Status report 3: Programmatic braided-wire shields in Discovery

Electro Magnetic Applications, Inc.

Prepared by Griffin Kowash

8 June 2023

**Objective:** To develop a tool to programmatically generate braided-wire shield models using the Discovery scripting API and evaluate transfer impedance properties.

**Status summary:** Helical braids can be programmatically constructed along arbitrary paths defined by parametric functions or sets of cartesian points. An “equidistant” option is available to reduce geometry distortion introduced by path parameterization. Validation against a reference model is ongoing.

**Updates since previous report:**

* Braids can be built along paths specified by the user, rather than being restricted to the z-axis.
* An optional “equidistant” setting can be toggled to eliminate distortion when the braid is constructed along a path.
* Code is available on GitHub to facilitate version control.

**Validation status:** Results from complete simulations of reference and test cases continue to show moderate discrepancies for a Gaussian current pulse. A 1 MHz sine wave current is currently being simulated to help diagnose the issue.

**Parameterized paths:** An example of a parameterized path geometry is shown below.

A heart drawn on a grid

Description automatically generated with low confidence A picture containing text, sketch

Description automatically generated

**Figure 1:** Full (left) and zoomed (right) view of a cable braid constructed along a parameterized path.

The script can construct a path using either a user-defined parametric function or a pre-calculated set of points.

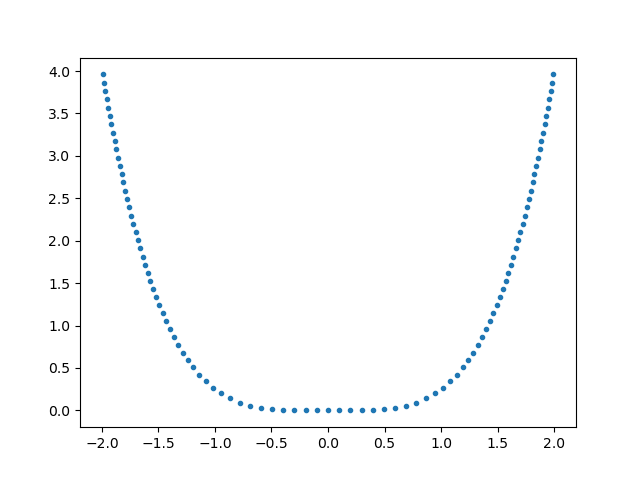
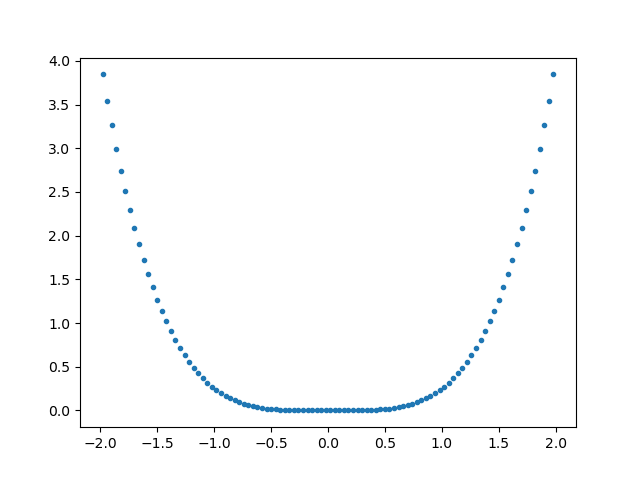
A problem presented by this feature is that braid dimensions can become distorted, since the “speed” at which a point travels along a parameterized path can vary from location to location. To address this, an “equidistant” option is available that re-models the path as a series of points with a constant separation in space. This allows the specified dimensions of the braid to be maintained along the full path.

A picture containing sketch, child art, drawing, screenshot

Description automatically generated A picture containing text, sketch, screenshot, design

Description automatically generated

**Figure 2:** Standard (left) and equidistant (right) parameterizations of a quartic curve. In the standard model, the rhombic gaps between carriers become longer further along the arms of the U-shape; in the equidistant model, the gap sizes largely remain the same.

****

**Figure 3:** Points composing the braid path shown in Figure 2. Points in the standard parameterization (left) are tightly spaced near the bottom of the curve and increasingly separated further away. In the equidistant parameterization (right), points are equally spaced.

The equidistance algorithm retains a small amount of error in regions of high curvature, causing neighboring points to be placed slightly closer or further than they should be. This error can be decreased by using a higher curve resolution and is unlikely to be a concern for normal use. The standard deviation in point separation for a quartic curve is provided below for different resolutions.

**Table 1:** Standard deviation in spacing between neighboring points along a path using the original and equidistant methods. The error is most prominent in regions of high curvature and decreases rapidly with increasing resolution.

|  |  |  |
| --- | --- | --- |
| **Curve resolution** | **Point spacing std (original)** | **Point spacing std (equidistant)** |
| 5 | 0.656 | 0.0159 |
| 10 | 0.541 | 0.00730 |
| 20 | 0.330 | 0.00205 |
| 50 | 0.148 | 1.51e-4 |
| 100 | 0.0765 | 1.94e-5 |
| 1000 | 0.00789 | 1.98e-8 |

**Next steps:**

* Evaluate simulation results for sine wave current.
* Make geometry measurements along equidistant parametric paths to verify that the equidistance algorithm maintains the specified braid characteristics.
* Investigate integrating path algorithm with Discovery sketching curves.
* Add surface plotting for parameterized braids.
* Integrate new functionality with Discovery API script.

**GitHub repository:** <https://github.com/GriffinKowash/Cable-braids>

**References:**

Vance, E.F. (1974.) *Shielding effectiveness of braided wire shields*. Stanford Research Institute, Interaction Note 172.