Optimal Allocation of Proposals to Reviewers to Facilitate Effective Ranking

Wade D. Cook
Schulich School of Business, York University, Toronto, Ontario M3J 1P3, Canada, wcook@schulich.yorku.ca

Boaz Golany
Faculty of Industrial Engineering and Management, Technion—Israel Institute of Technology,
Haifa 32000, Israel, golany@ie.technion.ac.il

Moshe Kress
Center for Military Analyses, POB 2250, Haifa 31021, Israel, and Operations Research Department,
Naval Postgraduate School, Monterey, California 93943, mkress@ie.technion.ac.il

Michal Penn, Tal Raviv
Faculty of Industrial Engineering and Management, Technion—Israel Institute of Technology, Haifa 32000, Israel
{mpenn@ie.technion.ac.il, tal.raviv@sauder.ubc.ca}

Peer review of research proposals and articles is an essential element in research and development processes worldwide. Here we consider a problem that, to the best of our knowledge, has not been addressed until now: how to assign subsets of proposals to reviewers in scenarios where the reviewers supply their evaluations through ordinal ranking. The solution approach we propose for this assignment problem maximizes the number of proposal pairs that will be evaluated by one or more reviewers. This new approach should facilitate meaningful aggregation of partial rankings of subsets of proposals by multiple reviewers into a consensus ranking. We offer two ways to implement the approach: an integer-programming set-covering model and a heuristic procedure. The effectiveness and efficiency of the two models are tested through an extensive simulation experiment.

1. Introduction

A large portion of current academic research is sponsored through various agencies and funds with specific interests in different areas of research. The sponsorship process typically starts with a call for proposals (CFP), which is distributed to the relevant community. Proposals are then submitted according to guidelines that appeared in the CFP. These proposals are sent for a peer review that serves as the core of the entire process. The referees who review the proposals are usually provided with some instructions on the norms and criteria that should be applied to gauge the quality of the submitted proposals. In most cases, each referee is asked to review a subset of the submitted proposals. (In extreme cases, each referee reviews a single proposal.) The reviews are collected by the body that issued the CFP, which uses some aggregation scheme to transform the individual evaluations into a single overall ranking.

The manner in which preferences over objects that need to be ranked (proposals, in our case) are expressed depends on the level of possible quantification. In some situations, cardinal or quantitative data on each of various attributes of the objects can be specified. In many practical applications, however, it is not possible to explicitly quantify the objects' values in a full cardinal format, and one must settle for the less-specific ordinal specification. In some situations, one can specify a complete "ranking" of N objects on an ordinal scale in vector format $A = (a_1, a_2, \ldots, a_N)$, where $a_i \in \{1, 2, \ldots, N\}$ is the rank position occupied by object i. When such a ranking A^k is supplied by each member k of a committee of K members, one can define a consensus of opinions in several ways (e.g., the median ranking as discussed in Cook and Seiford 1978). From a practical point of view, if N is large, a full ranking may prove difficult, and in many applications analysts often choose to use a Likert scale (typically five points) as the basis for eliciting preferences (see, e.g., Garg 1996).

One very common format for expressing preferences is to use pairwise comparisons. This mode of expression forces one to make a direct choice of one object over another when comparing two objects, rather than requiring one to compare *all* objects simultaneously. It is particularly attractive when a comparison of all the objects is not possible and only a partial ranking may

be supplied. In the case that an individual voter or committee member can only express preferences concerning a proper subset of the objects, then a partial ranking is the most information that this person can provide. In such a situation, vector representations, as discussed above, make little practical sense, and one must then default to pairwise comparisons.

Most peer reviews to date were based on cardinal rankings. However, some researchers (mainly in the social sciences and decision-analysis areas) have recently started to raise questions on the validity of the outcomes of such processes. In particular, several studies were devoted to analyzing the reliability of and the possible existence of various biases in such peer review processes (e.g., Cicchetti 1991, Hodgson 1995, Campanario 1998, Jayasinghe et al. 2001). They found low degrees of agreement among referees and various kinds of biases. Other studies (e.g., Dirk 1999) focused on the criteria that guide the referees' work and reported on a common language—a certain set of criteria that referees tend to use to evaluate research quality. However, as emphasized by Langfeldt (2001), these criteria are often interpreted or operationalized differently by various reviewers. Techniques that generate ordinal rankings on the basis of pairwise evaluations may provide some remedy to these difficulties because they are more straightforward and require less effort from the reviewers. As an illustration, consider two scenarios: (1) an evaluator who is asked to estimate the length of a single stick and (2) an evaluator who is asked to estimate which of two sticks is longer. Obviously, cardinal rankings are superior to ordinal rankings as they provide more refined information. The trouble is that sometimes the more refined information is practically very difficult to attain (e.g., measure the length of the single stick without any ruler at hand). Additionally, with cardinal rankings we are more exposed to biases that may stem from the "generosity" of the evaluators assigned to particular proposals. Suppose that we have three proposals, A, B, and C and two reviewers, R_1 and R_2. Proposals A and B are assigned to R_1. Assume that R_1, who tends to be less than generous in evaluations, prefers A over B and rates them (in a cardinal 1–10 scale) as 6 and 5, respectively. R_2, who tends to be much more generous in ratings, is assigned proposals B and C. This reviewer prefers B over C and rates them as 9 and 6.5, respectively. Clearly, the ranking A ≻ B ≻ C can be accepted by both reviewers, but by ranking according to the average cardinal rates, we obtain B ≻ C ≻ A. Because the number of evaluators for each proposal is not expected to be large, the likelihood that such phenomena will occur is not insignificant. Hence, in this paper we shall consider peer reviews in which reviewers are asked to provide ordinal, rather than cardinal, rankings of proposals.

This paper focuses on an important operational aspect of the peer review process that has been mostly neglected until now—the method by which proposals are assigned to specific referees. Referees are typically characterized according to their particular areas of expertise. Therefore, to obtain the most professional evaluation, some matching procedure should be implemented that will assign each proposal to the referee(s) that are most qualified to review it. However, the referees are also associated with (usually self-imposed) limits on the number of proposals they are willing to review or capable of reviewing within the specified time window. In general, one cannot expect the distribution of expertise areas within the pool of available referees to be uniform, because at any given period there are subareas that are more "popular" than others. Thus, implementation of an assignment procedure that is purely based on matching considerations is likely to lead to segregation of the proposals into subsets (in the extreme case, each proposal is a subset in its own right) where there is no overlap in the referees that review proposals across the different subsets. When this happens under ordinal-ranking settings, it severely restricts the validity of the overall ranking that will eventually be generated from the collection of individual referees' evaluation. The reason is quite simple—each partial ranking is limited to its relevant subset of proposals, and if such a subset has no overlap with another subset, we have no basis for comparison between them. For example, consider a subset of three excellent proposals sent to reviewers A and B, who evaluate them independently. Both reviewers rank order these proposals in decreasing order P_1, P_2, and P_3. Another subset of four rather mediocre proposals is sent to reviewers C, D, and E, who rank them (again in decreasing order) P_4, P_5, P_6, and P_7. Although the evaluations *within* each group of referees were consistent, the fact that there is no overlap between the two groups leaves the review board with an open question—how to combine the two separate rankings into an overall ranking.

The need to view proposals' evaluation from the perspective of partial rankings highlights the requirement for overlap among the subsets of proposals assigned to the various reviewers. Pairwise comparison data are specified in the form of binary preference matrices. This will mean that lack of overlap among the proposal subsets will result in zero entries (holes) in the matrix structure. In such cases, any final overall ranking is questionable.

The problem of aggregating individual rankings to create an overall ranking representative of the group is of longstanding interest in group decision making. It was first examined by Kemeny and Snell (1962) and later by Bogart (1975), who extended the structure to

partial orders. In particular, the problem of consensus ranking when preferences are represented in vector (rank order) format has been investigated extensively by many researchers, including Cook and Seiford (1978), Kirkwood and Sarin (1985), and Cook and Kress (1991), and various solution methods based on distance functions have been studied. Hence, we will not address this problem in the current paper.

The rest of this paper is organized as follows. In §2, we model the allocation problem through a set-covering (SC) formulation and explain how it generates allocations with the desired maximum overlap. But there might be situations with large numbers of proposals and reviewers where the SC formulation may become computationally intractable. Hence, we present and demonstrate a heuristic algorithm, based on "greedy" principles, which can be employed without difficulty even for very large problems. In §3, we report on a large set of numerical examples used to evaluate the exact and heuristic procedures proposed earlier. Section 4 concludes the paper.

2. Procedures for Allocating Proposals to Referees

Following the rationale given in §1, we seek an allocation of proposals to reviewers that uses the maximum available reviewing capacity while obtaining the maximum possible overlap in the evaluations. Our quantification of the term "overlap" is based on a pairwise perspective. Given N proposals, there are $\binom{N}{2}$ pairs. To avoid allocations that lead to mutually exclusive subsets of proposals, it is desirable that each pair be evaluated by at least one reviewer. Define $E = (e_1, e_2, \ldots)$ as the vector of *pair allocations*, where e_h, $h = 1, 2, 3, \ldots$ denotes the number of pairs evaluated by at least h reviewers. First, we prefer, when possible, to get allocation solutions where each and every pair is evaluated by at least one reviewer. Furthermore, for $h = 1, 2, \ldots$, our objective is to obtain allocations that are as balanced as possible. That is, we prefer solutions where each of the pairs is evaluated by about the same number of reviewers. Consequently, our objective is to maximize the utilization of the reviewers' capacity through a weighted number of pair allocations where the weights ensure a lexicographic minimization of the pair-allocations vector. This motivation can be demonstrated through the following illustrative example. Suppose that we have to allocate four proposals to three reviewers, and each reviewer is capable of reviewing three proposals ({1, 2, 4}, {2, 3, 4}, and {1, 3, 4}, respectively). But each reviewer is willing to review only two proposals. Allocating proposals {2, 4}, {2, 4}, and {3, 4} to the three reviewers, respectively, yields a pair-allocations vector $E = (2, 1, 0, 0, \ldots)$. On the other hand, the allocation {1, 2}, {2, 4}, and {3, 4} would yield a pair-

allocations vector $E = (3, 0, 0, \ldots)$, which is clearly preferred to the previous solution.

2.1. Set-Covering Integer-Programming Formulation

Let u_k be the number of proposals that referee k is willing to review. We assume that u_k is smaller than or equal to the number of proposals that referee k is *qualified* to review. Clearly, in any optimal solution each reviewer is assigned u_k proposals. We associate a collection $\mathcal{I}_k$ of proposal subsets for each reviewer. Each member $I \in \mathcal{I}_k$ contains u_k proposals that reviewer k is qualified to review. Our aim is to select, for each referee k, a single member of $\mathcal{I}_k$ so as to maximize the covering of the pairs $\{p, q\}$. Before presenting our set-covering binary integer-programming formulation (SCIP), some additional notations are needed.

Variables

x_k^I A binary variable whose value is 1 if referee k reviews the proposals according to subset $I \in \mathcal{I}_k$, and 0 otherwise.

t_{pq}^h A binary variable whose value is 1 if the number of referees that review the pair of proposals $\{p, q\}$ is exactly h.

Parameters

C_{kp}^I An indicator whose value is 1 if the combination of referee k and proposal p satisfy $p \in I$ for $I \in \mathcal{I}_k$, and 0 otherwise.

W^h A weight associated with "level" h. A selection of values for these weights that ensures a lexicographic preference structure is discussed in Proposition 2.1 below.

T_{pq} The number of referees capable of reviewing the pair of proposals $\{p, q\}$, $(p \neq q)$.

H The collection of all pairs of proposals $\{p, q\}$, $(p \neq q)$ for which there exists *at least* one referee who is qualified to review *both* proposals.

We now present the binary integer-programming formulation of the problem:

$$\text{(SCIP)} \quad \max \sum_{\{p, q\} \in H} \sum_{h=1}^{T_{pq}} W^h \cdot t_{pq}^h, \tag{1}$$

such that

$$\sum_{k=1}^{K} \sum_{I \in \mathcal{I}_k} C_{kp}^I \cdot C_{kq}^I \cdot x_k^I \geq \sum_{h=1}^{T_{pq}} h \cdot t_{pq}^h \quad \forall \{p, q\} \in H, \tag{2}$$

$$\sum_{I \in \mathcal{I}_k} x_k^I = 1, \quad k = 1, \ldots, K, \tag{3}$$

$$\sum_{h=1}^{T_{pq}} t_{pq}^h \leq 1 \quad \forall \{p, q\} \in H, \tag{4}$$

$$x_k^I, t_{pq}^h \in \{0, 1\}. \tag{5}$$

The objective (1) maximizes the weighted sum of the t_{pq}^h indicators. Note that $h = 0$ is excluded because W^0

is set to zero as explained in Proposition 2.1 below. The set of constraints (2), one for each pair, coupled with constraints (3) and (4), force the implication that $t_{pq}^h = 1$ means that h is the number of referees assigned to review the pair of proposals $\{p, q\}$. The set of constraints (3), one for each referee, ensures that exactly one subset I is chosen for each referee. Constraints (4), one for each pair of proposals, enforce that exactly one value of h is associated with each pair (strong inequality means that $h = 0$ is chosen). Finally, constraints (5) define the variables.

The proposition below states that an appropriate selection of the weights leads to a lexicographic minimization of the pair-allocations vector E.

PROPOSITION 2.1. *If the weights W^h are selected as positive increasing series with decreasing difference series, that is, $0 < W^1 < W^2 < \cdots$ and $W^h - W^{h-1} > W^{h+1} - W^h$ for all $h \geq 2$, then SCIP yields an assignment that minimizes the pair-allocations vector by lexicographic order.*

PROOF. First note that the objective function value (1) is uniquely determined by the pair-allocations vector E as $\sum_h (W^h - W^{h-1})e_h$ (with the convention that $W^0 \equiv 0$). Consider two assignments resulting in E and F vectors and assume that $E \succ_{\text{lex}} F$. That is, there is some t such that $e_h = f_h$ for all $h < t$ and $e_t > f_t$. Our claim follows from the fact that $\sum_h (W^h - W^{h-1})e_h > \sum_h (W^h - W^{h-1})f_h$. To prove our claim, it is enough to show that

$$\sum_{h \geq t}(W^h - W^{h-1})e_h > \sum_{h \geq t}(W^h - W^{h-1})f_h; \qquad (6)$$

that is,

$$(W^t - W^{t-1})(e_t - f_t) > \sum_{h \geq t+1}(W^h - W^{h-1})(f_h - e_h).$$

Now, by the way we selected W_h, we have that $(W^h - W^{h-1}) < (W^t - W^{t-1})$ for all $h > t$ and so it is enough to show that

$$(e_t - f_t) \geq \sum_{h \geq t+1}(f_h - e_h). \qquad (7)$$

But (7) always holds (as equality) by the facts $\sum_h e_h = \sum_h f_h = \sum_k \binom{u_k}{2} \equiv$ constant and $e_h = f_h$ for all $h < t$. □

An example of a series that meets the conditions of Proposition 2.1 is the harmonic series $W^h = \sum_{i=1}^h (1/i)$.

In many cases, it is sufficient to define the decision variable t_{pq}^h only for a few small values of h. This is because optimal solutions are approximately balanced, so the coverage of each pair is likely to be much smaller than T_{pq}. Moreover, if one prefers odd numbers of reviewers to be allocated for the proposal pairs to eliminate ties, t_{pq}^h can be defined only for odd values of h. By doing this, it is possible to reduce the dimension of our integer program.

Other refinements are possible. Suppose that there are many pairs for which there is not a single referee who can review both proposals. In this case, we

might like to ensure, as a secondary preference, that we "cover" the relations between the two parties in the pair through a third party (i.e., rely on the transitive rule). However, this may lead to a rather complicated formulation with many more variables and constraints, which would be significantly harder to solve.

Our numerical experiments indicate that the SCIP formulation can be solved to optimality for small to medium problems (see §3). However, larger instances, and in particular instances with large values of u_k, are more difficult to solve. The difficulty arises from the fact that the number of possible combinations for reviewer k who is qualified to review n_k proposals and willing to review u_k proposals is $\binom{n_k}{u_k}$. Thus, the number of x_k^I variables $\left(\sum_{k=1}^K \binom{n_k}{u_k}\right)$ may explode.

2.2. Heuristic (Greedy) Procedure
The basic principle of the heuristic procedure is to identify in each step a pair of proposals with the largest priority to be assigned and assign it (or at least one of its members) to the reviewer with the largest reviewing capacity from the group of reviewers who can review the said pair. The initial assignment priority for each pair is determined by T_{pq}—the larger the available number of reviewers that can review it, the smaller the priority is. Then, in each step, the priority is updated according to the number of reviewers who have already been assigned to the pair (or to one of its members).

To run the heuristic, we need to define the outcome measures:

n_{pq} The number of reviewers assigned to review both p and q.

n_{p-q} The number of reviewers who were assigned to review proposal p, were also qualified to review proposal q, have not yet exhausted their u_k capacity, but were not assigned to review proposal q.

The purpose of these outcome measures is to update the weights assigned to the pairs of proposals during the assignment process. The logic that underlines the specific indexing rule we use here to update the weights (see the prioritization step below) is as follows. The initial value for each weight is T_{pq}. Pairs with larger T_{pq} values are assigned higher weights, which means that there is a low priority to assign them to reviewers. As long as p and q were not assigned to the same referees, n_{pq} remains zero and the weight is not updated. When n_{pq} is positive, the weight is increased by $2 \cdot n_{pq}$, where the coefficient 2 reflects that the "worth" of each additional reviewer assigned to review the pair pq is double that of an additional reviewer who is capable of reviewing the pair. When n_{pq} is positive, we also account for indirect

comparisons between p and q. This is done through the square root of the product $n_{p-q} \cdot n_{q-p}$, where the square root function "compensates" for the product of the two relevant outcome measures, thus making their joint effect similar to that of n_{pq}. For example, suppose we start with $T_{pq} = 4$; consequently, the weight W_{pq} is also 4. Then, we assign both proposals to a certain reviewer, making $n_{pq} = 1$ and, consequently, $W_{pq} = 12$. Later, we make other assignments that lead to $n_{p-q} = n_{q-p} = 2$, causing the weight value to become $W_{pq} = 44$.

Part of the heuristic nature of the procedure stems from the fact that the measures n_{p-q} and n_{q-p} provide only an indication of possible indirect comparison. Assume that one of the reviewers who reviews p also reviews another proposal g. If one of the reviewers who reviews q also reviews g, then we have an indirect comparison. But this is not always the case. Finally, we note that there could be many other priority rules that one might use to satisfy the desired qualitative property of reducing a pair's priority as the number of direct and indirect comparisons increase.

The steps of the heuristic procedure are given below.
1. Prioritization of proposals
    (i) Compute for each pair $\{p, q\}$ a weight W_{pq} as follows:
$$W_{pq} = T_{pq} \cdot (1 + 2 \cdot n_{pq} \cdot (1 + 2 \cdot \sqrt{n_{p-q} \cdot n_{q-p}})).$$
    (ii) Order the pairs in a nondecreasing order of W_{pq}. Break ties arbitrarily.
2. Assignment of reviewers
    (i) Choose the first pair in the ordered list. If you encounter a pair for which there is no available referee, skip it.
    (ii) Select the referee with the largest u_k out of the referees capable of reviewing the selected pair. Assign both proposals of this pair to this referee and update the relevant u_k value.
3. Termination test
    (i) Decrease the number of proposals that the selected referee can read by either one or two (depending if both proposals were new to him).
    (ii) If there are no more available referees (i.e., $u_k = 0$ for all k), stop.
    (iii) Otherwise, return to Step 1.

An illustrative example of the heuristic procedure is given in the appendix.

3. Numerical Experiments

In this section, we present some numerical experiments to demonstrate the applicability of the proposed procedures. Our testing platform was a Pentium 4, 2 GHz with 512 MB RAM, running under

Table 1 **Four Classes of Test Problems**

Class	No. of proposals	No. of referees	u_k	$P(A_{kp} = 1)$
A	20	40	{3, 4, 5}	0.4
B	20	50	{3, 4, 5}	0.3
C	30	60	{3, 4, 5}	0.3
D	40	80	{3, 4}	0.25

Windows XP. To demonstrate the effectiveness (in terms of solution quality) and efficiency (in terms of computational times) of the two methods proposed for the assignment of proposals to reviewers, we constructed four classes of test problems with different numbers of reviewers, proposals, and reviewers' capacity (Table 1). The capacity of each reviewer was randomly drawn from the set specified in column u_k. The qualification of each reviewer with respect to each proposal was determined by a Bernoulli random variable with probability as specified in the far right column of the table (where $A_{kp} = 1$ if referee k is capable of reviewing proposal p, and 0 otherwise). For each class we generated 25 test problems.[1]

The heuristic algorithm was implemented in Matlab 6.1. The SCIP program was solved by CPLEX 8.0 on the same computer. The time limit was set to 30 minutes and the relative optimality tolerance was set to 10^{-5}. Solutions within this relative gap were considered optimal. The heuristic algorithm has always reached a solution in up to 160 seconds (even for the largest problems).[2]

The first column of Table 2 presents the percentage of the problems that were solved to optimality (feasibility in brackets) within the time limit (30 minutes). Note that only in group C we fail to solve two problems (out of the 25). These two problems were excluded from the statistics reported in the other columns. The average optimality gaps presented in the second column were calculated over all the problems for which a feasible solution was obtained. The value in brackets is the maximum optimality gap over all the 25 problems of the class. We used the harmonic series as weights for our mixed integer programming (MIP) formulation.

In the forth and sixth columns we present the average number of uncovered pairs and their share of the total number of proposal pairs (in parentheses) for both the MIP and the heuristic procedures. In the fifth and seventh columns we present the average number of proposals that were covered by at least one, two, three, and four reviewers, respectively. The average

Table 2   **Results of the Numerical Experiment**

	SCIP formulation				Heuristic	
Class	% solved to optimality (feasibility)	Average (max) optimality gap	Uncovered pairs	Covered pairs (E)	Uncovered pairs	Covered pairs (E)
A	95% (100%)	0.04% (0.55%)	0 (0%)	(190, 61, 0, 0)	18 (9%)	(172, 92, 14, 2, 0)
B	90% (100%)	0.00% (0.01%)	2 (1%)	(188, 98, 5, 0)	7 (4%)	(183, 81, 21, 4, 1)
C	10% (95%)	4.28% (12.84%)	81 (19%)	(354, 16, 0, 0)	134 (31%)	(301, 61, 9, 1, 0)
D	100% (100%)	0.00% (0.00%)	423 (54%)	(357, 0, 0, 0)	457 (59%)	(323, 30, 3, 0, 0)

lexicographic advantage of the optimal solution over the heuristic one is quite evident. Also, we note that in both methods and in virtually all our test problems, no proposal was reviewed by more than four reviewers.

The results demonstrate the usefulness of the integer-programming formulation for fairly large problems with up to 40 proposals and 80 referees, providing the number of possible assignments for each individual reviewer is not too large. Whenever the integer-programming formulation is able to produce a feasible solution within the time limitation, it is significantly superior to the solutions derived by the heuristic method, even when the optimality gap of the obtained solution is quite large. Nevertheless, the heuristic method is important for at least two reasons: (1) It is capable of quickly delivering a solution for large problems in which the integer-programming formulation fails to find a feasible solution in reasonable time, and (2) the feasible solutions it generates might serve as initial upper bounds to enhance the performance of the MIP solver.

4.   Conclusions

The process of reviewing, evaluating, and finally ranking research or research-related manuscripts (e.g., submissions to academic competitions or research proposals) is an integral part of academia. This process is based on peer review by researchers that usually perform this task on a voluntary basis. In many cases, the submissions are numerous and diverse in their subject topics and therefore require a large and diversified group of reviewers or judges. An important question in this context is how to assign the manuscripts to the various reviewers in the most effective and efficient way, considering their areas of expertise, their academic capabilities, and the number of manuscripts each reviewer is willing to review. Arguably, one cannot always expect that the mix of submitted manuscripts will conform to the available review capacities. Hence, a complicated set of trade-offs must be considered during the assignment phase.

We observe that in many peer review settings, where referees are required to provide a cardinal ranking of proposals, there are no clear norms for assessments, and there may be a large variation in what criteria the referees choose to emphasize and

how they emphasize them. In these settings, ordinal rankings provided by the referees may reflect the relative order of proposals better than cardinal rankings. Based on this observation, we propose a new approach for the proposals-to-reviewers assignment problem that provides a solution that maximizes the number of proposal pairs that are evaluated in a balanced way.

It is shown that the proposals-to-reviewers assignment problem can be represented as a set-covering problem, which can be solved quite easily for problems of moderate size. In general, and for very large numbers of proposals and reviewers, a simple yet efficient heuristic is proposed for solving the assignment problem. If every pair is reviewed by at least one reviewer, then connectivity among the proposals is guaranteed, and therefore a complete pairwise comparison aggregate matrix can be obtained.

The larger part of our input data is the binary matrix A_{kp}. This matrix may contain thousands of entries for a moderately sized problem (e.g., 3,200 entries for a problem with 40 proposals and 80 referees). We note that there is no need to enter this matrix manually. Instead, each proposal can be associated with one (or several) disciplinary area(s) out of a limited list of such areas. Similarly, each reviewer can state his or her expertise areas, out of the same list. Using these data, the creation of the matrix A_{kp} can be easily automated. Indeed, some journals have already adopted such a data-acquisition process through Web-based electronic submission systems.

Future research may explore mechanisms that will assist the review board to "negotiate" u_k values with the reviewers. Because the u_ks affect the SCIP model only implicitly (through the C_{kp}^I parameters), this will require the development of some special-purpose sensitivity analysis model. Another direction might be to extend the overlap concept from pairs to n-tuples with $n > 2$ (as the heuristic procedure attempts to do, at least implicitly, through the outcome measures n_{p-q} and n_{q-p}).

Appendix. Illustrative Example of the Heuristic Procedure

Consider the four-proposal, three-reviewer example we discussed at the beginning of §2.

Step 1. *Prioritization*. There are three pairs with a weight of 1 ({1, 2}, {1, 3}, {2, 3}) and three pairs with a weight of 2 ({1, 4}, {3, 4}, {2, 4}).

Step 2. *Assignment*. Assign proposals 1 and 2 to reviewer 1; update $u_1 = 1$; compute n_{pq}, n_{p-q}, n_{q-p}.

Step 1. *Prioritization*. The updated list of pairs (ordered in increasing weight values) is now as follows:

No.	Pair	Potential reviewers	n_{pq}	n_{p-q}	n_{q-p}	Weight
1	1, 3	3	0	1	0	1
2	2, 3	2	0	1	1	1
3	1, 4	1, 3	0	1	0	2
4	2, 4	1, 2	0	1	0	2
5	3, 4	2, 3	0	0	0	2
6	1, 2	—	1	0	0	3

Note that the pair (1, 2) now has no potential reviewers, because the single reviewer who was capable of reviewing both proposals (reviewer no. 1) has already received them for evaluation.

Step 2. *Assignment*. Assign proposals 1 and 3 to reviewer 3; update $u_3 = 1$; compute n_{pq}, n_{p-q}, n_{q-p}.

Step 1. *Prioritization*. The new list of pairs is now as follows:

No.	Pair	Potential reviewers	n_{pq}	n_{p-q}	n_{q-p}	Weight
1	2, 3	2	0	1	1	1
2	1, 4	1, 3	0	2	0	2
3	2, 4	1, 2	0	1	0	2
4	3, 4	2, 3	0	1	0	2
5	1, 2	—	1	1	0	3
6	1, 3	—	1	1	0	3

Step 2. *Assignment*. Assign proposals 2 and 3 to reviewer 2. Update $u_2 = 1$.

Step 1. *Prioritization*. The new list of pairs is now

No.	Pair	Potential reviewers	n_{pq}	n_{p-q}	n_{q-p}	Weight
1	1, 4	1, 3	0	2	0	2
2	2, 4	1, 2	0	2	0	2
3	3, 4	2, 3	0	2	0	2
4	1, 2	—	1	1	1	7
5	1, 3	—	1	1	1	7
6	2, 3	—	1	1	1	7

Step 2. *Assignment*. The top priority is now given to the pair 1, 4. Because the u_k values of all reviewers are now set to 1, we cannot assign the pair to any reviewer. Also, proposal 1 is already assigned to reviewers 1 and 3. Hence, we assign proposal 4 to reviewer 1 and update $u_1 = 0$.

Step 1. *Prioritization*. The new list of pairs is now

No.	Pair	Potential reviewers	n_{pq}	n_{p-q}	n_{q-p}	Weight
1	3, 4	2, 3	0	1	0	2
2	1, 4	3	1	2	0	4
3	2, 4	2	1	2	0	4
4	1, 2	—	1	1	1	7
5	1, 3	—	1	1	1	7
6	2, 3	—	1	1	1	7

Step 2. *Assignment*. The top priority is now given to the pair 3, 4. Again, it is impossible to assign them both to a single reviewer to whom none of them was already assigned. Because proposal 3 was already assigned to reviewers 2 and 3, we assign proposal 4 to reviewer 2 and update $u_2 = 0$.

Step 1. *Prioritization*. The new list of pairs is now

No.	Pair	Potential reviewers	n_{pq}	n_{p-q}	n_{q-p}	Weight
1	1, 2	—	1	1	1	7
2	1, 3	—	1	1	1	7
3	2, 3	—	1	1	1	7
4	2, 4	—	2	0	0	10
5	1, 4	3	1	1	1	14
6	3, 4	3	1	1	1	14

Step 2. *Assignment*. Now we need to skip the first four rows, as there are no remaining available reviewers to review these pairs. The top priority is now given to the pair 1, 4, where our only option is to assign proposal 4 to reviewer 3 and update $u_3 = 0$.

Step 3. *Termination*. At this point there are no more available reviewers, so we stop.

The final allocation, {1, 2, 4} → 1, {1, 3, 4} → 3, {2, 3, 4} → 2, is the unique optimal solution for this case.

References

Bogart, K. P. 1975. Preference structures II: Distances between asymmetric relations. *SIAM J. Appl. Math.* **29**(2) 254–262.

Campanario, J. M. 1998. Peer review for journals as it stands today. *Sci. Comm.* **19** 181–211.

Cicchetti, D. V. 1991. The reliability of peer review for manuscript and grant submissions: A cross-entropy investigation. *Behavioral Brain Sci.* **14** 119–186.

Cook, W., M. Kress. 1991. *Ordinal Information and Preference Structures: Decision Models and Applications.* Prentice-Hall, Englewood Cliffs, NJ.

Cook, W., L. Seiford. 1978. Priority ranking and consensus formation. *Management Sci.* **24**(16) 1721–1732.

Dirk, L. 1999. A measure of originality: The elements of science source. *Social Stud. Sci.* **29**(5) 765–776.

Garg, R. K. 1996. The influence of positive and negative wording and issue involvement on responses to Likert scales in marketing research. *J. Marketing Res. Soc.* **38**(3) 235–246.

Hodgson, C. 1995. Evaluation of cardiovascular grant-in-aid applications by peer-review—Influence of internal and external reviewers and committees. *Canadian J. Cardiology* **11**(10) 864–868.

Jayasinghe, U. W., H. W. Marsh, N. Bond. 2001. Peer review in the funding of research in higher education: The Australian experience. *Ed. Eval. Policy Anal.* **23**(4) 343–364.

Kemeny, J. G., L. J. Snell. 1962. Preference ranking: An axiomatic approach. *Mathematical Models in the Social Science*, Chapter 2. Ginn, Boston, MA, 9–23.

Kirkwood, C. W., R. K. Sarin. 1985. Ranking with partial information—A method and an application. *Oper. Res.* **33**(1) 38–48.

Langfeldt, L. 2001. The decision-making constraints and processes of grant peer review, and their effects on the review outcomes. *Social Stud. Sci.* **31**(6) 820–841.