A low-cost open-source automated shot peen forming system

Wassime Siguerdidjane, Farbod Khameneifar, Frédérick P. Gosselin

1. Hardware in context

Shot peen forming is a cold working process used in the aerospace industry to form large sheet metal panels. It leverages high-velocity impacts to induce compressive residual stresses in the material. These stresses lead to changes in the curvatures of the part. The industrial and academic implementations of the process in automated conditions usually demand industrial robots with large blasting cabinets. These solutions are expensive and require large spaces. While 6-axis industrial robots offer the possibility to work on complex geometries, most academic research in the field is conducted on simple plates [1–5] to validate proposed modelling and process planning solutions. This suggests that a simpler solution may be consid-
ered by most researchers in the field. In this article, we propose a low-cost open-source shot peen forming robot able to form small samples. The presented prototype was leveraged to validate experimentally our closed-loop automation scheme for shot peen forming.

2. Hardware description

The peening system is a 2-axis cartesian robot made from 2020 aluminum extrusions, powered by stepper motors and controlled by an Arduino Mega microcontroller with the Ramps1.4 shield, running the Marlin 3D printer firmware. The peening task is handled by a sandblasting nozzle fixed on the robot’s gantry, that projects the blasting media on the sample. The robot itself is housed in a plastic storage container to keep the media inside and make it reusable. This solution provides a table-top alternative to the standard solution of using an industrial robot arm housed in a blasting cabinet. The other main advantage to the proposed solution is the cost reduction, the peening robot costs about 400 USD, while industrial infrastructures usually involve a few 100 000 USD. Additionally, this solution lets the user choose the software with which the system interacts without license limitations or black box implementations. This is particularly interesting in the context of academic research since researchers usually want control over their hardware and software. Since the robot is controlled by an open-source 3D printer firmware, the user is free to modify it as required and the robot supports serial command transmission in the form of G-code. The solution is constructed out of a majority of off-the-shelf components and a few 3D printed parts, making it easy to replicate. Fig. 1 presents a photograph of the proposed peen forming system.

The presented hardware can be useful for users interested to:

• Perform in–house automated small scale shot peen forming of plates at low cost
• Perform automated sandblasting finishing on parts

3. Design files

The design files are available in the project repository at https://doi.org/10.17605/OSF.IO/Z5RVN, along with a G-code file sample to control the robot. The designed parts are all 3D printed using PLA. These parts serve to support drive components and protect the components, such as the rails, wheels and pulleys from the shot media. If not properly shielded from the shot, these components will either be damaged over time or their functions will be impaired during operation.

3.1. Design files summary

Table 1 presents the required design files. All parts in the design files table are 3D printed using PLA filament.

1. cover_motor_x.FCStd: a part covering the stepper motor shaft and the drive pulley for the x-axis. It provides a rigid covering to keep the flexible rail cover from interfering with the drive. The part is located in the X-axis motor sub-assembly (see Fig. 3). Approximately 7 m of filament is required to print this part;
2. belt_cover_x.FCStd: a part covering the belt and idler pulley for the x-axis. It also provides a rigid covering of the components. The part is located in the X-axis idler pulley sub-assembly (see Fig. 5). Approximately 11 m of filament is required to print this part;
3. belt_bearing_support_x.FCStd: a part used to support the idler pulley. The part is located in the X-axis idler pulley sub-assembly (see Fig. 5). Approximately 3 m of filament is required to print this part;

Fig. 1. Photos of the presented shot peen forming system presenting the lid of the container on and off.
4. **cover_center_x.FCStd**: a part covering the x-axis gantry. It provides a rigid covering of the gantry and keeps the flexible rail cover from tangling in the wheels. The part is located in the X-axis gantry sub-assembly (see Fig. 4). Approximately 6 m of filament is required to print this part;

5. **cover_motor_y.FCStd**: a part covering the stepper motor shaft and the drive pulley for the y-axis. It provides a rigid covering to keep the flexible rail cover from interfering with the drive. The part is located in the Y-axis motor sub-assembly (see Fig. 6). Approximately 7 m of filament is required to print this part;

### Table 1
Design file table.

| Design filename | File type | Open source license | Location of the file |
|-----------------|-----------|---------------------|----------------------|
| cover_motor_x.FCStd | CAD | GNU GPL v. 3 | https://doi.org/10.17605/OSF.IO/Z5RVN |
| belt_cover_x.FCStd | CAD | GNU GPL v. 3 | https://doi.org/10.17605/OSF.IO/Z5RVN |
| cover_center_x.FCStd | CAD | GNU GPL v. 3 | https://doi.org/10.17605/OSF.IO/Z5RVN |
| cover_motor_y.FCStd | CAD | GNU GPL v. 3 | https://doi.org/10.17605/OSF.IO/Z5RVN |
| belt_cover_y.FCStd | CAD | GNU GPL v. 3 | https://doi.org/10.17605/OSF.IO/Z5RVN |
| belt_bearing_support_y.FCStd | CAD | GNU GPL v. 3 | https://doi.org/10.17605/OSF.IO/Z5RVN |
| sandblaster_mnt.FCStd | CAD | GNU GPL v. 3 | https://doi.org/10.17605/OSF.IO/Z5RVN |
| sandblaster_cover.FCStd | CAD | GNU GPL v. 3 | https://doi.org/10.17605/OSF.IO/Z5RVN |

### Table 2
Mechanical bill of materials (For more detail refer to the csv file available in the repository at https://doi.org/10.17605/OSF.IO/Z5RVN).

| Component | Qt./build | Qt./pkg | cost/pkg (USD) | Qt. of pkg | Cost (USD) |
|-----------|-----------|---------|----------------|-----------|------------|
| M5X8 screw | 18        | 10      | $0.99          | 2         | $1.98      |
| M5X10 screw | 10       | 10      | $1.09          | 1         | $1.09      |
| M5X15 screw | 3        | 10      | $1.19          | 1         | $1.19      |
| M5X20 screw | 3        | 10      | $1.29          | 1         | $1.29      |
| M5X30 screw | 2        | 10      | $1.49          | 1         | $1.49      |
| M5X45 screw | 4        | 10      | $1.79          | 1         | $1.79      |
| M3X10 screw | 8        | 100     | $0.29          | 8         | $2.32      |
| M4X12 screw | 5        | 100     | $9.86          | 1         | $9.86      |
| M5 nut | 13        | 100     | $1.76          | 1         | $1.76      |
| M5 washer | 16        | 100     | $3.14          | 1         | $3.14      |
| M4 nut | 5         | 100     | $1.41          | 1         | $1.41      |
| M4 washer | 5        | 100     | $2.86          | 1         | $2.86      |
| 4 hole L bracket | 4     | 12      | $10.45         | 1         | $10.45     |
| 4 hole straight bracket | 2   | 20      | $13.67         | 1         | $13.67     |
| Mini V gantry | 3      | 1       | $34.99         | 3         | $104.97    |
| 2020 v-slot 500 mm | 5    | 1       | $5.49          | 5         | $27.45     |
| Nema 17 mounting bracket | 3 | 4       | $12.38        | 1         | $12.38     |
| Corner bracket | 8     | 20      | $10.72         | 1         | $10.72     |
| M5 t-nuts | 24        | 100     | $13.67         | 1         | $13.67     |
| GT2 belt kit | 1       | 1       | $20.11         | 1         | $20.11     |
| 1/4NPT quick-connect fitting | 1    | 1       | $4.42          | 1         | $4.42      |
| 1/4NPT push fitting | 2    | 10      | $27.23         | 1         | $27.23     |
| 1/4 inch OD PTFE tube | 1  | 1       | $11.21         | 1         | $11.21     |
| Sandblaster | 1      | 1       | $20.91         | 1         | $20.91     |
| Plastic storage bin | 1     | 1       | $17.04         | 1         | $17.04     |
| **Total** |           |         |                |           | **$324.41** |

| Component | Quantity | Cost per unit (USD) | Total cost |
|-----------|----------|---------------------|------------|
| Arduino Mega RAMPS kit | 1       | $50.48               | $50.48     |
| 12 V solenoid valve | 1       | $13.90               | $13.90     |
| 12 V 5A power supply | 1       | $10.68               | $10.68     |
| Arduino relay module | 1       | $7.23                | $7.23      |
| Grounded power chord | 1       | $11.02               | $11.02     |
| **Total** |           |                     | **$93.31** |

4. **cover_center_x.FCStd**: a part covering the x-axis gantry. It provides a rigid covering of the gantry and keeps the flexible rail cover from tangling in the wheels. The part is located in the X-axis gantry sub-assembly (see Fig. 4). Approximately 6 m of filament is required to print this part;

5. **cover_motor_y.FCStd**: a part covering the stepper motor shaft and the drive pulley for the y-axis. It provides a rigid covering to keep the flexible rail cover from interfering with the drive. The part is located in the Y-axis motor sub-assembly (see Fig. 6). Approximately 7 m of filament is required to print this part;
6. belt_cover_y.FCStd: a part covering the belt and idler pulley for the y-axis. It also provides a rigid covering of the components. The part is located in the Y-axis idler pulley sub-assembly (see Fig. 8). Approximately 4 m of filament is required to print this part;
7. belt_bearing_support_y.FCStd: a part used to support the idler pulley. The part is located in the Y-axis idler pulley sub-assembly (see Fig. 8). Approximately 3 m of filament is required to print this part;
8. cover_center_y.FCStd: a part covering the y-axis gantry. It provides a rigid covering of the gantry and keeps the flexible rail cover from tangling in the wheels. The part is located in the Y-axis gantry sub-assembly (see Fig. 7). Approximately 8 m of filament is required to print this part;
9. sandblaster_mnt.FCStd: a part used to mount the sandblasting nozzle to the gantry. The part is located in the X-axis gantry sub-assembly (see Fig. 4). Approximately 9 m of filament is required to print this part;
10. sandblaster_cover.FCStd: a part used to secure the sandblasting nozzle to the sandblasting mount. The part is located in the X-axis gantry sub-assembly (see Fig. 4). Approximately 8 m of filament is required to print this part.

4. Bill of materials (BOM)

This section presents the off-the-shelf parts and materials required to build the presented shot peen forming robot. Table 2 summarizes the parts and costs of the mechanical assembly, while Table 3 summarizes the electronic parts required as well as their cost. It details the number of parts required for the build as well as the number of parts in each package in which the component can be purchased at the location listed in the detailed BOM linked in the project repository. To stay concise, the suggested locations for purchase are only listed in the detailed file on the repository.

5. Build instructions

5.1. General instructions

The mechanical build is separated into sub-assemblies, which are then assembled to form the complete build. The electrical wiring instructions follow the mechanical build instructions. Fig. 2 presents a rendered representation of the global mechanical drive assembly of the robot. This provides the reader a general idea of the assembly before the build instructions of each sub-assembly are presented.
The 2020 extrusions can be cut to the desired length to accommodate the available storage bin. If the reader prefers to build a custom enclosure for the robot, it is also possible, though it was not considered necessary for the purposes of the project.

5.2. X-axis motor sub-assembly

This section describes the assembly of the X-axis motor sub-assembly. Fig. 3 presents an exploded view of the sub-assembly.

1. Insert 3 t-nuts on the x-axis 2020 extrusion, as shown on Fig. 3;
2. Mount a first corner bracket to the Nema 17 bracket using an M4X12 screw, an M4 washer, and an M4 nut;
3. Mount a GT2 drive pulley to the Nema 17 shaft. Secure the pulley with the included set screws;
4. Mount a Nema 17 stepper motor to the Nema 17 bracket with 4 M3X10 screws;
5. Thread an M3X8mm screw through the first corner bracket, the x-axis motor cover, and into the lone t-nut (Fig. 3). This step secures the motor to the extrusion and the cover;
6. Join two corner brackets to the motor cover and 2020 extrusion by threading two M3X8mm screws into the two remaining t-nuts. These two corner brackets will join the x-axis to the y-axis gantries.

5.3. X-axis gantry sub-assembly

This section describes the assembly of the X-axis gantry sub-assembly. Fig. 4 presents an exploded view of the sub-assembly.

1. Insert 4 M5 nuts in the sandblaster mount (Fig. 4);
2. Mount the sandblaster mount and gantry cover to the Openbuild mini V gantry using two M5X16mm screws threaded into the mini V gantry;
3. The sandblaster is secured by screwing the sandblaster mount cover to the sandblaster mount with 4 M5X45mm screws;
4. Loop the GT2 belt around the notches available on the mini V gantry. This belt will loop around the X-axis motor drive pulley and the X-axis idler pulley to provide drive to the x-axis gantry;
5. The X-axis gantry sub-assembly then slides on the x-axis 2020 extrusion.

5.4. X-axis idler pulley sub-assembly

This section describes the assembly of the X-axis idler pulley sub-assembly. Fig. 5 presents an exploded view of the sub-assembly.

---

5

![Fig. 3. X-axis motor sub-assembly.](image-url)
1. Mount a corner bracket to the idler pulley mount, using an M5x16mm screw and an M5 nut (Fig. 5);
2. Mount the idler pulley to the idler pulley mount with an M5x20mm screw threaded into the mount;
3. Slide the idler assembly cover on the X-axis 2020 extrusion;
4. Insert two M5 t-nuts in the X-axis 2020 extrusion;
5. Mount the corner bracket to the X-axis 2020 extrusion with an M5x8mm screw and an M5 t-nut;
6. Loop the X-axis drive belt around the idler pulley and fix it to the X-axis gantry;
7. Slide the cover over the idler assembly and fasten it and two corner brackets to the 2020 extrusion with two M5x8mm screws and M5 t-nuts. The two corner brackets will be used to connect this portion of the X-axis to the Y-axis gantry, similarly to the corner brackets of the motor cover assembly;
8. Cap the idler pulley assembly cover with its lid. The lid and the cover are printed with an interference fit, but for further sealing, silicone rubber caulk can be applied to the joint.

Fig. 4. X-axis gantry sub-assembly.

Fig. 5. X-axis idler pulley sub-assembly.
5.5. Y-axis motor sub-assembly

This section describes the assembly of the Y-axis motor sub-assembly. Fig. 6 presents an exploded view of the sub-assembly. Two instances of this sub-assembly are necessary. These steps need to be followed for the right and left Y-axis assemblies.

1. Mount the L-bracket to one slot of the Nema 17 bracket as shown in Fig. 6, using an M4X12mm screw, an M4 washer, and an M4 nut;
2. Mount the straight flat bracket to the other slot Nema 17 bracket as shown in Fig. 6, using an M4X12mm screw, an M4 washer, and an M4 nut;
3. Mount a GT2 drive pulley to the Nema 17 shaft. Secure the pulley with the included set screws;
4. Mount a Nema 17 stepper motor to the Nema 17 bracket with 4 M3X10 screws;
5. Insert two M5 t-nuts in a 2020 extrusion. This extrusion serves as a structural component, not as an axis rail;
6. Slide the 2020 extrusion in the square hole of the motor cover;
7. Insert an M5 t-nut in one of the Y-axis 2020 extrusions;
8. Slide the motor cover on the Y-axis 2020 extrusion;
9. Mount the straight flat bracket onto the structural 2020 extrusion by treading two M5X10mm screws, through two M5 washers and into the two M5 t-nuts;
10. Mount the L-bracket onto the Y-axis 2020 extrusion by treading an M5X10mm screw, through two M5 washers and into the M5 t-nut;
11. Loop the Y-axis drive belt around the GT2 pulley.

5.6. Y-axis gantry sub-assembly

This section describes the assembly of the Y-axis gantry sub-assembly. Fig. 7 presents an exploded view of the sub-assembly. Two instances of this sub-assembly are necessary. One assembly is fastened to the X-axis idler pulley assembly and the other to the X-axis motor assembly.

1. Loop the Y-axis drive belts to one side of each gantry and fasten them with zip ties;
2. Slide the gantry covers over the Openbuild mini V gantries;
3. Fasten the corner brackets from the X-axis assembly to the Openbuild mini V gantry with two M5X8mm screws through the gantry cover. If assembling the left Y gantry, fasten the brackets of the X-axis idler assembly. For the right assembly, fasten the brackets of the X motor assembly;

4. Slide the gantry assemblies on the Y-axis 2020 extrusions.

5.7. Y-axis idler pulley sub-assembly

This section describes the assembly of the Y-axis idler sub-assembly. Fig. 8 presents an exploded view of the sub-assembly. Two instances of this sub-assembly are necessary. These steps need to be followed for the right and left Y-axis assemblies.

1. Mount a GT2 idler pulley to the pulley bracket using an M5X30mm screw and an M5 nut;
2. Mount the pulley bracket to the pulley cover using an M5X20mm screw and an M5 nut;
3. Insert two M5 t-nuts in a 2020 extrusion. This extrusion serves as a structural component, not as an axis rail;
4. Mount the L-bracket to the structural 2020 extrusion by threading two M5X8mm screws into the t-nuts;
5. Slide the L-bracket through the slot in the pulley cover;
6. Insert an M5 t-nut in the Y-axis 2020 extrusion;
7. Slide the Y-axis 2020 extrusion in the pulley cover;
8. Mount the L-bracket to the Y-axis 2020 extrusion by threading an M5X8mm screw into the t-nut;
9. Loop the Y-axis drive belt around the idler pulley and into the belt slot on the gantry.
5.8. Mounting the robot motion assembly to the container

This section describes the mounting of the robot motion assembly to the storage container used as housing. Refer to Fig. 1 for a general view of the assembled robot in its housing.

1. Select the mounting height of the robot for all components to clear the lid when moving. This depends on the chosen storage container;
2. Trace four reference points to drill four holes in the container for mounting;
3. Drill four holes on the marked reference points;
4. Insert two M5 t-nuts in each of the two structural 2020 extrusions;
5. Thread four M5X8mm screws through the mounting holes in the storage container and into the M5 t-nuts.

5.9. Sealing and component protection

Shot peen forming generates dust. The goal of the robot housing is to keep most of the generated dust within. All components within the housing will be covered in dust and will be impacted by the projectiles. The rails, if left unprotected, will also accumulate shot media, which will get lodged in the wheels and impede the motion of the robot. This section presents the steps required to seal the robot housing and protect the components housed within it.
1. Adding insulation neoprene foam tape to the lid increases the quality of the seal, reducing the quantity of dust that can escape the housing;
2. The chosen storage bin featured integrated latches to secure the lid to the container. Alternatively, 3D printed latches can be designed for a storage bin that lacks such a feature. Also, binder clips can be used to secure the lid or to enhance the integrated latches for a more even sealing pressure;
3. The rails can be protected with traditional bellows, but in this project, we tested the low-cost alternative of using 4 mil polyethylene sheets. These sheets can be cut and joined with glue or tape, or welded with heat from a soldering iron to make sleeves that are mounted to the 3D printed covers with zip-ties.

5.10. Electrical wiring instructions

The instructions take into account that the user is familiar with Arduino microcontrollers. This project leverages an Arduino Mega fitted with a RAMPS 1.4 shield. The setup of the RAMPS shield, in terms of jumper and driver installation, is presented in the RAMPS documentation on the RepRap wiki [6]. The project uses the Marlin firmware to drive the system. The procedure to upload and set up the firmware is standard and detailed in the Marlin documentation [7]. For this project, we use the X and Z axis of the shield only, since only two degrees of freedom are necessary. A more complex 3-axis system could be designed and used for the same shot peen forming purpose, but for treating plates, a simple 2-axis system is sufficient. The motors of the Y-axis of the robot are connected to the Z-axis of the shield because two connection ports are available, with only one available port for the other axes (Fig. 9). The D9 port (Fig. 9) is used to control the solenoid valve and is represented in the firmware as a fan, which is a standard Marlin feature.

Fig. 10 presents a schematic of the wiring required for the project.

1. Connect the wires of the 12 V solenoid valve to the D9 port on the shield while respecting the polarity;
2. Connect the wires of the X-axis stepper motor to the X motor port on the shield. Upon testing, if the stepper motor rotates in the wrong direction, the orientation of the connection can be flipped;

---

**Fig. 9.** RAMPS 1.4 shield layout. Reproduced from [6].
3. Connect the Y-axis motors to the Z-axis ports on the shield. The Z-axis is used because it provides the ports to connect two motors at once. Therefore, the G-code used to control the robot will be in terms of X and Z motion;
4. Connect the 12 V 5A power supply to the 5A port on the shield;
5. Certain power supplies require the user to wire the power plug on terminals (for 110 V or 240 V). Always use caution when dealing with high voltage and perform the wiring with the power plug removed from the power socket;
6. The user is free to mount the solenoid valve and the power supply to the housing. It is better to keep the microcontroller away from the housing to protect it from potential electrostatic discharges;
7. The microcontroller can be controlled through a USB tether, though we recommend running the G-code using an SD card through a RAMPS display interface. The wiring of such display interface is documented in the RAMPS documentation [6].

6. Operation instructions

6.1. Safety

The operation of the shot peening robot requires the following safety considerations:

- Compressors and compressed air are sources of concentrated energy and are dangerous. Never operate compressed air equipment over the manufacturer's pressure rating and make sure that the pressure hoses used are not compromised;
- The cover of the robot housing needs to be closed at all times when the compressed air solenoid is activated;
- Protective goggles are to be worn at all times during the operation of the robot;
- Since the blasting media can generate dust upon impact with the sample, we advise the user to wear a dust mask or a respirator when operating the robot;
- Some blasting media can generate static electricity when passing through the hoses; some minor static shocks are to be expected when touching any conductive surface after handling the robot. Surrounding electronics should be protected against static shocks. For this reason, the Arduino is kept away from the robot housing. The use of rubber gloves will reduce this inconvenience and protect the surrounding electronics from static shock when being handled.

6.2. Operation

1. As a safety precaution, make sure that the robot is disconnected from the wall outlet and the air pressure source when preparing the robot for use. Always wear safety goggles when operating the machine.
2. Fill the bottom of the robot housing with your media of choice such as steel shot peening media, ceramic media, walnut shell blasting media, sand, sodium bicarbonate (baking soda), etc. Different media will offer different results in terms of treatment effectiveness or surface finish. Make sure the housing contains enough media to allow the intake hose to be submerged in media during the complete operation. The quantity of media is determined empirically.

3. Place your sample in the appropriate fixture to hold it during the treatment. This fixture depends on the shape of the sample.

4. Plug in the machine and turn on the main power switch. This will start the power supply and power the Arduino Mega. After a few seconds, the Marlin loading screen will appear. Once loaded, Marlin displays the status screen.

5. The robot is controlled via a G-code file that the user should place on an SD card. A sample G-code file is provided in the project repository. The SD card can be inserted in the reader offered by the display suggested in the electrical BOM in Table 3.

6. The rotary encoder of the display needs to be pressed to access the menu. The user can navigate the Marlin menu by rotating the encoder and pressing it to make a selection.

7. Close the lid of the robot housing.

8. Connect the intake solenoid valve to your compressed air source. Air should not be free to flow at this point.

9. Navigate to the "Print from SD card" menu of Marlin using the display rotary encoder.

10. Select the G-code file containing the sequence of operations.

11. The sample G-code provided begins by opening the solenoid valve to let the compressed air into the system. Since the wires of the valve are connected to the fan control pins on the Ramps 1.4 shield, Marlin requires the M106 command to activate it.

12. The robot will then begin to follow the motion dictated by the G-code file.

13. Once all motion instructions are completed by the robot, the sample G-code contains an M107 command that closes the solenoid valve, stopping the compressed air flow.

14. As a safety precaution, disconnect the compressed air hose from the solenoid valve.

15. The treated sample can now be retrieved from the fixture.

16. If no further treatment is required, the main power switch can be turned off and the robot can be unplugged.

7. Validation and characterization

The current section demonstrates an example of the deflections that are attainable with the shot peen forming robot. The deflection of a shot-peened aluminum plate depends on its size, the shot media, the location of treated regions, the exposure time, the air pressure, and the air flow rate. The hardware was used in our research work on the automation of shot peen forming with a 18.9 L air compressor which can provide an air flow rate of up to 141.6 L/min. The presented examples were conducted on two plates of 76.2 mm × 76.2 mm × 1.04 mm 6061-T6 aluminum uniformly treated. Both samples were incrementally peened for 2, 4, 6, 8, and 10 passes and measured between each pass. The shot peening media used for these tests was Zirshot Z425 ceramic media. One sample was peened with an air pressure of 344.7 kPa and the other with 517.1 kPa. Both samples were peened with a constant nozzle travel speed of 5 mm/s, using a zig-zag trajectory. The robot path was designed to extend beyond the samples’ area to keep accelerations and decelerations from happening over the plates. The plates were kept in a constraining jig to keep them flat during the peening process. Fig. 11 shows the resulting shape of the shot-peened sample. The sample treated at 517.1 kPa is presented on the left, and the sample treated at 344.7 kPa is presented on the right. Fig. 12 presents the deflections of the samples as a function of the number of peening passes. This demonstrates that the low-cost hardware is able to form deflections of up to 2.39 mm on 1.01 mm thick plates.

![Fig. 11. The final 76.2 mm × 76.2 mm × 1.04 mm shot peened plates after 10 passes, with the 517.1 kPa sample on the left and the 344.7 kPa sample on the right.](image-url)
Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was funded by Aerosphere Inc., the Fonds de recherche du Québec - Nature et technologies and the ministère de l’Économie, de la Science et de l’Innovation [Funding reference number LU-210888]. The authors would like to acknowledge the contribution of Tanuj Deshmukh and Jovan Kostenov, whose internships were centered on the fine-tuning of the hardware.

References

[1] P.A. Faucheux, F.P. Gosselin, M. Lévesque, Simulating shot peen forming with eigenstrains, J. Mater. Process. Technol. 254 (2018) 135–144, https://doi.org/10.1016/j.jmatprotec.2017.11.036.
[2] A. Gariépy, S. Larose, C. Perron, P. Bocher, M. Lévesque, On the effect of the orientation of sheet rolling direction in shot peen forming, J. Mater. Process. Technol. 213 (6) (2013) 926–938, https://doi.org/10.1016/j.jmatprotec.2013.01.003.
[3] L.V. Grasty, C. Andrew, Shot peen forming sheet metal: Finite element prediction of deformed shape, Proc. Inst. Mech. Eng. Part B: J. Eng. Manuf. 210 (4) (1996) 361–366, doi: 10.1243/0954410991808335.
[4] H.Y. Miao, M. Levesque, F. Gosselin, Shot peen forming pattern optimization to achieve complex target shapes: the inverse problem (Apr 2021), doi:10.31224/osf.io/a2frz.
[5] W. Siguerdidjane, F. Khameneifar, F.P. Gosselin, Efficient planning of peen-forming patterns via artificial neural networks, Manuf. Lett. 25 (2020) 70–74, https://doi.org/10.1016/j.mfglet.2020.08.001.
[6] RepRap, Ramps 1.4 — reprap, [Online; accessed 29-June-2021] (2021).https://reprap.org/mediawiki/index.php?title=RAMPS_1.4&oldid=189010.
[7] Marlin, Marlin firmware, [Online; accessed 29-June-2021] (2021).https://github.com/MarlinFirmware/Marlin.

Fig. 12. The deflection of uniformly peened 76.2 mm x 76.2 mm x 1.04 mm 6061-T6 plates as a function of the number of peening passes, for air pressures of 344.7 and 517.1 kPa with Zirshot Z425 ceramic shot media.