AN LP RELAXATION PROCEDURE FOR ANNOTATING POINT FEATURES USING INTERACTIVE GRAPHICS

Robert G. Cromley
University of Connecticut
Storrs, CT 06268

ABSTRACT

The development of computer-assisted mapping procedures has made the design of maps a normative problem in the processing and communication of information. However, mathematical programming techniques have rarely been used by cartographers in the optimal design of maps. Instead, heuristics have been employed. A modified linear programming technique is used here to operationalize the annotation of point features. An interactive relaxation procedure is utilized to avoid the problem of having to define algebraically the set of cartographic restrictions on the placement of labels. Because all constraints in the point annotation problem are graphic in nature, any violated constraint in the relaxed problem can be visually detected on a screen image. The viewer interactively adds violated constraints to the algebraic problem until no violations appear on the screen image. This approach is promising as many cartographic rules are partially qualitative and human interaction is an appropriate method to resolve conflicts.

INTRODUCTION

As cartographers have defined mapping procedures for the accurate and efficient graphic display of information, the concept of designing an optimal map by machine has been proposed by various researchers. The map as a normative model for processing information has been used in determining class intervals (Jenks and Caspall, 1971; Monmonier, 1973), line generalization (Douglas and Peucker, 1973), gray-tone selection (Smith, 1980), and more recently annotation (Ahn and Freeman, 1983; Freeman and Ahn, 1984). With few exceptions, though, cartographers have not used techniques of optimization from mathematical programming but instead have opted for heuristics as cartographic problems have been difficult to define as a linear or nonlinear program. This paper is an initial investigation into the integration of linear programming procedures with interactive graphics to overcome some of the difficulties of operationalizing the optimal annotation of point features.

The interface between mathematical programming and computer graphics holds promise for the solution of many optimization problems in cartography and spatial analysis that may be more difficult to solve using either technique in isolation of the other. Many algebraic constraints of linear and nonlinear programs used in spatial analysis can be transformed into a graphic display. Thus, constraints can be modelled graphically and the set of feasible solutions identified visually. In this manner, algebraic programs have the potential to be solved by inspection on a graphics terminal. Composite mapping is one such graphic technique that has been applied to site location problems. Conversely, mathematical programs can find solutions that minimize or maximize some utility function over space that are not recognizable by the human eye. By combining these two techniques, it is possible to synthesize the positive attributes of each. The point annotation problem is amenable to this approach because it has the properties of a mathematical program although the final solution has a graphic form. First, a discussion of the point feature annotation problem is presented and then a linear programming (LP) relaxation procedure is defined to solve the problem.

MACHINE ANNOTATION OF POINT FEATURES

The annotation of point features is just one of three name placement problems in mapping. The positions of names for area features, line features, and point features must be determined in unison although a feature name with a smaller degree of freedom with respect to its placement is usually placed before names with a larger latitude in their positioning. The study is restricted here to just the point feature problem for ease of exposition and will be expanded in the future.

While the positioning of names on a map is somewhat subjective, cartographers have attempted to standardize the rules for an aesthetically pleasing yet informative map. Imhof (1975) has formulated a set of guidelines for general map annotation that has been refined and expanded by Freeman and Ahn for the point feature problem (Ahn and Freeman, 1983; Freeman and Ahn, 1984). Their rules are summarized here as: 1) names should be horizontal in an east-west orientation, 2) each name should be near but not too close to the point feature that it annotates, and 3) name placement above and to the right of the point is generally preferred. Although numerous name positions are possible with respect to each point, Figure 1 presents six of the more preferred positions. While the rank ordering of positions in this figure is subjective, it is illustrative of a potential preference function. This utility function could be determined by a panel of experts.

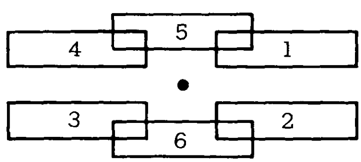

Figure 1. An example set of name positions

The crux of annotating point features is not where to position the name with respect to its point but where it should be positioned relative to the location of other point labels (Freeman and Ahn, 1984, p. 556). Names may need to be repositioned until a final placement pattern is determined. For example, in Figure 2 the preferred name placement for point four conflicts with the name placement for point three. However, moving point four's name to another position may conflict with the placement of names for other points. Therefore, the positions of several names may need to be readjusted before the final name pattern for all four points is determined.

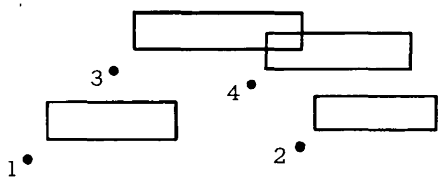

Figure 2. An example positioning conflict

This adjustment process has been resolved by constructing a graph of possible name conflicts (Freemand and Ahn, 1984). Each node in the graph represents a point; two nodes are connected if their name placement areas overlap. Each connected component of the graph is processed independently to resolve any potential conflicts within the area represented by the component. As each node is processed, its branches to other points are examined to determine the position of its name. If no name can be placed, the procedure backtracks to a previous node and the positions of placed labels are altered to accommodate the new point.

This resolution of overlap problems may be indeterminant with respect to the preference for name placement. The resolution to the positioning conflict in Figure 2 may have required the relocation of point three's name or alternatively the relocation of names for points one, two, and four. Therefore, the prefernce function for individual points should be extended to a utility function for the entire distribution of names. One pattern should be preferred over another; this problem can be overcome within the context of linear programming.

AN LP RELAXATION PROCEDURE

The problem of annotating point features is amenable to solution by linear programming techniques because preferences for the possible positioning of a point label have been outlined and a general set of rules (i.e. no names should overlap) that constrain the final distribution of of names has been identified. Position preferences can be modelled as the objective function of the program and the general set of rules forms the constraint set. The range of potential name positions, as previously discussed, is limited to one of six places for each point (see Figure 1). Each position is assigned a priority weight from one to six based on the position's preference ranking where one is the most desirable position and six is the least. The objective of the point feature annotation problem is to produce a map that minimizes the sum of the priority weights for the entire label distribution. This goal is expressed algebraically as:

$$\text{Minimize} \quad \sum_{i=1}^{n} \sum_{j=1}^{6} W_{ij} X_{ij} \; ; \qquad (1)$$

where, n is the number of points to be annotated;
W_{ij} is the priority weight of the jth position for the ith point;
and, X_{ij} is a decision variable that determines whether the jth position is chosen for the ith point.

The constraint set for this program is defined by the following two rules: 1) each point must be assigned a label, and 2) no label can overlap or intersect another point or label. The first rule can be modelled algebraically by the following inequality:

$$\sum_{j=1}^{6} X_{ij} \geq 1 \qquad \text{for all i.} \qquad (2)$$

This inequality forces at least one of the positions to be selected for each point and no more than one would be chosen given the minimizing goal of the objective function. Using just equations (1)-(2) as an LP, the optimal solution would produce a map where the label for each point would be assigned to its most preferable position. While finding the solution to this problem is trivial, it could satisfy the broader annotation problem if no labels overlapped. In a clustered or dense point distribution, however, this would be an unlikely occurance.

The modelling of the overlap rule is more difficult than the assignment requirement. Usually, linear programs are executed on a computer in batch mode because as a normative model once the parameters have been set, the solution is deterministic. Unfortunately, the point annotation problem does not lend itself to solution by this approach. The overlap contraints for labelling point features define a set of geographic relationships that cannot be determined without analyzing a graphic display of all possible placement combinations. The final constraint set may contain too many inequalities for efficient solution and some violations of the overlap rule may be accidentally overlooked and thus omitted from the LP model.

The model presented here utilizes a relaxation procedure to avoid these problems. The general philosophy of relaxation techniques is to solve initially a simpler problem with fewer constraints than the original one. If the optimal solution to the relaxed problem does not violate any of the omitted constraints, then it is also the optimal solution to the original problem (Lasdon, 1970, p. 268). Whenever constraints are violated, they are explicitly included and the problem is iteratively solved until no relaxed constraint is violated. Normally, this iterative process is transparent to the user as violated constraints are identified algebraically by the program. Because overlap constraints in the point annotation problem are graphic in nature, each violated relaxed constraint in the program has a graphic manifestation on a map image. This suggests that an interactive graphic approach can be used where violated constraints are detected visually by the user and denoted for inclusion as an explicit constraint in the next iteration. For this model, the following algebraic inequality is used to implement each overlap restriction:

$$X_{ir} + X_{jk} \lesssim 1 \; ; \qquad (3)$$

where, X_{ir} denotes the rth label position of the ith point;
and, X_{jk} is the kth label position of the jth point.

This inequality only permits one of the two overlapping positions to be present in the next iteration while the objective function searches for the next best preference pattern of point labels. This process continues until the user cannot detect any more overlap violations on the screen image.

This LP relaxation procedure was written in FORTRAN for implementation on a Tektronix 4012 graphics terminal at the University of Connecticut. To improve the overall aesthetic appearance of the final map display, the user is allowed to make minor adjustments to any of the map parameters used by the program. Additionally, the user has the option to modify the system objective function for choosing the optimal annotation distribution. The utility function discussed above minimizes total priority weight; although this goal will try to place names in an overall system design, individual points may have their worst label position selected. An alternative utility function would be to place names such that the worst placement for any single point is avoided. This minimax criterion is used frequently in public facility location and can easily be implemented by powering the priority weights for each point.

SUMMARY

The interface between mathematical programming and computer graphics has the potential to solve problems in cartographic design and spatial analysis. Whenever system constraints are graphic in nature, they can be modelled as such and solved using a relaxation procedure. If any constraints are violated in the relaxed problem, they can be interactively included and the problem solved again until no relaxed constraints are detected. While this approach was applied here to the point annotation problem, this man-display-program has appeal whenever system rules are partially qualitative by nature and human interaction is an appropriate method for resolving any conflicts. The system also has the flexibility to use different utility functions to find the most aesthetically pleasing name distribution.

ACKNOWLEDGMENTS

The author wishes to thank Jeanne Mengwasser for final preparation of all figures.

REFERENCES

Ahn, J. and Freeman, H. 1983, A Program for Automatic Name Placement: Proceedings, AUTO-CARTO VI, Vol. 2, pp. 444-453.

Douglas, D. and Peucker, T. 1973, Algorithms for the Reduction of the Number of Points Required to Represent a Digitized Line or its Caricature: The Canadian Cartographer, Vol. 10, pp. 112-122.

Freeman, H. and Ahn, J. 1984, AUTONAP - An Expert System for Automatic Map Name Placement: Proceedings, International Symposium on Spatial Data Handling, Vol. 2, pp. 544-569.

Imhof, E. 1975, Positioning Names on Maps: The American Cartographer, Vol. 2, pp. 128-144.

Jenks, G. and Caspall, F. 1971, Error on Choroplethic Maps: Definition, Measurement, Reduction: Annals, the Association of American Geographers, Vol. 61, pp. 217-244.

Lasdon, L. Optimization Theory for Large Systems. 1970. New York: The Macmillan Company.

Monmonier, M. 1973, Analogs Between Class-Interval Selection and Location-Allocation Models: The Canadian Cartographer, Vol. 10, pp. 123-131.

Smith, R. 1980, Improved Areal Symbols for Computer Line-Printed Maps: The American Cartographer, Vol. 7, pp. 51-57.