



**HAL**  
open science

## Slicer and process improvements for open-source GMAW-based metal 3-D printing

Yuenyong Nilsiam, Paul Sanders, Joshua Pearce

► **To cite this version:**

Yuenyong Nilsiam, Paul Sanders, Joshua Pearce. Slicer and process improvements for open-source GMAW-based metal 3-D printing. *Additive Manufacturing*, 2017, 18, pp.110-120. 10.1016/j.addma.2017.10.007 . hal-02111397

**HAL Id: hal-02111397**

**<https://hal.science/hal-02111397>**

Submitted on 26 Apr 2019

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

## Slicer and Process Improvements for Open-source GMAW-based Metal 3-D Printing

Yuenyong Nilsiam<sup>1</sup>, Paul Sanders<sup>2</sup>, and Joshua M. Pearce<sup>1,2,\*</sup>

1. Department of Electrical & Computer Engineering, Michigan Technological University, Houghton MI
2. Department of Materials Science & Engineering, Michigan Technological University, Houghton MI

corresponding author:

1400 Townsend Drive

Houghton, MI 49931-1295

906-487-1466

[pearce@mtu.edu](mailto:pearce@mtu.edu)

### Abstract

Low-cost gas metal arc welding (GMAW)-based 3-D printing has proven effective at additive manufacturing steel and aluminum parts. Early success, however, was based on hand-writing G-code, which is inadequate for the majority of potential users. To enable automated slicing a 3-D model and generating G-code for an acceptable path for GMAW 3-D printing, this paper reports on upgrading of the free and open source *CuraEngine*. The new slicer, *MOSTMetalCura*, provides the following novel abilities necessary for GMAW 3-D printing: i) change the perimeter metric from width to track count, ii) avoid movement that overlaps previous weld beads, iii) have infill start immediately after the perimeter finished and in the direction that eliminates translations, iv) add a variable pause between layers to allow for substrate cooling, v) configure GPIO pins to turn on/off the welder, and vi) set optimized wire feed speed and voltage of the welder based on printing speed, layer height, filament diameter, and tool track width. The process for initiating these changes are detailed and the new slicer is used to help improve the function of the printer for ER70S-6 steel. To find the printing function with the smallest bead width based on volume of material, the line width, layer height, and printing speed are varied to provide wire feed speed calculated by *MOSTMetalCura*, then the settings are used to print 3-D models. The results of 3-D printing three case study objects of increasing geometric complexity using the process methodology improvements presented, which show resolution of 1mm bead widths.

**Keywords:** Open-source slicer, metal 3-D printing, low cost metal 3-D printer, open-source metal 3-D printer, GMAW 3-D printing

### 1. Introduction

Additive manufacturing with 3-D printing has matured beyond simple rapid prototyping [1-5] to small-batch production [6-8] and distributed manufacturing [9-13]. Some of the most industrially interesting 3-D printing is that of metals, which include laser sintering and melting [14-18] and electron beam melting [19-21]. These systems are mature and such industrial-grade additive manufacturing machines can be prohibitively expensive (e.g. > US\$500,000 - US\$1.5 million), which is beyond the reach of consumers and small and medium sized enterprises (SMEs) [22]. Recently progress has been

made in upgrading the popular open-source self-replicating rapid prototyper (RepRap) 3-D printer designs [23-25] into a low-cost open-source metal 3-D printer [26]. This metal 3-D printer uses a low-cost gas-metal arc welder as a fixed printer head and is controlled by an open-source Arduino micro-controller [27,28]. Opposite the motion of most polymer 3-D printers where the printer head moves, the stage of the metal 3-D printer holds a re-usable substrate and moves during the printing [29,30]. The power of the welder can also be monitored with open source hardware and software [31,32]. An open-source firmware called *Franklin* [33] is used to control the metal 3-D printer by translating G-code into controlling signals. G-code is a numerical control programming language that is commonly used for controlling automated machine tools and it can be manually written or generated by slicer programs. Low-cost GMAW-based 3-D printing has proven effective at both steel and aluminum [26, 34].

This early success was based largely on hand-coding G-code for relatively simple geometries. To be functional for a wide array of users, a slicer software must be capable of slicing a 3-D model into layers and then generating G-code for the path of each layer. Expensive metal 3-D printing systems come with their own proprietary slicers. Unfortunately, the free and open source slicers such as *Cura* [35] and *Slic3r* [36] are made primarily for polymer based 3-D printing. In order for a slicer program to support the open-source metal 3-D printer, some functions need to be added to existing polymer based slicers including: i) the ability to change the perimeter metric from width of the perimeter to track count because the line width is constant, ii) the ability to avoid movement that will run over the previously laid weld bead (polymer printers can handle this contact but this is not possible for metal as the solidified metal surface does not have the give of warm polymer layer), iii) have infill starts immediately where the last segment of the perimeter finished and in the direction that eliminates translations to reach previously unfilled areas, iv) add an option to pause between layers to allow for substrate cooling and ability to set the pause time, v) the capability to configure the General-Purpose Input/Output (GPIO) pins (to turn on and off the welder), and vi) set the optimized wire feed speed and voltage of the welder based on the printing speed, layer height, filament diameter, and the width tool track. To provide this new functionality for open-source metal 3-D printing, the open source *CuraEngine* [37], has been upgraded here to *MOSTMetalCura*. The process for doing this and the changes made are first detailed and then used to enable better control over the printer to help improve the function of the printer for steel using printing speed, the wire feed speed, and the voltage of the welder. The optimized function would result in an accurately printed part with high observational resolution (smallest bead width) and a smooth surface. To find the optimized function, the following experiments are done: the line width, layer height, and printing speed are varied to provide wire feed speed calculated by the slicer *MOSTMetalCura*, then the settings would be used to optimally print 3-D models. Other 3-D printing quality factors are assessed for optimization of the slicing algorithm as well including shield gas parameters, temperature and humidity. Even though, these factors are not used directly by the slicer, they are crucial for metal 3-D printing. The 3-D printing results of three case study objects of increasing geometric complexity are presented and discussed.

## 2. Background

Today, metal 3-D printing is a popular topic, there is a rapid growth of journal papers about metal 3-D printing in search results from Google Scholar [38]. Even with the high costs of commercial metal 3-D printers and their maintenance, many companies still pay the price due to their ability to

perform rapid prototyping [38,39]. A low-cost open-source metal 3-D printer will lower the barrier for the technology to be accessible to individuals and small and medium enterprises (SMEs). This will allow more people to fabricate customized 3-D objects and will be rapid growth and improvements by open-source communities around the world. The metal 3-D printer RepRap [26] is open-source hardware inspired by the *Rostock*, which is a *Deltabot RepRap* and is controlled by open-source software, *Franklin* [33]. *Franklin* is the control system for the low-cost metal 3-D printer which contains two parts. First, the firmware that controls the printer and exchanges information with the web-based server on a host computer in the same network. Second, the web interface that handles the G-code from a slicer then communicates with the *Franklin* Firmware, which can control the motors, temperature, and GPIO pins (see Figure 1). One of the GPIO pins is used in the low-cost metal 3-D printer to turn the welder on or off. Opposite from the original *RepRap* printers, the metal 3-D printer has the welder as the printer head fixed in a single position and the three-axis stage moving according to the printing path. Weld-based 3-D printing is relatively inexpensive and produces solid adhesion between layers with low porosity, but there are constraints in resolution and surface quality [29]. To solve the problem, gas metal arc welding (GMAW) is used for the low-cost metal 3-D printer.

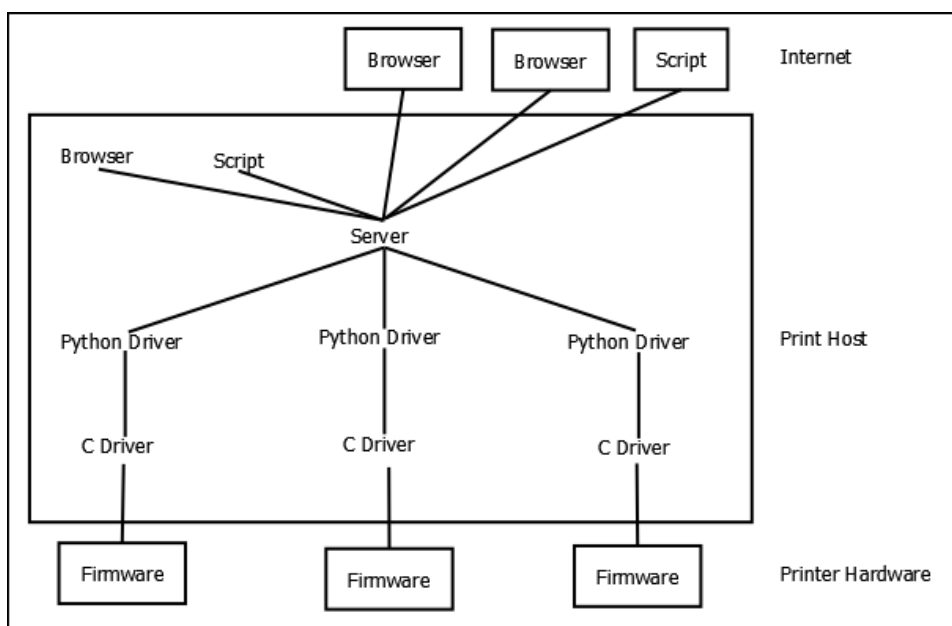


Figure 1. The architecture of *Franklin* and schematic of the workflow from users to the printer.

*Cura*, one of the most popular open source slicing programs, was developed by Daid (David Braam) and was released on the AGPLv3 license. Two main components of *Cura* are the GUI (Graphic User Interface) and the slicing engine called *CuraEngine* [40,41]. The GUI part is written in Python and features a 3-D model file viewer, the ability to send a .STL file to *CuraEngine* as well as receive G-code from, and connecting with a 3-D printer and sending the G-code to it. The *CuraEngine* is written in C++ and is the core for the slicing process. Working directly with the *CuraEngine* also gives us the capability to have more control over the slicing process through settings in the configuration file (*fdmprinter.json*).

There are several fundamental differences between 3-D printing with GMAW-based metal deposition and plastic deposition using FFF (which the existing open source program can support). Polymer based extruders enable retraction of the filament, which stops the flow of plastic on the work piece. This enables the printing of multiple closed surfaces that are not connected in each z step (e.g. the tines of a fork facing up). In addition, polymer based 3-D printers can tolerate the print head passing back over a printed area without further plastic deposition at the same z height. The freshly printed polymer layers provide only a small resistance to the moving print head even in the delta design where the head is attached only with magnetic bearings. On the other hand, GMAW-based metal printers, cause significant divergence from optimal print parameters in the initial arc, so ideally they print continually only pausing between layers if cooling is necessary. In addition, a solid steel workpiece that may have small z inhomogeneities is not conducive to letting the print pass back over printed lines while printing on the same z height. The work piece as well as the printer may be damaged. Thus, a toolpath from a slicer provided for a polymer 3-D printer (where these two constraints do not exist) is not acceptable for GMAW 3-D printing. An upgrade of existing open source slicers is necessary.

### 3. Methods

#### 3.1 Open Source Cura

Open-source software, such as *Cura*, provides access to all the source code and the freedom to modify it. The CuraEngine starting code was taken from *GitHub* with the version number 15.06 [37]. In order to modify the source code, the structure and the processes of the program need to be known and understood. The process flow of the *CuraEngine* can be found in Figure 2.

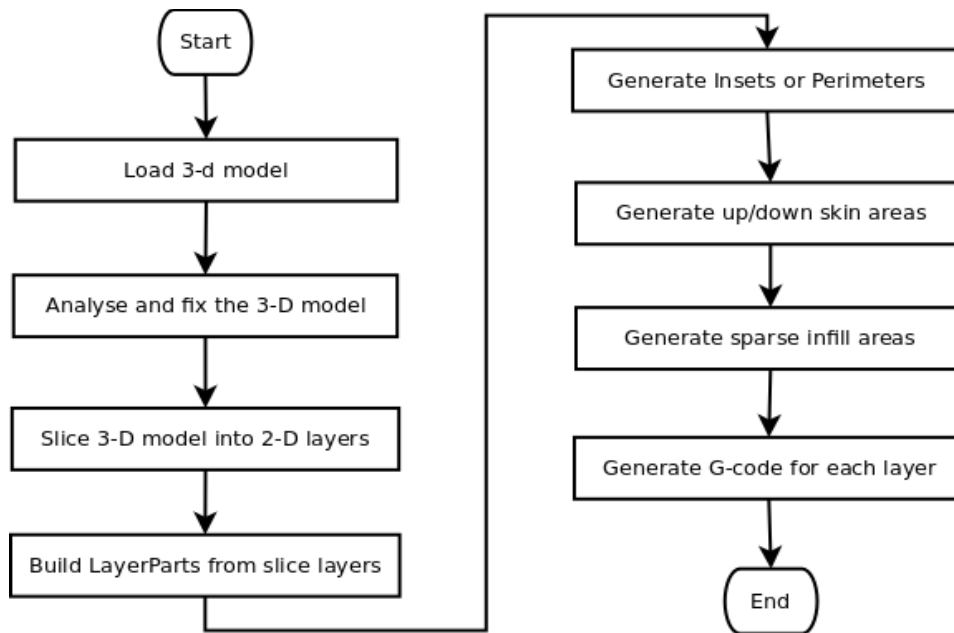


Figure 2. The process flow of *CuraEngine* (Ultimaker/CuraEngine, 2016)

*LayerParts* are isolated parts in a layer. For example, a simple table with 4 legs is sliced into layers. The lowest layer would have 4 parts in the layer (see Figure 3.). Skins are areas that suppose to be fully filled with material. Usually the top and bottom of a model would be fully filled. Infill are areas inside the model and either can be fully filled, partially filled or left empty. However, in metal 3-D printing both skin and infill are always fully filled.

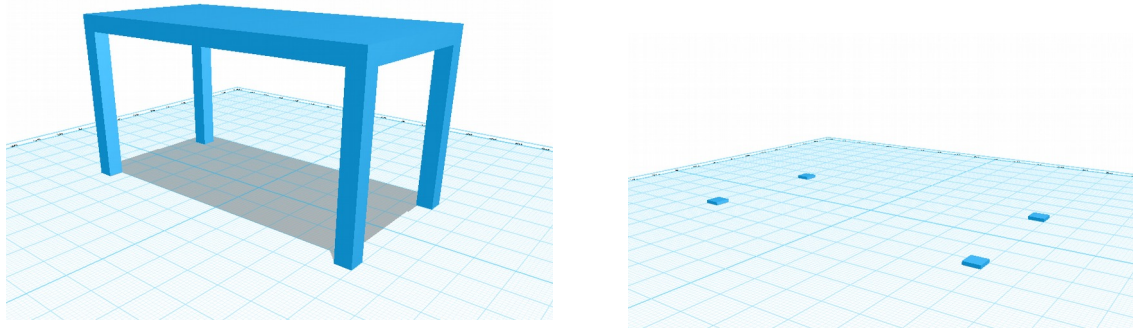


Figure 3. A table 3-D model and its *LayerParts* in the first layer.

### 3.2 Altering Cura

Working directly with *CuraEngine* provides more control compared to using *Cura* because more settings can be changed via the configuration file, *fdmprinter.json*. These changes are summarized in Table 1.

Table 1. Summary of new settings in *MOSTMetalCura*.

Setting	Purpose
<i>machine_metal_printing</i>	To toggle between polymer and metal slicing
<i>wall_line_count</i>	To set the perimeter metric
<i>top_bottom_pattern</i> and <i>fill_pattern</i> set to concentric	To avoid running over the previous laid weld bead and also to start the infill immediately after the last segment of the perimeter
<i>machine_welder_on_gcode</i> and <i>machine_welder_off_gcode</i>	To turn the welder on and off
<i>machine_min_dist_welder_off</i>	To prevent the welder from cycling on/off more than necessary

First a setting called *machine\_metal\_printing* is added to the configuration file. This setting is a boolean data type. If it is set to true, then the generated G-code would be appropriate for the open-source metal 3-D printer. If it is set to false, then the generated G-code would be the same as using



original *CuraEngine* without using other additional settings for MOST's metal 3-D printer. A boolean member called *isMetalPrinting* is added to class *gcodeExport* in order to keep the setting value of *machine\_metal\_printing*. Next, the perimeter metric can be changed from width to track count as the width is not controllable as in FFF of polymers by changing the setting named *wall\_line\_count* in the configuration file. This number would define how many track count for the perimeter. To avoid running over the previous laid weld bead and also to start the infill immediately after the last segment of the perimeter is done, the parameters *top\_bottom\_pattern* and *fill\_pattern* should be set to *Concentric*, so all printing would be in concentric pattern (see Figure 4). The concentric pattern would make the printing go around either form the outside to the inside or vice versa. The pattern is switched and repeated every other layer.

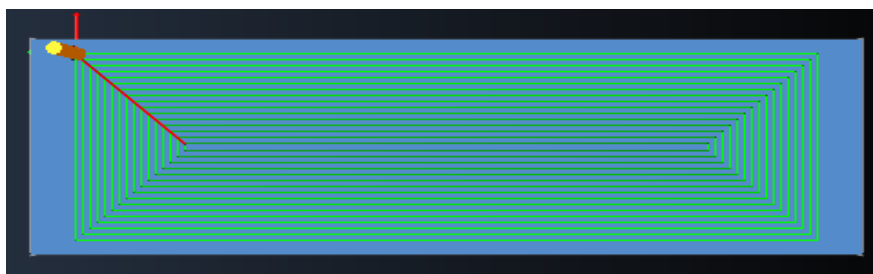


Figure 4. Concentric pattern of a layer of block (101.6 mm  $\times$  31.75 mm  $\times$  31.75 mm).

Next, for a GMAW metal 3-D printer, the welder can be turned on or off via G-code, so the settings *machine\_welder\_on\_gcode* and *machine\_welder\_off\_gcode* were added to the configuration file. The data type of these settings are string and the value of each setting is G-code commands to turn on and turn off welder through GPIO pin [33]. A default value of *machine\_welder\_on\_gcode* is *G4 P0\nM42 P1 S1\n*. *G4* is G-code command for the printer to be still doing nothing and *P0* means for zero millisecond. *\n* is newline command. *M42* is switching general purpose I/O pin command and *M42 P1 S1* means set pin 1 to value 1 [42]. A default value of *machine\_welder\_off\_gcode* is *G4 P0\nM42 P1 S0\n* and can be interpreted in the same way as previous example. A boolean member called *isWelding* is added to class *gcodeExport* to keep a status of the welder. Two string variables are added to the class, *welder\_on* to store setting value from *machine\_welder\_on\_gcode* and *welder\_off* to store setting value from *machine\_welder\_off\_gcode*. In function *writeMove()* of *gcodeExport* class, if *extrusion\_mm3\_per\_mm* is greater than 0.000001, *isMetalPrinting* is true, and *isWelding* is not true, then the *welder\_on* would be inserted into the generated G-code. If *extrusion\_mm3\_per\_mm* is lesser than 0.000001, then it is only a traveling without printing and the *welder\_off* would be inserted into the generated G-code.

In order to set the values of *welder\_on* and *welder\_off*, two methods or functions called *setWelderOn()* and *setWelderOff()* were added to *gcodeExport* class according to the object-oriented programming concept. These methods would be the way to set the values of the attributes of *gcodeExport* object or class. *fffProcessor* class reads values of *machine\_welder\_on\_gcode* and *machine\_welder\_off\_gcode* from the configuration file then assigns those values to the attributes *welder\_on* and *welder\_off* of *gcodeExport* class through the methods, *setWelderOn()* and *setWelderOff()*. This is how objects interacts with each other. In the same way, method

*setIsMetalPrinting()* and *setIsWelding()* are for setting values of *isMetalPrinting* and *isWelding* attributes. These methods are implemented in *gcodeExport.cpp*. The initial value of *isWelding* attribute is set to false.

When a movement of the substrate is for traveling, not printing or welding, actually the welder should be turned off. However, to avoid turning off and on the welder unnecessarily, the setting named *machine\_min\_dist\_welder\_off* is added to the configuration file. The data type of the setting is double type and the unit is in millimeter. If a traveling distance is less than the setting value (10 millimeters by default), then it will travel without turning the welder off. A double data type member called *min\_dist\_welder\_off* is added to *gcodeExport* class to store value from *machine\_min\_dist\_welder\_off* setting. In function *writeMove()* of *gcodeExport* class, before *welder\_off* will be inserted into the generated G-code, the travel distance has to be greater than *min\_dist\_welder\_off* value. The method *setMinDistWelderOff()* was added to *gcodeExport* class, so other classes can set the value of attribute *min\_dist\_welder\_off*.

The pause between layers for substrate cooling is added as an option in the configuration file. The parameter named *machine\_layer\_pause* is a boolean data type. If it is set to true, then the *machine\_layer\_pause\_gcode*, *machine\_layer\_pause\_time*, and *machine\_layer\_increase* values would be inserted in a generated G-code between each layer. The *machine\_layer\_pause\_gcode* is added to the configuration file as well and would be used only if the value of *machine\_layer\_pause* is true. In function *writeGCode()* of *fffProcessor* class, if *machine\_layer\_pause* value is true, then at the end of each layer the *machine\_welder\_off\_gcode* and *machine\_layer\_pause\_gcode* values would be added to the generated G-code. A default value of *machine\_layer\_pause\_gcode* is *G4 P*, of *machine\_layer\_pause\_time* is 60000, and of *machine\_layer\_pause\_increase* is 20 which tells the printer to do nothing for 60,000 milliseconds and increase this pause time by 20 percent every layer. Similarly, future work could utilize this flexibility in the code to use IR sensor feedback to begin a new layer when the part has reached an acceptably low temperature.

In addition, there are some functions (or options) that are not available for GMAW-based 3-D metal printers, so there is no need to include those G-code commands. In function *writeTemperatureCommand()* of *gcodeExport* class, before inserting G-code commands about extruder temperature, the checking of metal printing would be checked. If it is the metal printing, then those extruder temperature G-code commands would not be inserted.

Finally, the version name string is defined in *settings* class at the top of the file. So it was changed to *MOSTMetalCura*. In *CmakeLists.txt*, the name of the software and library links are set in that file, so they were changed from *CuraEngine* to *MOSTMetalCura*. At the top of a generated G-code file, this version text would be added as a comment. Likewise, important information are added as comments after the version text which include line width, layer height, printing speed, material diameter, material volume per second, recommended voltage and wire feed speed for the welder (Millermatic 190 MIG welder). This information was added in the function *writeGCode()* of *fffProcessor* class.

### 3.3 GMAW 3-D Printing with MOSTMetalCura

The new *MOSTMetalCura* slicing program is tested on a *RepRap* GMAW-based 3-D printer previously described (Nilsiam, et al., 2015). Based on the previous works [26,29], ER70S-6 steel



welding wire with 0.024 inches or 0.6 mm diameter was used in the experiment. The welder was Millermatic 190 MIG welder. To optimize the wire feed speed, the expected and actual volume of material are calculated, so that they are matched. The expected volume of steel can be calculated in Equation (1) and the actual volume that coming out from the welder is calculated by Equation (2).

$$E_v = L_w \times L_h \times P_s \quad (1)$$

Where  $E_v$  is expected volume in cubic millimeters per second ( $\text{mm}^3/\text{s}$ ).  $L_w$  is the welding bead width in mm,  $L_h$  is the height of each layer in mm, and  $P_s$  is the printing speed or the movement speed of the substrate in mm per second in this case.

$$C_{aw} = \pi \times \left( \frac{D_w}{2} \right)^2$$

$$W_s = \frac{(202.3 - 25.0)}{(100 - 20)} \times W_n - 19 = (2.216 \times W_n) - 19 \quad (2)$$

$$A_v = C_{aw} \times W_s$$

Where  $C_{aw}$  is the cross-sectional area of the wire in square millimeters ( $\text{mm}^2$ ).  $D_w$  is the diameter of the steel welding wire in mm.  $W_s$  is the actual wire feed speed in millimeters per second ( $\text{mm}/\text{s}$ ) and  $W_n$  is a unitless of the wire speed setting on the welder.  $A_v$  is the actual volume of steel wire from the welder in cubic millimeters per second ( $\text{mm}^3/\text{s}$ ). According to Miller website [43], the wire feed speed is between 60 to 600 inches per minute. On a scale from 10 to 100 of the wire speed setting, each for 10 seconds of running and the measurement of the wire feed rate can be shown in Figure 5. The measurement of the wire length at scale 10 is discarded because it is given the same speed as scale 20 and the speed is only changed after scale 20. Then the  $W_s$  is derived from the length values. The optimized wire feed speed is found if an expected volume equals to an actual volume and the calculation is done by *MOSTMetalCura*.

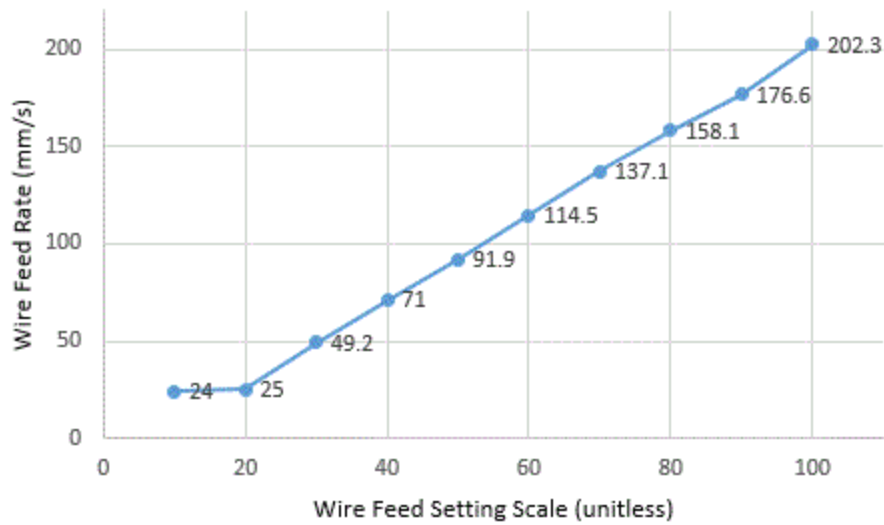


Figure 5. The wire feed rate for each scale setting.

To test this approach the 3-D prints were made with the following 3-D models, demonstrating increased geometric complexity of extruded shapes (see Figure 6-8). The simple block (101.6 mm × 31.75 mm × 31.75 mm), chisel (140 mm × 20 mm × 20 mm) and gear (60 mm diameter × 10 mm) were used because their geometries do not include any bridges or overhangs. All STL files for these 3-D models are available [44].

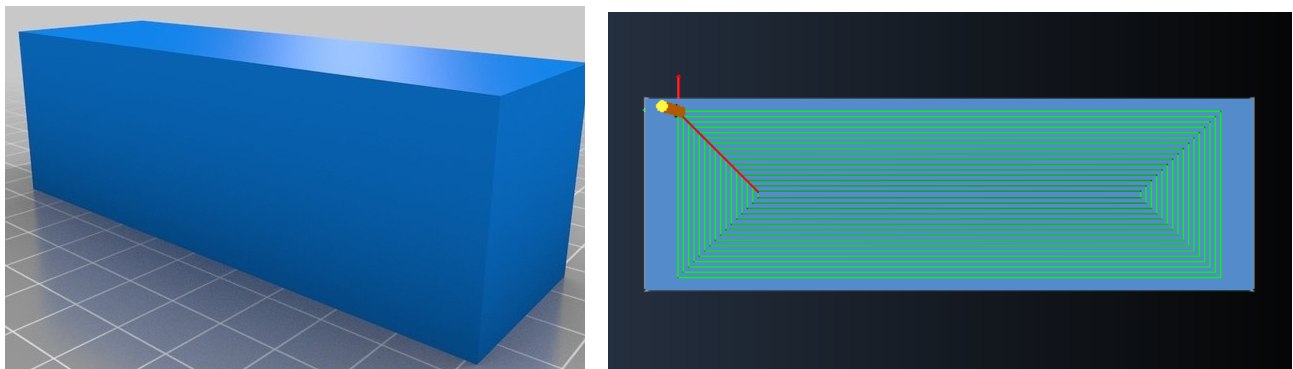


Figure 6. a) 3-D model of block (grid lines are every 10mm) and b) a slice showing the path for 3-D printing using *MOSTMetalCura*.

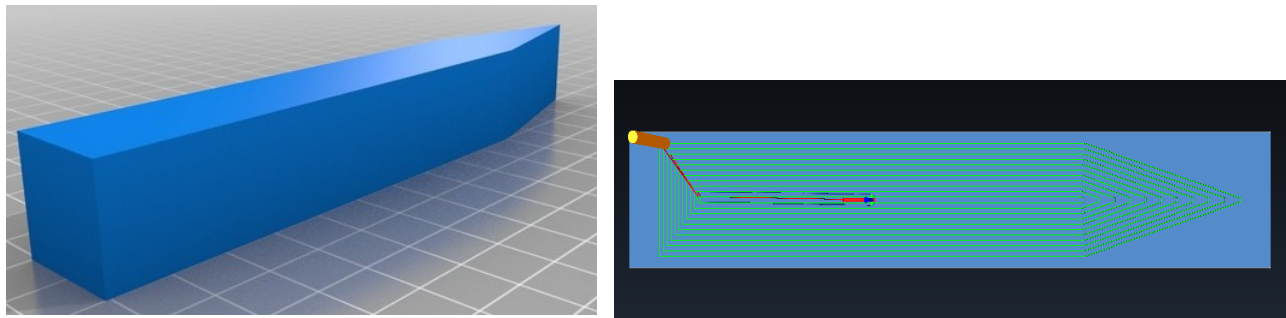


Figure 7. a) 3-D model of chisel (grid lines are every 10mm) and b) a slice showing the path for 3-D printing using *MOSTMetalCura*.

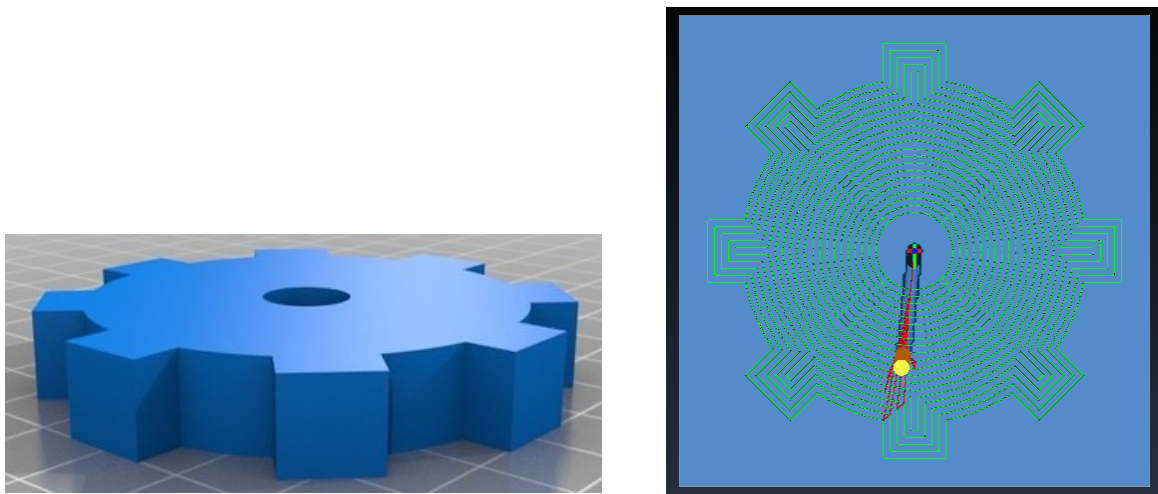


Figure 8. a) 3-D model of gear (grid lines are every 10mm) and b) a slice showing the path for 3-D printing using *MOSTMetalCura*.

All experiments are conducted using Millimatic 190 MIG welder. The distance between the nozzle and the substrate should be around 8 to 10 millimeters. The use of RC25 shield gas, which is a mix of 25% carbon dioxide ( $\text{CO}_2$ ) and 75% argon (Ar), would be tested at different level of pressures (20-60 CFH). Temperature and humidity would be measured using digital temperature and humidity monitor ( $\pm 0.05$  Celsius and  $\pm 0.5$  Percent) and assessed. When the printing speed is set in the configuration file, then the wire feed speed and voltage setting would be calculated and put in comment of a generated G-code file by the *MOSTMetalCura*. The ER70S-6 steel welding wire with 0.024 inches or 6 millimeters diameter is used in all experiments. According to preliminary experiments [45], the recommended voltage setting for the welder is 5 volts. The quality of the 3-D printed objects were quantified with digital calipers ( $\pm 0.1$  mm).

#### 4. Results and Discussion

The output of the slicer program is a G-code file. The header would contain important information as in the following text:

```
;Generated with Cura_SteamEngine MOSTMetalCura
;////////////////////////////////////
;Infill line width: 0.98 mm.
;Layer height: 1 mm.
;Printing speed: 7 mm/s
;Material diameter: 0.6 mm.
;Expected Material: 6.86 mm3/s.
;////////////////////////////////////
;Recommended welder (Millermatic 190) settings
;Voltage: 5 volts
;Wire speed: 19.61 - 21.61
;////////////////////////////////////
```

These are comments in the G-code file which gives information about line width, layer height, printing speed, material diameter, expected material in mm<sup>3</sup> per second, and the recommended voltage and wire feed speed settings for the welder, specifically for Millermatic 190 MIG welder.

The line width is set in the configuration file and is set close to one millimeter, but a little smaller or a little bigger line width,  $\pm 0.02$  range, might help fit all paths in one layer better. The paths can be viewed by one of the G-code viewer, such as the open source CAMotics [46].

The settings for GMAW 3-D printing which were evaluated here are line width, printing speed, voltage and wire feed speed. These process variables are interdependent, which makes overarching process rules difficult to derive. The approach taken here is to obtain a set of reasonable process parameters that results in successful print and then to systematically vary one of the variables at a time up and down to find an optimum while observing and identifying a physical reason for the increased or decreased print quality with that variation.

With the setting of 1 millimeter layer height, 0.98 millimeter bead width, and the printing speed of 7 millimeters per second, the calculated wire feed scale is 25.3 mm/s. The printed results of the gear model were observationally poor (see Figure 9). The height of the printed dots was measured by digital caliper to about 2 millimeters ( $\pm 0.1$ ). However, the diameter of the printed gear was measured to about 60 millimeters as expected. According to these results, the layer height was change to 2 millimeters and the rest of the parameters remain the same, the new calculated wire feed scale was about 30 and the printed results were significantly improved over Figure 9 as shown in Figure 10.



Figure 9. The result of printed gear (60 mm diameter) with 1 mm layer height.



Figure 10. The result of printed gear (60 mm diameter) with 2 mm layer height.

A problem created with higher speed is it results in a rough printed surface for higher layers. The printing speed at 10 mm per second causes a surface non-uniformity after the third layer (see Figure 11). Another problem is that it is not easy to set wire speed on the welder to suit every printing speed. For example, if the printing speed is changed to 8 mm per second, then the wire speed number would need to be around 34.1 which is hard to set it to be the same every time because the wire feed speed setting on the welder does not provide scale details between the tens (10-100). It should be noted that users can add a finer physical scale to the welder to more accurately set the wire speed.

The default configuration of slicing for plastic 3-D printing is using *Lines* pattern which would cause more turning off the welder and traveling in each layer for metal 3-D printing. In the default slicing, there is no pause between layers, so the heat would be accumulated as the printing is on going and that would cause an unexpected shape and poor quality of surface.



Figure 11. Poor surface quality after third layer of printing speed 10 mm/s on block (101.6 mm × 31.75 mm).

When the optimization of settings were found after systematically varying line width, printing speed, voltage and wire feed speed: a layer height of 2 millimeters, line width of 1 millimeter, printing speed of 7 millimeters per second, voltage of 5 volts, and wire feed scale of 30. The steel 3-D printed of a few layers of block, chisel and gear are shown in Figure 12-14. The few layers of printed objects are shown here to demonstrate visual beads/ line widths. The finished 3-D printed gear is shown in Figure 15 and the machined gear is shown in Figure 16.





Figure 12. 3 layers of 3-D printed block (101.6 mm × 31.75 mm).



Figure 13. 3 layers of 3-D printed chisel (140 mm × 20 mm × 20 mm).

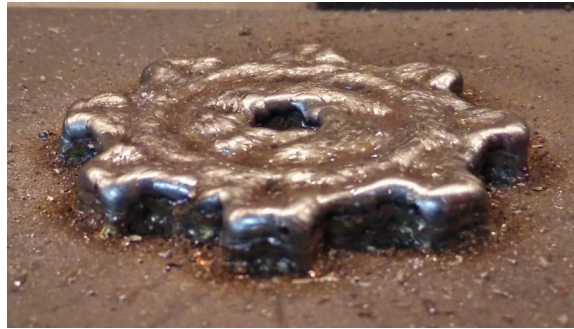


Figure 14. 3 layers of 3-D printed gear (60 mm diameter).



Figure 15. The finished 3-D printed gear (60 mm diameter) (5 layers).



Figure 16. The finished machined gear (60 mm diameter).

From Figure 14 and 15, the surface of the printed part gets rougher as the layer number is increased, because of the substrate effect. Specifically, any imperfection on the lower layer will be continuously propagated in all the following layers often this imperfection is aggravated. This still enables near net shape printing as seen in Figure 16, which shows the gear after machining to be functional.

Several issues can arise in GMAW-based 3-D printing, which can be overcome with understanding of the shield gas settings, printer geometries, and heat loss from the substrates as shown below.

The shield gas flow rate was found to be optimized at 30 CFH (cubic foot per hour). If the shield gas flow rate is lower than optimal it will affect the quality of welding bead and adhesion due to not enough shield gas to cover the welding area from exposure to oxygen and nitrogen in the air. On the other hand, if the shield gas flow rate is higher than optimal it will cause a rough surface of the printed part because the melting metal can be blown away from the intended location [47]. The correct shield gas flow rate will help with spatter control, heat control, and adhesion quality [48]. The weld gun needs to be perpendicular to the substrate, so the shield gas can cover around the welding area. If the weld gun is inclined to one side of the nozzle, then some of shield gas will be blocked by the nozzle and cause brown oxide problem on one side of the printed part as shown in Figure 17. This error can be caused by the weight of the cord that connected to the weld gun pulling the weld gun down to lean to one side of the nozzle. In order to fix this problem, a mechanical support was added to hold the cord of the weld gun or a more rigid gun holding assembly can be used.

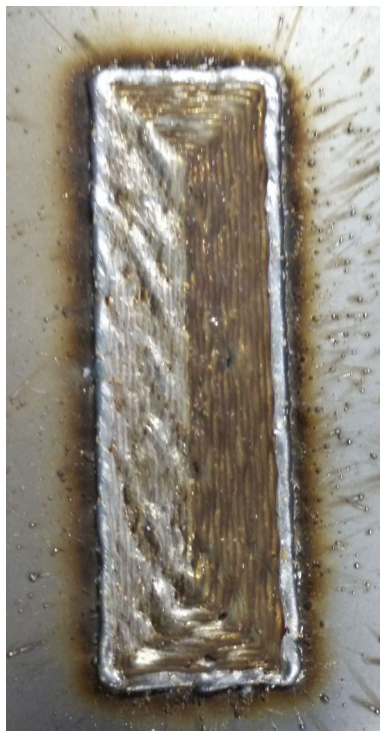


Figure 17. With optimal settings, brown oxide due to one side nozzle blocking shield gas (101.6 mm × 31.75 mm).

Brown oxidation during the printing can also occur from high humidity printing environments, leaking gas lines, or old gas tanks. Dehumidification of the printing room, keeping the room at certain and suitable temperature (25 degree Celsius), non-porous gas lines and new RC25 gas tanks can reduce oxidation to next to nothing.

Substrate deformation can also be a problem if heat is built up during printing causing an imbalanced thermal distribution as shown in Figure 18. When such a imbalance in thermal distribution of a substrate causes bending of the substrate during GMAW 3-D printing, it can deform the printed part and in extreme cases destroy the print. To help reduce the heat imbalance in the substrate, the aluminum plate (thickness of 0.249 inches or 6.325 mm) was placed on the cement board under the substrate. The results of experiments show that adding aluminum plate helped reduce and eliminate substrate deformation. However, the larger the part being printed, the longer the delay in-between layers is needed to dissipate heat built up (and thus imbalanced thermal distribution) in the part from GMAW deposition.



Figure 18. Thermal induced substrate deformation which is indicated by the angle symbol (101.6 mm × 31.75 mm). When corrected the substrate remains flat.

If the temperature of the part is too high during printing it will increase the surface roughness of the next layer. As more layers are printed, the heat transfer from the printed part will become slower, so the pause after the layer needs to be longer. The heat is trapped in the center of the printed part, so the effectiveness of the aluminum plate is reduced. In *MOSTMetalCura*, the pause time can be set in the configuration file, but the needed pause time is variable and dependent on the size of a printed part and is not known for the first time of printing and object. So the pause time between layers was set to 60 seconds in *MOSTMetalCura*, then it was manually paused and resumed in *Franklin* between layers. During the pause time, the nozzle can be checked and cleaned to make sure it is not clogged with spatter that can block the shield gas. Future work could use thermal modeling of any part geometry and material selection to automatically calculate the time needed for pausing.

According to the fact that the wire feed speed cannot be automatically controlled by the metal 3-D printer due to the use of a low-cost welder, the fixed width tool track or fixed bead width is used for metal 3-D printing. The change of the speed can change the width of the bead, but the shape of the



bead cannot be controlled to be repeatable. To be able to print the wanted shape the bead width is minimized, which in these experiments was found to be about 1 millimeter. However, the smaller the bead width the more tool tracks are necessary to cover a given area. In addition, to increasing print time this also adds more heat to the part during the welding process. Thus, printing large models is still challenging due to the heat problem inside a printed part when the surface and volume ratio becomes less and less as more layers added. A 3-D steel printed part keeps the heat inside itself and slowly releases the heat. It is much slower compared to the heat release rate of the carbon steel substrate that is on an aluminum plate. It takes about 5 minutes for a substrate to release the heat, but it took about 20 minutes for a layer of the printed block to cool to about 30°C. Future work could utilize an active cooling system to reduce the pause time between layers and accelerate the part production process.

A complexity print geometry has a significant impact to the printing quality with GMAW 3-D printing. A 3-D model with a lot of small details, a lot of small different curves or patterns in each layer or parts that smaller than 1 millimeter results in low quality of surface and shape because the limitation of the smallest bead width is 1 millimeter. The diameter of the welding wire (0.6 mm) limits the bead width as the welding is melting the wire on to a substrate so there would always be some spreading out of that melting steel. The smallest uniform welding beads in experiments using the setup and materials described is about 1 millimeter. For an overhang or a bridge in a model, a support can be added, but it might require machining to remove the support or the use of different metals that can be preferentially removed by a chemical or high temperature process. Thus, many of the free and open-source 3-D models available on-line are not suitable for metal 3-D printing using GMAW technology at this time. The size of the welding bead also causes the smoothness between layers is also deficient resulting in a staircase effect would be seen on a printed part of complicated 3-D models with a shallow slope of less than 45 degrees. For example, consider a beveled gear [49]. When printed at 30 degrees it has an obvious staircase effect and shape problem when it was metal 3-D printed (see Figure 19). When the angle is changed to 45 degrees the stair case effect is reduced but it still has a shape problem at the top part due to the gear teeth being too small (see Figure 20). Both examples have gap problem on the surface because the size of the models are not in a full millimeter. However, the author provided an SCAD file, that the user can edit to increase the size of the beveled gear and generate a new model appropriate for GMAW metal 3-D printing. This can be seen in Figure 21. After the beveled gear is printed, it would require more machining until the gear is appropriate to use for a high tolerance application. Another option would be to design a 3-D model for metal 3-D printing with an open-source software 3-D CAD modeler, such as OpenSCAD [50]. The 3-D model for metal 3-D printing should be in full millimeter of length and if there is a slope it should not be less than 45 degrees if this is possible option. For an overhang or a bridge in a model, a support can be added, but it might require machining to remove the support.

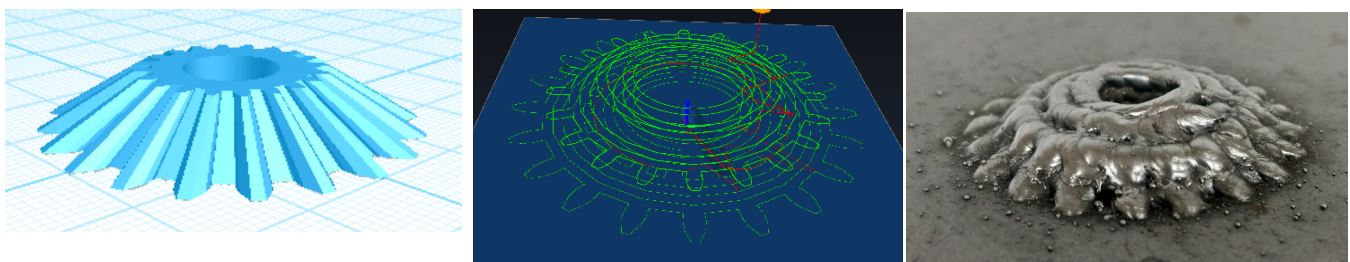


Figure 19. A beveled gear 3-D model with 30 degrees angle (grid lines are every 10mm), its generated path and metal 3-D printed.

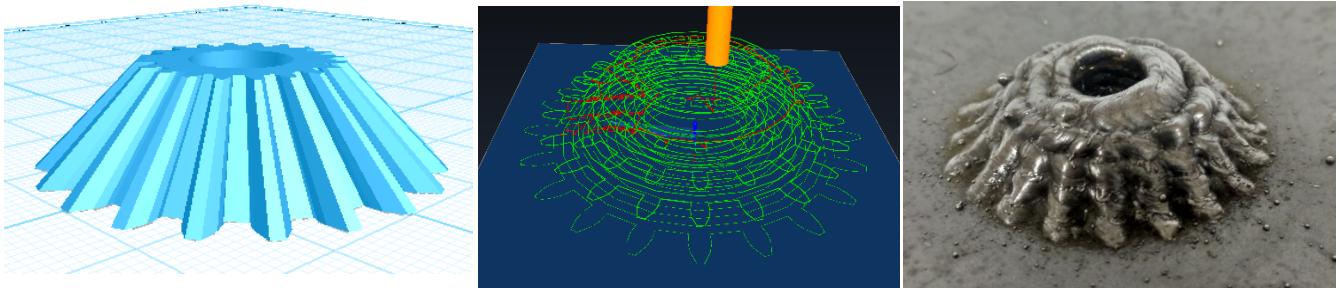


Figure 20. A beveled gear 3-D model with 45 degrees angle (grid lines are every 10mm), its generated path and metal 3-D printed.

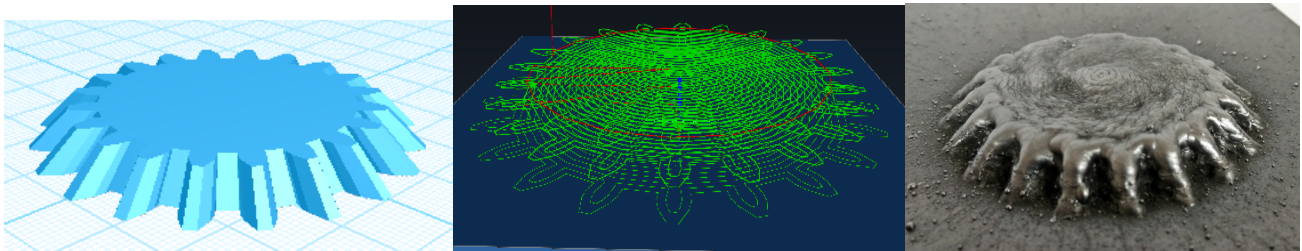


Figure 21. A beveled gear 3-D model (grid lines are every 10mm), its generated path and metal 3-D printed.

For the future work, research can focus on reducing the layer height of each layer. One option is to mill out the surplus layer height [51-55], which helps reduce the layer height and also removes the rough surface, which results in a better quality as it enables first layer quality on every layer. However, this will increase the printing time in total and create more waste. In addition, methods to extract heat inside a steel printed part during printing needs to be researched more in order to mitigate or solve the problem. The most obvious solution to this problem is to use this technology primarily for larger parts.

## 5. Conclusions

The open-source slicer software called *MOSTMetalCura* for metal 3-D printing has been implemented successfully and reported in this paper. It was customized from the open-source slicing engine named *CuraEngine*. Many customizable settings were added to the new slicer to support the metal 3-D printing. The output G-code is specifically for the low-cost open-source metal 3-D printer developed by MOST, however, it can be adjusted to support other metal 3-D printers as well. The paper also reports how to optimize the settings for the metal 3-D printing. The optimized settings for the low-cost open-source metal 3-D printer are 7 millimeters per second printing speed, 2 millimeters of layer height, about 1 millimeter of bead width, 5 volts of welding voltage, 28-31 on scale of wire feed speed, and 30 CFH of RC25 shield gas. The result shows the high resolution of 3-D printed steel



simple gear.

## References

1. Gebhardt, Andreas, (2003) Rapid Prototyping. Hanser, Munich.
2. Noorani R, (2006) Rapid prototyping: principles and applications. John Wiley & Sons Incorporated.
3. Cooper K, (2001) Rapid prototyping technology: selection and application. CRC press.
4. Islam MN, Pramanik A, Slamka S, (2016) Errors in different geometric aspects of common engineering parts during rapid prototyping using a Z Corp 3D printer. *Progress in Additive Manufacturing*, 1-9. doi: 10.1007/s40964-016-0006-7
5. Boparai KS, Singh R, Singh H (2016) Modeling and optimization of extrusion process parameters for the development of Nylon6–Al–Al<sub>2</sub>O<sub>3</sub> alternative FDM filament. *Progress in Additive Manufacturing*, 1-14. doi: 10.1007/s40964-016-0011-x
6. Bak D (2003) Rapid prototyping or rapid production? 3D printing processes move industry towards the latter. *Assembly Automation*, 23(4), 340-345.
7. Petrick IJ, Simpson TW (2013) 3D printing disrupts manufacturing: how economies of one create new rules of competition. *Research-Technology Management*, 56(6), 12-16.
8. Lasagni F, Vilanova J, Perriñán A, Zorrilla A, Tudela S, Gómez-Molinero V (2016) Getting confidence for flying additive manufactured hardware. *Progress in Additive Manufacturing*, 1-11. doi: 10.1007/s40964-016-0014-7
9. Lipson H, Kurman M (2013) *Fabricated: The new world of 3D printing*. John Wiley & Sons.
10. Wittbrodt BT, Glover AG, Laureto J, Anzalone GC, Oppliger D, Irwin JL, Pearce JM (2013) Life-cycle economic analysis of distributed manufacturing with open-source 3-D printers. *Mechatronics*, 23(6), 713-726.
11. Mota C (2011, November) The rise of personal fabrication. In *Proceedings of the 8th ACM conference on Creativity and cognition*, 279-288. ACM.
12. Gwamuri J, Wittbrodt BT, Anzalone NC, Pearce JM (2014) Reversing the Trend of Large Scale and Centralization in Manufacturing: The Case of Distributed Manufacturing of Customizable 3-D-Printable Self-Adjustable Glasses. *Challenges in Sustainability*, 2(1), 30-40.
13. King DL, Babasola A, Rozario J, Pearce JM (2014) Development of mobile solar photovoltaic powered open-source 3-D printers for distributed customized manufacturing in off-grid communities. *Challenges in Sustainability*, 2(1), 18-27.
14. Laeng J, Stewart JG, Liou FW (2000) Laser metal forming processes for rapid prototyping-A review. *International Journal of Production Research*, 38(16), 3973-3996.
15. Lewis GK, Schlienger E (2000) Practical considerations and capabilities for laser assisted direct metal deposition. *Materials & Design*, 21(4), 417-423.
16. Santos EC, Shiomi M, Osakada K, Laoui T (2006) Rapid manufacturing of metal components by laser forming. *International Journal of Machine Tools and Manufacture*, 46(12), 1459-1468.
17. Delgado J, Ciurana J, Serenó L (2011) Comparison of forming manufacturing processes and selective laser melting technology based on the mechanical properties of products: In this work, the superior property of the selective laser melting technology is presented by comparing four real parts manufactured using forming processes and selective laser melting technology and

- analysed for tension, compression and flexural. *Virtual and Physical Prototyping*, 6(3), 167-178.
18. Cooper F (2016) Sintering and additive manufacturing: “additive manufacturing and the new paradigm for the jewellery manufacturer”. *Progress in Additive Manufacturing*, 1-15. doi: 10.1007/s40964-015-0003-2
  19. Heintl P, Rottmair A, Körner C, Singer RF (2007) Cellular titanium by selective electron beam melting. *Advanced Engineering Materials*, 9(5), 360-364.
  20. Gaytan SM, Murr LE, Medina F, Martinez E, Lopez MI, Wicker RB (2009) Advanced metal powder based manufacturing of complex components by electron beam melting. *Materials technology*, 24(3), 180-190.
  21. Murr LE, Gaytan SM, Ramirez DA, Martinez E, Hernandez J, Amato KN, Wicker RB (2012) Metal fabrication by additive manufacturing using laser and electron beam melting technologies. *Journal of Materials Science & Technology*, 28(1), 1-14.
  22. Peels J (2014) Metal 3D printing: From lab to fab. *Inside 3DP*. <http://www.inside3dp.com/metal-3d-printing-lab-fab/>. Accessed 23 May 2014
  23. Sells E, Smith Z, Bailard S, Bowyer A, Olliver V (2009) RepRap: The Replicating Rapid Prototyper: Maximizing Customizability by Breeding the Means of Production. in F. T. Piller, M. M. Tseng (Eds.), *Handbook of Research in Mass Customization and Personalization: Strategies and concepts*. World Scientific, 1:568-580.
  24. Jones R, Haufe P, Sells E, Iravani P, Olliver V, Palmer C, Bowyer A (2011) RepRap—the replicating rapid prototyper. *Robotica*, 29(01), 177-191.
  25. Bowyer A (2014) 3D printing and humanity's first imperfect replicator. *3D printing and additive manufacturing*, 1(1), 4-5.
  26. Anzalone GC, Zhang C, Wijnen B, Sanders PG, Pearce JM (2013) A low-cost open-source metal 3-D printer. *IEEE Access*, 1, 803-810.
  27. Banzi M, Shiloh M (2014) *Getting Started with Arduino: The Open Source Electronics Prototyping Platform*. Maker Media, Inc..
  28. Arduino - Home (2015) <https://www.arduino.cc/>. Accessed 1 November 2015
  29. Haselhuhn AS, Gooding EJ, Glover AG, Anzalone GC, Wijnen B, Sanders PG, Pearce JM (2014) Substrate release mechanisms for gas metal arc weld 3D aluminum metal printing. *3D Printing and Additive Manufacturing*, 1(4), 204-209.
  30. Haselhuhn AS, Wijnen B, Anzalone GC, Sanders PG, Pearce JM (2015) In situ formation of substrate release mechanisms for gas metal arc weld metal 3-D printing. *Journal of Materials Processing Technology*, 226, 50-59.
  31. Pinar A, Wijnen B, Anzalone GC, Havens TC, Sanders PG, Pearce JM (2015) Low-cost open-source voltage and current monitor for gas metal arc weld 3D printing. *Journal of Sensors*. doi:10.1155/2015/876714
  32. Nilsiam Y, Haselhuhn A, Wijnen B, Sanders P, Pearce JM (2015) Integrated Voltage—Current Monitoring and Control of Gas Metal Arc Weld Magnetic Ball-Jointed Open Source 3-D Printer. *Machines*, 3(4), 339-351.
  33. Wijnen B, Anzalone GC, Haselhuhn AS, Sanders PG, Pearce JM (2016) Free and open-source control software for 3-D motion and processing. *Journal of Open Research Software*, 4(1).
  34. Haselhuhn AS, Buhr MW, Wijnen B, Sanders PG, Pearce JM (2016) Structure-property

- relationships of common aluminum weld alloys utilized as feedstock for GMAW-based 3-D metal printing. *Materials Science and Engineering: A*, 673, 511-523.
35. Cura 3D Printing Slicing Software (2015) <https://ultimaker.com/en/products/cura-software>. Accessed 1 November 2015
  36. Slic3r (2015) <http://slic3r.org/>. Accessed 1 November 2015
  37. Ultimaker/CuraEngine (2015) <https://github.com/Ultimaker/CuraEngine>. Accessed 1 November 2015
  38. Laplume AO, Petersen B, Pearce JM (2016) Global value chains from a 3D printing perspective. *Journal of International Business Studies*. doi:10.1057/jibs.2015.47
  39. Berman B (2012) 3-D printing: The new industrial revolution. *Business horizons*, 55(2), 155-162. <https://doi.org/10.1016/j.bushor.2011.11.003>
  40. Cura - RepRapWiki (2016) <http://reprap.org/wiki/Cura>. Accessed 4 March 2016
  41. daid/Cura (2016) <https://github.com/daid/Cura>. Accessed 4 March 2016
  42. G-code - RepRapWiki (2015) <http://reprap.org/wiki/G-code>. Accessed 12 October 2015
  43. Miller MIG Welders - MIG Welding and GMAW Welding Machines - MillerWelds (2015) <https://www.millerwelds.com/equipment/welders/mig-gmaw/millermatic-190-mig-welder-m00487>. Accessed 18 October 2015.
  44. MOST metal slicing examples <https://osf.io/6u5sp/> Accessed 26 January 2017.
  45. Clark, R., Lund, P., Sanders, P., Pearce, JM. Weld bead performance metrics for arc weld-based additive manufacturing of ER70S-2 steel and 316L stainless steel (to be published).
  46. CAMotics (2015) <http://camotics.org/>. Accessed 5 December 2015
  47. Lancaster JF (1984) The physics of welding. *Physics in technology*, 15(2), 73.
  48. Suban M, Tušek J (2001) Dependence of melting rate in MIG/MAG welding on the type of shielding gas used. *Journal of Materials Processing Technology*, 119(1), 185-192.
  49. Thornburg, D. (2014). <https://www.youmagine.com/designs/beveled-gear>. Accessed 20 September 2016
  50. OpenSCAD (2016) <http://www.openscad.org/>. Accessed 4 March 2016
  51. Choi DS, Lee SH, Shin BS, Whang KH, Song YA, Park SH, Jee HS (2001) Development of a direct metal freeform fabrication technique using CO<sub>2</sub> laser welding and milling technology. *Journal of Materials Processing Technology*, 113(1), 273-279.
  52. Xiong X, Haiou Z, Guilan W (2008) A new method of direct metal prototyping: hybrid plasma deposition and milling. *Rapid Prototyping Journal*, 14(1), 53-56.
  53. Krassenstein B (2014) VDK6000, Incredible 6-axis Metal 3D Printer, Milling Machine, Laser Scanner Unveiled. <https://3dprint.com/10079/vdk6000-robotic-work-environment/>. Accessed 14 November 2016
  54. Anderton J (2014) 3D Printing and 5-Axis Machining Combined in One Machine [ENGINEERING.com](http://www.engineering.com/AdvancedManufacturing/ArticleID/8778/3D-Printing-and-5-Axis-Machining-Combined-in-One-Machine.aspx). <http://www.engineering.com/AdvancedManufacturing/ArticleID/8778/3D-Printing-and-5-Axis-Machining-Combined-in-One-Machine.aspx>. Accessed 14 November 2016
  55. Sodick Inc (2016) One-Process Metal 3D printing & milling machine - Today's Motor Vehicles. <http://www.todaysmotorvehicles.com/product/sodick-3d-print-milling-machine-092916/>. Accessed 14 November 2016.