INTERACTIVE GRAPHICS AND THE REPRESENTATION OF NON-CARTESIAN WOVEN TEXTILE STRUCTURES

J. A. Hoskins and M. W. King
Department of Computer Science and
Department of Clothing and Textiles
University of Manitoba
Winnipeg, Manitoba

ABSTRACT

A microcomputer based interactive system for the creation and display of the data file corresponding to a non-cartesian woven textile structure is examined. Comparisons with the system for representing conventional cartesian fabrics are made and illustrative examples are provided.

Nous présentons dans cet article un système interactif, sur micro-ordinateur, de création et de représentation graphique de fichier de données simulant un textile tissé à structure non-cartésienne. On fait la comparaison de ce nouveau système avec celui de la représentation cartésienne des tissus conventionnels. Quelques exemples illustrent cette comparaison.

1. *Introduction.*

Conventionally woven structures are formed by the interlacement of two sets of strands which lie orthogonal to one another, with the resulting intersections of these two strands lying in a cartesian plane [Figure 1]. Traditionally, the data structure for this intersection information has been formulated as a matrix of black and white squares, called a point diagram [Figure 2], representing the only two possible types of intersections. A microcomputer interactive graphical display system for the analysis and visual representation of the design data has been developed and described [1]. This interactive graphical environment has however dealt exclusively with cartesian structures in which all of the strands in each of the two perpendicular directions lie parallel to one another.

Another interesting class of woven structures consists of those fabrics produced by crossed weaving. In cross woven structures such as leno, gauze [2] and sprang [3], the paths of the lengthwise strands are no longer parallel, in that these strands are allowed to cross over each other between intersections with the crosswise elements. This technique produces a fabric exhibiting a particular stability of structure while still maintaining an open quality and is used extensively for decorative clothing fabric, for industrial uses such as screens and sieves, [4] and for stabilizing the selvedges of shuttleless loom fabric [5].

While the data structure for crossed woven fabrics can still be represented as a binary or bit matrix, a radical re-interpretation of this matrix is required. The purpose of this paper is to examine the characteristics of such a data structure as well as to describe a microcomputer based system for storing and displaying crossed weave design data.

2. *Application Data Structure.*

Crossed woven structures actually consist of two distinct types of design rows, namely interlacement and twist rows. Interlacement rows are ones in which the warp strands intersect with a weft strand which lies perpendicular to them. The structure of the rows can be described using a binary representation, where a "1" corresponds to a warp strand lying on top of a weft strand and a "0" corresponds to a weft strand lying on top of a warp strand.

Twist rows, on the other hand, deal strictly with interactions between adjacent pairs of warp strands, with four types of interaction being possible. Two adjacent warp strands can remain parallel to each other with no twisting taking place, the left-hand strand can twist over the right-hand strand or the right-hand strand can twist over the left-hand strand. The fourth possibility is that the left-hand strand is untouched and the right-hand strand interacts with the adjacent strand to its right. Since only four types of interaction are possible for each pair of warp strands, two bits can be used to represent each warp pair in a twist row.

Clearly, twist and interlacement rows are mutually exclusive. That is, a twist row will contain no warp/weft intersections while an interlacement row will exhibit no warp twisting. Each of these rows can therefore be designated interlacement or twist by a single entry in a binary "twist" vector, where a "1" represents a twist row and a "0" represents an interlacement row.

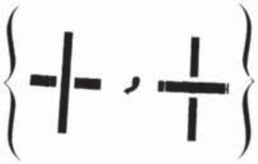
Superficially then, the application data structure for crossed woven fabrics is identical to that for conventionally woven cartesian structures. However, whereas each element of a conventional point diagram can be interpreted completely independently of all the other entries in the matrix, the application data structure is now context sensitive.

Firstly, if the twist vector entry for a given row is "0", then each matrix element in that row is interpreted individually, as one of the two possible warp/weft intersections. Otherwise, if the twist vector entry for the row is "1", then the matrix elements for that row are interpreted as two place binary numbers in decimal form. Further, the elements of a twist row are paired 1 and 2, 3 and 4, ..., $2i-1$ and $2i$ until or unless a pair corresponding to the fourth type of interaction is encountered. At this point, the parity of the pairing changes to $2i$ and $2i+1$, and continues thus to the end of the row or until changed again. Elements in a twist row are therefore also dependent for their interpretation upon all of the previous entries in that row.

3. *Graphical Display.*

The graphical display of crossed woven structures involves a mapping of the bit matrix data structure to two sets of ordered graphic output primitives or tiles, with the value of the twist vector at a given row determining which set of tiles is to be indexed.

Two tiles are required to represent the possible intersections in an interlacement row, as contained in the ordered pair:



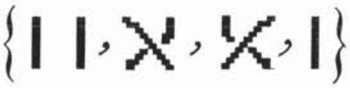
An index (I) into this set of tiles can be computed, using the formula

$$I = d_{i,j} + 1 \quad d_{ij} \in D = \left\{\begin{array}{l}\text{all binary elements}\\ \text{in the application}\\ \text{data structure}\end{array}\right\}$$

$$i \in \{\text{interlacement rows}\}.$$

For example, the binary sequence 1 0 0 1 correspondings to the following interlacement representation:

Four graphic output primitives are required to represent the possible interactions which can occur between warp ends in a twist row (i.e. when the value of the twist vector is "1" at that row). These tiles are given by the ordered 4-tuple:



The index (I) into this ordered set of tiles is computed by the formula

$$I = (2d_{ij} + d_{ij+1} + 1) \bmod 4 + 1$$

$$d_{ij}, d_{ij+1} \in D = \left\{\begin{array}{l}\text{all elements in the applica-}\\ \text{tion data structure}\end{array}\right\}.$$

$$i \in \{\text{twist rows}\}.$$

The fourth interaction corresponds to leaving the first of a pair of warp strands untouched and considering the interaction between the second strand of the pair and its neighbour to the right. This amounts to a change of parity of the column index (j) of the elements d_{ij} in the indexing formula and applies from the middle of the current pair of strands to the end of the sequence or until another of this type of interaction occurs.

For example, the binary sequence 0 1 1 1 0 0 1 0 0 1 1 is mapped to the appropriate twist row graphical representation as follows:

$$\begin{array}{lccccc} \text{SEQUENCE:} & \underbrace{0\ 1} & \underbrace{1\ 1} & \underbrace{0\ 0} & \underbrace{1\ 0}\ 0 & 1\ 1 \\ \text{I :} & 3 & 1 & 2 & 4\underbrace{\quad}_{2} & \underbrace{\qquad}_{1} \end{array}$$

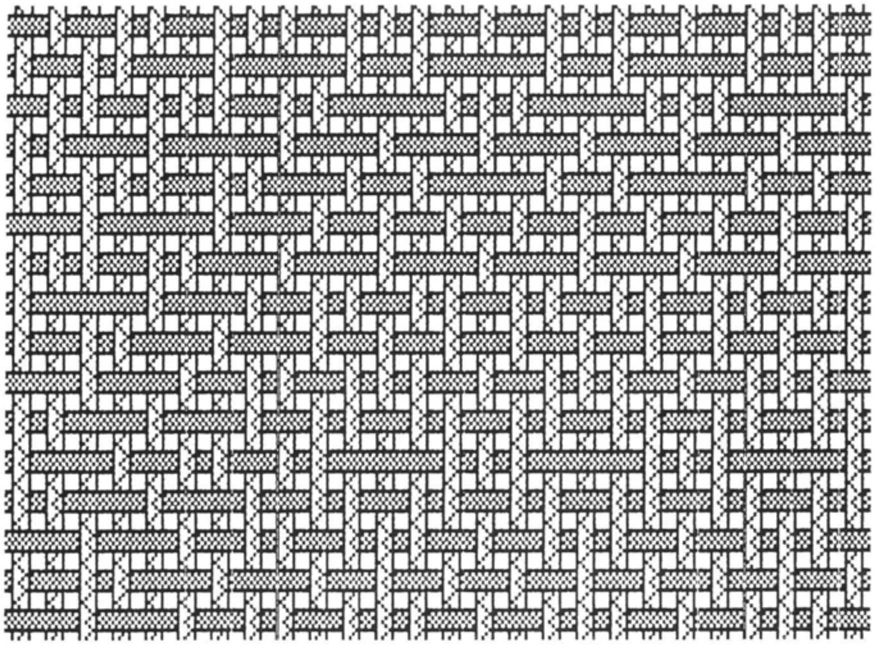


Figure 1

Conventionally Woven Textile Structure

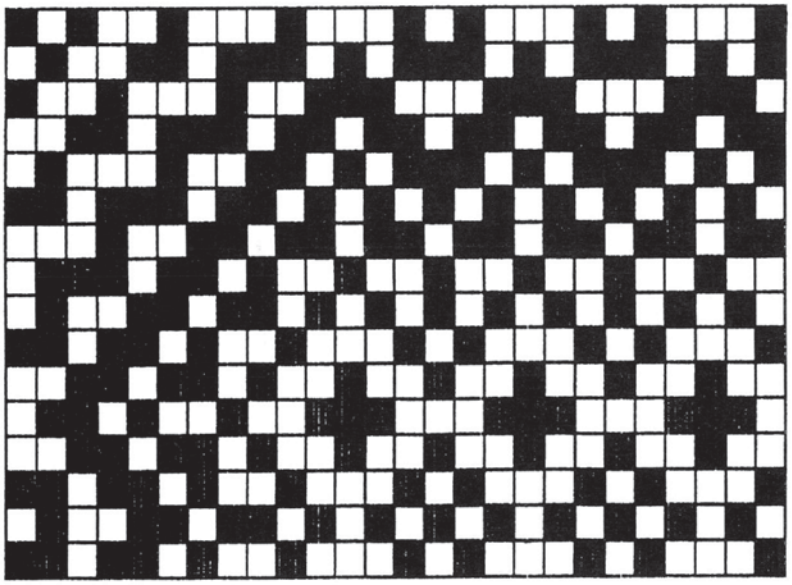


Figure 2

Point Diagram

Clearly, for this operation to be valid, the neighbouring strand cannot already be a part of a twisted pair of strands. This will necessitate a range check on the index value of the right neighbouring pair, if the fourth graphic tile is to be drawn in response to a user specification rather than a mapping of the current data structure. This will be discussed further in the Data Input Section.

4. Data Input.

In the case of conventionally woven structures, the binary matrix data structure provides a visually meaningful representation of the interaction between warp and weft elements. Elements can therefore be entered directly into the data structure. At each stage the user receives appropriate graphical feedback as to what effect the new element has on the overall textile structure. This is not however the case with crossed woven fabrics, where the binary matrix data structure does not provide an immediate graphical representation of the fabric but must be interpreted in a more complex, context sensitive manner. This necessitates an alternative form of data input for crossed woven textiles.

The most appropriate and meaningful method of data entry is to look at a graphics screen and draw the appropriate tiles, using some simple consistent interaction sequence [6], such as key presses. The tile which is chosen is interpreted by the program and the database updated.

Precise placement of the tiles is achieved by moving a cursor around the screen in unit movements, using the standard directional key pad. The width of the units is dynamically determined under program control. A unit move in an interlacement row takes the cursor from one intersection to the next, with the width being the distance between two warp ends. A unit move in a twist row, on the other hand, takes the cursor from between two warp strands to midway between the next adjacent pair of strands, with the width being twice the distance between two warp ends. Twist rows also require the ability to move over half a unit to accommodate changes to the parity of pairing.

There are six possible tiles which can be drawn, two in the interlacement rows and four in the twist rows. Interlacement tiles cannot be drawn in twist rows and vice versa. The conversion of the data structure to a graphic display therefore requires the use of either six keys plus range checking or a single key interpreted in context. The second solution requires less memorization on the part of the user and is the one which has been chosen.

If the cursor is positioned in an interlacement row, as indicated by some symbol such as "+" displayed in that row position in the twist vector [see Figure 3], then a single key can be used to toggle from one possible intersection representation to the other. The key presses thus form a cyclic group of order 2. If the cursor is positioned on a twist row, as indicated by a symbol such as "×" in the twist vector, the same key can be used to cycle through the four possible types of graphic tiles. These key presses now form a cyclic group of order 4, with dynamic re-definition of the fourth element. As previously discussed, the fourth twist interaction changes the parity of the pairing and requires that the pair of warp strands immediately to the right not be twisted. If they are twisted, the fourth key press makes no change to the structure display, but flashes the cursor to indicate that the key press has been accepted.

In addition to updating the structure display, and corresponding database, the twist vector must be addressable as well. Once again, a single key, the same one as before, can be used to toggle between the two types of entries in this vector. The structure display and twist vector are treated as separate input areas. An additional key is used for each of these areas, to indicate where data input is to take place. Cursor positioning and movement is now restricted to the specified area and the data entry key is interpreted in the appropriate context.

Before data entry can take place it is necessary that the screen and database be initialized. This initialization can either be to a previously created and stored structure, in which case data entry corresponds to editing this structure, or to some initial or foundation structure. In the second case, it is convenient to initialize the structure matrix and twist vector to zero--all bits off. This corresponds to a "fabric" where all of the rows are interlacement rows and, at every intersection the weft strand lies on top of the warp strand. This is of course not a single cohesive fabric but is instead a reducible [7] structure in which the strands separate into two disjoint planes [Figure 4].

Obviously, if the twist vector entry for one of these rows is changed to a "×" on the screen and a "1" in the database, the entire structure row corresponding to this entry must be re-interpreted as a twist row and re-drawn. In this case, the entirely zero twist row, corresponds to all warp pairs twisted with the left strand over the right strand. Changes induced by updating the data structure area are,

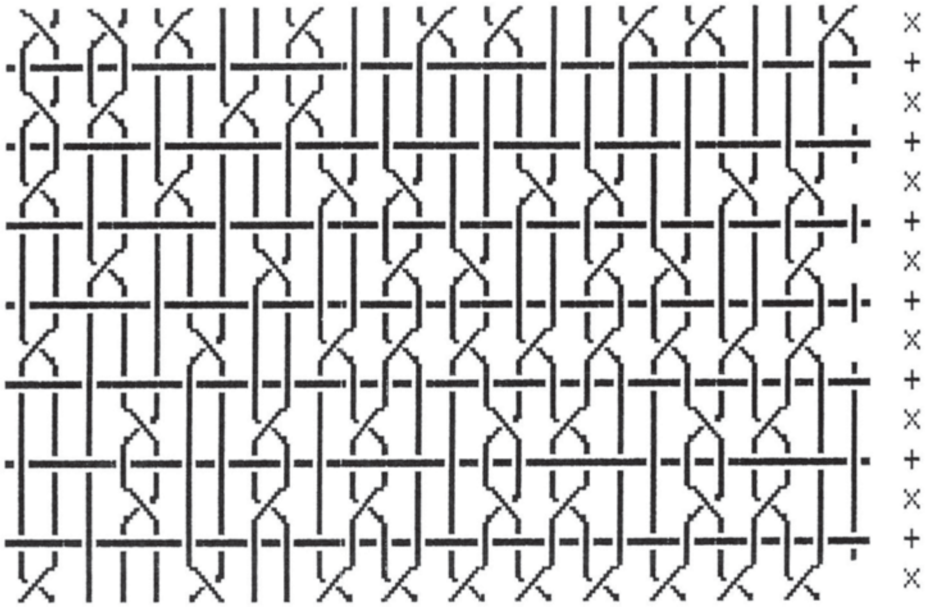


Figure 3

Crossed Woven Textile Structure

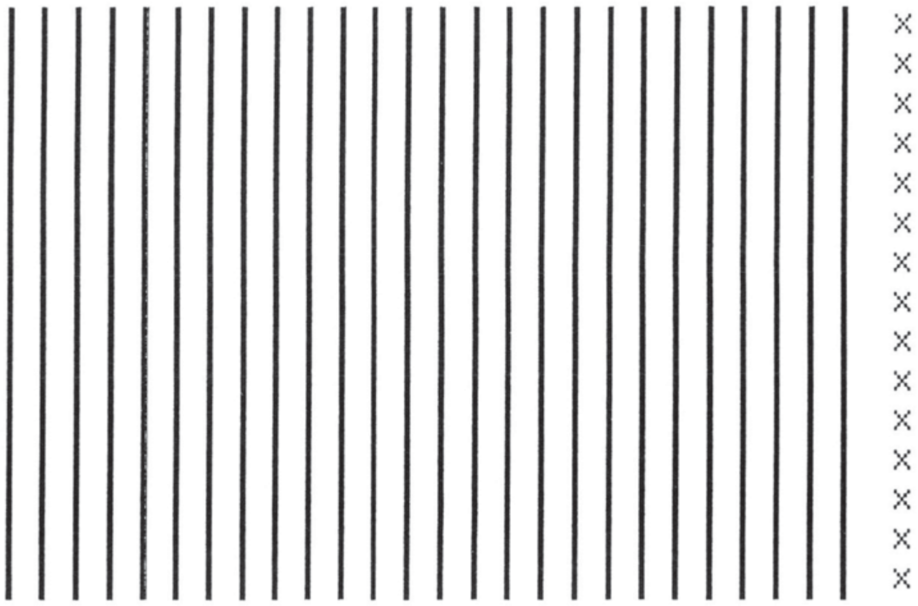


Figure 4

Foundation Structure

on the other hand, entirely local, with the exception of the fourth twist interaction. This action potentially necessitates a re-interpretation of all the data elements in the row, lying to the right of and including the pair of elements being addressed.

5. *Conclusion.*

In conclusion, crossed woven textile structures can be stored efficiently as bit matrices but this type of data structure is not however visually meaningful and is therefore kept invisible to the user. Interpreting the data and displaying the appropriate graphic image represents an interesting problem in mapping data to graphic output primitives, particularly since the mapping is completely context sensitive.

REFERENCES

[1] J.A. Hoskins and M.W. King; *Interactive Design of Woven Textiles;* Proceedings of the International Computer Color Graphics Conference; Tallahassee, Florida; March, 1983.

[2] H. Nisbet; *Grammar of Textile Design;* (3rd ed.); Benn; London; 1927; pp. 211-247.

[3] Peter Collingwood; *The Techniques of Sprang: Plaiting on Stretched Threads;* Watson-Guptill Publications; New York; 1974.

[4] Z. J. Grosicki; *Watson's Advanced Textile Design: Compound Woven Structures;* (4th ed.); Newnes-Butterworths; London; 1977; pp. 207-210.

[5] J.J. Vincent; *Shuttleless Looms;* The Textile Institute; Manchester; 1980; pp. 13-20.

[6] J.D. Foley and A. Van Dam; *Fundamentals of Interactive Computer Graphics;* Addison-Wesley Publishing Company; Philippines; 1982; pp. 55-56.

[7] J.A. Hoskins; *Factoring Binary Matrices: A Weaver's Approach;* Lecture Notes in Mathematics, *952* (Springer-Verlag), pp. 300-326.