PSYCHOMETRIKA—VOL. 66, NO. 3, 405–420
SEPTEMBER 2001

COMPACT INTEGER-PROGRAMMING MODELS FOR EXTRACTING SUBSETS OF STIMULI FROM CONFUSION MATRICES

MICHAEL J. BRUSCO

MARKETING DEPARTMENT
FLORIDA STATE UNIVERSITY

STEPHANIE STAHL

TALLAHASSEE, FLORIDA

This paper presents an integer linear programming formulation for the problem of extracting a subset of stimuli from a confusion matrix. The objective is to select stimuli such that total confusion among the stimuli is minimized for a particular subset size. This formulation provides a drastic reduction in the number of variables and constraints relative to a previously proposed formulation for the same problem. An extension of the formulation is provided for a biobjective problem that considers both confusion and recognition in the objective function. Demonstrations using an empirical interletter confusion matrix from the psychological literature revealed that a commercial branch-and-bound integer programming code was always able to identify optimal solutions for both the single-objective and biobjective formulations within a matter of seconds. A further extension and demonstration of the model is provided for the extraction of multiple subsets of stimuli, wherein the objectives are to maximize similarity within subsets and minimize similarity between subsets.

Key words: combinatorial data analysis, confusion matrix, integer programming.

Introduction

Many psychological experiments are characterized by a set of objects, $S = \{1, 2, \ldots, n\}$, that represents both stimuli and responses. Under such circumstances, the $n \times n$ confusion matrix, $\mathbf{C}$, is a convenient method for representing the experimental data. Typically, the rows of the confusion matrix correspond to the stimuli, the columns correspond to the responses, and entries, c_{ij}, represent the number of instances (or proportion of instances) for which response j was given for stimulus i. Entries on the main diagonal of $\mathbf{C}$ represent the correct *recognition* of the stimuli, whereas off-diagonal elements indicate incorrect responses or confusion among the stimuli. Confusion matrices are quite common in a variety of psychological areas including visual alphabetic letter recognition (Appelman & Mayzner, 1982; Dawson & Harshman, 1986; Townsend, 1971; Townsend & Ashby, 1982), speech and/or auditory recognition (Hodge & Pollack, 1962; Miralles & Cervera, 1995; van Son & Pols, 1999), taste recognition (Hettinger, Gent, Marks, & Frank, 1999), tactile recognition (Loomis, 1990; Vega-Bermudez, Johnson, & Hsiao, 1991), odor discrimination (Kent, Youngentob, & Sheehe, 1995; Youngentob, Markert, Mozell, & Hornung, 1990), lip reading (Manning & Shofner, 1991; Massaro, Cohen, & Gesi, 1993), and ergonomic design of instrument panels (Moore, 1974).

Confusion matrices can be been analyzed using unidimensional seriation and scaling methods (Baker & Hubert, 1977; Hubert, 1974; Hubert, 1976; Hubert, Arabie, & Meulman, 1997; Hubert & Schultz, 1976), multidimensional scaling (Kruskal, 1964; Townsend, 1971; Yougnentob et al., 1990; Zielman & Heiser, 1996), and hierarchical cluster analysis (Hettinger et al., 1999; Smith & Jones, 1975; Smith, Wilson, & Jones, 1975). Hubert (1987) reviewed a variety of potential applications of the quadratic assignment paradigm regarding the analysis of confusion

Requests for reprints should be sent to Michael J. Brusco, Marketing Department, College of Business, Florida State University, Tallahassee, FL 32306-1110. E-Mail: mbrusco@cob.fsu.edu

0033-3123/2001-3/2000-0813-A $00.75/0

matrices. These include compactness and isolation of subsets, as well as evaluation of symmetry, subset selection and seriation. A large body of confusion-matrix research has focused on the "constant ratio rule" (Clarke, 1957; Hodge & Pollack, 1962; Rich, 1971; Townsend & Landon, 1982). This rule specifies that if Experiment 1 and Experiment 2 are precisely the same in every way except that the set of stimuli (S_1) in Experiment 1 is a proper subset of the set of stimuli (S_2) in Experiment 2 ($S_1 \subset S_2$), then the confusion matrix of Experiment 1 will reflect the same proportions as observed in the submatrix of Experiment 2.

In this paper, we focus on the combinatorial optimization problem associated with the extraction of a subset of stimuli from a confusion matrix. Theise (1989) carefully identifies the assumptions of the subset extraction problem, which include the constant ratio rule. Instead of evaluating clusterings or reorderings of rows and columns of a confusion matrix, the problem posed by Theise is concerned with the extraction of a fixed number of stimuli from a confusion matrix such that the resulting subset contains minimum total confusion. Such problems occur in the redesign of automobile controls (Green & Pew, 1978), the selection of directional symbols that will be clear to passengers (Zwaga & Boersema, 1983), and the identification of a subset of push-buttons that will be clear to equipment operators (Moore, 1974). Other examples might include the design of computer keyboards, the programming of buttons for fast-food restaurant cash registers, and the reduction of a large inventory of perfumes or paint colors to a more manageable size.

Theise (1989) presents several linear 0-1 integer-programming models for subset extraction problems. However, considerably improved mathematical formulations are possible and that is the focus of our paper. There are at least three important reasons for developing improved formulations: (a) an increase in the size of confusion matrices to which optimal subset extraction methods can be applied, (b) reduced storage requirements and computational effort in obtaining optimal solutions and (c) greater flexibility in analysis objectives and constraints. We present a compact integer linear programming formulation that provides a substantial reduction in the number of variables and constraints relative to the models suggested by Theise. We also present an extension of the formulation to a biobjective subset extraction problem that considers both confusion and recognition in the objective function. We further extend our modeling approach to an important problem described by Heiser (1988), in which multiple subsets are extracted such that the similarity within subsets is large and the similarity between subsets is small. These compact models increase the feasibility of integer programming methods for subset extraction problems.

In the next section of this paper, we present single-objective mathematical programming formulations for the problem of extracting a subset of minimum total confusion and demonstrate our new formulation using an empirical interletter confusion matrix from Heiser (1988, p. 41). Section 3 presents a biobjective subset extraction model that considers both total confusion and total recognition in the objective function. A demonstration of the biobjective model is also provided in section 3. In section 4, we present a model for extracting multiple subsets of stimuli from a matrix, along with a corresponding demonstration for the interletter confusion matrix. Limitations and extensions are presented in section 5 and the paper concludes with a brief summary in section 6.

Mathematical Programming Formulations for the Subset Extraction Problem

A Quadratic 0-1 Formulation

The subset extraction problem is concerned with the selection of a subset of m stimuli from a master set of n stimuli ($1 < m < n$) such that total confusion is minimized. Consistent with Theise (1989), confusion consists of incorrect responses for stimuli ($c_{ij} + c_{ji}$), as well as undecided responses for each stimuli ($u_i, i = 1, \ldots, n$). As explained by Theise (1989), a stimulus that is selected for inclusion in the subset should be appropriately penalized for undecided responses to that stimulus. In some instances, such as the push-button data set studied by Moore (1974) and Theise (1989), the number or proportion of undecided responses might be available and important to the analysis. In other cases, undecided responses might be prohibited or not relevant to the analysis. Defining $x_i = 1$ if stimulus i is selected for inclusion in the subset and 0 otherwise ($i = 1, \ldots, n$), the subset extraction problem can be modeled as a quadratic 0-1 integer programming problem (hereafter Q1) as follows:

$$\text{Minimize } Z = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} (c_{ij} + c_{ji}) x_i x_j + \sum_{i=1}^{n} u_i x_i. \tag{1}$$

$$\text{Subject to} \quad \sum_{i=1}^{n} x_i = m, \tag{2}$$

$$x_i \in \{0, 1\} \qquad \forall i = 1, \ldots, n. \tag{3}$$

The objective function (1) of Q1 represents total confusion associated with the extracted subset. The constant $c_{ij} + c_{ji}$ represents the total confusion between stimulus i and stimulus j. This term is only counted in the objective function if both stimulus i and j are selected in the subset (i.e., $x_i = x_j = 1$). Constraint (2) guarantees that the number of selected stimuli is equal to m and the constraints in (3) place binary restrictions on the decision variables. Kuo, Glover, and Dhir (1993) demonstrate that the NP-complete clique problem (Garey & Johnson, 1979) is reducible to Q1 and thus problem Q1 is NP-hard.

There are at least two possible strategies for generating optimal solutions for modest sized versions of Q1. One approach is to tackle directly problem Q1 using an enumerative algorithm (Comley, 1996; Hansen, 1972; Taha, 1972). Another option, which was employed by Theise (1989), is the conversion of Q1 into a linear 0-1 programming problem (Glover, 1975; Glover & Woolsey, 1974). The relative efficacy of enumerative and transformed linear approaches is difficult to gauge. The general consensus is that neither approach can claim superiority in all cases (Glover, 1975) and that certain problem instances can be particularly difficult for certain algorithms (Comley, 1996). However, one pragmatic advantage of the linearization approach is the widespread availability of commercial branch-and-bound integer linear programming software. Kuo et al. (1993, p. 1174) observed that the enumerative quadratic methods "...have not undergone the intensive refinements of linear zero-one methods, nor have they found widespread use in real-world applications". For these reasons, we focus on the linearization approach throughout the remainder of this paper, while recognizing that direct quadratic methods, dynamic programming, and branch-and-bound techniques might also provide viable solution approaches.

Theise's (1989) Integer Linear Programming Formulation (P1 and P1-E)

Theise (1989) presented a linearization of Q1 that required the definition of an additional set of binary decision variables; $y_{ij} = 1$ if stimulus i and stimulus j are both selected for inclusion in the subset and 0 otherwise ($1 \le i < j \le n$). Theise's (1989) integer linear programming model (hereafter P1) for extracting a subset of stimuli from a confusion matrix is as follows:

$$\text{Minimize } Z_1 = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} (c_{ij} + c_{ji}) y_{ij} + \sum_{i=1}^{n} u_i x_i. \tag{4}$$

$$\text{Subject to} \quad x_i + x_j - y_{ij} \le 1 \qquad \text{for } 1 \le i < j \le n, \tag{5}$$

$$-x_i - x_j + 2y_{ij} \le 0 \qquad \text{for } 1 \le i < j \le n, \tag{6}$$

$$y_{ij} \in \{0, 1\} \qquad \text{for } 1 \le i < j \le n, \tag{7}$$

and (2) and (3).

The objective function (4) of P1 is comparable to (1) except that the quadratic term $(x_i x_j)$ is replaced with y_{ij}. It should be noted that y_{ij} variables are necessary only for pairs with $c_{ij} + c_{ji} > 0$. In other words, if the total confusion between i and j is zero, then there is no penalty to be collected in the objective function even when both are included in the subset. This feature allows for considerable variable and constraint reduction when the confusion matrix is sparse. Constraint set (5) requires, for each (i, j) pair, that $y_{ij} = 1$ if $x_i = x_j = 1$. In other words, if stimulus i and stimulus j are both included in the subset, then the total confusion between them must be included in the minimization of (4) via $y_{ij} = 1$. Constraint set (6) ensures that $y_{ij} = 0$ unless $x_i = x_j = 1$. We observe that, for the case in which $c_{ij} + c_{ji} > 0$ for all i and j, that P1 consists of $n + n(n-1)/2$ integer decision variables and $n(n-1) + 1$ constraints, not including the binary restrictions imposed in (3) and (7).

There are several possible enhancements to the formulation proposed by Theise. First, (6) is not necessary because the nature of the objective function is such that it would always be detrimental for any y_{ij} to assume a value of one. Theise (1989) did not report any computational benefits (e.g., better lower bounds, stronger branching, etc.) from including (6), and our limited experience revealed that the significant increase in the number of constraints actually hurt solution efficiency. Additionally, constraint set (5), in conjunction with the nonnegativity of $c_{ij} + c_{ji}$, obviates the need to impose the integer restrictions on the y_{ij} variables in (7). To minimize total confusion, y_{ij} will equal zero if $x_i + x_j \le 1$, but must equal exactly 1 if $x_i + x_j = 2$. Hereafter, we refer to the model corresponding to P1 with the two enhancements noted above as P1-E. Although these enhancements result in far fewer integer variables (only n instead of $n + n(n-1)/2$), a large number of continuous variables and constraints remain in the model. Fortunately, using ideas developed by Glover (1975), significant reductions in problem size can be realized.

An Improved Integer Linear Programming Formulation (P2)

The improved linear programming formulation (hereafter P2) for the subset extraction problem is based on the work of Glover (1975) and Kuo et al. (1993). We define the following quantities:

$$w_i = \begin{cases} \sum_{j=i+1}^{n} (c_{ij} + c_{ji}) x_j | x_i = 1 \\ 0 | x_i = 0 \end{cases} \qquad \text{for } i = 1, \ldots n-1. \tag{8}$$

If stimulus i is included in the selected subset of stimuli, then w_i represents the total confusion between stimulus i and all other *selected* stimuli with indices greater than i. Otherwise, w_i assumes a value of zero. The sum of the w_i values represents the total confusion among all stimuli in the subset, excluding the undecided responses. To collect values of w_i accurately in a linear programming formulation, we note that an obvious upper bound on w_i is the sum of the $\min(m, n-i)$ largest values of $(c_{ij} + c_{ji} | j > i)$, for $1 \le i \le n-1$. Denoting these upper bounds as $\alpha_i (1 \le i \le n-1)$, a compact formulation (P2) is as follows:

$$\text{Minimize } Z_2 = \sum_{i=1}^{n-1} w_i + \sum_{i=1}^{n} u_i x_i. \tag{9}$$

$$\text{Subject to} \quad \sum_{j=i+1}^{n} (c_{ij} + c_{ji}) x_j - \alpha_i (1 - x_i) - w_i \le 0 \qquad \text{for } 1 \le i \le n-1, \tag{10}$$

$$w_i \ge 0 \qquad \text{for } 1 \le i \le n-1, \tag{11}$$

and (2) and (3).

The objective function (9) of P2 represents total confusion within the subset. Constraint set (10) assures that the continuous w_i variables can equal their lower bounds of zero if $x_i = 0$, but must

TABLE 1.
A problem-size comparison for alternative single-objective formulations of the subset extraction problem

	Model P1	Model P1-E	Model P2
Binary variables	$n + (n(n-1)/2)$	n	n
Continuous variables	—	$n(n-1)/2$	$n-1$
Constraints	$n(n-1)+1$	$(n(n-1)/2)+1$	n

equal the sum of confusions between i and j ($j > i$ and $x_j = 1$) when $x_i = 1$. Constraint set (11) places nonnegativity restrictions on the w_i variables. The reduction in problem size when moving from P1 (or P1-E) to P2 is tremendous. Model P2 requires only n integer variables, $n - 1$ continuous variables, and n constraints. For a confusion matrix associated with the 26 letters of the alphabet (assuming all $c_{ij} + c_{ji} > 0$), model P1 would require 351 integer variables and 651 constraints, whereas model P2 only requires 26 integer variables, 25 continuous variables, and 26 constraints. We note that the LP formulation of P2 is not as restrictive as P1 or P1-E. In other words, the difference between the LP relaxation objective value and the integer optimal objective value is typically larger for P2 than for P1. However, the ease of solving LPs during the branch-and-bound process generally enables much faster identification and confirmation of the integer optimal solution. Table 1 presents a problem-size summary for P1, P1-E, and P2.

A Demonstration of P2 Using an Empirical Interletter Confusion Matrix

All computational results reported in this paper were obtained using a 400 MHz Pentium II PC with 128 MB of random access memory. Matrix generation programs were written in Fortran and formulations were solved using the mixed-integer-linear-programming solver associated with ILOG CPLEX, Version 6.5 (ILOG, 1999).

Our goal was to select a confusion matrix of moderate size and complexity that would be representative of confusion matrices observed in practice. Theise (1989) applied model P1 to a confusion matrix ($n = 25$) originally reported by Moore (1974), which corresponded to pushbuttons for sorting equipment used by the British postal service. The total confusion matrix (Theise 1989, p. 296) is rather sparse, consisting of only 92 instances in which $c_{ij} + c_{ji} > 0$ out of a total of 300 such off-diagonal sums. As a result, the P1 formulation required only 117 binary variables (92 y_{ij} variables and 25 x_i variables) and Theise reported that the solution for $m = 12$ required approximately 10 minutes using Hyper/Lindo PC on a 386 PC. We formulated and solved this same problem using P2. With the speed of contemporary computer platforms and our new formulation, the total time to read and solve this formulation was less than one-half of one CPU second. Therefore, we concluded that it was desirable to identify a more challenging confusion matrix.

We turned our attention to a confusion matrix associated with judgments regarding the similarity of the $n = 26$ (capital) letters of the alphabet (van der Heijden, Malhas, & van den Roovaart, 1984). The confusion matrix published by van der Heijden et al. (1984, p. 86) represents confusion as proportions, whereas Heiser (1988, p. 41) published the same data using the raw confusion frequencies. We used the raw frequency matrix in our demonstrations, which is presented in Table 2. This matrix does not contain undecided responses and therefore the second term of (9) is not considered in our analysis. We believe that this matrix was a good candidate for analysis for two reasons. First, empirical interletter confusion matrices are widespread in the psychological literature (Dawson & Harshman, 1986; Manning & Shofner, 1991; Townsend, 1971; Vega-Bemudez et al., 1991). Second, Heiser provided a description of a subset extraction problem associated with the confusion matrix.

We formulated and solved P1, P1-E, and P2 for values of m ranging from 2 to 25. The optimal objective values, CPU times, number of nodes, and number of iterations for each problem are reported in Table 3.

TABLE 2.
An empirical interletter confusion matrix obtained by van der Heijden et al. (1984) and replicated from Heiser (1988, p. 41)

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	760	7	2	2	5	10	7	24	19	48	62	16	6	32	7	9	5	78	13	6	1	5	10	29	8	29
B	22	471	9	64	16	2	180	17	1	12	8	2	2	12	44	14	77	110	105	2	11	2	9	0	3	5
C	2	2	854	7	59	17	121	1	7	3	4	22	0	4	20	9	16	13	9	11	4	4	0	1	3	7
D	5	42	7	816	5	4	12	4	3	51	3	2	2	10	50	72	27	21	17	4	19	2	5	1	7	9
E	5	50	26	7	420	220	41	25	45	10	24	35	3	19	8	66	10	80	28	38	8	5	2	4	12	9
F	13	12	13	2	68	440	15	38	63	22	25	21	10	20	4	222	3	83	9	60	10	5	9	2	15	16
G	8	35	60	21	21	2	719	11	2	5	1	5	1	11	61	11	124	41	28	8	15	4	3	0	0	3
H	18	21	3	12	8	13	10	497	42	38	18	16	121	130	8	14	7	80	5	28	21	7	62	2	15	4
I	2	4	13	2	18	30	3	10	729	98	5	82	3	4	3	8	1	5	8	111	7	5	1	5	19	24
J	10	5	4	6	2	8	5	31	93	776	14	29	6	16	5	8	2	8	7	25	55	16	18	7	22	22
K	3	2	11	3	10	27	3	12	17	18	462	13	13	17	0	18	1	57	6	27	7	26	26	208	137	76
L	10	6	16	5	23	5	1	26	177	62	17	644	6	13	6	10	6	10	11	54	26	12	17	5	13	19
M	6	13	6	4	7	11	5	181	7	14	66	4	525	86	7	21	6	62	7	29	17	8	47	33	25	3
N	23	14	1	7	1	2	5	26	6	10	130	16	33	593	1	5	4	130	15	7	6	24	31	72	34	4
O	1	19	118	110	3	5	184	1	5	11	1	4	0	5	460	24	170	15	28	4	17	2	3	1	5	4
P	22	27	20	39	21	101	18	26	2	10	6	6	7	4	18	681	8	106	25	7	15	5	6	4	7	9
Q	6	17	127	99	1	4	220	4	1	13	2	9	1	3	302	5	299	13	34	0	19	4	5	1	4	7
R	26	26	18	23	18	43	55	48	0	7	16	3	6	28	28	157	50	533	53	13	10	3	9	8	15	4
S	8	81	51	12	89	24	245	7	16	12	18	10	2	16	19	20	33	121	361	21	7	5	2	4	5	11
T	5	2	4	3	5	83	2	16	202	43	9	37	5	14	2	30	1	13	3	624	8	15	2	7	39	26
U	5	4	14	15	3	2	44	13	5	34	6	25	14	30	25	8	17	6	3	7	754	59	98	2	7	0
V	1	2	3	6	0	16	5	5	30	25	21	9	3	30	7	15	3	18	4	16	57	634	51	20	218	1
W	49	17	1	6	4	8	7	82	5	16	189	11	107	99	9	24	8	153	10	3	15	6	224	120	15	12
X	2	1	0	1	2	6	2	2	11	19	103	8	6	27	1	4	0	26	5	16	3	44	13	604	253	41
Y	2	1	0	2	0	27	0	6	24	28	14	5	4	19	1	26	1	8	2	37	5	32	8	71	788	89
Z	11	1	2	3	9	7	3	4	13	34	18	15	1	5	3	4	1	8	12	17	3	7	4	33	70	912

TABLE 3.
Branch-and-bound integer programming results for the interletter confusion data using models P1, P1-E, and P2

Subset Size	Objective Value, Z_2^*	CPU Time			B & B Nodes			B & B Iterations		
		P1	P1-E	P2	P1	P1-E	P2	P1	P1-E	P2
$m = 2$	0	.65	.57	.08	23	22	6	83	45	16
$m = 3$	9	.91	.73	.21	29	32	131	211	216	206
$m = 4$	41	3.42	1.79	.65	305	227	478	1669	1444	860
$m = 5$	81	4.16	2.21	.78	358	294	591	2334	2190	1302
$m = 6$	154	6.73	2.65	1.99	833	339	1652	6005	3403	3350
$m = 7$	278	9.24	5.44	2.67	1323	939	2135	11089	9449	5650
$m = 8$	424	10.96	5.66	5.28	1450	745	5181	15860	10520	11166
$m = 9$	633	13.19	7.71	3.38	1566	1073	2781	21632	16873	7861
$m = 10$	866	14.77	8.17	3.31	1692	990	2897	25011	17383	7184
$m = 11$	1157	15.99	8.15	3.44	1529	814	2940	29995	17840	6973
$m = 12$	1567	14.54	8.67	4.45	947	712	3646	24016	17872	9531
$m = 13$	2008	12.44	8.41	4.48	549	565	1937	11810	11836	12121
$m = 14$	2488	9.36	5.16	2.19	392	395	1434	9012	8913	3975
$m = 15$	2994	7.37	3.09	1.49	193	167	751	4701	4097	2229
$m = 16$	3595	6.53	2.62	.99	89	96	386	2340	2605	1108
$m = 17$	4275	5.16	1.54	.39	32	32	172	1134	1060	508
$m = 18$	5103	4.27	1.21	.40	11	18	150	575	681	451
$m = 19$	6114	4.80	1.08	.36	7	9	153	532	536	369
$m = 20$	7181	3.68	.75	.20	11	8	76	544	510	189
$m = 21$	8280	3.65	.75	.14	10	9	47	451	434	134
$m = 22$	9427	2.78	.60	.08	2	2	10	405	405	72
$m = 23$	10669	.28	.28	.04	0	0	2	366	366	53
$m = 24$	12110	.28	.25	.04	0	0	0	342	342	51
$m = 25$	13688	.25	.23	.04	0	0	0	345	345	30

The results in Table 3 indicate that total confusion is a non-decreasing function of m. In some instances, the optimal subset for $m + 1$ is equal to the optimal subset for m, plus one additional stimulus. However, in other instances, there is a considerable difference between the optimal subset for m and the optimal subset for $m + 1$. For example, the optimal subset for $m = 13$ is {A, B, C, D, F, L, M, S, T, U, V, X, Z}. In the optimal subset for $m = 14$, stimuli {F, S} are replaced with {E, P, Q}.

The results in Table 3 also suggest that, relative to P1, optimal subsets were extracted more efficiently when using P1-E. In many instances, solutions to P1-E were obtained in less than one-half of the time required to solve P1. In all cases, solutions to P2 were obtained in less time than either P1 or P1-E and, in most cases, solution time for P2 was less than one-half the time of its nearest competitor (P1-E). The CPU time required to solve the P2 formulations never exceeded 5.28 seconds. Problems for $6 \leq m \leq 15$ required at least one CPU second, whereas optimal solutions for all other problems were obtained in less than one second.

A Biobjective Integer Linear Programming Formulation for Subset Extraction

The Biobjective Integer Linear Programming Formulation (P3)

Theise (1989) proposed a biobjective variation of the subset extraction problem. Specifically, he offered an integer goal-programming formulation that focused on the maximization of total recognition (correct responses for stimuli) subject to a constraint that ensured that total confusion was below a maximum allowable value, h. Although Theise's (1989) model is interesting from a theoretical standpoint, it does not provide a convenient mechanism for exploring the tradeoffs between total recognition and total confusion for problems of practical size. As a plausible alternative for studying these tradeoffs, we propose using a weighted biobjective integer-programming model. Similar optimization models have recently proven successful in analyzing multiobjective seriation problems for asymmetric proximity matrices (Brusco & Stahl, 2001). The biobjective integer programming formulation (hereafter P3) of interest here is as follows:

$$\text{Minimize } Z_3 = g_1\left[\sum_{i=1}^{n-1} w_i + \sum_{i=1}^{n} u_i x_i\right] - g_2\left[\sum_{i=1}^{n} c_{ii} x_i\right]. \tag{12}$$

$$\text{Subject to } \quad \sum_{i=1}^{n-1} w_i + \sum_{i=1}^{n} u_i x_i \leq h, \tag{13}$$

$$\text{and } (2), (3), (10), \text{ and } (11).$$

Problem P3 is a direct extension of P2 and focuses on the minimization of a weighted combination of total confusion and total recognition (12). The second component of (12), weighted total recognition, is subtracted from the objective function because we are attempting to minimize total confusion, yet maximize total recognition. The parameters g_1 and g_2 should be used within the context of a convex combination where both parameters are nonnegative. Constraint (13) can be included if it is desirable to place an upper bound on total confusion, as in Theise's biobjective model.

Demonstration of P3 Using an Empirical Interletter Confusion Matrix

As a demonstration for problem P3, we consider the interletter confusion matrix under the assumption of a desired subset size of $m = 13$ and no prespecified aspiration level for total confusion (i.e., $h = \infty$). We used the CPLEX branch-and-bound code to solve P3 for each of 11 weighting schemes (g_1, g_2) beginning with (1, 0), and subsequently decreasing g_1 by .1 and increasing g_2 by .1 until arriving at the weighting scheme (0, 1). The results of the study are reported in Table 4.

Table 4 indicates that modest sacrifices in confusion (recognition) can lead to significant improvements in recognition (confusion). For example, a 3.6% improvement in recognition can be realized for only a .3% increase in confusion by moving from a weighting scheme of (1, 0) to (.9, .1). By moving from a weighting scheme of (0, 1) to (.3, .7), a 20.5% reduction in confusion

TABLE 4.
Branch-and-bound integer programming results for interletter confusion data using model P3. All biobjective results correspond to a subset size of $m = 13$ with no aspiration level

Confusion weight, g_1	Recognition weight, g_2	Objective Value, Z_3	Total Confusion	Total Recognition	B & B Nodes	B & B Iterations	CPU Time
1.0	0.0	2008	2008	8399	1819	12917	4.88
.9	.1	943.6	2015	8699	1292	8646	3.32
.8	.2	−127.8	2015	8699	1138	5633	2.80
.7	.3	−1210.1	2152	9055	787	4168	2.14
.6	.4	−2364.6	2261	9303	420	1630	1.14
.5	.5	−3521	2261	9303	314	917	.70
.4	.6	−4677.4	2261	9303	210	647	.44
.3	.7	−5880.5	2684	9551	57	146	.17
.2	.8	−7132.4	2978	9660	12	48	.08
.1	.9	−8396.2	2978	9660	1	19	.06
0.0	1.0	−9691.0	3376	9691	0	13	.02

can be achieved at the expense of only a 1.4% reduction in recognition. Regions of the biobjective solution space can be explored in greater detail by refinement of the weights.

Table 4 also shows that the total CPU time required to solve the P3 formulations ranged from .02 to 4.88 seconds. The solution to the problem associated with a weighting scheme of (0, 1) is trivial because it is only necessary to select the m stimuli with the largest diagonal elements in the subset. Solution times gradually increased as more weight was placed on total confusion, reaching a maximum at the weighting scheme of (1, 0).

Demonstration of P3 with Total Confusion Aspiration Levels

Our next demonstration concerns the biobjective subset extraction problem ($m = 13$) using different aspiration levels for total confusion. For example, suppose that upon further examination of Table 4, the quantitative analyst was interested in finding a subset that provided the largest total recognition subject to a constraint that total confusion does not exceed more than roughly 110% of its minimum value of 2008 (e.g., $h = 2200$). One way to answer this question is to solve model P3 using the weighting scheme (0, 1) and including (13) with $h = 2200$. Table 5 presents results for different values of h. These results were obtained using a weighting scheme (.0001, .9999) to provide better values of total confusion in case of multiple optima. In other words, for $h = 2200$, there might be two subsets that provide the same total recognition, yet different levels of total confusion. The weighting scheme (0, 1) would be indifferent to these two solutions, but the weighting scheme with a very small positive value of g_1 would yield the subset with the smaller total confusion as the unique optimal solution.

The results in Table 5 reveal that the solution corresponding to weights of (.7, .3) in Table 4, does not provide the largest total recognition, given an upper bound on confusion of $h = 2200$. It is possible to increase recognition up to 9120 while remaining within the bound for confusion at 2189. This demonstrates the utility of analyst-specified aspiration levels that can be used to optimize one criterion subject to the aspiration constraint on the other criterion.

One of the most striking aspects of Table 5 is that the biobjective formulations become increasingly difficult to solve as the value of h is decreased. For $3000 \geq h \geq 2300$, the total CPU time was less than one second. However, the CPU time was at least 3.94 seconds for $2050 \geq h \geq 2010$, with a maximum of 6.71 seconds for $h = 2010$.

TABLE 5.
Branch-and-bound integer programming results for interletter confusion data using model P3. All results correspond to a subset size of $m = 13$, a total confusion weight of $g_1 = .0001$ and a total recognition weight of $g_2 = .9999$

h	Z_3^*	Total Confusion	Total Recognition	B & B Nodes	B & B Iterations	CPU Time
2010	−8397.96	2008	8399	6062	24150	6.71
2020	−8697.93	2015	8699	4085	17546	4.95
2050	−8697.93	2015	8699	3286	13237	3.94
2100	−8870.90	2083	8872	2495	8782	2.93
2200	−9118.87	2189	9120	947	3256	1.51
2300	−9301.84	2261	9303	199	647	.46
2400	−9355.83	2380	9357	211	537	.41
2500	−9444.81	2489	9446	89	224	.19
2600	−9444.81	2489	9446	265	614	.54
2700	−9549.78	2684	9551	31	87	.17
2800	−9553.77	2755	9555	23	64	.13
2900	−9590.76	2832	9592	11	39	.08
3000	−9658.74	2978	9660	1	19	.05

Extraction of Multiple Subsets from a Confusion Matrix

An Integer Linear Programming Model for Extracting Multiple Subsets (P4)

In some psychological experiments, the interrelationships among the extracted stimuli are of interest. Heiser (1988) described a subset extraction problem for the empirical letter confusion matrix obtained by van der Heijden et al. (1984). The problem was to select four pairs of letters from the confusion matrix such that similarity within the pairs was large and similarity between the pairs was small. This problem is more complicated than simply extracting a subset of $m = 8$ from the $n = 26$ stimuli because of the interest in the relationship between pairs of stimuli within the subset. For the purposes of our mathematical model, similarity between stimuli is reflected in the total confusion between those two stimuli. The problem is viewed as selecting four subsets, each of size $m = 2$, from the confusion matrix. A generalization of P1 for this problem requires an extremely large number of variables and constraints, and is not a viable alternative for problems of practical size. A generalized variation of P3 (hereafter P4) for extracting T subsets of sizes $m_t \geq 2(1 \leq t \leq T)$ from a confusion matrix requires the following variable definitions:

$x_{it} =$ 1 if stimulus i is selected for inclusion in subset t, 0 otherwise, for $1 \leq i \leq n$ and $1 \leq t \leq T$;
$v_{it} =$ if stimulus i is selected for subset t, then v_{it} is the total confusion between stimulus i and all other selected stimuli $j\,(j > i)$ in the same subset, otherwise $v_{it} = 0$, for $1 \leq i \leq n-1$ and $1 \leq t \leq T$;
$w_{it} =$ if stimulus i is selected for subset t, then w_{it} is the total confusion between stimulus i and all other selected stimuli $j\,(j > i)$ in the subsets other than t, otherwise $w_{it} = 0$, for $1 \leq i \leq n-1$ and $1 \leq t \leq T$.

We also define the following parameters:

$\beta_{\text{it}} =$ an upper bound on v_{it}, which is computed as the sum of the $\min(m_t - 1, n - i)$ largest values of $(c_{ij} + c_{ji} | j > i)$, for $1 \leq i \leq n-1$ and $1 \leq t \leq T$;
$\gamma_{\text{it}} =$ an upper bound on w_{it}, which is computed as the sum of the $\min((\sum_{p=1}^{T} m_p) - m_t, n - i)$ largest values of $(c_{ij} + c_{ji} | j > i)$, for $1 \leq i \leq n-1$ and $1 \leq t \leq T$;

With these variables and parameters, model P4 is formulated as follows:

$$\text{Maximize: } Z_4 = g_1 \sum_{i=1}^{n-1} \sum_{t=1}^{T} v_{it} - g_2 \sum_{i=1}^{n-1} \sum_{t=1}^{T} w_{it}. \tag{14}$$

Subject to

$$v_{it} - \beta_{it} x_{it} \leq 0 \qquad \text{for } 1 \leq i \leq n-1, 1 \leq t \leq T, \tag{15}$$

$$v_{it} - \sum_{j=i+1}^{n} (c_{ij} + c_{ji}) x_{jt} \leq 0 \qquad \text{for } 1 \leq i \leq n-1, 1 \leq t \leq T, \tag{16}$$

$$-w_{it} + \gamma_{it} x_{it} + \sum_{j=i+1}^{n-1} \sum_{r \neq t} (c_{ij} + c_{ji}) x_{jr} \leq \gamma_{it} \qquad \text{for } 1 \leq i \leq n-1, 1 \leq t \leq T, \tag{17}$$

$$\sum_{i=1}^{n} x_{it} = m_t \qquad \text{for } 1 \leq t \leq T, \tag{18}$$

$$\sum_{t=1}^{T} x_{it} \leq 1 \qquad \text{for } 1 \leq i \leq n, \tag{19}$$

$$x_{it} \in \{0, 1\} \qquad \text{for } 1 \le i \le n, 1 \le t \le T, \tag{20}$$

$$v_{it}, w_{it} \ge 0 \qquad \text{for } 1 \le i \le n-1, 1 \le t \le T. \tag{21}$$

The objective function (14) of P4 consists of two components. The first component represents total confusion within subsets. The second component represents total confusion between subsets. This second component is subtracted from the objective function because the goal is to maximize similarity within subsets, yet minimize similarity between subsets. Although a biobjective exploration of convex combinations of g_1 and g_2 could be conducted, we limit our analysis to equal weights of $g_1 = g_2 = 1$.

Constraints (15) and (16) ensure that v_{it} will assume a value of zero if stimulus i is not selected for inclusion in subset t, but can assume a value equal to the sum of stimulus i's confusion with other selected stimuli $j\,(j > 1)$ in subset t if that stimulus is included in the subset. Constraints (17) ensure that similarity between subsets will be appropriately collected. If stimulus i is not selected for inclusion in subset $t\,(x_{it} = 0)$, then w_{it} can assume a value of zero, which is desirable from the standpoint of the objective function. However, if $x_{it} = 1$, then w_{it} must be at least as large as the sum of stimulus i's confusion with all other selected stimuli $j\,(j > i)$ in other subsets $r\,(r \ne t)$. Although it is possible to replace the β_{it} and γ_{it} terms in (15) and (17) with some arbitrary large value, M, we have observed that the tighter bounds greatly improve solution efficiency and are important to the formulation. Constraints (18) require that exactly m_t stimuli are selected for each of the T subsets. Constraints (19) guarantee that a stimulus is not selected for more than one subset. Constraints (20) place binary restrictions on the x_{it} variables, and (21) impose nonnegativity restrictions on the v_{it} and w_{it} variables.

Model P4 consists of nT integer decision variables, $2T(n-1)$ continuous variables, and $3T(n-1)+n+T$ constraints. This formulation is larger and more complex than P2 and P3 because of the modeling of multiple interrelated subsets. However, because P4 is constructed using the same type of compact representation of variables and constraints, it is a plausible approach for the modest sized confusion matrices often found in practice.

Demonstrations and Analysis of P4 Using an Empirical Interletter Confusion Matrix

We first applied P4 to the problem addressed by Heiser (1988), which was to extract $T = 4$ pairs of size $m_t = 2(1 \le t \le T)$ from the $n = 26$ letters in the confusion matrix. The optimal solution to this formulation required more than one million integer iterations, approximately 6 minutes of CPU time, and a maximum storage of 5MB of RAM. The extracted letter pairs were O-Q, H-M, E-F, and X-Y. Three of these pairs are the same as those identified by Heiser. His fourth pair was K-X instead of X-Y. It is possible that alternative weighting schemes, or transformations of the similarity matrix might yield different results. To test this theory, we noted that Heiser (1988, p. 42) applied a probability model based on the work of Shepard (1957) and Luce (1963), in order to obtain a symmetric similarity matrix, **D**, based on van der Heijden et al. (1984) interletter confusion matrix. We applied P4 to this similarity matrix by replacing the $(c_{ij} + c_{ji})$ terms in the formulation with d_{ij}. The extracted subsets of letter pairs were the same as those obtained when we applied the model directly to the confusion matrix. It was interesting to note, however, that the CPU time required to solve P4 based on **D** was only about 2 minutes.

In addition to the similarity of the pairs selected based on the solution of P4 to the pairs obtained by Heiser (1988), it is encouraging to note that pairs have a logical interpretation. The similarity of letters within the pairs is rather obvious. For example, the circular nature of O and Q, or the fact that E and F are characterized by a vertical bar on the left and two or more horizontal bars. Even the X-Y and K-X pairs share a notable feature in that they all have intersecting lines in their midsections. Closer inspection also reveals a lack of similarity between pairs. None of the letters in three of the pairs remotely resemble O or Q. Like the letters E and F, the letters H and M have a vertical bar on the left, but they also have a vertical bar on the right. It generally seems that P4 resulted in a selection of a similar letters within pairs, but a lack of similarity of letters between pairs.

We also applied P4 to the problem of extracting $T = 4$ subsets of size $m_t = 3(1 \leq t \leq T)$. By moving from pairs to triples, the solution space and problem complexity is significantly increased. For the interletter confusion matrix, the optimal solution for P4 required nearly seven million integer iterations, almost 42 minutes of CPU time, and a maximum storage of 15 MB of RAM. The extracted letter triples were E-F-P, G-O-Q, I-J-L, and K-X-Y. When we applied model P4 to **D**, two of the selected triples were different, and solution time was only 23 minutes. The extracted letter triples were E-F-P, G-O-Q, X-Y-Z, and H-M-W.

Limitations and Extensions

Limitations of the Models and Applications to other Confusion Matrices

The primary limitation of the models presented in this paper is that, for certain data sets, branch-and-bound integer programming methods consume significant memory storage and CPU time. The feasibility of integer programming methods for subset extraction is affected by two critical factors: (a) the size of the confusion matrix, which affects the number of variables and constraints in the models, and (b) the structure of the confusion matrix. We have observed that, for dense confusion matrices with little variability among the entries, the models presented in this paper can be very difficult to solve even for $n \approx 30$. However, dense confusion matrices with significant variability among the entries (e.g., the interletter confusion matrix from van der Heijden et al. (1984)) are relatively easy to solve. Under such conditions, we have successfully applied our models to randomly generated confusion matrices as large as $n = 50$.

We have applied the models presented in this paper to various empirical confusion matrices (Manning & Shofner, 1991; Moore, 1974; Rothkopf, 1957; Vega-Bermudez et al., 1991). The largest and most challenging of these was the Morse code confusion data ($n = 36$) reported by Rothkopf (p. 97). We applied models P2 and P3 to this confusion matrix for conditions comparable to those associated with Tables 2, 3, and 4 (detailed results are available from the authors). Although we were always able to obtain optimal solutions for the Morse code data, the CPU times were 10–15 minutes for some of the problems. Model P4 failed in implementation for the Morse code data because of computer memory limitations. We believe that the increased difficulty associated with solving problems for the Morse code data stems from two factors: (a) a larger number of variables and constraints for the Morse code data, and (b) less variability among the confusion entries in the Morse code data. To test this hypothesis partially, we squared the confusion entries in the Morse code data prior to running model P4. As a result, we were able to successfully extract four pairs (6 CPU minutes) and four triples (2 CPU hours) from the "squared" Morse code confusion data.

Despite the squaring of the confusion measures, the subsets extracted for the Morse code data were easily interpreted. The four pairs selected based on the solution to P4 were ($E = \bullet$, $T = -$), ($G = --\bullet$, $O = ---$), ($B = -\bullet\bullet\bullet$, $6 = -\bullet\bullet\bullet\bullet$), and ($9 = ----\bullet$, $0 = -----$). Again, the similarity within pairs is evident. The pairs E-T, G-O, and 9-0 all have the same number of elements in the symbol and differ only by one element. For the remaining pair, the symbol for 6 appends a dot to the end of the symbol for B. The lack of similarity between pairs is also clear. The E-T, G-O, and 9-0 pairs are well separated based on the fact that they have 1, 3, and 5 elements in their symbols respectively. The B-6 pair and 9-0 pair are differentiated based on the fact that the symbols consist mostly of dots, whereas the symbols for 9 and 0 consist mostly of dashes. Comparable findings regarding similarity within and between triples of letters were observed. The four triples were ($A = \bullet-$, $I = \bullet\bullet$, $N = -\bullet$), ($G = --\bullet$, $O = ---$, $W = \bullet--$), ($B = -\bullet\bullet\bullet$, $X = -\bullet\bullet$, $6 = -\bullet\bullet\bullet\bullet$), and ($1 = \bullet----$, $9 = ----\bullet$, $0 = -----$).

Modeling Enhancements and Alternative Solution Approaches

As noted earlier in the paper, there are alternative optimal solution approaches for subset extraction problems, including quadratic enumerative algorithms and dynamic programming. Hubert, Arabie and Meulman (2001) have developed an extensive variety of dynamic programming algorithms for related problems. Their dynamic programming paradigm could be modified easily to develop a recursive scheme for subset selection, which could then be compared to the integer-programming approach. Our emphasis on branch-and-bound integer programming is based on the accessibility of software and the flexibility of incorporating different criteria and constraints. In this paper, we have explicitly modeled such criteria as minimizing total confusion, maximizing total recognition, maximizing confusion within subsets, and minimizing confusion between subsets. Criteria such as minimizing the maximum confusion in a subset, or maximizing the minimum confusion in a subset, can also be modeled using integer programming. The formulations in this paper can also be augmented to incorporate problem-specific constraints. Constraints can be imposed to require that, if a particular stimulus is included in a subset, then one or more other stimuli are included in the same subset. Alternatively, constraints can be added to guarantee that certain subsets of stimuli are not included in the same subset. In short, there are many possible variations of objective criteria and constraints that can be accommodated by integer programming formulations.

It is also important to recognize that there are several strategies that might improve the performance of branch-and-bound integer programming methods for the formulations proposed in this paper, including specialized branching strategies and cutting planes. Another approach is to use heuristic methods to provide an initial feasible integer solution, as well as a bound on the objective function. We have developed a simulated annealing heuristic for subset extraction problems that provides excellent starting solutions and can improve computational efficiency for larger problems. The heuristic is also an efficient and effective solution procedure in its own right, and can provide good solutions for much larger confusion matrices ($n > 100$).

Summary

This paper has presented single-objective and biobjective linear 0-1 formulations for extracting a subset of stimuli from a confusion matrix. These formulations are far more compact than previously proposed formulations for the same problem. We successfully solved both single-objective and biobjective problems for an empirical interletter confusion matrix using branch-and-bound integer programming. All the ILP formulations solved in a reasonable amount of microcomputer CPU time (.02 to 6.71 seconds) and did not consume a significant amount of memory for storage of the branch-and-bound tree. We also extended the basic modeling approach to the problem of extracting T subsets of sizes m_t ($1 \leq t \leq T$) such that within subset similarity is large and between subset similarity is small. The model was successfully applied to the extraction of letter pairs and triples from the interletter confusion matrix. Although these problems were considerably larger and more difficult to solve, the resulting subsets of pairs and triples were easy to interpret, and the pairs were similar to those obtained by Heiser (1988). Limitations, other applications, possible enhancements, and alternative approaches to the proposed integer programming models were also discussed.

References

Appelman, I.B., & Mayzner, M.S. (1982). Application of geometric models to letter recognition: Distance and density. *Journal of Experimental Psychology: General, 111*, 60–100.

Baker, F.B., & Hubert, L.J. (1977). Applications of combinatorial programming to data analysis: Seriation using asymmetric proximity measures. *British Journal of Mathematical and Statistical Psychology, 30*, 154–164.

Brusco, M.J., & Stahl, S. (2001). An interactive approach to multiobjective combinatorial data analysis. *Psychometrika, 66*, 5–24.

Clarke, F.R. (1957). Constant-ratio rule for confusion matrices in speech communication. *Journal of the Acoustical Society of America*, *29*, 715–720.

Comley, W.J. (1996). QUAD01: A data-structured implementation of Hansen's quadratic zero-one programming algorithm. *European Journal of Operational Research*, *95*, 420–426.

Dawson, M.R., & Harshman, R.A. (1986). The multidimensional analysis of asymmetries in alphabetic confusion matrices: Evidence for global-to-local and local-to-global processing. *Perception & Psychophysics*, *40*, 370–383.

Garey, M.R., & Johnson, D.S. (1979). *Computers and intractability: A guide to the theory of NP-completeness*. New York, NY: W.H. Freeman.

Glover, F. (1975). Improved linear integer programming formulations of nonlinear integer problems. *Management Science*, *22*, 455–460.

Glover, F., & Woolsey, R.E. (1974). Converting the 0–1 polynomial programming problem to a 0–1 linear program. *Operations Research*, *22*, 180–182.

Green, P., & Pew, R.W. (1978). Evaluating pictographic symbols: An automotive application. *Human Factors*, *20*, 103–114.

Hansen, P. (1972). Quadratic 0–1 programming by implicit enumeration. In F.A. Lootsma (Ed.), *Numerical methods in nonlinear optimization* (pp. 265–278). San Diego, CA: Academic Press.

Heiser, W.J. (1988). Selecting a stimulus set with prescribed structure from empirical confusion frequencies. *British Journal of Mathematical and Statistical Psychology*, *41*, 37–51.

Hodge, M.H., & Pollack, I. (1962). Confusion matrix analysis of single and multidimensional auditory displays. *Journal of Experimental Psychology*, *63*, 129–142.

Hettinger, T.P., Gent, J.F., Marks, L.E., & Frank, M.E. (1999). A confusion matrix for the study of taste perception. *Perception & Psychophysics*, *61*, 15–21.

Hubert, L.J. (1974). Some applications of graph theory and related nonmetric techniques to problems of approximate seriation: The case of symmetric proximity measures. *British Journal of Mathematical and Statistical Psychology*, *27*, 133–153.

Hubert, L.J. (1976). Seriation using asymmetric proximity measures. *British Journal of Mathematical and Statistical Psychology*, *29*, 32–52.

Hubert, L.J. (1987). *Assignment methods in combinatorial data analysis*. New York, NY: Marcel Dekker.

Hubert, L.J., Arabie, P., & Meulman, J. (1997). Linear and circular unidimensional scaling for symmetric proximity matrices. *British Journal of Mathematical and Statistical Psychology*, *50*, 253–284.

Hubert, L., Arabie, P., & Meulman, J. (2001). *Combinatorial data analysis: Optimization by dynamic programming*. Philadelphia, PA: Society for Industrial and Applied Mathematics.

Hubert, L.J., & Schultz, J. (1976). Quadratic assignment as a general data analysis strategy. *British Journal of Mathematical and Statistical Psychology*, *29*, 190–241.

ILOG. (1999). *ILOG CPLEX 6.5 User's manual*. Mountain View, CA: Author.

Kent, P. F., Youngentob, S.L., & Sheehe, P.R. (1995). Odorant-specific spatial patterns in mucosal activity predict perceptual differences among odorants. *Journal of Neurophysiology*, *74*, 1777–1781.

Kruskal, J.B. (1964). Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, *29*, 1–27.

Kuo, C., Glover, F., & Dhir, K.S. (1993). Analyzing and modeling the maximum diversity problem by zero-one programming. *Decision Sciences*, *24*, 1171–1185.

Loomis, J.M. (1990). A model of character recognition and legibility. *Journal of Experimental Psychology: Human Perception and Performance*, *16*, 106–120.

Luce, R.D. (1963). Detection and recognition. In R.D. Luce, R. Bush, & E. Galanter (Eds.), *Handbook of mathematical psychology, Vol. 1* (pp. 103–189). New York, NY: Wiley.

Manning, S.K., & Shofner, E. (1991). Similarity ratings and confusability of lipread consonants compared with similarity ratings of auditory and orthographic stimuli. *American Journal of Psychology*, *104*, 587–604.

Massaro, D.W., Cohen, M.M., & Gesi, A.T. (1993). Long-term training, transfer, and retention in learning to lipread. *Perception & Psychophysics*, *53*, 549–562.

Miralles, J.L., & Cervera, T. (1995). Voice intelligibility in patients who have undergone laryngectomies. *Journal of Speech and Hearing Research*, *38*, 564–571.

Moore, T.G. (1974). Tactile and kinaesthetic aspects of push-buttons. *Applied Ergonomics*, *5*, 66–71.

Rich, R. (1971). Constant ratio rule for confusion matrices from short-term memory experiments. *British Journal of Psychology*, *62*, 27–35.

Rothkopf, E.Z. (1957). A measure of stimulus similarity and errors in some paired-associate learning tasks. *Journal of Experimental Psychology*, *53*, 94–101.

Shepard, R.N. (1957). Stimulus and response generalization: A stochastic model relating generalization to distance in psychological space. *Psychometrika*, *22*, 325–345.

Smith, P.T., & Jones, K.F. (1975). Some hierarchical scaling methods for confusion matrix analysis: II. Applications to large matrices. *British Journal of Mathematical and Statistical Psychology*, *28*, 30–45.

Smith, P.T., Wilson, A.W., & Jones, K.F. (1975). Some hierarchical scaling methods for confusion matrix analysis: I. Theory and applications for small matrices. *British Journal of Mathematical and Statistical Psychology*, *28*, 1–29.

Taha, H.A. (1972). A Balasian-based algorithm for zero-one polynomial programming. *Management Science, Series B*, *18*, 328–343.

Theise, E.S. (1989). Finding a subset of stimulus-response pairs with minimum total confusion: A binary integer programming approach. *Human Factors*, *31*, 291–305.

Townsend, J.T. (1971). Theoretical analysis of an alphabetic confusion matrix. *Perception & Psychophysics*, *9*, 40–50.

Townsend, J.T., & Ashby, F.G. (1982). Experimental test of contemporary mathematical models of visual letter recognition. *Journal of Experimental Psychology: Human Perception and Performance*, *8*, 836–864.

Townsend, J.T., & Landon, D.E. (1982). An experimental and theoretical investigation of the constant-ratio rule and other models of visual letter confusion. *Journal of Mathematical Psychology*, *25*, 119–162.

van der Heijden, A.H.C., Malhas, M.S.M., & van den Roovaart, B.P. (1984). An empirical interletter confusion matrix for continuous-line capitals. *Perception & Psychophysics*, *35*, 85–88.

van Son, R.J.J.H., & Pols, L.C.W. (1999). Perisegmental speech improves consonant and vowel identification. *Speech Communication*, *29*, 1–22.

Vega-Bermudez, F., Johnson, K.O., & Hsiao, S.S. (1991). Human tactile pattern recognition: Active vs. passive touch, velocity effects, and patterns of confusion. *Journal of Neurophysiology*, *65*, 531–546.

Youngentob, S.L., Markert, L.M., Moxell, M.M., & Hornung, D. E. (1990). A method for establishing a five odorant identification confusion matrix task in rats. *Physiology and Behavior*, *47*, 1053–1059.

Zielman, B., & Heiser, W.J. (1996). Models for asymmetric proximities. *British Journal of Mathematical and Statistical Psychology*, *49*, 127–146.

Zwaga, H.J., & Boersema, T. (1983). Evaluation of a set of graphic symbols. *Applied Ergonomics*, *14*, 43–54.

Manuscript received 4 MAY 2000
Final version received 19 SEP 2000