Spreadsheets as Notational Environment for Paper Weaving

Jonathan Skjøtt
jonathan@jskjott.com
Takram
London, United Kingdom

ABSTRACT

This work explores how spreadsheets can serve as the foundation for the creation of a hybrid design medium for paper weaving. The case for using spreadsheets as a model for the development of environments for designing paper weaving patterns is made. Firstly, because both spreadsheets as well as the woven form of paper weaving designs have cells as their elementary substrate. Secondly, because it allows for exploration of the design space. And thirdly, as it enables use by end-users with varying levels of experience. An environment, VisiWeave, implementing the points mentioned is presented and explored.

CCS CONCEPTS

• **Human-centered computing** → *User interface programming.*

KEYWORDS

spreadsheets, weaving, end-user programming

ACM Reference Format:
Jonathan Skjøtt. 2020. Spreadsheets as Notational Environment for Paper Weaving. In *Companion Proceedings of the 4th International Conference on the Art, Science, and Engineering of Programming (<Programming'20> Companion), March 23–26, 2020, Porto, Portugal.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3397537.3397545

This paper first gives an introduction to paper heart weaving as it is practiced in the Nordic countries, then it will introduce and explain a prototype notational environment, VisiWeave, which exemplifies how spreadsheets can be useful as a model for making tools for creating paper weaving designs.

1 THE CRAFT OF WEAVING PAPER HEARTS

Woven hearts are made by interlacing two double-layered pieces of paper, each of which had grooves cut into them following some pattern. When two such pieces of paper are woven together, a pattern will arise formed by the combination of the two separate pieces. The "hello world" of woven hearts forms a 3 x 3 grid criss-cross pattern. The process of making this heart is shown below.

Looking at fig. 1 we can see that all hearts have an unwoven and a woven form. The process of making always involves first creating two instances of the unwoven forms and, secondly, manifesting the final form by combining the two unwoven forms.

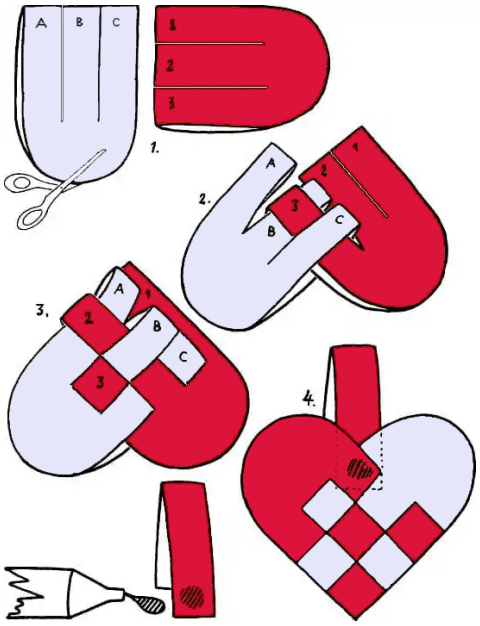

Figure 1: A popular depiction of the process of weaving - the "hello world" heart. Notice the row and column designations of the two forms being combined.

In other words, when making woven hearts one has to work with two distinct levels of abstraction or representation. Each of the two representations contains all the information to create the other representation, though it, at times, can be hard to deduce how one representation relates to the other.

Currently, the biggest challenge associated with designing a new weaving pattern lies in trying to visualise what the woven form will look like based on the unwoven forms. The current design process mostly consists of drafting several versions of the unwoven forms while trying to figure out what they will look like when

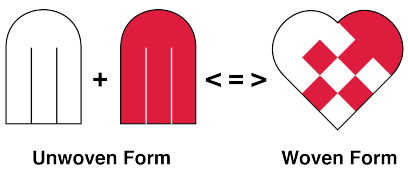

Figure 2: The two forms woven hearts can take.

<Programming'20> Companion, March 23–26, 2020, Porto, Portugal

ACM ISBN 978-1-4503-7507-8/20/03…$15.00
https://doi.org/10.1145/3397537.3397545

combined. When you try to make more complex hearts it becomes increasingly hard or unfeasible to visualise what the interlacing of the two unwoven forms will look like.

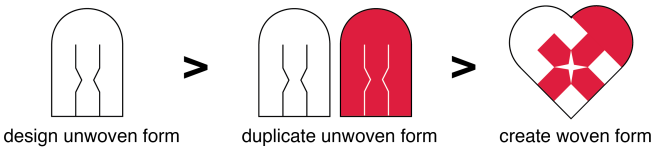

Figure 3: The woven heart design process. The final form of seemingly simple hearts can become challenging to visualize.

The idea to try to and create an environment for creating weaving designs based on spreadsheets grew out of this particular frustration with how the current design process limits one to work with the unwoven forms - it too was a response to not being able to find a satisfying way to navigate the design challenges through the use of physical media.

1.1 Embracing Computational Media to Change the Working Representation

Combining computational media with traditional paper crafts allow for the creation of a hybrid design medium. Merging paper and computation to enable design has a long history within the field of computational origami and has also been demonstrated recently in Hyonjoo Oh and Mike Eisenberg's work on Paper Mechatronics [6]. Oh describes this medium as being a "reappraisal of traditional educational papercrafts in combination with accessible mechanical, electronic, and computational elements".[6]

In the context of paper weaving, as the appearance of the heart in its final form is mostly what matters, computational media can be used to enable the designer to work directly with the woven form. Then, the unwoven form could be generated when the pattern itself is done and all that is left is the weaving activity.

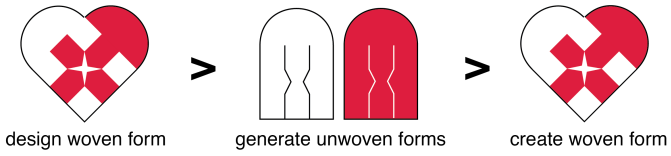

Figure 4: A better design process.

Being able to work directly with the woven form would eliminate the need to try and translate, in one's head or with other tools, between several forms throughout the design process. However, often the problem with working on a higher level of abstraction is that it can be hard to create an accessible interaction model which still allows for a full exploration of the creative space.

The remaining part of this paper is about how spreadsheets as an interaction model enables designers to work directly with the woven form while still having access to a large design space.

1.2 Woven Paper Hearts as Spreadsheet

As is exemplified in figure 1, the unwoven form of a paper heart contains a number of bands. When two forms are woven together they form a grid-like structure with rows and columns. It is possible to assign cell coordinates to the areas in the grid based on which bands they are formed by.

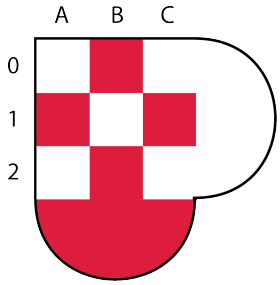

Figure 5: Column and row assignment to the cells of a woven heart.

The combination of the two unwoven forms using the plain weave technique produces a grid-like sheet marked by altering colors. The plain weave is perhaps the simplest weaving technique, the strings or bands criss-cross under and over each other.

With coordinates based on a grid-like structure it is possible to drop the heart into a spreadsheet as has been done in figure 6.

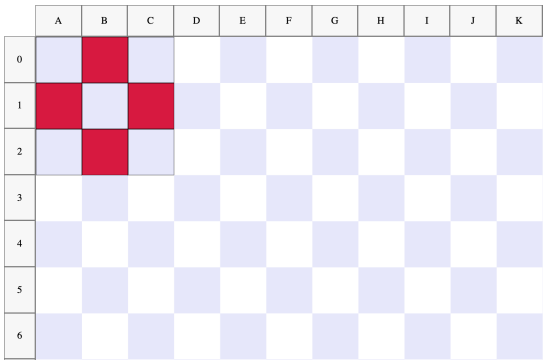

Figure 6: The "hello world" heart shown in the VisiWeave spreadsheet.

Having shown how a paper weaving design can be fit into a spreadsheet, the VisiWeave environment will now be introduced and the reasons for using spreadsheets will be explained along with the relevant parts of the prototype.

2 VISIWEAVE - A PROTOTYPE NOTATIONAL ENVIRONMENT FOR PAPER WEAVING

VisiWeave is a prototype made to explore and illustrate the potential of using spreadsheets as a notational environment for paper-weaving design. The name of the prototype is inspired by VisiCalc, the first spreadsheet software whose release in 1979 sparked great change in many domains. Hopefully, like VisiCalc was for

calculation, VisiWeave will be the first of many spreadsheet-like environments made for weaving.

Just like in spreadsheet software, VisiWeave has a vast expanse of cells known as the tabular canvas as well as a formula bar. In addition to these two interface elements the environment also has an expandable panels on the left and right side of the interface: on the left is an interpreter for declaring functions and macros and on the right is an export pane with the woven and unwoven forms as well as a button for printing out the unwoven form when it is time to physically weave the design. Both the formula bar and the interpreter accepts a minimal dialect of LISP.

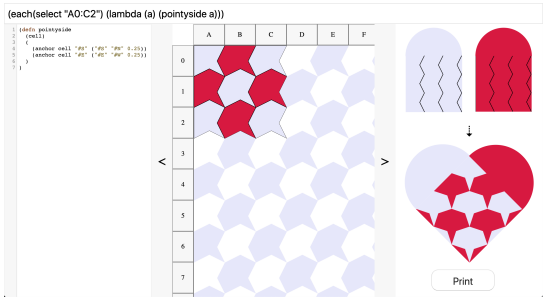

Figure 7: The VisiWeave interface with all panels open.

Three main concerns drove the decision to make spreadsheets the main source inspiration in designing VisiWeave. Firstly, both spreadsheets as well as the woven form of paper weaving design have cells as their elementary substrate, secondly it allows for exploration of the design space, and thirdly it enables use by end-users with varying levels of experience.

2.1 The Cell as the Notational Unit

The core difference between between the ontology of a cell in a regular spreadsheet versus in VisiWeave lies in how cells can take on a larger variety of shapes. Cells do not have to be rectangular. The core idea here is that the shape of any cell in VisiWeave is determined by the value of the cell.

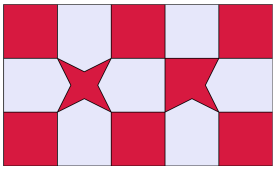

Figure 8: In VisiWeave the shape of cells do not have to be rectangular.

A significant part of creating a notation relates to the way in which cells are shaped. As this work is being done with the end goal of making a weaving environment which is accessible to end-users the notation needed for shaping cells should, ideally, be easy to understand and work with. The way it was approached in this work was inspired by Islamic star patterns geometry where patterns are created through interactions between rays generated from select points on a grid [4]. The existing system is based on the cardinal directions being laid out on a compass rose. In any given cell there are 8 reference points laid out like a compass. Changes to the sides of the cell will be made referring to the lines which are drawable between the points.

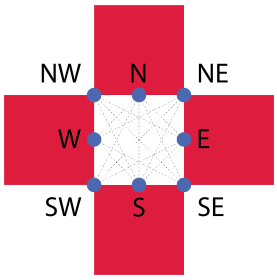

Figure 9: A representation of the reference point system.

To change the shape of one of the cell's sides one can provide a letter, say N for north, two reference points, say W and NE, and where in between those two points, starting from the first mentioned reference point, a point should be added.

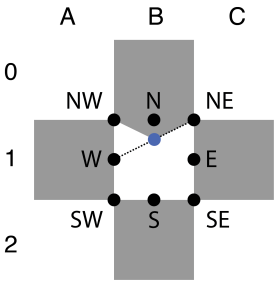

Figure 10: Placing a point halfway between W and NE, altering the upper side of B1.

In the prototype environment this change would be made by calling the function below.

The anchor function takes a cell name, a cell side, and list containing two reference points and a fraction. Then, starting from the first reference point, the line between the two reference points is traversed before a point is added to the path forming the selected side when the fraction of the distance between the points has been traversed.

2.2 Enabling Exploration

Many of the functionalities of spreadsheets enable exploration of the space of possible weaving designs. Many of these are based on the tabular canvas and how it invites addition, editing, and play. There is always space for another set of rows and columns, enabling another exploration or experiment. In VisiWeave this means that several designs can live in the same sheet.

There are also ways the spreadsheet model has been augmented to support exploration in ways which go beyond the affordances of regular spreadsheets. For example, selection in VisiWeave also determine what is considered for printing or export.

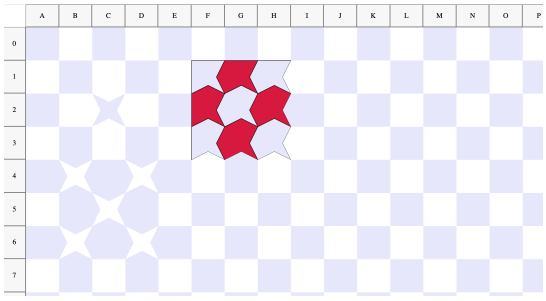

Figure 11: The tabular canvas can contain many weaving designs at any given time.

As can be seen in figure 12, only selected cells are rendered in the export pane. In figure 12, the cells J0 through L2 are selected, this means that the woven and unwoven form seen in the export pane are the forms produced if only those nine cells were to be used for a heart design.

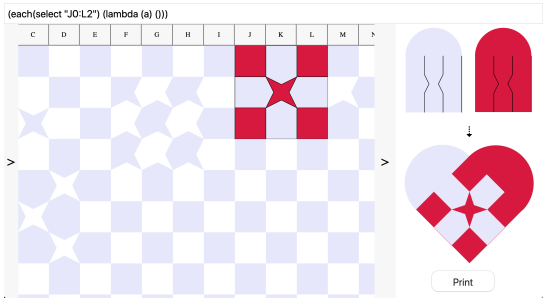

Figure 12: When a part of the VisiWeave spreadsheet is selected the woven and unwoven parts are rendered in the export pane.

2.3 Making Environment Usable by a Diverse Set of End-users

The environment was made to cater to both novel and more experienced users in how the interpreter invites long form programming as contrasted with the formula bar inviting the input of concise chunks of commands.

Novice users can do just fine without using the interpreter - they will have access to a large expressive space without having to write their own functions. However, they always have the options of opening up the interpreter, which accepts the same language as the formula var. The interpreter allow more experienced users to write their own functions and macros which they can then share with other users.

2.4 Generating Unwoven Forms

It is not too hard to translate the digital representation of the woven form to into the unwoven forms. It is a matter of iterating through the cells and adding each of the four sides of the cell to the corresponding unwoven representation.

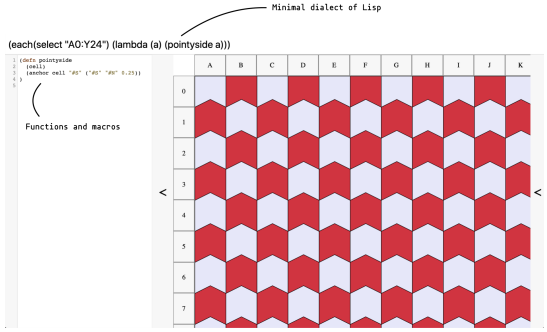

Figure 13: Both the formula bar and the interpreter accept a minimal dialect of LISP.

3 DISCUSSION AND FUTURE DIRECTIONS

3.1 Relationship to Other Kinds of Weaving

This paper has presented an environment for making woven paper heart designs. Paper weaving is different from other types of weaving in a few ways. Generally less strings are used in paper weaving - the "hello world" heart uses 6 strings in total which is much less than the number of strings used by even the most compact fabric looms.

VisiWeave has so far not worked with other weaving techniques than the plain weave. Altering the order of interlacing of strings is the main way other kinds of weaving produce unique styles. Other scholars, such as Lea Albaugh [1], have made excellent strides in illustrating what is possible when computational tools are used to change the order of interlacing in fabric weaving.

3.2 Weavability

Currently the system does not guide the user in telling them how hard it will be to physically weave the design they are creating. This means that users might design patterns which are unfeasible or simply impossible to put together in real life. These issues would mostly arise because the columns or rows become to thin at places - making it impossible to weave without tearing the paper.

3.3 Notatinal for Changing Interlacment

The current prototype does not allow the user to change the interlacement pattern of their design. The user is forced to see their design as it would look if woven with a plain weave. Work is needed to explore how the weaving pattern might be manipulated in a spreadsheet-based interface.

3.4 Spreadsheets for Visual and Generative Design Work

There are a number of visual tasks which might benefit for a spreadsheet-like interaction model. One might look at the extensive history of grids and geometric patterns within design and visual art and see how spreadsheets could serve as the foundation for the creation of generative design tools.

3.5 Recursive Spreadsheets and Other Extensions of the Medium

Alongside the exploration of which other visual domains VisiWeave might be useful in, there is a need to consider the ways in which spreadsheets should change in response to more visual tasks. One exploration which is needed is that of having sheets inside sheets representing the nested structures found in both weaving and visual grids.

3.6 Bi-directional Evaluation

A promising direction of work for this environment would be to draw inspiration from some of the many visual programming environments implementing bidirectional evaluation such as Ronin by the design studio Hundred Rabbits [2], Sketch-n-Sketch by Mikaël Mayer and Ravi Chugh [3], or Libfive by Matt Keeter [5]. These tools try and make it possible to generate code based on interactions with the graphical interface, but likewise allow interact with the interface by means of writing code. Like VisiWeave, many of the tools use LISP as the end-user facing language and work with many of the same geometric primitives.

3.7 Improving the Notation

The notation introduced in this paper is still a work in progress. I was not chosen based on an extensive consideration of the different forms a notation might take. Therefore, a future line of work could more deeply consider what ontology might make most sense - this is particular relevant for cells.

3.8 Engage and Document Usage

VisiWeave as an environment has currently only been tested within the small community in which it was developed. A promising direction of future research would be to see how a diverse set of end-user communities might use the tool.

ACKNOWLEDGMENTS

I want to thank Xiaofan Liang for providing invaluable feedback throughout this process.

REFERENCES

[1] Lea Albaugh. *Weaving Notation*. Sept. 2018.
[2] Rekka Bellum Devine Lu Linvega. *Ronin Software*. 2017. URL: https://hundredrabbits.github.io/Ronin/.
[3] Brian Hempel, Justin Lubin, and Ravi Chugh. "Sketch-n-Sketch: Output-Directed Programming for SVG". In: *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* (2019).
[4] Craig S. Kaplan. "Islamic Star Patterns from Polygons in Contact". In: *Proceedings of Graphics Interface 2005*. GI '05. Victoria, British Columbia: Canadian Human-Computer Communications Society, 2005, pp. 177–185. ISBN: 1568812655.
[5] Matt Keeter. *Libfive Software*. 2018. URL: http://libfive.com/.
[6] Hyunjoo Oh et al. "Paper Mechatronics: A Design Case Study for a Young Medium". In: June 2015. DOI: 10.1145/2771839.2771919.