Algorithmic Generation of Chinese Lattice Designs

S. Y. Lee and K. M. Tiong

Abstract—Chinese lattice designs commonly appear on traditional Chinese windows, doors and decorative ornaments. The underlying structure of Chinese lattice designs are mathematically interesting and present wonderful opportunities for detailed investigation. A closer look reveals that many of these Chinese lattice designs can be algorithmically designed using some simple procedures. In this paper, we look at the algorithmic design procedure of some Chinese lattice designs sourced from Daniel Sheets Dye’s definitive book on Chinese lattices using the turtle graphics functionality of MuPAD®, a MATLAB® Symbolic Math Toolbox. We also generate random multi-colored Chinese lattice designs.

Index Terms—Algorithmic, chinese lattice design, turtle graphics, MuPAD®.

I. INTRODUCTION

Chinese lattice designs are mathematically interesting structural art. They can be commonly found on traditional Chinese windows, doors and decorative ornaments. Chinese lattice designs are found in Far East countries like China, Korea, Japan and even in Vietnam. Chinese lattices are made of wood, which is not long-lasting, and thus many Chinese lattices have not survived the passing of time. However, evidence of Chinese lattices is found in pictures or reliefs on old bronzes, ancient ceramics and things found in graves. Some physical examples of Chinese lattice designs can be found in [1] and [2]. Interestingly, Chinese lattice designs are now even used in modern architectural designs as a building façade [3].

There are very few publications on Chinese lattice designs. The most important work on Chinese lattice designs is Daniel Sheets Dye’s book [4] which contains the most comprehensive description and collection on Chinese lattice designs constructed between 1000BC and 1900AD. Dye classified Chinese lattices into 26 groups according to the common visual features.

Investigations on Chinese lattice designs have previously centered on the perspective of shape grammar that describes the composition procedures of lattice designs and generates existing lattice designs as well as new hypothetical designs of similar style [5]-[7]. In several papers, Majewski and Wang investigated Chinese lattice designs from an algorithmic [1], [8] and symmetry groups point of view [2], [9]. The algorithmic investigation using MuPAD® turtle graphics in [1] was introductory in nature and limited to only three examples of Chinese lattice designs.

In this paper, we looked at more Chinese lattice designs from the different group classifications of Dye (12 groups were selected with a total of 25 Chinese lattice designs). The Chinese lattice designs were then algorithmically generated using MuPAD® turtle graphics. We added a further element by coloring the Chinese lattices.

II. MuPAD® TURTLE GRAPHICS

A. Turtle Graphics

Turtle graphics can be simply described as the mimicking of finite straight line movements by an abstract pen through a sequence of commands.

B. MuPAD® Commands

The following table from [10] shows the turtle graphics command of MuPAD®.

<table>
<thead>
<tr>
<th>Command</th>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left(angle)</td>
<td>left(angle)</td>
<td>Turn left given angle (in</td>
</tr>
<tr>
<td>Right(angle)</td>
<td>right(angle)</td>
<td>Turn right given angle (in radians)</td>
</tr>
<tr>
<td>Forward(angle)</td>
<td>forward(length)</td>
<td>Draw forward a line of given length</td>
</tr>
<tr>
<td>Up</td>
<td>penUp()</td>
<td>Take the pen up</td>
</tr>
<tr>
<td>Down</td>
<td>penDown()</td>
<td>Take the pen down</td>
</tr>
<tr>
<td>Push</td>
<td>push()</td>
<td>Save the current stage of the turtle</td>
</tr>
<tr>
<td>Pop</td>
<td>pop()</td>
<td>Move the turtle to the last stage</td>
</tr>
<tr>
<td>LineColor(color)</td>
<td>setLineColor(color)</td>
<td>Change the current path color</td>
</tr>
</tbody>
</table>

To familiarize with MuPAD®’s turtle graphics, simple examples of implementation can be examined in [11].

III. ALGORITHMIC PROCESS OF CHINESE LATTICE DESIGNS

When representing the Chinese lattice designs algorithmically, the first step is to determine the design’s basic building block. The basic building block may consist of more than one simpler building block. Each simpler building block will be drawn using a procedure. The procedures for the simpler building blocks will be combined to form the basic building block. This combined procedure will then be repeated with for-loop to build the whole Chinese lattice design structure.

We describe the algorithmic process through the Out-Lock example below.

The basic building block of the Out-Lock pattern in Fig. 1 is shown in Fig. 2. It can be observed that it is made up of many rectangular shapes arranged in a certain order. Table II...
shows the MuPAD® codes for drawing the basic building block for the Out-Lock pattern.

<table>
<thead>
<tr>
<th>MuPAD® code</th>
<th>Description</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>x1:=proc(U)</td>
<td>begin</td>
<td>Produces a rectangle with sides of two and one units</td>
</tr>
<tr>
<td>forward(U,2);</td>
<td>U::right(PI/2);</td>
<td></td>
</tr>
<tr>
<td>forward(U,1);</td>
<td>U::right(PI/2);</td>
<td></td>
</tr>
<tr>
<td>forward(U,2);</td>
<td>U::right(PI/2);</td>
<td></td>
</tr>
<tr>
<td>forward(U,1);</td>
<td>U::right(PI/2);</td>
<td></td>
</tr>
<tr>
<td>return(U)</td>
<td>end:</td>
<td>2</td>
</tr>
</tbody>
</table>

x2:=proc(U) begin U::push(); forward(U,4); U::right(PI/2); forward(U,1); U::right(PI/2); forward(U,4); U::right(PI/2); forward(U,1); U::pop(); return(U) end:

Produces a rectangle with sides of two and one units

y1:=proc(U) local i,j; begin U::push(); for i from 1 to 4 do x2(U); Upforward(U,5); U::right(PI/2); end_for; Upforward(U,1); U::right(PI/2); Upforward(U,1); U::left(PI/2); for j from 1 to 4 do x1(U); Upforward(U,3); U::right(PI/2); end_for; U::pop(); return(U) end:

In this procedure, two for-loop are used. The first for-loop produces the pattern on the right (above). The second for-loop produces the pattern on the right (below). The whole code will then be combined to produce the pattern in Fig. 2.

In the code above (Table III), we defined a new command “Upforward” where the abstract pen moves a certain finite length without drawing the line. This is to simplify the pen up – move – pen down movements which are repeatedly used. The codes are shown in Table IV. To produce a multi-colored effect for the Chinese lattice, we randomized the line color with codes described in Table V.

The complete program codes for this Out-Lock pattern is given in the Appendix. The program generates a multi-colored Chinese lattice as in Fig. 3.

**IV. CHINESE LATTICES DESIGNS GENERATED**

Following the principles and steps in Section III, we
generated the following Chinese lattice designs.

Fig. 4. Parallelogram.

Fig. 5. Octagon or octagon square.

Fig. 6. Single focus frames.

Fig. 7. Wedge-lock.

Fig. 8. Presentation.

Fig. 9. Out-lock.

Fig. 10. Parallel waves.

Fig. 11. Opposed waves.

Fig. 12. Recurving Wave.
V. CONCLUSION

Many Chinese lattice designs can be successfully generated through an appropriate algorithmic process. For this type of Chinese lattices, the lattice must be broken into smaller basic building blocks. This requires a sharp observation of the Chinese lattice design’s underlying structure. It should be noted however that not all Chinese lattices can be generated using this algorithmic process. Rustic ice-ray and symmetrical ice-ray lattices are two examples where MuPAD®’s turtle graphics may not be able to construct successfully. A discussion of these is beyond the scope of this paper.

A randomized coloring of the lines in a Chinese lattice’s design produces appealing multi-colored Chinese lattices.

APPENDIX

Complete MuPAD® codes to generate the Out-Lock pattern in Fig. 3

\[ \text{forward} := \text{proc}(V, \text{dist}) \]
\[ \text{local} \ a, b, c; \]
\[ \text{begin} \]
\[ a := \text{frandom}(); \]
\[ b := \text{frandom}(); \]
\[ c := \text{frandom}(); \]
\[ V::\text{setLineColor}([0,0,0]); \]
\[ V::\text{forward}(\text{dist}); \]
\[ \text{return}(V) \]
\[ \end: \]

\[ \text{Upforward} := \text{proc}(V, \text{dist}) \]
\[ \text{begin} \]
\[ V::\text{penUp}(); \]
\[ V::\text{forward}(\text{dist}); \]
\[ V::\text{penDown}(); \]
\[ \text{return}(V) \]
\[ \end: \]

\[ x1 := \text{proc}(U) \]
\[ \text{begin} \]
\[ \text{forward}(U,2); \]
\[ U::\text{right}(\pi/2); \]
\[ \text{forward}(U,1); \]
\[ U::\text{right}(\pi/2); \]
\[ \text{forward}(U,2); \]
\[ U::\text{right}(\pi/2); \]
\[ \text{forward}(U,1); \]
\[ U::\text{right}(\pi/2); \]
\[ \text{return}(U) \]
\[ \end: \]

\[ x2 := \text{proc}(U) \]
\[ \text{begin} \]
\[ U::\text{push}(); \]
\[ \text{forward}(U,4); \]
\[ U::\text{right}(\pi/2); \]
\[ \text{forward}(U,1); \]
\[ U::\text{right}(\pi/2); \]
\[ \text{forward}(U,4); \]
\[ U::\text{right}(\pi/2); \]
\[ \text{forward}(U,1); \]
\[ U::\text{right}(\pi/2); \]
The program codes for the other Chinese lattice designs produced in this paper can be furnished upon request from the authors.

REFERENCES


S. Y. Lee was formerly an undergraduate student at the School of science and technology, Universiti Malaysia Sabah. His undergraduate dissertation in 2008 was an investigation on Chinese lattice designs using MuPAD® and he graduated with a B.Sc. in Mathematics with Computer Graphics under the supervision of the second author. He now works as a programmer.

K. M. Tiong is currently an assistant professor at the faculty of arts and social sciences, The University of Nottingham Malaysia Campus at Semenyih, Selangor, Malaysia. He holds a B.Sc.Ed. (Hons) (Mathematics and Physics) from University of Malaya and M.Sc. (Mathematics) from Universiti Teknologi Malaysia. His current research interests involve mathematical patterns, mathematical games and statistics education.