3d Wire Bending Sculptor

Alexander Clapacs
Cornell University
Ithaca, New York
ac927@cornell.edu

Adler Faulkner
Cornell University
Ithaca, New York
aif28@cornell.edu

John Tregurtha
Cornell University
Ithaca, New York
jt548@cornell.edu

Rudy Winkler
Cornell University
Ithaca, New York
rgw66@cornell.edu

INTRODUCTION

We can all envision a product designer rapidly prototyping some product and needing to get a feel for the dimensions of an object they are considering making. He or she could 3D print a part, but that is often time consuming and expensive, and its degree of precision is often unnecessary in the early stages of prototyping. They could also laser cut the part they're considering making, but at such an early stage of a product's design life cycle similar arguments against 3D printing hold for laser cutting. Instead, the designer can take up a piece of wire, and in near real time have a 3d wireframe of their design to hold, observe, and then iterate on. Once done with this preliminary iteration, the wire used in it can be re-straightened for future use in another iteration or another project. This process is suitable for early design iterations, but building wireframes that represent some design quickly and relatively accurately is no trivial task. Ideally, a designer would be able to take advantage of the flexibility and abstractness of wireframe models without sacrificing accuracy or speed.

Outside of the world of product design there are other motivations for constructing models out of wire. There is an entire artistic community centered around constructing sculptures out of wire[14,15,16]. While there are valid reasons to continue sculpting wire art by hand, there is certainly a place for automated wire sculpting as well. Given the difficult nature of dealing with complex wire sculptures by hand, an automatic wire sculptor could serve as a tool that helps artists iterate on designs as well as potentially expand the set of things artists are able to make in general.

Thus, we set out to build a 3D wire bender that meets the needs of both product designers and artists that we've covered. We started with a simple design that focused on constructing accurate 3D wireframe models both quickly and accurately. There has been some previous work in this field[1,3,9] so with our design we sought to improve on the DIY baseline provided by [9]. Furthermore, none of the wire benders we found support wire with a gauge as small as we used, so our bender is theoretically capable of processing models with a higher level of detail than other wire benders available. The bender can be constructed to build either 2D or 3D wire sculptures, and the mechanism to do so can come in many different forms; for instance it could be as simple as a single arm controlled by a motor, or could be as complicated as multiple bending spools that are controlled by multiple motors and sensors. Because we decided to focus on precision and speed rather than usability, we did not design our prototype to be able to handle spools of wire nor bend with multiple arms. Instead, we simplified this aspect of our prototype and only used segments of straight wire as our material and designed our prototype to only have one bending mechanism.

No matter how usable, accurate, and quick the bender is, the abstracted instructions remain the same for all bending machines: feed the wire through a specified amount and bend the wire a specified degree. These instructions could come in as many forms, such as G-code converted from existing 3D software (similar to how 3D printers work), or vector files converted into basic instruction lists, or even hand curated instructions straight from a text file. We avoided building a GCode interpreter because there was no clear way to process GCode properly without a software heavy approach. For example, many models could easily be bent into the machine while the model is being bent. So, we saw detecting how to bend such shapes properly as intrinsically linked to interpreting GCode, and software of that complexity is outside the scope of our goals. To give instructions to the machine, we created a 2D drawing interface that converts a drawing into an instruction set for our bender. Additionally, we curated 3D models by building instruction sets by hand. Overall, there is space for improvement on the software aspect of our prototype while we feel our hardware is a contribution to the wire bender space. We can extrude and bend with significant levels of accuracy, process models quickly, and process small gauge wire capable of a high level of detail unlike other wire benders that have previously been created.

RELATED WORK

A reasonable amount of work has been done in the field of wire bending. The features and capabilities of wire bending machines that have been developed thus far are highly dependent on the use case of the machine, as one would expect. We've found that, in general, wire bending machines are mostly used for industrial purposes, but are used in DIY settings as well. We've also seen that the design of a wire bending machine is dependent on the diameter and tensile strength of the wire that is expected to be used [2]. Further impacting design decisions are, the length, dimensionality, and complexity of the shapes one expects to create with a wire bending machine. The shapes one expects to create impact both the type of wire and the design of the mechanism that bends the wire [1,2,3]. We will speak in more detail about the impacts of these choices in the Industrial Wire Bending section. Later, we will also discuss the different types of ways wire can be bent, as that information is critical in the choice of our wire bending mechanism.

Industrial Wire Bending

The wire benders we've found that were designed for industrial purposes tend to be designed for a more specific use case and have more powerful capabilities specific to that use case [2]. There are different configurations of the bending mechanism that offer distinct advantages. For example, BLM Group's single-head wire bending machine is better at processing shorter, complex, 3D shapes, while their double-headed wire bending machines are better at "medium-to-long length parts with many bends, symmetric parts, parts that close back on themselves or wire wound resistors" [3]. The choice of the wire bending mechanism has many implications in the capabilities of a wire bending machine that are not immediately obvious. For example, Pensa's two main products, the D.I.Wire Pro and D.I.Wire Plus, handle wire dimensions of 0.7mm-4.8mm and 1.6mm-3.2mm respectively. Furthermore, the Pro has a maximum bend angle of 180 degrees and "roll bend[s]" smooth curves, while the Plus has a maximum bend angle of 135 degrees and can only "bump bend" smooth curves [10]. In general, the advantages of professional-grade over hobbyist wire bender are intuitive: professional-grade benders process wire faster, are capable of processing a wider range of wire types and sizes, and are more precise than hobbyist-level wire benders like the D.I.Wire Plus [10]. We've also found wire benders that people have built from scratch, which in general offer more rudimentary features than the industrial grade products. In spite of their more primitive capabilities, DIY wire bender designs provide useful direction for our own designs.

DIY Wire Bending

Given that we will be rapidly prototyping our own wire bender, we can learn much from the decisions of other DIY wire benders. The company Pensa appears to have gotten to it's current position as an industrial producer of wire benders with a DIY rapidly prototyped wire bender that they call D.I.Wire Bender. The design of the D.I.Wire Bender is rather simple: there is a spool of wire which serves as the source of the wire, next there is a mechanism that straightens the wire, and that mechanism feeds the wire into the piece that is responsible for bending the wire [9]. We used this design as our source of inspiration and we sought to improve on the capabilities of the simple design of the D.I.Wire Bender.

Software

Many industrial wire bending companies use proprietary software [1,2,3], so it is hard to tell what they use specifically to translate designs into machine instructions and how they guarantee the optimal bending path given some arbitrary model. The D.I.Wire project states that they translate basic Vector files into instructions such as "Feed 50mm, bend 90 degrees, feed 100mm…" [4]. Depending on the complexity of designs, it is possible to just write manual instructions and feed them to the machine (Similar to the D.I.Wire [4]). Making things as simple as a cube and as complex as most geometric 3d shapes could easily be made manually by writing out all of the necessary instructions. Generally, the more complex the shape the more complex the software we will need. In our approach we borrowed both from the D.I.Wire Bender and the industrial grade wire benders. For some 3D models we manually built an instruction set and push them to our wire bender; we also developed a 2D drawing interface that processes a drawing into custom instructions for our machine.

Wire Sculpting as Art

Wire sculpting art lies at the other end of the spectrum from industrial bending. Rather than optimizing machines to repeatedly create relatively simple structures, wire sculpting artists craft one-off complex designs. In most cases, the wire sculpting process for these artists is "manual", with the use of handheld tools and welders allowing for much more intricate or complex creations. These creations range from large 3d pieces[15], to smaller 2d pieces[16], and pieces with varying levels of intricacy [14].

These artistic examples are almost always made by integrating multiple wires or segments into a single piece, either by welding or some other connection technique. The relevance to our project is that we can take inspiration from these works, even if we cannot approach the same level of complexity. If the simple industrial bends give us a low bound for potential bending goals, then a lifesize, 1000-wire 3d portrait of Jimi Hendrix[14] gives an upper bound.

DESIGN

The basic construction of the wire bending machine contains 3 primary parts: A feeder that pushes the wire through the machine, and a bender which bends the wire at specific angles, and a 3D rotator to move the bender around the axis of the wire. We were able to eliminate two components, the source and the straightener, from our initial design by making a critical design decision. By selecting to bend only wire that was already straight and manually inserting new pieces, we did not need a source component to hold a spool of wire and did not need anything to straighten the wire. The main goal of our bender is to take in, process, and then bend sets of instructions that correspond to either 2D or 3D models. In order to interact with the bender, we developed a web based application to send instructions to the RedBear controlling our wire bender. The general flow of using the prototype is as follows: manually feed a piece of wire into the machine, then either draw something on our drawing interface that converts 2D drawings into instruction sets for the machine or click on one of the buttons that corresponds with a preset set of 3D model instruction sets, and then hit "Bend It!".

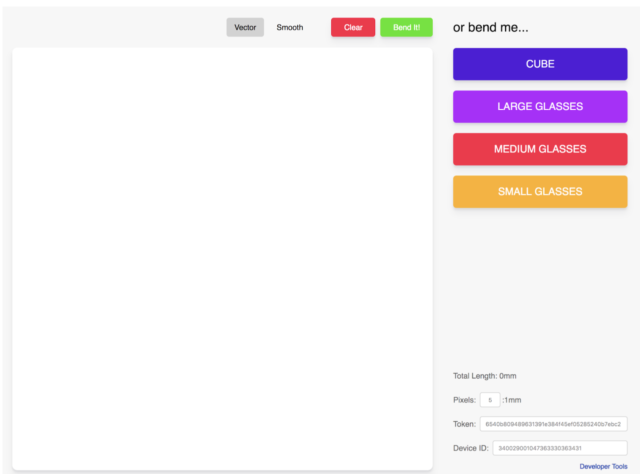

Figure 1: The web application we developed in order to interface with the bender. The large white area is where a user can draw a 2D design they wish to be bent. On the right side are four preset sets of 3D model instructions.

The first component we designed and tested was the bender. We envisioned a bender with 3 core requirements; be strong enough to bend our 1/16" welding wire, be able to bend 90° in either direction, and be able to retract or move out of the way of the wire as it is fed by the feeder. Due to the fact the we had a very strong servo, the strength requirement of the system relied heavily on weak points that existed in the multiple mounting and mating pieces we created. We iterated on these parts multiple times each to ensure their strength and precise measurements were adequate. The full bending head and servo were secured to our platform with a stacked layer of laser cut acrylic rectangles. We elected to use stacked acrylic because of its strength and the ease with which we could adjust the stack's height. The bottom piece of this stack of acrylic was a single larger piece with mounting holes for our servo.

The servo came with a number of servo horns (small plastic attachments to be placed on the servo shaft). Initially, we had screwed a screw directly into one of the holes on the servo horn. The screw would then come in contact with the wire as the servo turned and would bend the wire. We quickly found that the servo horn was too weak for this, and did not allow us to bend accurate angles. We also needed to mount a solenoid to the servo somehow (more details on what the solenoid is used for later), and the screw into the servo horn did not provide a solution to this.

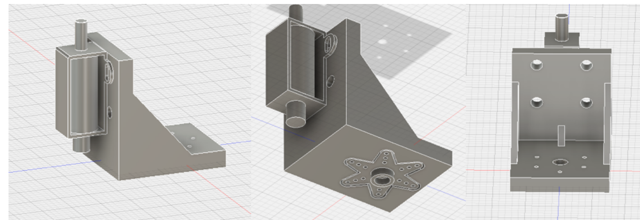

Figure 2: Multiple angles of the L bracket used to secure the solenoid to the servo horn.

To resolve the issue with the screw, we 3d printed a bracket that attaches to the servo horn (Figure 2). This bracket allowed us to screw the horn to the bender head on multiple locations, and thus removing any weakness we had when bending with a single arm servo horn and the screw. This bracket also allowed us to attach a solenoid to the servo. This solenoid served as the bending head and was what actually pushed on the wire to bend it. The solenoid can retract and protrude on command, and has approximately ¼" of range between these two settings. This allowed us to bend in both directions because, depending on which direction we wanted to bend, we simply retracted the solenoid, moved the servo to the other side of the wire, and then protruded the solenoid and continued bending in the new direction. The servo was a 180° degree servo, and thus was always capable of bending 90° in both directions. This simultaneously allowed us to feed more wire to be bent without the wire running into the bending head. This design worked well in testing, and we solidified the design by calculating actual vs. expected bending angles based on the geometry of the servo and the bending head.

We made the decision to build out one of our secondary goals, which was to bend wire in 3 dimensions instead of just two. This decision required us to add an additional component, the 3D rotator. Hence, why 1 of our 3 primary components of our final prototype is the 3D rotator. The 3D rotator allowed the bending mechanism to rotate around a third axis, this axis being the wire itself. To rotate the bending mechanism, we originally designed a geared system with one gear being mounted on our central feeding tube, such that the wire would be the center point of this gear. The other gear was mounted onto a stepper motor, which was offset from the feeding tube. Our initial design for this system involved 3D printing mounting pieces to mate our stepper motor to pre-purchased gears. This immediately hit obstacles, because the purchased gears had mounting holes that were very imprecise, which would have made 3D printing very challenging and unreliable. Before even spending the time to 3D print these parts, we moved immediately to laser-cut gears. Laser-cutting the gears meant that we could directly size the fit and mating of both gears. After a handful of sizing iterations, we arrived at a set of gears that fit snugly with slight sanding, and secured these with Loctite glue (Figure 3 & 4).

Testing of this setup alleviated worries that the gears might not be strong or secure enough to rotate the (relatively) heavy bending mechanism. The gears performed as expected, and handled the torque. The final part of this implementation was to calculate step-to-angle ratios based on our gearing and the properties of the motor. To do this, we simply used our known gear tooth ratios and the existing motor information. Once we calculated the translation from steps to degrees of rotation, we were able to fully test our prototype's 3D bending capabilities. We could rotate to any angle with a high degree of precision, and the entire system affected by 3D rotation is more than capable of handling the load such rotation imposed.

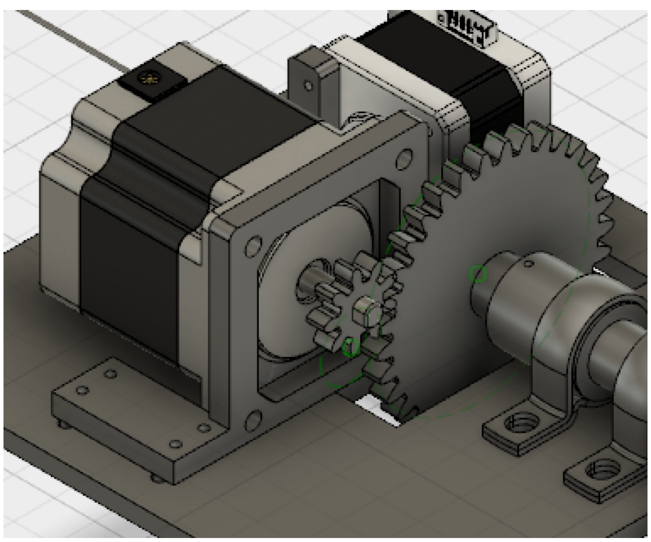

Figure 3: A close up of the model of the 3D rotator motor and gears assembled with our final design.

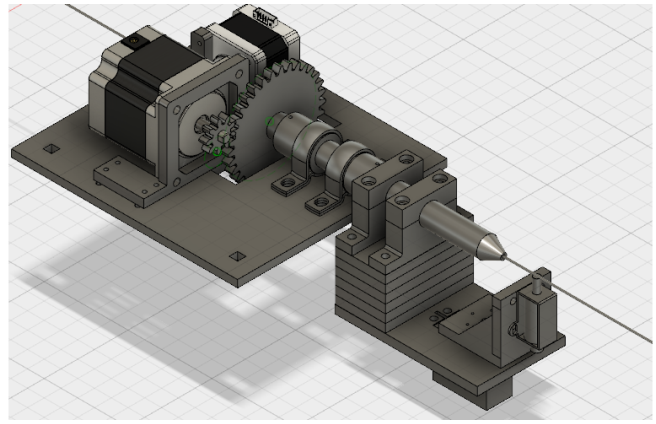

Figure 4: A zoomed out perspective of figure 1 to conceptualize the 3D rotator in our final design.

After making the design decision to use straightened wire, we created the first implementation of the feeder component. This initial implementation had two sub components. One piece was a 3D printed wheel, mounted on a stepper motor, which acted as the driving force to feed the wire. The second piece was a bearing, mounted on a long bolt, and placed directly adjacent to the 3D printed wheel. By making the outer face of the 3D printed wheel concave, we were able to place the two pieces together such that the bearing was flush to the concave face of the wheel. Our thinking was that the tight fit, along with friction tape wrapped around both pieces, would be enough friction to consistently feed wire through the pieces and to the bender. As soon as we began doing test-feeds of wire, we realized that here wasn't enough friction from the tight fit to feed a consistent amount of wire with every step; the wheel and bearing would often slip on the wire when the stepper activated.

With advice from our TA Olav, we looked to the feeding mechanism of the PrintrBot for inspiration in redesigning our feeder. The PrintrBot uses a heavy spring-loaded mechanism and toothed aluminum wheel to generate lots of friction on the material being fed. We were fortunate enough to receive a fully functional PrintrBot feeding mechanism to test with our aluminium wire. One concern that we had with using such a feeding mechanism was that the metal teeth of the PrintrBot extruder would not be able to bite into the aluminum wire as it does with filament for 3D printers. This concerned us because we assumed the relative hardness of the metal teeth to the printer filament is the what generates friction and enables it to extrude the filament. These concerns were alleviated, though, when we found the metal teeth are hard enough relative to the aluminum wire to bite into the wire as well, thus generating enough friction. After again mounting the mechanism to our stepper, we tested the setup.

One unforeseen integration problem that this new extruder introduced was a slight rotation of the wire with every step of the extruder's stepper motor. This was an issue because it effectively mapped any 2D model we were bending into a relatively unpredictable 3D space, thus making our bender inaccurate. We thought we'd avoid such a problem by simplifying the source mechanism of our wire with our straight strands rather than our spool; ironically, had we gone with the spool approach the weight of the spool would likely have prevented this slow rotation of the wire. Nonetheless, we thought the PrintrBot extruder was our best option, and we liked the idea of using already straightened wire. So, we designed and 3D printed a piece that we could attach to each wire that we feed into the machine to stabilize it and prevent rotation (Figure 5).

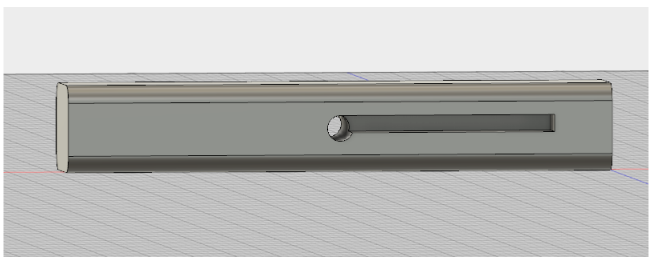

Figure 5: The wire stabilizer.

Once we resolved the issue of rotating wire, the PrintrBot mechanism worked perfectly for all of our criteria. It reliably fed the wire, it was fabricated to fit with an existing stepper motor shaft, and it was spring loaded such that new wire could be loaded with a simple lever press. Once we approximated the amount of wire extruded per step of the stepper motor, we were able to extrude wire with a high degree of precision. This solved all of the existing issues, and so we settled on using this setup in our final implementation.

Videos

Feeder - https://youtu.be/d37LDRYocrw

Bender - https://youtu.be/DfYo5F-UPMw

3D Rotator - https://youtu.be/MCQfxSHRWU0

Cube Demonstration - https://youtu.be/cv3JMS-BVlY

Schematics

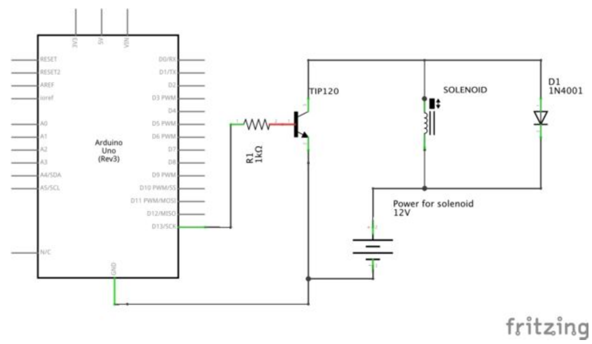

Figure 6: Schematic showing how our solenoid is powered and controlled.

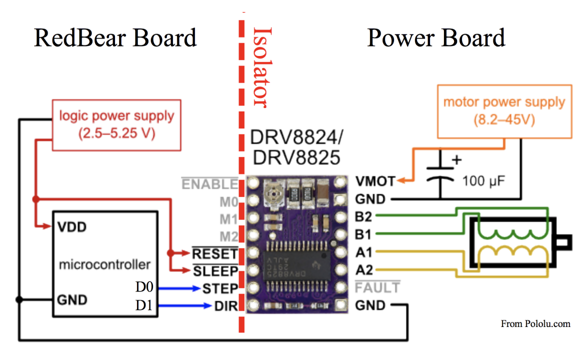

Figure 7: Schematic showing how our two stepper motors are powered and controlled. Both of our stepper motors were powered by 8.2 volts.

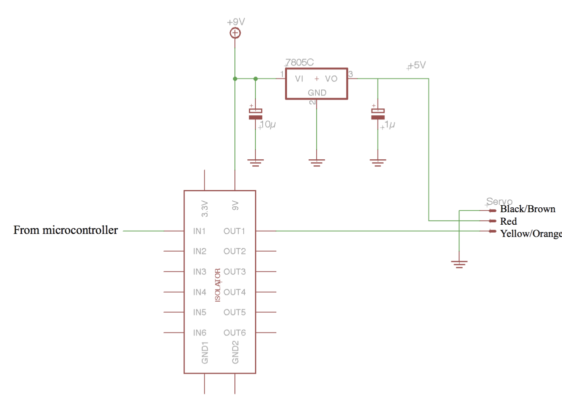

Figure 8: Schematic showing how our servo motor is powered and controlled.

FUTURE WORK

Though we were happy with our final prototype there were some areas for improvement that we're aware of and these areas provide a platform for future work in this space. The focus of our project was on the hardware aspect of wire benders, but the future work to be done both for our project and the space in general involves both hardware and software.

As far as software goes, there are multiple areas for improvement for our prototype as well as the industry standard as far as we understand it. For our software, the first, most basic next step for us is to implement a GCode interpreter for 2D GCode files. While 3D GCode is an additional order of magnitude of complexity because of the need to be aware of where the model will be in space, 2D GCode is a feasible first step to expand the capabilities of our prototype. Extending this to 3D would likely require the development of an algorithm that computes the optimal path of bending given an arbitrary GCode file. In this context the optimality of such a bending sequence would have to take into account where the physical model will be in space with respect to the bender, in order to avoid bending the model into the machine. Additionally, it would also need to take into account how close the computed path is to the actual model represented by the GCode file. With these two additional wrinkles, the problem becomes much harder, both in practice and from a Computer Science perspective. Being able to keep track of a 3D model in space in relation to the wire bender is a graphics problem at heart and would require software of the same order of complexity as 3D modeling software. Finding the optimal path given that information of 3D space is also likely an NP-Complete problem, in the same vein as the traveling salesman problem. That being said, this problem there is a set of 3D models for which this problem is possible, and creating software that can compute the optimal path would be a major contribution to this space.

If we assume that we could interpret 3D GCode models properly via the assumptions in the above paragraph, then there is another extension to this space that applies directly to our prototype. The drawing interface we created only supports 2D drawing, but there is software that exists that supports 3D drawing as well. Thus, with a 3D GCode interpreter developed for wire benders adding a 3D drawing interface is a natural extension to wire benders as well.

The extension of work to the software in the space of wire benders is generally applicable to wire benders in general, but the future work for hardware we will explore specifically through our prototype.

The first clear improvement we could make to our wire bender is to support spools of wire rather than needing to feed in individual pieces of straight wire for each wire frame we want to create. Though we were satisfied with using straight pieces for our prototype, needing to feed in a new piece of wire for every print is not very user friendly. This bottleneck also limited the amount of models that our bender could create because the length of our wire was limited as well. Having the capability to process spools of wire would theoretically allow our machine to process models of arbitrary length.

Another area our prototype could be improved is with the bender mechanism. Our prototype was limited by a defective servo that had a smaller range of angles than was advertised; nonetheless, our bender was only capable of making 90 degree bends in either direction. We could have increased this range by either getting a non-defective servo, moving the bender head closer to the nozzle on the steel tube, or by only bending in one direction and using our additional axis of rotation to allow for us to bend in any direction we want. We considered implementing the last idea for our final prototype, but we were concerned that enabling sharper bends would increase the likelihood that a model is bent into the wire bender which can cause the bend to fail. Thus, we saw this as an issue intrinsically related to the software problems we spoke about earlier, and that exponential increase in complexity was too much for our final prototype. But, this is a clear place where our bender could be improved.

Another issue that we were running into with our bender head was related to the range of motion of the solenoid that comes into contact with the wire. As we bent longer models, sometimes the weight of the wire would cause the wire coming out of the steel tube to come into contact with the solenoid even when it was in the down position. When this problem occurred, unexpected things happened and the model we were bending would fail. One clear way to improve this aspect of our design would be to use a solenoid with a larger range than the one we used. This problem also hints at a deeper issue with all wire benders we've encountered, though, and is a potentially interesting space for wire benders to be improved. None of the wire benders that we found with our research had any sort of support mechanism for the model that was being bent. We can't be certain if other wire benders run into the same issue of the wire weight disrupting the model being bent, but their hardware does not account for that issue. Thus, we think it's reasonable to assume that even industrial wire benders have failure cases related to this issue. So, we think that some mechanism that could stabilize and support the weight of the model being bent has the potential to improve the quality of wire benders in general, particularly those that are optimized for thinner, weaker, more malleable wire. Whether this support would come in the form of a platform or arms that move with the model given the instructions being processed, our wire bender certainly would be improved by this additional support. It's possible that this problem presents a deeper issue with the current design paradigm for wire benders. The current paradigm is quite similar to 3D printers in that both printers and wire benders extrude some amount, and then move mechanisms on different axes to create shapes. Perhaps this paradigm is ill suited to deal with wire in general, and a different approach entirely would provide a significant improvement in the space of wire benders. One way to break out of the paradigm would be to extrude the entire length, or larger portions, of wire needed for a model to be bent, and then have 1 or multiple bending mechanisms manipulate the length of extruded wire onto a platform of some sort. Though we don't have a clear idea of exactly how to improve on the current design paradigm, through our experience it's clear there is a deficit that this paradigm presents. If one considers [14] or [15] as the upper bound for the complexity of wire models that could be created, then there is a clear gap between the state of the art wire benders and the upper bound of wire models that can be created. We know that our prototype isn't yet capable of creating models as complex or heavy as those, and we're nearly certain there aren't any wire benders yet in existence that can create them either. Perhaps the best way to create a machine that is capable of creating models on that order of magnitude of complexity is to break out of the design paradigm that is currently in place.

REFERENCES

1. AIM, Product Page http://www.aimmachines.com/products/
2. AIM, Common Features of Wire Benders, http://www.aimmachines.com/products/
3. BLM Group, Wire Bending Machines, http://www.blmgroup.com/en/wire-bending
4. John Baichtal, (12 May 2012), Arduino-Powered Wire Bending Machine, https://makezine.com/2012/05/03/arduino-powered-wire-bending-machine/
5. Make it Extreme, (25 Jan 2017), Homemade Roller Bender https://www.youtube.com/watch?v=EaIqlG1vHs8
6. Master Spring and Wire Form Co., Wire Form Technical Design Resources, http://www.masterspring.com/spring-wire-form-technical-resources/wire-form-resources/
7. Maciej Majewski, (11 May 2017), Wire Journey, https://www.youtube.com/watch?v=OXRr3_msfd4
8. Nargesa, (19 Dec 2012), Horizontal Hydraulic Press Brake Nargesa PP200, https://www.youtube.com/watch?v=TEPEwYjt08s
9. Pensa, (2015), DIWire Bender Tutorial, http://www.instructables.com/id/DIWire-Bender/
10. D.I.Wire, Pensa Labs, D.I.Wire Specs https://static1.squarespace.com/static/590cc5e3f7e0ab1d6d7cd88d/t/59b06e45f14aa1a0007faa0b/1504734790542/_DIWire-Comparison-8_5x11_05242017.pdf
11. Pensa, (6 Feb 2017) , Introduction to D.I.Wire Pro https://www.youtube.com/watch?v=KbD6qDIsmpo
12. Simplify3D, 3D Printing G-Code Tutorial, https://www.simplify3d.com/support/articles/3d-printing-gcode-tutorial/
13. Taymer, (2017), Cable and Wire Printing http://www.taymer.com/products/cable-and-wire-printing/
14. Life-size multi-wire Hendrix, by Ivan Lovatt http://www.ivanlovattsculpture.com/portraits-in-wire.html
15. Scale wire Subaru, by Benedict Radcliffe http://www.benedictradcliffe.co.uk/Japanese-Classic
16. 2d Cupid for Tiffany's, by Gavin Worth http://www.gavinworth.com/tiffany.html