Journal of Classification 2:239-254 (1985)

Journal of
Classification
©1985 Springer-Verlag New York Inc.

Investigation of Proportional Link Linkage Clustering Methods

William H. E. Day
Memorial University of Newfoundland

Herbert Edelsbrunner
Technische Universität Graz

Abstract: Proportional link linkage (PLL) clustering methods are a parametric family of monotone invariant agglomerative hierarchical clustering methods. This family includes the single, minimedian, and complete linkage clustering methods as special cases; its members are used in psychological and ecological applications. Since the literature on clustering space distortion is oriented to quantitative input data, we adapt its basic concepts to input data with only ordinal significance and analyze the space distortion properties of PLL methods. To enable PLL methods to be used when the number n of objects being clustered is large, we describe an efficient PLL algorithm that operates in $O(n^2 \log n)$ time and $O(n^2)$ space.

Keywords: Algorithm complexity; Algorithm design; Integer link linkage clustering method; Monotone invariance; SAHN clustering method; Space distortion.

1. Introduction

In the twenty years since Sokal and Sneath's classic treatise (1963) on numerical taxonomy appeared, tremendous advances have occurred in developing and analyzing methods for the objective construction of hierarchical classifications of objects. Typically the input to such clustering methods is a symmetric, nonnegative, real-valued matrix $D = [d_{xy}]$, where $d_{xy} = 0$ if and only if $x = y$; d_{xy} is interpreted as a quantitative measure of dissimilarity between objects x and y. As output these clustering methods usually generate a dendrogram or what Johnson (1967) calls a *hierarchical clustering*

This work was partially supported by the Natural Sciences and Engineering Research Council of Canada and by the Austrian Fonds zur Förderung der wissenschaftlichen Forschung.

Authors' addresses: William H. E. Day, Department of Computer Science, Memorial University of Newfoundland, St. John's, Newfoundland, Canada A1C 5S7. Herbert Edelsbrunner, Institute für Informationsverarbeitung, Technische Universität Graz, Schiesstattgasse 4a, A-8010 Graz, Austria.

scheme (HCS). An HCS is a sequence $P_0, \ldots, P_w$ of partitions of the objects being studied in which P_0 is the disjoint partition, P_w is the conjoint partition, and P_i is a refinement (in the usual sense) of P_j for all $0 \leqslant i < j \leqslant w$. Associated with each P_i is a nonnegative real number α_i, called its *level*, where $\alpha_0 = 0$ and $\alpha_i \leqslant \alpha_j$ for all $0 \leqslant i < j \leqslant w$. The HCS is *nonoverlapping* because the blocks, or clusters, of each partition are mutually exclusive; it is *hierarchical* (in the sense of Johnson, 1967) because two clusters in different partitions either are mutually exclusive or can be ordered by set containment. Within this context, a hierarchical clustering method is simply a function that maps each dissimilarity matrix into a corresponding HCS.

Many well-known hierarchical clustering methods share characteristics deriving from a paradigmatic algorithm that can be used to construct the HCS. They are *agglomerative* methods: starting with the disjoint partition P_0, they group objects into successively fewer and fewer clusters, arriving eventually at the conjoint partition P_w. They are *sequential* methods in which the same algorithm is used iteratively to generate P_{i+1} from P_i for all $0 \leqslant i < w$. They are *pair-group* methods: at each iteration exactly two clusters are agglomerated into a single cluster. Sneath and Sokal (1973) use the acronym SAHN to designate clustering methods that are sequential, agglomerative, hierarchical, and nonoverlapping. Readers wishing more information on SAHN clustering methods can consult Anderberg (1973), Everitt (1980), Legendre and Legendre (1983a), Sneath and Sokal (1973), or any standard numerical taxonomy text.

SAHN clustering methods can often be defined algorithmically simply by providing a precise specification of step 3 in the paradigmatic algorithm in Table 1. Since step 2 agglomerates clusters i and j to form the new cluster (ij), step 3 must calculate for every other cluster k the revised dissimilarity $\delta_{(ij)k}$ between (ij) and k. Thus $\delta_{(ij)k} = min\{\delta_{ik}, \delta_{jk}\}$ for the *single linkage* (or *minimum* or *nearest neighbor*) clustering method (Florek et al. 1951; Lukaszewicz 1951; McQuitty 1957; Sneath 1957); $\delta_{(ij)k} = \max\{\delta_{ik}, \delta_{jk}\}$ for the *complete linkage* (or *maximum* or *furthest neighbor*) clustering method (Sørensen 1948); and $\delta_{(ij)k} = (\sum_{x \in (ij)} \sum_{y \in k} d_{xy}) \ / \ (|(ij)| \cdot |k|)$, where $|z|$ denotes the cardinality of a set z, for the *group average* (or *UPGMA* or *unweighted arithmetic average*) clustering method (Rohlf 1963). This algorithmic approach to specifying families of SAHN methods is developed in early contributions by Ward (1963), Johnson (1967), Wishart (1969), and Anderberg (1973, chapter 6).

Clustering methods must be selected carefully since they presume dissimilarity structure that may be absent in any given application: "usually more is significant than just the ordering, but it is rash to employ without careful justification methods ... which assume this" (Jardine and Sibson 1971, p. 92). When data comprising a dissimilarity matrix have only ordinal significance (situation considered by Bromley (1966) and Johnson (1967)),

TABLE 1

Paradigmatic SAHN Clustering Algorithm

```
begin
    Initialize $\alpha_o \leftarrow 0$; $P_o \leftarrow \{\{s\}:s \in S\}$; $\Delta \leftarrow D$
    for m = 0 to n-2 do
        begin
1.          Search $\Delta$ for a closest pair {i,j} of distinct clusters in $P_m$
2.          Agglomerate clusters i and j into (ij) at level $\alpha_{m+1} = \delta_{ij}$ to obtain $P_{m+1}$ from $P_m$.
3.          Update $\Delta$ to reflect agglomeration of i and j into (ij).
        end
end
```

Note: S is the set of n objects to be clustered. $D = [d_{ij}]$ is the original matrix of interobject dissimilarities. $\Delta = [\delta_{ij}]$ is the matrix of intercluster dissimilarities.

the clustering method used should be invariant to monotone increasing transformations of the dissimilarity matrix. Johnson (1967) uses this argument in preferring single linkage and complete linkage to clustering methods such as group average. Monotone invariance plays an important role in the work of Jardine and Sibson (1968, 1971) and Sibson (1970, 1972). A plethora of monotone invariant clustering methods have since been proposed. Hubert (1972) develops a method that is invariant to hypermonotone increasing transformations of the dissimilarity matrix. Graph-theoretic methods advocated by Ling (1972), Hubert (1973, 1974, 1976, 1977), and Matula (1977) are based on the rank ordering of data in the dissimilarity matrix. Hubert (1973) and D'Andrade (1978) describe methods based on monotone invariant goodness-of-fit statistics. Janowitz (1978a, 1978b, 1981) develops an elegant order-theoretic model of clustering methods and specializes it to the consideration of monotone invariant methods (1979a, 1979b).

Although single and complete linkage are monotone invariant clustering methods, single linkage tends to agglomerate objects into relatively few large straggly clusters, while complete linkage tends to agglomerate objects into relatively many compact spherical clusters (Hubert and Schultz 1975). Sneath (1966) provides alternatives to these extremes by introducing the *proportional link linkage* (U-PLL) family of monotone invariant SAHN methods, where the parameter U is any real number such that $0 < U \leqslant 1$.

These methods can be defined by specifying step 3 in the paradigmatic SAHN algorithm. If

$$J = \lceil\, U \cdot |(ij)| \cdot |k| \,\rceil \quad , \tag{1}$$

where $\lceil x \rceil$ denotes the least integer not less than x, then the dissimilarity $\delta_{(ij)k}$ between the agglomerated cluster (ij) and any other cluster k is the *J-th* smallest of the set $\{d_{xy} \,:\, x \in (ij), y \in k\}$ of interobject dissimilarities. 1-PLL corresponds to complete linkage. There is no fixed value of U for which U-PLL is single linkage; but if one fixes instead the number n of objects being clustered, U-PLL is single linkage for any $U \leqslant 4 \,/\, n^2$. Sneath (1966) speculates that .5-PLL may be the "median link" method Kendrick and Proctor (1964) use in their analysis of the Fungi Imperfecti, and it may also be J. D. Carroll's "average method based on medians" (Johnson 1967) and D'Andrade's "median method" (1978). Furnas (1980) defines two median methods for his investigation of the structural representation of two-class data: both methods use the (unique) middle ranked interobject dissimilarity if their number is odd, but when their number is even, the *minimedian* (respectively, *maximedian*) clustering method selects the lesser (respectively, greater) of the two middle-ranked dissimilarities. The minimedian method corresponds to .5-PLL, but the maximedian method has no correspondent in the PLL family. Legendre and Legendre (1983a) encourage the use of U-PLL in ecological applications, and they use .75-PLL in constructing a benthonic classification (1983b). P. Legendre and his colleagues also modify the U-PLL algorithm to impose on the resulting HCS constraints of time contiguity (Legendre, Baleux, and Troussellier 1984; Legendre, Dallot, and Legendre 1985) or space contiguity (Legendre and Legendre 1984).

Sneath (1966) provides a second alternative to single and complete linkage by introducing the *integer link linkage* (K-ILL) family of monotone invariant clustering methods, where the parameter K is any positive integer. These methods can be defined in terms of the paradigmatic SAHN algorithm in a manner identical to that of U-PLL except that equation (1) is replaced by $J = \min\{\, K, |(ij)| \cdot |k| \,\}$. 1-ILL is single linkage. There is no fixed value of K for which K-ILL is complete linkage; but if one fixes instead the number n of objects being clustered, K-ILL corresponds to complete linkage whenever $K \geqslant n^2/4$. We know of no published investigations that use K-ILL for nontrivial values of K. The explanation may be that the paradigmatic SAHN algorithm for K-ILL can generate hierarchies in which *reversals* in partition levels (i.e., $\alpha_i > \alpha_j$ with $i < j$) occur when $K > 1$; Figure 1 exhibits such a reversal in an application of 2-ILL. Although single reversals may be dismissed simply as minor irritations, other pathological cases are more difficult to ignore; for example, Figure 2 exhibits an avalanche of reversals in an application of 2-ILL.

Dissimilarity Matrix D

$$\begin{array}{c|cccc|} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 1 & 2 & 2 \\ 2 & 1 & 0 & 4 & 4 \\ 3 & 2 & 4 & 0 & 3 \\ 4 & 2 & 4 & 3 & 0 \\ \hline \end{array}$$

Hierarchical Clustering

i	**Level α_i**	**Partition P_i**
0	0	{{1}, {2}, {3}, {4}}
1	1	{{1,2}, {3}, {4}}
2	3	{{1,2}, {3,4}}
3	2	{{1,2,3,4}}

Figure 1. A reversal caused by the 2-ILL clustering method. Clusters {3} and {4} agglomerate at level $\alpha_2 = 3$; clusters {1,2} and {3,4} then have dissimilarity 2 and agglomerate immediately at level $\alpha_3 = 2$. Since $\alpha_2 > \alpha_3$, this hierarchy is not a hierarchical clustering scheme.

These reversals demonstrate that when K-ILL is defined for $K > 1$ by the paradigmatic SAHN algorithm, its output is not guaranteed to be an HCS so that it is not a well-defined hierarchical clustering method. The integer link linkage concept can be salvaged by clustering method models that prohibit reversals; Janowitz (1979a), for example, incorporates integer and proportional link linkages in his model of type I agglomerative monotone equivariant clustering methods.

Dissimilarity Matrix D

$$
\begin{array}{c|cccccccc|}
 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\
\hline
1 & 0 & 1 & 6 & 2 & 2 & 6 & 6 & 6 \\
2 & 1 & 0 & 6 & 6 & 6 & 6 & 6 & 6 \\
3 & 6 & 6 & 0 & 1 & 6 & 3 & 3 & 6 \\
4 & 2 & 6 & 1 & 0 & 6 & 6 & 6 & 6 \\
5 & 2 & 6 & 6 & 6 & 0 & 1 & 4 & 4 \\
6 & 6 & 6 & 3 & 6 & 1 & 0 & 6 & 6 \\
7 & 6 & 6 & 3 & 6 & 4 & 6 & 0 & 5 \\
8 & 6 & 6 & 6 & 6 & 4 & 6 & 5 & 0 \\
\hline
\end{array}
$$

Hierarchical Clustering

i	**Level α_i**	**Partition P_i**
0	0	{{1}, {2}, {3}, {4}, {5}, {6}, {7}, {8}}
.	.	.
.	.	.
4	5	{{1,2}, {3,4}, {5,6}, {7,8}}
5	4	{{1,2}, {3,4}, {5,6,7,8}}
6	3	{{1,2}, {3,4,5,6,7,8}}
7	2	{{1,2,3,4,5,6,7,8}}

Figure 2. An avalanche of reversals caused by the 2-ILL clustering method.

2. Clustering Space Distortion

It is important to understand when, and why, reversals occur since they cause clustering methods based on the paradigmatic SAHN algorithm to be ill-defined. If one visualizes clusters as points in an appropriate conceptual space, then a reversal in hierarchical clustering can be interpreted as a distortion of space in the vicinity of the newly formed cluster. As clusters i and j (with dissimilarity δ_{ij}) agglomerate into cluster (ij), the space near (ij) may appear to contract with respect to cluster k so that (ij) has diminished dissimilarity with respect to k. If the space contraction is such that $\delta_{(ij)k} < \delta_{ij}$, then a reversal in partition levels results. Lance and Williams (1966, 1967) give intuitive descriptions of distortion, contraction, conservation, and dilation of space. Hubert and Schultz (1975) investigate space distortion by analyzing the number and type of clusters formed by clustering methods at comparable levels. DuBien and Warde (1979) and Rohlf (1977) independently propose formal definitions of various space distortion concepts.

Although Rohlf (1977) and DuBien and Warde (1979) each define the concepts of space contraction, conservation, and dilation, their definitions reflect a preoccupation with SAHN methods that generally are not invariant to monotone increasing transformations of the dissimilarity matrix. Thus DuBien and Warde (1979, pp. 37-38) discuss using space distorting clustering methods with "some measure of distance for clustering data sets for which it is 'semi-reasonable' to assume at least an interval scale of measurement for the variables comprising each data point". However, Rohlf (1977, p. 9) explicates the original concepts provocatively: "Given three points i, j, and k, the dissimilarity between k and the cluster i∪j could (depending upon the clustering method) tend to be less than, equal to, or greater than the average dissimilarity between k and the members of the cluster. Lance and Williams (1967) classify such methods as space contracting, space conserving, and space dilating methods, respectively." The dependency of these definitions on average dissimilarity precludes their use with data having only ordinal significance, but they are easily adapted to this more general context in the following way.

Suppose a monotone invariant clustering method is implemented by an appropriate specialization of the paradigmatic SAHN algorithm. Assume clusters i and j are selected in step 1 for agglomeration into (ij) at level δ_{ij}. Let k be any cluster such that $k \neq i$ and $k \neq j$. The clustering method is called *space contracting* if, for all such i, j, and k, $\delta_{(ij)k}$ is not greater than the minimedian dissimilarity between objects in (ij) and objects in k. It is called *space dilating* if, for such i, j, and k, $\delta_{(ij)k}$ is not less than the minimedian dissimilarity between objects in (ij) and objects in k. It is *space conserving* if it is both space contracting and space dilating, and it is *space distorting* if it is not space conserving. With these definitions, the minimedian clustering method (Furnas 1980) is the only space conserving method, and the maximedian method is very mildly space dilating. The U-PLL methods are space contracting when $U \leqslant .5$, space dilating when $U \geqslant .5$, space conserving when $U = .5$, and space distorting when $U \neq .5$. The K-ILL methods exhibit more complex behavior. 1-ILL is single linkage and so is space contracting. For $K > 1$, the type of space distortion for given i, j, and k depends on K's ranking relative to that of the corresponding minimedian. As a consequence, Sneath (1966, p.5) observes that "an Integer Link Linkage has a special relation to Proportional Link Linkages: it corresponds to different degrees of Proportional Link Linkage depending on the number of [objects] in the clusters. An Integer Link Linkage would start as similar to Complete Linkage and become more and more like Single Linkage as the Sorting Level changed down the dendrogram." Thus K-ILL methods for $K > 1$ are space distorting without being either space contracting or space dilating.

Any space distorting clustering method that permits reversals is characterized by a violation of monotonicity in the sequence of partition levels. Using the notation of the preceding paragraph, a clustering method is called *monotonic* (or *monotone nondecreasing*) if $\delta_{(ij)k} \geqslant \delta_{ij}$ for all such clusters i, j, and k (Lance and Williams 1966). Rohlf (1977) then calls a clustering method *super space contracting* if it is not monotonic; consequently, a clustering method permits reversals if and only if it is super space contracting. The space contraction definitions have a basic difference in form: a method is space contracting if an inequality involving $\delta_{(ij)k}$ always holds, whereas a method is super space contracting if a different inequality involving $\delta_{(ij)k}$ is violated. Milligan (1979) and Batagelj (1981) give a characterization of monotonicity for SAHN methods that are combinatorial in the sense of Lance and Williams (1966, 1967); while Diday (1983) obtains analogous results for Jambu and Lebeaux' combinatorial generalization (1983). Since the K-ILL and U-PLL families are not combinatorial (Legendre and Legendre 1983a, p. 237), we give characterizations that identify the super space contracting methods among their members.

Theorem 1 *A K-ILL clustering method is monotonic if and only if $K = 1$.*

Proof If $K = 1$ then K-ILL is the monotonic single linkage clustering method.

If $K > 1$ then we describe the dissimilarity matrix of an example for which K-ILL exhibits a reversal. Let m be the least even integer such that $K + 2 \leqslant m^2$, and let $\{1, \ldots, 2m\}$ be the set of objects being clustered. The dissimilarity matrix $D = [d_{xy}]$ initially has zeros on, and ones off, the main diagonal. We modify entries above the main diagonal in three regions, with changes below to maintain symmetry. The region with entries d_{xy}, where $1 \leqslant x \leqslant m < y \leqslant 3m/2$, has $(K - 1)$ entries with value 2 and

$(m^2/2 - K + 1)$ entries with value 4. The region with entries d_{xy}, where $1 \leqslant x \leqslant m$ and $3m/2 < y \leqslant 2m$, has $(K - 1)$ entries with value 2 and $(m^2/2 - K + 1)$ entries with value 4. The region with entries d_{xy}, where $m < x \leqslant 3m/2 < y \leqslant 2m$, has all entries with value 3. Figure 1 shows a resulting dissimilarity matrix when $K = 2$.

When the K-ILL algorithm operates on this dissimilarity matrix, the last two partitions it generates exhibit the following reversal:

$$\alpha_{n-2} = 3,\ P_{n-2} = \{\{1, \ldots, m\}, \{m+1, \ldots, 2m\}\}\ ;$$

$$\alpha_{n-1} = 2,\ P_{n-1} = \{\{1, \ldots, 2m\}\}\ .$$

Thus K-ILL is super space contracting. ●

Theorem 2 *U-PLL clustering methods are monotonic for all $0 < U \leqslant 1$.*

Proof Suppose step 1 of the U-PLL algorithm selects clusters i and j to agglomerate, and let k be any cluster such that $k \neq i$ and $k \neq j$. For any clusters r and s, define c_{rs} to be the number of dissimilarities in the set $\{d_{xy}\colon x \in r, y \in s, d_{xy} < \delta_{ij}\}$. Since $\delta_{ik} \geqslant \delta_{ij}$, we have $c_{ik} < \lceil U \cdot |i| \cdot |k| \rceil$ so that $c_{ik} < U \cdot |i| \cdot |k|$. Similarly, $c_{jk} < U \cdot |j| \cdot |k|$. By substitution we obtain $c_{(ij)k} = c_{ik} + c_{jk} < U \cdot |(ij)| \cdot \ |k| \leqslant \lceil U \cdot |(ij)| \cdot |k| \rceil$ so that $\delta_{(ij)k} \geqslant \delta_{ij}$. ●

3. An Algorithm for U-PLL Clustering

Although U-PLL clustering methods have desirable monotone invariance and space distortion properties, they should also have algorithms using reasonable time and space resources to solve reasonably nontrivial problems. To evaluate the complexities of such algorithms, we measure *problem size* by the number n of objects to be clustered, and we describe an algorithm's *time* (respectively, *space*) *complexity* by a function $f(n)$ expressing, for each n, the largest amount of time (respectively, space) the algorithm needs to solve any problem instance of size n. In describing the asymptotic behavior of such positive valued functions, we say that $f(n)$ is $O(g(n))$ whenever there exists a positive real constant c such that $f(n) \leqslant c \cdot g(n)$ for all large positive n. Readers wishing detailed information on the analysis of algorithm complexity can consult the classic textbook by Aho, Hopcroft, and Ullman (1974) or the excellent survey by Weide (1977).

An extensive literature exists on the computation of SAHN clustering methods. Sibson (1973) describes a single linkage algorithm based on the efficient extension of a hierarchical clustering of m objects to one of $(m + 1)$ objects; the algorithm requires $O(n^2)$ time and $O(n)$ space. Defays (1977) uses this approach to obtain a complete linkage algorithm with the same asymptotic behavior. Recent advances establish algorithms requiring $O(n^2)$ time for all well-known combinatorial SAHN clustering methods (Day and Edelsbrunner 1984; Murtagh 1983, 1984). By comparison, comparatively little information is available on the complexities of algorithms for U-PLL methods. Furnas (1980) gives an algorithm for minimedian (.5-PLL) and maximedian methods that requires $O(n^4)$ time and $O(n^2)$ space. Execution times provided by Legendre (personal communication) suggest that his U-PLL algorithm requires $O(n^3)$ time. One can also analyze the complexity of the paradigmatic SAHN algorithm when it is specialized to the U-PLL computation. In any reasonable implementation, step 1 searches $\binom{n-m}{2}$ entries in matrix Δ and step 3 analyzes $O(mn)$ entries in matrix D. Since these steps dominate loop execution, the algorithm requires $O(n^3)$ time. The matrices cause any reasonable implementation of the algorithm to require $O(n^2)$ space. These bounds on asymptotic behavior of the paradigmatic SAHN algorithm for U-PLL are unlikely to be improved by implementation details.

Although U-PLL clustering methods are defined by specializing the paradigmatic SAHN algorithm, other algorithms can generate a U-PLL HCS from a dissimilarity matrix. We describe in Table 2 a new SAHN algorithm for U-PLL clustering that exemplifies Anderberg's *sorted matrix approach* to hierarchical clustering (1973). His sorted matrix algorithm for single linkage (p. 149) is a restatement of Kruskal's greedy algorithm (1956) to compute the minimum spanning tree of an edge-weighted complete undirected graph. Hubert (1973) gives a sorted matrix algorithm for the complete linkage method. These algorithms build an HCS by processing each dissimilarity matrix element exactly once in nondecreasing order of dissimilarity. They comprise two phases: an initialization phase in which matrix elements are placed in processing order; and a construction phase in which matrix elements are processed to generate the HCS. In our sorted matrix algorithm for U-PLL, these phases require $O(n^2 \log n)$ and $O(n^2)$ time, respectively, and so the entire U-PLL algorithm requires only $O(n^2 \log n)$ time. Figure 3 contrasts this asymptotic behavior with those of related clustering algorithms.

The sorted matrix algorithm for U-PLL is written in an almost self-evident high-level language called *Pidgin ALGOL* (Aho, Hopcroft, and Ullman 1974, pp. 33-39). It comprises an initialization phase (line 1) and an HCS construction phase (line 2). Two basic ideas underlie the algorithm design: process interobject dissimilarities in nondecreasing order; and maintain sufficient information to test efficiently for cluster agglomeration. Array $N = [n_i]$ maintains a count n_i of the number of objects in each cluster i in partition P_m. Array $C = [c_{ij}]$ maintains a count c_{ij} of the number of interobject dissimilarities d_{xy} encountered with x in cluster i, y in cluster j, and i less than j. If d_{xy} causes c_{ij} to be incremented and if $c_{ij} \geqslant \lceil U \cdot n_i \cdot n_j \rceil$

TABLE 2

Sorted Matrix SAHN Clustering Algorithm for U-PLL

```
begin
1.  Initialize N, C, Q and Z
2.  for m = 0 to n - 2 do
        begin
            found ← false
            repeat
3.              ($d_{xy}$ ,x,y) ← MIN(Q)
4.              i ← FIND(x,Z); j ← FIND(y,Z)
5.              if i ≠ j then
                    begin
                        $c_{ij}$ ← $c_{ij}$ + 1
                        if $c_{ij} \geq \lceil U \cdot n_i \cdot n_j \rceil$ then found ← true
                    end
            until found
6.          Agglomerate clusters i and j at level $\alpha_{m+1} = d_{xy}$ to obtain $P_{m+1}$ from $P_m$.
7.          UNION (i,j,k,Z)
8           Update N and C to reflect agglomeration of i and j into k.
        end
end
```

(line 5), then partition P_{m+1} can be constructed (line 6) by agglomerating i and j in $O(n)$ time. Initializing N and C (line 1) requires $O(n^2)$ time, while updating them at each execution of line 8 requires $O(n)$ time.

Since the sorted matrix algorithm for U-PLL requires interobject dissimilarities in nondecreasing order, it constructs (line 1) from the dissimilarity matrix $D = [d_{xy}]$ a priority queue (Aho, Hopcroft, and Ullman 1974, pp. 147-152) Q of elements (d_{xy}, x, y), $1 \leqslant x < y \leqslant n$, where d_{xy} is called the element's label. Elements of Q are made available (line 3) in nondecreasing order of label value; the MIN operation returns an element with least label value and deletes it from Q. Q can be implemented as an array of elements; it's $O(n^2)$ entries are sorted into proper order (line 1) in $O(n^2 \log n)$ time, while each MIN execution (line 3) requires only constant time.

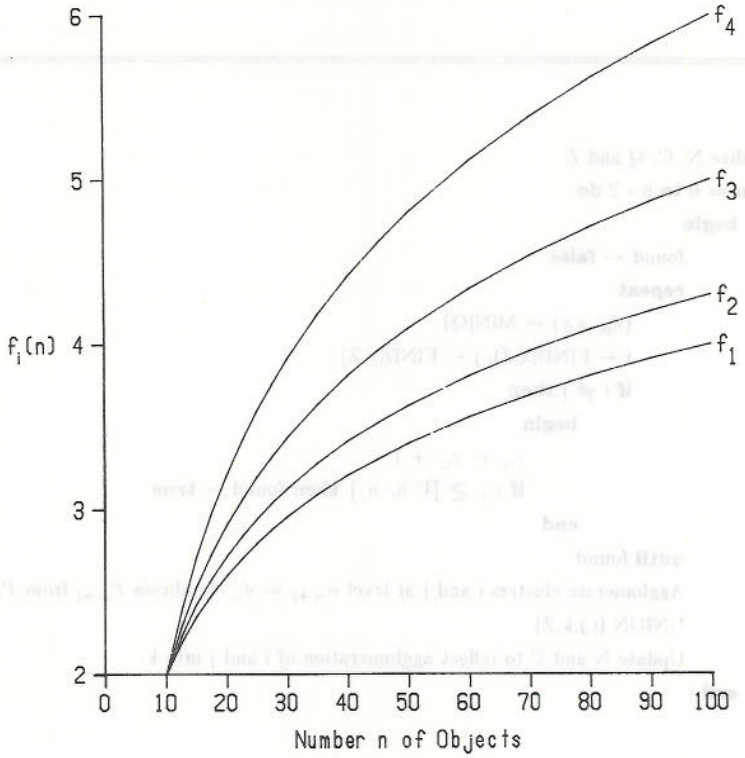

Figure 3. Asymptotic behaviors of clustering methods. The curves f_i represent functions $f_i(n) = \log_{10}(G_i(n))+c_i$, the constants c_i providing normalization at $n = 10$. $G_1(n) = n^2$ is the time complexity of Sibson's single linkage algorithm (1973): it also is a trivial lower bound on the time complexities of clustering algorithms having dissimilarity matrices as input. $G_2(n) = n^2 \log_{10} n$ and $G_3(n) = n^3$ are time complexities of the sorted matrix and paradigmatic SAHN algorithms for U-PLL. $G_4(n) = n^4$ is the time complexity of a minimedian (.5-PLL) algorithm (Furnas 1980).

When the sorted matrix algorithm for U-PLL processes the element (d_{xy}, x, y) returned by MIN (line 3), it must identify which clusters contain objects x and y. This can be done by maintaining the partition P_m as a set Z of mutually exclusive subsets (the clusters) subject to FIND and UNION operations. *FIND(x, Z)* returns the name of the cluster in Z of which x is currently a member. *UNION(i, j, k, Z)* replaces clusters i and j in Z with the agglomerated cluster k. Z can be implemented as an array $Z = [z_x]$ of size n in which z_x is the name of the cluster containing object x (Aho, Hopcroft, and Ullman 1974, pp. 124-125). Initialization of Z (line 1) requires $O(n)$ time; each FIND execution (line 4) requires only constant time, while each UNION execution (line 7) can be accomplished in $O(n)$ time by scanning Z sequentially.

Now we can complete the time complexity analysis of the sorted matrix algorithm for U-PLL. The initialization phase (line 1) is dominated by the $O(n^2 \log n)$ time required to sort Q. Analysis of the HCS construction phase (line 2) breaks into two parts. Each execution of the **repeat** clause (lines 3-5) requires constant time; since the clause is invoked $O(n^2)$ times, the **repeat** statement requires $O(n^2)$ time overall. Each execution of lines 6-8 requires $O(n)$ time; since these lines are invoked $(n-1)$ times, they require $O(n^2)$ time overall. Since the construction phase thus requires $O(n^2)$ time, the entire algorithm is dominated by the initialization phase and requires $O(n^2 \log n)$ time overall.

4. Conclusion

We investigated properties of two families of SAHN clustering methods that Sneath (1966) introduced as integer link linkage (K-ILL) and proportional link linkage (U-PLL). Since these methods are invariant to monotone increasing transformations of dissimilarity matrix input, they may be useful in psychology, ecology, and other areas where clustering data typically have only ordinal significance. We defined basic concepts to describe the clustering space distortion exhibited by monotone invariant SAHN algorithms. We established that every nontrivial K-ILL method is super space contracting so that reversals may occur in the clustering hierarchy, while every U-PLL method is monotone so that reversals are impossible. We showed that each U-PLL method exhibits a type of clustering space distortion characterized by U. We described a new U-PLL algorithm that requires $O(n^2 \log n)$ time and $O(n^2)$ space to cluster n objects. This asymptotic behavior is better than that of other known U-PLL algorithms, but we do not know if it is the best possible.

There remain challenging unsolved research problems concerning U-PLL clustering methods. Since these methods exhibit reasonable, predictable, space distortion properties, they should next be subjected to quantitative comparisons with other monotone invariant clustering methods. As for algorithmic considerations, we believe it would be of considerable interest to design U-PLL algorithms with time complexities superior to that of the sorted matrix algorithm we described. Since such algorithms may not exist, we also would welcome any improved lower bounds on the time complexities of all U-PLL algorithms. Our goal is to discover a U-PLL algorithm with time complexity $O(f(n))$, and a lower bound $g(n)$ on the time complexities of all U-PLL algorithms, such that $f(n) = O(g(n))$; such an algorithm may rightfully be called optimal for U-PLL clustering.

References

AHO, A.V., HOPCROFT, J.E. and ULLMAN, J.D., (1974), *The Design and Analysis of Computer Algorithms,* Reading, Massachusetts: Addison-Wesley.

ANDERBERG, M.R., (1973), *Cluster Analysis for Applications,* New York: Academic Press.

BATAGELJ, V., (1981), "Note on Ultrametric Hierarchical Clustering Algorithms," *Psychometrika, 46,* 351-352.

BROMLEY, D.B., (1966), "Rank Order Cluster Analysis," *British Journal of Mathematical and Statistical Psychology, 19,* 105-123.

DAY, W.H.E., and EDELSBRUNNER, H., (1984), "Efficient Algorithms for Agglomerative Hierarchical Clustering Methods," *Journal of Classification, 1,* 7-24.

D'ANDRADE, R.G., (1978), "U-statistic Hierarchical Clustering," *Psychometrika, 43,* 59-67.

DEFAYS, D., (1977), "An Efficient Algorithm for a Complete Link Method," *Computer Journal, 20,* 364-366.

DIDAY, E. (1983), "Inversions en Classification Hiérarchique: Application à la Construction Adaptative d'indices d'agrégation," *Revue de Statistique Appliquée, 31,* 45-62.

DuBIEN, J.L., and WARDE, W.D., (1979), "A Mathematical Comparison of the Members of an Infinite Family of Agglomerative Clustering Algorithms," *Canadian Journal of Statistics, 7,* 29-38.

EVERITT, B., (1980), *Cluster Analysis* (second edition), London: Heinemann.

FLOREK, K., LUKASZEWICZ, J., PERKAL, J., STEINHAUS, H., and ZUBRZYCKI, S., (1951), "Taksonomia Wroclawska," *Przeglad Antropologiczny, 17,* 193-211.

FURNAS, G.W., (1980), "Objects and Their Features: The Metric Representation of Two-Class Data," Ph.D. dissertation, Department of Psychology, Stanford University, Stanford, California.

HUBERT, L., (1972), "Some Extensions of Johnson's Hierarchical Clustering Algorithms," *Psychometrika, 37,* 261-274.

HUBERT, L., (1973), "Monotone Invariant Clustering Procedures," *Psychometrika, 38,* 47-62.

HUBERT, L.J., (1974), "Some Applications of Graph Theory to Clustering," *Psychometrika, 39* 283-309.

HUBERT, L., (1977), "A Set-Theoretical Approach to the Problem of Hierarchical Clustering," *Journal of Mathematical Psychology, 15,* 70-88.

HUBERT, L., and BAKER, F.B., (1976), "Data Analysis by Single-Link and Complete-Link Hierarchical Clustering," *Journal of Educational Statistics, 1,* 87-111.

HUBERT, L., and SCHULTZ, J., (1975), "Hierarchical Clustering and the Concept of Space Distortion," *British Journal of Mathematical and Statistical Psychology, 28,* 121-133.

JAMBU, M., and LEBEAUX, M.-O., (1983), *Cluster Analysis and Data Analysis,* New York: Elsevier Science.

JANOWITZ, M.F., (1978a), "An Order Theoretic Model for Cluster Analysis," *SIAM Journal on Applied Mathematics, 34,* 55-72.

JANOWITZ, M.F., (1978b), "Semiflat L-Cluster Methods," *Discrete Mathematics, 21,* 47-60.

JANOWITZ, M.F., (1979a), "Monotone Equivariant Cluster Methods," *SIAM Journal on Applied Mathematics, 37,* 148-165.

JANOWITZ, M.F., (1979b), "Preservation of Global Order Equivalence," *Journal of Mathematical Psychology, 20,* 78-88.

JANOWITZ, M.F. (1981), "Continuous L-Cluster Methods," *Discrete Applied Mathematics, 3,* 107-112.

JARDINE, N., and SIBSON, R., (1968), "A Model for Taxonomy," *Mathematical Biosciences, 2,* 465-482.

JARDINE, N., and SIBSON, R., (1971), *Mathematical Taxonomy,* New York: John Wiley.

JOHNSON, S.C., (1967), "Hierarchical Clustering Schemes," *Psychometrika, 32,* 241-254.

KENDRICK, W.B., and PROCTOR, J.R., (1964), "Computer Taxonomy in the Fungi Imperfecti," *Canadian Journal of Botany, 42,* 65-88.

KRUSKAL, J.B., (1956), "On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem," *Proceedings of the American Mathematical Society, 7,* 48-50.

LANCE, G.N., and WILLIAMS, W.T., (1966), "A Generalized Sorting Strategy for Computer Classifications," *Nature, 212,* 218.

LANCE, G.N., and WILLIAMS, W.T., (1967), "A General Theory of Classificatory Sorting Strategies. 1. Hierarchical Systems," *Computer Journal, 9,* 373-380.

LEGENDRE, L., and LEGENDRE, P., (1983a), *Numerical Ecology,* Amsterdam: Elsevier Scientific.

LEGENDRE, L., and LEGENDRE, P., (1983b), "Partitioning Ordered Variables into Discrete States for Discriminant Analysis of Ecological Classifications," *Canadian Journal of Zoology, 61,* 1002-1010.

LEGENDRE, P., BALEUX, B., and TROUSSELLIER, M., (1984), "Dynamics of Pollution-indicator and Heterotrophic Bacteria in Sewage Treatment Lagoons," *Applied and Environmental Microbiology, 48,* 586-593.

LEGENDRE, P., DALLOT, S., and LEGENDRE, L., (1985), "Succession of Species Within a Community: Chronological Clustering, with Applications to Marine and Freshwater Zooplankton," *American Naturalist, 125,* 257-288.

LEGENDRE, P., and LEGENDRE, V. (1984), "Postglacial Dispersal of Freshwater Fishes in the Quebéc Peninsula," *Canadian Journal of Fisheries and Aquatic Sciences, 41,* 1781-1802.

LING, R.F., (1972), "On the Theory and Construction of k-Clusters," *Computer Journal, 15,* 326-332.

LUKASZEWICZ, J., (1951), "Sur la liaison et la division des points d'un ensemble fini," *Colloquium Mathematicum, 2,* 282-285.

MATULA, D.W., (1977), "Graph Theoretic Techniques for Cluster Analysis Algorithms," in *Classification and Clustering,* ed. J. van Ryzin, New York: Academic Press, 95-129.

McQUITTY, L.L., (1957), "Elementary Linkage Analysis for Isolating Orthogonal and Oblique Types and Typal Relevancies," *Educational and Psychological Measurement, 17,* 207-229.

MILLIGAN, G.W., (1979), "Ultrametric Hierarchical Clustering Algorithms," *Psychometrika, 44,* 343-346.

MURTAGH, F., (1983), "A Survey of Recent Advances in Hierarchical Clustering Algorithms," *Computer Journal, 26,* 354-359.

MURTAGH, F., (1984), "Complexities of Hierarchic Clustering Algorithms: State of the Art," *Computational Statistics Quarterly, 1,* 101-113.

ROHLF, F.J., (1963), "Classification of Aedes by Numerical Taxonomic Methods (Diptera: Culicidae)," *Annals of the Entomological Society of America, 56,* 798-804.

ROHLF, F.J., (1977), "Computational Efficiency of Agglomerative Clustering Algorithms," Report RC 6831, IBM T. J. Watson Research Center, Yorktown Heights, New York.

SIBSON, R., (1970), "A Model for Taxonomy. II," *Mathematical Biosciences, 6,* 405-430.

SIBSON, R., (1972), "Order Invariant Methods for Data Analysis," *Journal of the Royal Statistical Society Series B, 34,* 311-349.

SIBSON, R., (1973), "SLINK: An Optimally Efficient Algorithm for the Single-Link Cluster Method," *Computer Journal, 16,* 30-34.

SNEATH, P.H.A., (1957), "The Application of Computers to Taxonomy," *Journal of General Microbiology, 17,* 201-226.

SNEATH, P.H.A., (1966), "A Comparison of Different Clustering Methods as Applied to Randomly-spaced Points," *Classification Society Bulletin, 1,* 2-18.

SNEATH, P.H.A., and SOKAL, R.R., (1973), *Numerical Taxonomy,* San Francisco: W. H. Freeman.

SOKAL, R.R., and SNEATH, P.H.A., (1963), *Principles of Numerical Taxonomy,* San Francisco: W. H. Freeman.

SØRENSEN, T., (1948), "A Method of Establishing Groups of Equal Amplitude in Plant Sociology Based on Similarity of Species Content and its Application to Analyses of the Vegetation on Danish Commons," *Biologiske Skrifter, 5,* 1-34.

WARD, Jr., J.H., (1963), "Hierarchical Grouping to Optimize an Objective Function," *Journal of the American Statistical Association, 58,* 236-244.

WEIDE, B., (1977), "A Survey of Analysis Techniques for Discrete Algorithms," *Computing Surveys, 9,* 291-313.

WISHART, D., (1969), "256 Note. An Algorithm for Hierarchical Classifications," *Biometrics, 25,* 165-170.