SOME NEW MAP DATA ENCODING SCHEMES. (U)
1978 H FREEMAN AFOSR-76-2937
UNCLASSIFIED AFOSR-TR-78-0946 NL
Some new schemes for encoding map data are introduced. The schemes can be regarded as generalizations of the well known 8-direction chain coding scheme. Instead of being limited to 8 types of links for approximating a curve, the new schemes possess 16, 24, 32, 48, or even more link types. The new schemes permit increased smoothness of representation, exhibit greater precision, and require less processing time for comparable resolution than present methods.

1. INTRODUCTION

For more than a decade now there has been a strong trend to digitize geographic map data and enter it into a computer data base. The main advantages of having geographic data in computer form are the ease of updating, the possibility of automated information retrieval, and the convenience of rapid computer processing for a variety of purposes. Geographic map data is inherently spatial in nature and consists of regions, lines, and points, together with their associated names and attributes. Although it is not necessary that the computer data structure for a geographic data base emulate the annotated line drawings of a map, for many purposes it has in fact been found advantageous to do so. As a result even when stored in a computer data base, geographic data is often represented in the form of large line drawings that bear a one-to-one correspondence to the familiar map representation. Maps contain a large amount of data, and it is important to pay attention to the efficient representation of such data if the computer storage and processing is to be economical. We shall describe here some new schemes for encoding the large and complex line structures that characterize map data.

A well known scheme for the computer representation of line-drawing data is the so-called chain coding scheme [1,2]. In this scheme an overlaid square lattice is assumed and the lines of the drawing are represented by sequences of straight-line segments connecting nodes of the lattice lying closest to the lines. In passing from one node to the next, there are 8 allowed directions, and the concatenated line segments are all of length 1 or √2 (times the lattice spacing). The scheme has found wide acceptance for map data encoding, mainly because of its inherent simplicity and the ease with which efficient processing algorithms can be developed for it. We shall here show that the basic (i.e., 8-direction) chain code can be generalized to codes having a much larger number of allowed directions and that such codes, in spite of their increased complexity, may have definite advantages for certain applications [3].

In selecting a scheme for encoding map data, one must evaluate it with respect to the following five criteria: (1) compactness, (2) precision, (3) smoothness, (4) ease of encoding and decoding, and (5) facility for processing. The relative weights to be assigned to each of these criteria vary somewhat with the intended application. If the purpose of the map encoding is primarily storage or transmission, compactness is likely to be of paramount importance as it directly determines the required amount of computer memory or channel bandwidth (or transmission time). Precision is important if quantitative aspects of the encoded data are of particular interest. Smoothness will be of significance if the encoded data is ever to be displayed directly, that is, without extensive output interpolation. For applications involving extensive processing, a coding scheme that facilitates the processing task (i.e., requires less computer time) will, of course, be particularly attractive.

2. GENERALIZED CHAIN CODES

In the basic (8-point) chain code, the next node (i, j) in sequence for a given present node (r, s) must be one of the 8 nodes that are 1 or √2-distant, i.e., such that max. |r-i|, |s-j| ≤ 1. Thus in Fig. 1, for a given node A, the permissible next nodes in the basic chain code are the nodes numbered 0 through 7. All of these nodes lie on the boundary of a square of side 2 and centered at A. We shall refer to this square boundary as "ring 1." Let us now consider a coding scheme in which the "next" node may be any node in ring 1 or in ring 2. (Ring 2 consists of nodes 8 through 23 in Fig. 1). These are the nodes for which max. |r-i|, |s-j| = 1 or 2. A chain based on such a 24-point scheme may contain links of length 1, √2, 2, 2√2, and 2√3. Also there will be a total of 16 allowed directions (determined by the nodes of ring 2). A curve encoded with this scheme will exhibit finer angular quantization and contain fewer segments than one encoded in the 8-point scheme. Finer angular quantization will yield improved smoothness. Fig. 2 shows a curve encoded in both the 3-point scheme (a) and the 24-point scheme (b).

*The term "ring" is used here in the same sense as in "boxing ring".

*The work reported on here was supported by the Air Force Office of Scientific Research, Directorate of Mathematical and Information Sciences, under Grant AFOSR 76-2937.
AIR FORCE OFFICE OF SCIENTIFIC RESEARCH (AFOSR)
NOTICE OF TRANSMITTAL TO DDC
This technical report has been reviewed and is approved for public release IAW AFR 190-12 (7b). Distribution is unlimited.
A. D. BLOOE
Technical Information Officer
DISCLAIMER NOTICE

THIS DOCUMENT IS BEST QUALITY PRACTICABLE. THE COPY FURNISHED TO DDC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.
In forming a chain coding scheme, we may use any number of rings, in any combination. Thus we may form a chain code based solely on ring 2. It will have 16 permissible directions and its links will be of length 2, i.e., \( \sqrt{2} \) and \( \sqrt{8} \). Its angular quantization will be either 18.4° or 26.5°. It differs from the 24-point code in that steps of length 1 or \( \sqrt{2} \) are not allowed. As a result there may be difficulty in obtaining a closed chain to correspond to a closed curve; that is, the end points of a 16-point encoded chain may be 1 or \( \sqrt{2} \) units apart without the availability of links of such lengths for closing this gap. For example, if one draws in Fig. 1 a line segment from node A to node 23 and from node 23 to node 1 (both permissible 16-point line segments), the end points, nodes A and 1, will be a distance \( \sqrt{2} \) apart. Although this lack of completeness may be objectionable to the purist, in practice it is of minor consequence since a chain can always be closed by some sacrifice in precision. Thus for the previous 2-link chain, drawing the second link from node 23 to node 9 instead of to node 1 will permit closing the chain with a link from 9 to A. The 16-link scheme has been previously proposed for use with digital plotters [6].

In returning to the 24-point code we note that (since the 8-point code is subsumed in it) it has all the features of the 8-point code of being able to follow fine detail (small radii of curvature) with short segments of length 1 and \( \sqrt{2} \) but in addition has longer segments of length 2, \( \sqrt{8} \), and \( 2\sqrt{2} \) for “taking bigger steps” when the curvature is more gentle. These larger steps can be taken with an angular quantization roughly twice as fine as that of the 8-point scheme. Presumably, with the foregoing in mind, a 48-link scheme utilizing rings 1, 2 and 4 should be even better.

The coding matrices corresponding to 4-, 8-, 16-, 24-, 32- and 48-point codes are shown in Fig. 3. Note that the coding matrix for the 32-point code consists of rings 1 and 3. This code thus has the ability to take relatively long, fine-angle steps but, because of ring 1, can also follow small detail. The 48-point code of Fig. 3 (g) consists of the complete rings 1 and 2, and the partial ring 4. In ring 4, those nodes for which one coordinate has value 3 have been omitted. If ring 2 were also eliminated, a 32-point code would result (consisting of ring 1 and the partial ring 4) that would have an

![Fig. 1. The different node rings surrounding the given Node A: 0 - 7 (ring 1), 8 - 23 (ring 2), 24 - 47 (ring 3), etc. Ring 1 is shown bold.](image1)

![Fig. 2. Two different chain encodings of the same curve: (a) 8-point code, (b) 24-point code.](image2)
excellent long-distance capability and yet retain
the ability to follow fine detail. The rules
governing the node relations for the codes in
Fig. 3 are shown in Fig. 4.

Fig. 3. Coding matrices for the 4-, 8-, 16-, 24-, 32-, and 48- point chain codes. (Two different versions of the 48-point code are shown).

3. QUANTIZATION

One of the appealing features of the 8-point
code has been its simplicity - for quantization,
for encoding, and for processing. As we go to
higher-order codes, the complexity of these tasks
increases. Let us examine first the quantiza-
tion problem. In Fig. 5(a), the so-called grid
intersection method for the 8-point code is
illustrated. One traces along the curve, and at
each intersection between curve and superimposed
grid, the node closest to the intersection is
selected as next node. The method assures that

4-Point Chain: \(|r - i| + |s - j| = 1\)
3-Point Chain: \(\max |r - i|, |s - j| = 1\)
16-Point Chain: \(\max |r - i|, |s - j| = 2\)
24-Point Chain: \(\max |r - i|, |s - j| = 1 \text{ or } 2\)
32-Point Chain: \(\max |r - i|, |s - j| = 1 \text{ or } 2\)
48-Point Chain: \(\max |r - i|, |s - j| = 1 \text{ or } 2 \text{ or } 3\)
48-Point Chain: \(\max |r - i|, |s - j| = 1 \text{ or } 2 \text{ or } 3\)

Fig. 4. Adjacent-node relationships for the 4-, 8-, 16-, 24-, 32-, and 48-point chain codes.

on average approximately 41 per cent of the links
in a chain will be of length \(\sqrt{2}\) \[7\]. An alter-
nate quantization scheme is the so-called square-
box scheme shown in Fig. 5(b), where the next
node is selected on the basis of a square box
"capture area" surrounding each node. The latter
scheme, however, yields then only 4-point coded
chains \[7\].

In Fig. 5(c) we show how the grid-intersec-
tion scheme has been extended to the 24-point
code. In determining the next node, one first
looks for the intersection between the curve and
ring 2. The closest ring-2 node is identified;
however, before it can be taken as the next node,
it is necessary to determine whether the curve
intersects ring 1 within limits set by the grid
midpoints to either side of the identified ring-
2 node. In Fig. 5(c), for curve A the ring-2
node is 17. Its limits in ring 1 are located at
the 1/4 and 3/4 points between nodes 1 and 2
(note the dashed lines). If the curve intersects
ring 1 within these limits, the ring 2 node is
the valid next node. Thus in Fig. 5(c), node
17 is a valid next node for curve A, but node 9
is not a valid next node for curve B. For curve
B, the next node must be taken from ring 1 (it
will be node 1).
both of the latter codes, addition of 2 to each code value will cause a 90-degree counter-clockwise rotation (subject to appropriate limit checks to assure remaining in the same ring). Some different coding conventions are illustrated in Fig. 7 (a) and (b), which have some advantages over those of Fig. 6. Proposed coding assignments for the 32-point and the two 48-point codes of Fig. 3 are shown in Figs. 8 and 9.

The number of bits required to represent a curve in the different chain-code systems varies considerably. In Fig. 10 we show a closed contour quantized into the 4-point, 8-point, 16-point, 24-point, and 32-point systems. If we use a distinct code word for each link, we shall require 2, 3, 4, 5, and 5 bits per link, respectively, for these codes. The results are shown in Table I.

The coding assignments in Fig's. 6 through 8 all associate a unique number with each allowed line segment. A curve quantized into a chain of line segments can thus be uniquely described by a string of numbers, and such a representation implies a fixed orientation (but not position) on the coding lattice. For a curve to be "well-quantized", the change in orientation from link to link should normally be

**Fig. 5.** Quantization schemes for the 4-, 8-, 24-, and 32-point chain codes. (a) 8-point grid-intersect quantization, (b) 4-point square-box quantization, (c) 24-point grid intersect quantization, and (d) 32-point grid intersect quantization.

The quantization scheme for the 32-point code (based on rings 1 and 3) is shown in Fig. 5 (d). Appropriate limits must be satisfied for rings 3, 2, 1 (in that order). In the figure, curve A satisfies all limits associated with node 9 and node 9 thus becomes the next node. However, node 16 cannot be selected for curve B because the associated ring-2 and ring-1 limits are not satisfied. One should note that, although the 32-point code utilizes only rings 1 and 3, for the purpose of quantization, all rings of lower order must be considered. The quantization procedure for higher-order codes is similar.

4. ENCODING

For the 8-point chain code, the coding convention is well known and is shown in Fig. 6(b). In Fig. 6 (a) we show the corresponding convention for the 4-point chain code. Possible conventions for the 16- and 24-point codes are shown in Fig. 6 (c) and (d), respectively. For
Table 2. Chain Difference Coding Scheme for 8-Point Code.

<table>
<thead>
<tr>
<th>Link Diff.</th>
<th>Code Word</th>
<th>No. of Bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>+1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>-1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>+2</td>
<td>31</td>
<td>4</td>
</tr>
<tr>
<td>-2</td>
<td>32</td>
<td>4</td>
</tr>
<tr>
<td>+3</td>
<td>331</td>
<td>6</td>
</tr>
<tr>
<td>-3</td>
<td>332</td>
<td>6</td>
</tr>
<tr>
<td>INIT</td>
<td>3330</td>
<td>7</td>
</tr>
<tr>
<td>CTRL</td>
<td>3331</td>
<td>7</td>
</tr>
</tbody>
</table>

In the table, the link difference "0" means continuation in the same direction, "1" means that the succeeding link is the next one in a counterclockwise sense, etc. INIT is used, with 3 additional bits, to set the initial link direction. CTRL is a control flag, analogous to the combination 04 in the conventional chain code.

Table 1. Bit Requirements for the Different Chains of Fig. 10.

<table>
<thead>
<tr>
<th>System of Links</th>
<th>Bits per Link</th>
<th>Total No. of Bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>124</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>87</td>
<td>3</td>
</tr>
<tr>
<td>16</td>
<td>44</td>
<td>4</td>
</tr>
<tr>
<td>24</td>
<td>48</td>
<td>5</td>
</tr>
<tr>
<td>32</td>
<td>38</td>
<td>5</td>
</tr>
</tbody>
</table>

In small. (A succession of many large changes in link orientation would imply that the curve was not quantized finely enough to preserve the detail of interest [9].) This suggests the presence of strong link-to-link coherence in any well-quantized chain, and this coherence can be utilized for compressing the corresponding number string. The most obvious and simple scheme is that of using first differences. For the case of the conventional 8-point chain, this leads to the so-called chain difference code [7] shown in Table 2.

Fig. 7. Alternate code assignments for the 16- and 24-point codes.

Fig. 8. Code assignments for the 32- and 48-point chain codes.
(See Fig. 7(a) for interpretation of the code words.) If written in the chain difference code shown in Table 3, the chain will require 21 3-bit words, 15 6-bit words, 1 9-bit word, and one 12-bit initial-direction word, for a total of 174 bits. In comparison, the full chain code requires 38(5) = 190 bits.

Fig. 9. 48-point chain code constituted from rings 1, 2, and a partial ring 4. and is also used (with additional digits) to indicate a valid link change of 24.

A chain difference code for the 87-link chain

which is illustrated in Fig. 10(c), is given by

The difference-code chain contains 81 2-bit code words (0, +1, or -1). 5 4-bit code words (=2 or -2) and, of course, an initial-direction code (INIT 3) of 10 bits to set the initial direction equal to link direction 3. The total number of bits thus is 2(81) + 5(4) + 10 = 192. In comparison, the 8-point chain requires 3(87) = 261 bits.

Let us now examine what will happen if we apply the chain-difference concept to one of the higher-order chains. say, to the 32-point chain. The 38-link, 32-point chain of Fig. 10(f) is given by the sequence

(19, 1, 10, 27, 11, 3, 2, 9, 25, 2, 11, 4, 3, 3, 10, 25, 2, 25, 9, 25, 2, 30, 20, 30, 22, 30, 14, 14, 6, 21, 4, 5, 4)

Table 3. Chain Difference Coding Scheme for 32-Point Code

<table>
<thead>
<tr>
<th>Link Change Word</th>
<th>Link Change Word</th>
<th>No. of Bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>L=0</td>
<td>70</td>
<td>S=0</td>
</tr>
<tr>
<td>LS/SL</td>
<td>71</td>
<td>CTRL</td>
</tr>
<tr>
<td>S=1</td>
<td>0</td>
<td>S=1</td>
</tr>
<tr>
<td>L=1</td>
<td>2</td>
<td>L=1</td>
</tr>
<tr>
<td>L=2</td>
<td>4</td>
<td>L=2</td>
</tr>
<tr>
<td>L=3</td>
<td>72</td>
<td>L=3</td>
</tr>
<tr>
<td>L=4</td>
<td>74</td>
<td>L=4</td>
</tr>
<tr>
<td>L=5</td>
<td>62</td>
<td>L=5</td>
</tr>
<tr>
<td>L=6</td>
<td>64</td>
<td>L=6</td>
</tr>
<tr>
<td>S=2</td>
<td>66</td>
<td>S=2</td>
</tr>
<tr>
<td>L=7</td>
<td>600</td>
<td>L=7</td>
</tr>
<tr>
<td>L=8</td>
<td>602</td>
<td>L=8</td>
</tr>
<tr>
<td>L=9</td>
<td>604</td>
<td>L=9</td>
</tr>
<tr>
<td>L=10</td>
<td>606</td>
<td>L=10</td>
</tr>
<tr>
<td>L=11</td>
<td>770</td>
<td>L=11</td>
</tr>
<tr>
<td>S=3</td>
<td>772</td>
<td>S=3</td>
</tr>
<tr>
<td>L=12</td>
<td>774</td>
<td>S=4</td>
</tr>
</tbody>
</table>

Legend:
L=1 - next link is 1st long (>2) link in positive (CCW) sense
S=1 - next link is 1 short (<2) link in positive (CCW) sense
LS/SL - next link is short/long but in same direction as previous long/short link (i.e., change in link length only)
Negative sign indicates negative (CW) sense.

Unlike the chain difference code for the 9-point code, the chain difference code for the 32-point code can be structured in a number of different ways. The code shown in Table 3 was selected after careful study of the probabilities of occurrence of the different link-to-link transitions. The objective is, of course, to maximize first the occurrence of code words of length 3, and next those of length 6. Unless the mean code word length for an encoded curve lies appreciably below 5 there is no advantage over the regular 32-point code.

5. COMPARATIVE CODE CHARACTERISTICS

We observe that for the contour of Fig. 10a, we obtained a 26.4 per cent saving (261 to 192) when converting from the 8-point chain code to the corresponding chain difference code. For the same contour, when converting from the 32-point chain representation to the corresponding chain-difference code, the saving is only 8.4 per cent (190 to 174). The explanation, of course, is that the 32-point code is in itself much more
efficient than the 8-point code and hence there
is less left to be gained by going to the dif-
ference coding scheme. Essentially similar
results are found for the 16-, 24-, and 48-point
codes. We make the interesting observation that
the number of bits required to represent a curve
does not vary materially with the coding scheme
used if we are prepared to use difference schemes
to compress the codes. Apparently, if we are
looking for advantages that will establish one
code as being superior to another, we must look
elsewhere than at compactness of representation.

Inspection of Fig. 10 shows that the smoothness
of the representation tends to increase with in-
crease in the order of the chain code. This is
due to the increased angular resolution of the
higher-order codes. When the different chains
of Fig. 10 were shown to a number of observers,
the majority selected the 32-point representation,
(f), as the "smoothest" and most faithful rendi-
tion of the original curve given in (a). Smooth-
ness is a largely subjective concept, difficult
to quantify. The superiority of the codes of
order 16 or greater over the 4- and 8-point
codes is, however, quite apparent.

To obtain some measure of the relative preci-
sion for the different codes, the perimeter, en-
closed area, and area error were determined for the contour and chains of Fig. 10.
Since the curve is free-form, the area had to be
computed by using a secondary lattice, one fifth
the size of the lattice used for the chains.
The area error is obtained by counting both interior and exter-
ior error as positive.

With respect to the complexity of any process-
ing algorithms, there is very little difference
among the different chain codes. The higher-
order codes merely require larger lookup tables
for the algorithms; there is virtually no dif-
ference in computation time per link. However,
since processing time is proportional to the
number of links in a chain, the higher-order
chains will, in fact, be processed much faster.
Thus, for example, since the 32-point chain of
Fig. 10 (f) has only 38 links whereas the cor-
responding 8-point chain, (c), has 87 links, the
former will be processed in less than half the
time required for the latter.

As described in Section 3, the quantization
procedures become progressively more involved as
we go to the higher-order chain codes. To a
somewhat lesser extent this is also true for the
"de-quantization" (i.e., display) procedures.
Specialized hardware can be employed for quanti-
ization to alleviate this problem. In any case,
however, the increased effort here is not likely
to outweigh the significant advantages gained in
smoothness, precision, and overall computation
time.

6. CONCLUSION

A set of higher-order chain codes has been
described which permits the use of 16, 24, 32 or
even more link types for representing map con-
tours. The higher-order codes offer the possi-
bility of increased precision, greater smooth-
ness, and reduced computation time for pro-
cessing map data. The advantages over the well
known 8-point chain code are appreciable. There
is also some gain in compactness of representa-
tion; however, this is of less significance
since similar gains can be obtained in other ways.

REFERENCES

1. H. Freeman, "On the Encoding of Arbitrary
Geometric Configurations", IRE Trans., GC-10,

2. "Computer Processing of Line-
Drawing Images", Computing Surveys, 6 (1),
March 1974, 57-97.

3. "Comparative Analysis of Line-
Pattern Coding Schemes", Conf. on Formal
Psychophysical Approaches to Visual Percep-

tons Using a Quasi-Euclidean Distance",

5. H. Rademacher, Lectures on Elementary Number

6. V. V. Athani, "The 16-Vector Algorithm for
Computer Controlled Digital x-y Plotter",
IEEE Trans. Computers, C-24, (3), August
1975, 831-839.

7. H. Freeman, "A Technique for the Classifica-
tion and Recognition of Geometric Patterns",
Proc. 3rd Int'l. Congress on Cybernetics,
Namur, Belgium, 1961, 348-369.

8. H. Freeman and J. Glass, "On the Quantization
of Line Drawing Data", IEEE Trans. Systems
1969, 70-79.

<table>
<thead>
<tr>
<th>Code</th>
<th>Perimeter</th>
<th>Area</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>172.8</td>
<td>372.0</td>
<td>16.6</td>
</tr>
<tr>
<td>8</td>
<td>103.2</td>
<td>310.5</td>
<td>11.5</td>
</tr>
<tr>
<td>16</td>
<td>97.4</td>
<td>306.5</td>
<td>13.1</td>
</tr>
<tr>
<td>24</td>
<td>98.1</td>
<td>301.8</td>
<td>10.2</td>
</tr>
<tr>
<td>32</td>
<td>97.4</td>
<td>307.0</td>
<td>9.1</td>
</tr>
<tr>
<td>Curve</td>
<td>96.6</td>
<td>310.3</td>
<td></td>
</tr>
</tbody>
</table>

Table 4. Perimeter and Area Data Computed for the Chains of Fig. 10.
Fig. 10. Comparison of original contour (a) with different chain types: (b) 4-point, (c) 8-point, (d) 16-point, (e) 24-point, and (f) 32-point chain.
Some new schemes for encoding map data are introduced. The schemes can be regarded as generalizations of the well known 8-direction chain coding scheme. Instead of being limited to 8 types of links for approximating a curve, the new schemes possess 16, 24, 32, 48, or even more link types. The new schemes permit increased smoothness of representation, exhibit greater precision, and require less processing time for comparable resolution than present methods.