity of the device. In order words, it is used as a gate that controls the amount of power (Watts) that is loaded at the output wire. Other mechanism can be used, such as a solid-state relay (SSR). This is an electronic switching device that has no moving parts (it does not wear out, to the opposite of electromechanical relays). Using a SSR, the control signal must be coupled to the controlled circuit in a way that provides galvanic isolation between the two circuits. The type of SSR used in this research utilizes optical coupling. When it switches on a photosensitive diode turns on a back-to-back MOSFET to switch the load. Both methods have been tested, getting more stable results using the solid-state relay.

This load is then charged on a 0.64mm Nichrome wire, which in order to work at 60W, it has to have a length of 800mm, having a resistance of 3.5 ohms, enough to achieve temperatures up to ~300C. This wire wraps a barrel in its total length to ensure a steady distribution of heat.

Another type of metal has been tested, in this case, 0.33 mm thickness stainless steel wire. For the same resistance, a 1500mm length was needed.

**1.1.4 Feeding system**

Another crucial system in order to have consistency while printing is the motor that feeds the filament into the nozzle. There is a large array of possibilities. This research explores one of the easiest, familiar and hackable approaches. A regular 1.75mm diameter 3D printer MK8 stepper motor is hacked so it can work for 3mm diameter filaments. By drilling the barrel with a slightly bigger drill bit, we are able to remove the Teflon tube and break the bottleneck shape that the barrel has at one of its edges.

The flow rate the stepper provides must be in sync with the motion of the robotic arm. The toolpath segments must correspond to a different feeding speeds depending on the direction of the toolpath. For instance, if a horizontal segment is being printed, and it represents the connection with the base or prior printed segments, the motion of the robot should decrease while the ratio steps/time of the stepper motor should increase. When printing vertically, the motion of the robot should be slightly bigger than the feed rate, as it would create tension in the segment, avoiding sagging effects. When printing the end of a vertical or diagonal segment, the robot must wait to provide with enough time cooling the filament in order to continue. Depending on the situation, the stepper must continue feeding the filament at a small rate, or should stop the extrusion (if the current node is the end of a continuous segment or the following segment is shorter than the previous one).

**1.1.5 Solidifying filament**

The third crucial aspect during the fabrication process is the transition of molten to rigid state. A cooling system is needed to be able to make this process happen at the same time as the printing is happening. For this system, we use the pneumatic system the robot has internally to inject air at a desired pressure. This variable becomes really important to master as minor changes produce major consequences and therefore, undesired geometries that has little fidelity with the design process. Many iterations have been analyzed and prototyped. A four tube air injection is chosen, at 30 mm distance from the TCP (Tool Center Point).

**1.1.6 Tool Orientation**

The tool is taught in ways that the Z elongation of the tool definition is perpendicular to the ground, being the ground the Work Object base, normally, with positive Z pointing up, following the right hand rule.

For large-scale prototypes and test, the tool has been inclined generally when working with more than one work object, and normally when the tip of the nozzle had to touch ground conditions, avoiding self-intersections.

Although for each of the three case studies, the extruder has always been positioned perpendicular to the work object, in parallel ongoing study cases, the tool is being tested tilted.

**1.2 Machine exploration**

Understanding the logic and mechanism before designing and fabricating a tool is efficiently effective. During much part of the beginning of the research, the acquisition of smaller scale 3D printing machines provided with enough knowledge for future fabrication steps. Prior researches have demonstrated that hacking regular 3D printers can provide with a faster printing process, speeding up the fabrication time up to 10 times compared to traditional layer-based printing (Mueller, Im et al, 2014).

All of the testing done to learn time and speed ratios were printed on Hatchbox Alpha 3D printer, a printer that activates the nozzle using six vertically actuated arms. This printer became an essential tool for the research, as all the nozzle protectors of the robotic tool are printed using ABS in this machine.

The natural next step was the acquisition of a 3D pen. Some wireframe structures were tested and prototyped to analyse flow rates, anchoring points over printed nodes and motion rates (this is not remarkably accurate as moving the hand a constant pace during the extent of the test is not natural). Also, the control of straight segments was of a high value to later apply in the robot’s toolpath process. Besides printing tests, opening the 3D pen was utterly useful to understand the mechanics of this process.

**1.3 Electro mechanic components**

**1.3.1 A4988 Micro stepper motor**

An Arduino Uno controls the feeding system. As described above, an MK-8 commercial stepper motor was used to feed the nozzle. Between the Arduino code that send different signals that are translated into on, off, different ratio steps/time, and choosing the direction of the steps (counter-clock wise, or clock wise), we use an A4988 micro-stepping driver for controlling bipolar stepper motors, and has a built-in translator for easy operation. This means that it is possible to control the stepper motor with 2 pins from this controller, one manages rotation direction and the other controls the steps.

The driver has five different step resolutions: full step, half, quarter, eight and sixteenth step. Also, it has a potentiometer for adjusting the current output, over-temperature thermal shutdown and crossover-current protection.

Its logic voltage is from 3 to 5.5V and the maximum current per phase is 2A if good addition cooling is provided or 1A continuous current per phase without heat sink or cooling.

The eight-step resolution has demonstrated to work more accurately to the results we were looking for. The control of the feeding system is a negotiation between the driver resolution and the delay on the steps loop in the code of the Arduino board.

**1.3.2 Hot End**

The melting point of the print media is compounded by a print nozzle and a heater mounted in an aluminium block. The hot end consists in regular commercialized aluminium barrels drilled with a slightly bigger drill bit than the orifice so it removes the Teflon tube that lives inside and breaks the capped end of the barrel. This allows to have a diameter of ~3mm, enough to make the 2.85mm filament pass through. The barrel is wrapped in Nichrome or stainless steel wire, with such diameter that provides with 12VDC 60W. This is enough to keep the nozzle to a melting temperature within the range of type of thermoplastics tested (PLA, PLA+, ABS, PETG). To make a longer nozzle, the join of commercial barrels is needed through a nut. This has the downside of not having the wire wrapped at the areas where the nut threads the barrel. This could be failsafed by wrapping the nut, although it won’t provide with the same temperature at this area.

Isolating the barrel is needed in order to keep the temperature stable. Due to the fact that the cooling system’s hoses point at the end of the nozzle with enough pressure to drop the temperature at the totality of the length of the barrel, it needs to be safely wrapped by kapton tape (polyimide tape that supports high temperatures), and heating block cotton made from heat-resistant ceramic fiber. Last step is to protect the insulation layers from the cooling system with a windscreen. This windscreen has the support to connect the 4 teflon tubes that provide cool air coming from the pneumatic system. Numerous iterations have been done in this last apparatus, all of them, printed in regular 3D printers using ABS thermoplastic, as it supports higher temperatures than PLA. Results have shown that the distance between the air hoses and tip of the nozzle are best between 250mm to 500mm.

**1.3.3 Temperature measurement**

Adjacent to the barrel wrapped with Nichrome wire, a temperature sensor needs to be attached. Same Kapton tape used to isolate and to set the heating wire, is used to stick a Thermistor. This is a resistor that changes value (non-linearly) based on the temperature. The type of thermistor used is an NTC or *negative temperature coefficient.* NTC thermistors decreases resistance as temperature rises.Due to the fact that microcontrollers do not have a resistance-meter built in, but a voltage reader (known as an analog-digital-converter), converting resistance into a voltage is required. Adding another resistor and connecting them in series will work, so when the resistance changes, the voltage changes too, according to a simple voltage-divider equation. Keeping one resistor fixed is recommended.

For instance, if the fixed resistor is 10k and the variable resistor is R, the voltage output (Vo) is:

**Vo = R / (R + 10K) \* Vcc**

Where **Vcc** is the power supply voltage (3.3V or 5V)

It is connected to a microcontroller. When measuring a voltage (**Vi**) into an Arduino ADC, the result is a float number.

**ADC value = Vi \* 1023 / Varef**

Combining the two (**Vo** = **Vi**):

ADC value = **R / (R + 10K) \* Vcc \* 1023 / Varef**

If Vcc (logic voltage) is the same as the ARef, analog reference voltage, the values cancel out.

ADC value = **R / (R + 10K) \* 1023**

Finally, **R** (the unknown resistance) is the result:

**R = 10K / (1023/ADC - 1)**

However, Arduino boards are naturally noisy, and interferences may vary the result, making it not scientifically accurate. An implemented solution is to use the 3.3V voltage pin as an analog reference. The 5V power supply comes from a computer’s USB, making the signal noisier (as more than one task in parallel is being sent) than the 3.3V power supply (it goes through a secondary filter or regulator stage). An alternative is to take more readings and average them. This is especially useful as some readings fluctuate to some peak readings, out of a natural range. The effects are diminished with more readings.

In order to convert resistance to temperature, a simplified B parameter equation of the Steinhart-Hart equation is used.



Where:

To: 25C = 298.15 K (room temperature),

B: 3950 (Coefficient of thermistor),

Ro: 10Kohm (Resistance at room temp)



Figure 1. ABS 3D printed nozzle windshield



Figure 2. ABS 3D printed nozzle windshield



Figure 3. ABS 3D printed nozzle windshield



Figure 4. ABS 3D printed nozzle windshield

**1.3.5 Robot-Arduino communication**

In order to coordinate and synchronize the functions of the robot and the Arduino, a communication between them is required. This is done through the DSQC 651, which is a circuit board normally mounted inside the robot controller, although it could also be mounted in an external I/O module. The combi I/O unit handles digital and analog communication between the robot system and any external systems (more information can be found in the 3HAC020676 ABB IRB6400TR official manuals).

I/O stands for Inputs/Outputs. Given the time for this research, only digital inputs are used, in other words, the communication only happens in one direction, from the robot to the Arduino. The wiring is made through an ATI QC-110 plate. Particularly, the interchange of information between both machines is directly related to the geometrical information at both scales, meso and micro scale. The robot is drawing the geometry, therefore, it has information about the location of each target. This means that depending on the direction of the printing path, the stepper motor should act accordingly. For instance, if it is printing upwards, the velocity of the stepper motor should be slightly slower than the motion of the robot to tense the segment. On the contrary, printing downwards might require quicker steps.

The robot emits electrical pulses with different lengths that are related to the target position, thus, to the geometrical pattern. The Arduino then receives this pulse, and interprets it accordingly, actuating the motor. Is critical to work with ranges instead of fixed values, as naturally, this communication is noisy enough to make the transmission not legible. This research works within ranges of 10000 microseconds. Among the totality of instructions, there are: slow extrusion, quick extrusion, stop or reverse.

**1.3.6 Algorithms for fabrication**

A tectonic design approach knowing the affordances and limitations of the material along with a tool design approach, which takes into account the limitations of the fabricating tool are recommended when using this technique. However, anticipating constrains does not ensure a clean and failsafed fabrication process. The feedback loop between design and fabrication processes actuates in order to prune unforeseen problems. The main challenge in spatial printing is that the nozzle has to respect already printed material in order to prevent collisions. About half of the algorithm code is definitions that deal with fabrication constrains. For instance, in the algorithm used for case study III, a detector for collisions should be checked between floors. In order to be efficient, instead of particular checks, simply an offset in the Z direction, higher than the height of the pattern, is given at the first and last points. This ensures that we never collide with already printed geometries during travelings.

Figure 5. Toolpath without avoidCollision definition

Figure 6. Toolpath without avoidCollision definition

At a micro-scale (pattern scale), another consideration needs to be taken. Same logic as the example above is followed. However, in order to be efficient, this is taken at the time of the design of the pattern, so future feedback loops are avoided. The taper of the nozzle establishes the first constrain when designing a pattern. For instance, two vertical segments must have a separation that equals the width of the nozzle. Another example, while it is always possible to print upwards or horizontal (regardless direction, angles), printing downwards become impossible when the slant of the print is steeper than the taper of the nozzle.



Figure 7. Toolpath without avoidCollision definition