Alternative Optimization Models for Reviewer Allocation in Peer Review Systems

by

Ali YEŞİLÇİMEN

A Thesis Submitted to the
Graduate School of Sciences and Engineering
in Partial Fulfillment of the Requirements for
the Degree of

Master of Science

in

Industrial Engineering

Koç University

January 15, 2015

Koç University

Graduate School of Sciences and Engineering

This is to certify that I have examined this copy of a master's thesis by

Ali YEŞİLÇİMEN

and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by the final
examining committee have been made.

Committee Members:

Assoc. Prof. Emre Alper Yıldırım (Advisor)

Prof. Metin Türkay

Asst. Prof. Sinem Çöleri Ergen

Date:

ABSTRACT

Today, most of the clinical and academic research are financed by the funding organizations. These organizations employ a selection process to select the proposals that are likely to have the highest return for the society. At this point, the selection of the proposals to be funded becomes crucial.

The main objective of this thesis is to propose exact and heuristic approaches to increase the success rate of the selection of better proposals for funding. We focus on the process of proposal evaluation by referees. The scope of this thesis is to develop solution methodologies to assign proposals to referees so as to maximize a certain set of criteria subject to a set of constraints. We develop two integer linear programming models. For larger instances, we propose heuristic methods.

We conduct computational experiments on randomly generated instances in an attempt to assess the performances of our exact and heuristic approaches. In comparison with previously proposed exact and heuristic approaches, our computational experiments reveal the promising performance of our solution approaches.

ÖZETÇE

Günümüzde araştırmaların büyük bir bölümü bilimsel araştırmalara destek veren kuruluşlar tarafından finanse edilmektedir. Bu kuruluşlar, finansal destek arayışında olan araştırmacıların, proje önerilerini bir seçim sürecinden geçirerek, çıktısı en iyi olabilecek projelere maddi destek sağlamaktadırlar.

Bu tezde, finansal destek sağlanacak araştırmaların seçiminin daha başarılı olmasını sağlamak amacıyla kesin ve sezgisel çözüm yöntemleri geliştirilmiştir. Projemizde temel olarak proje önerilerinin değerlendirme sürecinde görev alan hakemlere atanması üzerinde durduk. Bu atamaları bir takım kısıtlamalara bağlı kalarak belirli bir kriteri eniyileyecek şekilde gerçekleştiren çözüm yöntemleri geliştirdik. Bu kapsamda, iki adet tamsayılı doğrusal programlama modeli geliştirdik. Büyük problemler için sezgisel yöntemler oluşturduk.

Çözüm yöntemlerimizin performansını ölçmek amacıyla rastgele oluşturulmuş test problemlerini çözdük ve daha önce geliştirilmiş optimizasyon modelleri ve sezgisel yaklaşımlara kıyasla umut veren sonuçlar elde ettik.

ACKNOWLEDGMENTS

This work was supported in part by TÜBİTAK Grant 109M149.

I would like to express my deepest gratitude to my advisor, Assoc. Prof. Emre Alper Yıldırım for his guidance and support.

I would like to thank Prof. Metin Türkay and Asst. Prof. Sinem Çöleri Ergen for accepting to read and review this thesis.

I would like to express my deepest gratitude to my family.

I am grateful to all of my friends. I wish to express my special thanks to Çağın Ararat for his everlasting support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

1.1 Definition of Peer Review

Peer review is the evaluation of a work or performance by a group of people in a related profession. It is based on the concept that the value of a work or performance is assessed by qualified members who are experts in the same area. As the word "peer" in peer review clearly implies; the work will be submitted to the colleagues of the owner of the work, who are generally not close to her. These experts apply predefined norms and criteria to objectively gauge the quality of the work in question.

It is thought that peer review has been an evaluation method since ancient Greece. The first documented description of a peer-review process is in a book called Ethics of the Physician written by Ishaq bin Ali al-Rahwi [Spier, 2002]. He stated that it is the duty of a visiting physician to take notes of a patient's condition at each visit. When the patient had been cured or had died, the notes of the physician were examined by a local council to decide whether the physician performed according to standards. If their reviews were negative, the practicing physician could be sued for damages by a maltreated patient. At that time, this control system was used in health care to encourage physicians to give better service. Today peer review systems are used for many purposes such as scholarly evaluation of scientific publications, funding of research proposals, accreditation, promotion, and certification.

1.2 Peer Review Systems in Academia

Peer review systems are extensively used in scientific research to assess the value of scientific studies. There are mainly two types of peer review that take place in scientific research. The first one is used when researchers submit proposals of their upcoming or ongoing research projects to request funding. In this case, the main decision maker is the funding organization, which applies a selection process based on peer review to fund a limited number of research projects among all submitted ones. The second type of peer review is used to determine if a completed research project merits publication in a scientific journal or a conference proceeding. Editors of scholarly journals or conference organizers need to decide whether the paper will be published considering the contribution, impact, novelty, and accuracy of the content of the manuscript.

1.3 Peer Review Based Selection Process in Scientific Research Funding

Most of the current academic research projects are financially supported by governments, corporations or foundations. A research project obtains funding after a competitive process, in which potential research projects are evaluated and the most promising ones are selected.

The selection process typically starts with a Call for Proposals. Research proposals are submitted according to guidelines determined by the funding agencies. After the application deadline, proposals are sent to a group of reviewers who are provided with some metrics that should be applied to assess the quality of the submitted research projects. In most cases, each referee is asked to review a subset of all proposals. Finally, the individual reviews of the referees are collected by the funding agency and some aggregation scheme is applied to select the projects that will be funded [Cook et al., 2005].

1.4 Importance of Peer Review Process in Scientific Research Funding

After the industrial revolution, technological progress has been the main source of the economic development. For this reason, research and development (R&D) funding is continuously expanding. Each year governments, corporations, and foundations spend billions of dollars to support research projects.

Over the past 60 years, R&D spending by government, industry and all other sources has shown a dramatic increase. Figure 1.1 tracks total U.S. spending on R&D in both nominal terms and real terms [National Science Board, 2012]. The blue line shows the nominal R&D spending over years 1950 to 2010 while the red line shows the inflation adjusted values with 2005 dollars. As we can observe, R&D spending in the US has been rising steadily.

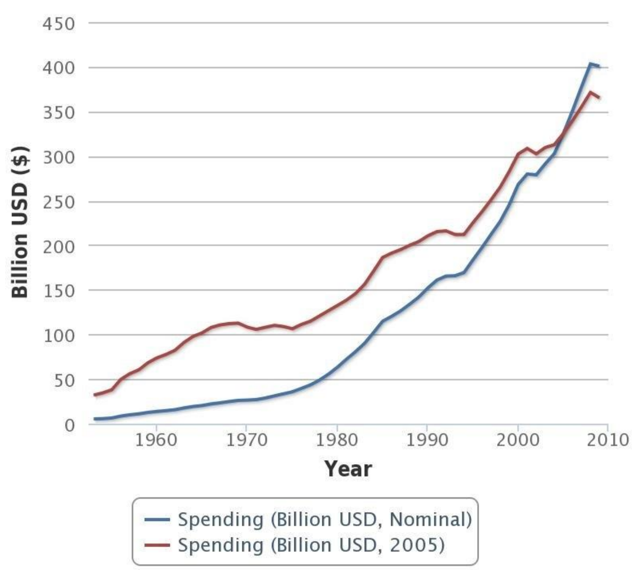

Figure 1.1: Total US R&D Spending

Figure 1.2 shows the proportions of R&D spending sources over the same time horizon [National Science Board, 2012]. As we can clearly observe, private industry has become more interested in funding R&D over the years. The dominance of private industry suggests that scientific funding is actually regarded as an investment. At this point, the evaluation of potential outcomes of research projects becomes crucial for funding organizations in order to maximize their return on investment. This observation shows the importance of peer review systems in scientific research funding.

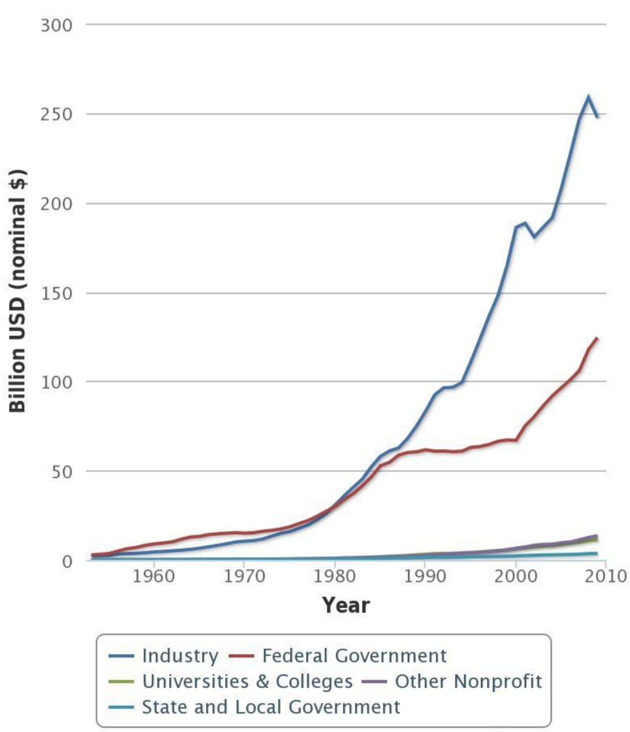

Figure 1.2: Sources of funding for US R&D

Federal Government is also an important source for R&D funding in the United States. In 2013, the US President Barack Obama made a statement on government support on scientific research funding at the 150^{th} annual meeting of the National Academy of Sciences in Washington DC. Obama said: “One of the things that I have tried to do over these last four years, and will continue to do over the next four years, is to make sure that we are promoting the integrity of our scientific process. Not just in the physical and life sciences, but in psychology and anthropology and economics and political science, all of which are sciences". President Obama also underlined the importance of peer review process to get the maximum outcome from the research projects that are funded.

1.5 Reliability of Peer Review Systems

In scientific research funding, peer review systems are used to determine which research projects will receive funding. As previously mentioned, this selection system is crucial for the society to get the maximum benefit from the resource allocated for scientific research. On the other hand, peer review is often criticized for being subjective and not reliable.

In 1981, Cole and Simon forwarded 150 recently reviewed proposals from the National Science Foundation to an expert panel chosen from the members of the National Academy of Sciences. The results of these two panels showed a moderate correlation (0.60-0.66) and there was a disagreement on nearly one quarter of the proposals. This study showed that reviewers are more likely to agree on excellent proposals than moderate ones [Cole et al., 1981].

Over the past 30 years, researchers have conducted dozens of studies to test the reliability and objectivity of peer review systems. Moreover, there have been several studies offering solutions to overcome shortcomings of the peer review systems. In this thesis, we focus on an important operational aspect of the peer review process - the assignment of proposals to referees. We present alternative exact and heuristic methods and give a detailed analysis of these methods.

The rest of the thesis can be summarized as follows. In Chapter 2, we give a literature survey. In Chapter 3, we define our problem and present our optimization models. For large problem instances, we present our heuristic models in Chapter 4. We present the results of our computational experiments in Chapter 5. Finally, we conclude our thesis in Chapter 6.

Chapter 2

LITERATURE REVIEW

As we stated, funding organizations use a selection process to determine the research projects that will be funded. This process consists of three main stages:

1. **Assignment:** After the application deadline, each referee is asked to review a subset of the submitted proposals. The assignment of proposals to the referees is the first stage of the selection process.

2. **Ranking:** Each referee evaluates the assigned proposals with respect to a predetermined set of criteria and returns a ranking.

3. **Aggregation:** Funding organization collects the individual evaluations of the referees and uses an aggregation scheme to create an overall ranking of all submitted proposals.

In this thesis, we deal with the problem of assigning the proposals to the referees by considering the integrity of the whole process described above. In the following parts of this chapter, we will investigate the studies on ranking, assignment and aggregation stages, respectively. Finally, we will present our focus and contribution of our research.

2.1 Studies on Ranking

After the evaluations, referees return a ranking of the proposals they review. In practice, there are several ranking methods for expressing preferences among proposals and these ranking methods have been the focus of several studies as they highly affect the accuracy of the decisions made in peer review systems. A commonly used method is cardinal ranking, where referees assign numerical grades in a predetermined range for each proposal they review. In peer review process, however, *cardinal ranking* is not practical because it is usually very difficult to quantify the value of a proposal.

Another common ranking method is *ordinal ranking*, where each reviewer returns a complete ranking of the proposals. For example, if a reviewer is assigned to review three proposals labeled as X, Y, and Z, she returns a complete ranking such as $Y > Z > X$ rather than assigning a numerical grade for each proposal. Cook *et al* compare cardinal and ordinal rankings and state that "Cardinal rankings are superior to ordinal rankings as they provide more refined information. The trouble is that more refined information is practically very difficult to attain in general" [Cook et al., 2005] . Since a fair cardinal ranking is very difficult, ordinal ranking is a more practical method than cardinal ranking in peer review processes. On the other hand, if the number of proposals a referee needs to review is large, it may be very difficult to give a complete ranking.

One other commonly used method is *pairwise comparison*, where a referee makes a decision for each pair of proposals. For example, if a referee is assigned to review three proposals labeled as X, Y, and Z, then she states her preference for each pair of proposals (X, Y), (X, Z), and (Y, Z) rather than giving a complete ranking of the three proposals.

The way the reviewers express their preferences may create some inconsistencies during the aggregation of the individual evaluations. This situation is explained in Cook *et al* 2005 [Cook et al., 2005]. In the example given in Table 2.1, there are two reviewers (A and B) who review three proposals (X, Y, and Z).

Reviewers/Proposals	***X***	***Y***	***Z***	**Generosity**
A	5	4		Moderate
B		8	7	High
Average	5	6	7	

Table 2.1: Sample Reviewer Evaluations in Cardinal Ranking

In cardinal ranking settings, referee A prefers X over Y and grades them as 5 and 4, respectively. On the other hand, referee B who is known to be more generous than referee A prefers Y over Z and rates them as 8 and 7, respectively. If we aggregate the individual evaluations based on cardinal rankings, we sort the proposals according to their average grades and we obtain a ranking given by $Z > Y > X$. However, if we apply pairwise comparison, we end up with a completely different overall ranking. Partial rankings of referees A and B are given by $X > Y$ and $Y > Z$, respectively. In this setting, if we apply the transitivity property for aggregation, we will obtain an overall ranking given by $X > Y > Z$, which is precisely the opposite of the previous ranking.

Although most of the peer review systems to date are based on cardinal ranking, several researchers have criticized the reliability of these processes. Studies conducted by Cicchetti [Cicchetti, 1991], Hodgson [Hodgson, 1995], Campanario [Campanario, 1998], and Jayasinghe *et al* [Jayasinghe et al., 2001] showed that cardinal ranking system causes various biases in such peer review systems [Cook et al., 2005]. In this thesis, we therefore build our solution methodology based on pairwise comparison ranking method.

2.2 Studies on Assignment

The assignment of proposals to referees is a difficult decision problem. First of all, each proposal is associated with particular research fields. Therefore, a referee must have competence in the related research areas to be eligible to review a specific proposal. Moreover, there are limits on the number of proposals each referee is willing to review and the assignment must also honor these capacity constraints.

In most of the studies, this problem is modeled as a generalized assignment problem. Benferhat and Lang [Benferhat and Lang, 2001], Tian *et al* [Tian et al., 2002], Merelo-Guervos *et al* [Merelo-Guervos et al., 2003], Janak *et al* [Janak et al., 2006], Sun *et al* [Sun et al., 2007], Schirrer *et al* [Schirrer et al., 2007] are several examples of this approach. These studies are based on the maximization of the total proficiency coefficient, which is a value in a predetermined range representing the proficiency of a reviewer for evaluating a specific proposal. In these studies, a proficiency coefficient matrix and referee capacities are the input parameters. The objective is to maximize the total proficiency coefficients of the referee-proposal assignments. In Table 2.2, an example of a proficiency coefficient matrix is given for three referees (Referees A, B, C) and four proposals (W, X, Y, Z). We note that proficiency coefficients are between 0 and 1, where 0 indicates that referee is not qualified to review the related proposal and 1 indicates that the referee is an expert. In this example, each referee is willing to review two proposals.

Reviewers/Proposals	$\boldsymbol{W}$	$\boldsymbol{X}$	$\boldsymbol{Y}$	$\boldsymbol{Z}$
$\boldsymbol{A}$	1	0.8	0.3	0
$\boldsymbol{B}$	0.8	0.7	0	0.4
$\boldsymbol{C}$	0	0.3	0.6	0.5

Table 2.2: Sample Proficiency Coefficient Matrix

In the optimal solution, the proposals W and X are assigned to referees A and B, and proposals Y and Z are assigned to referee C. This solution yields an objective value (total proficiency coefficient) of 4.4. However, as Cook [Cook et al., 2005] states, this implementation is purely based on matching considerations and leads to segregation of the proposals into subsets where there is no overlap. For this reason, it severely restricts the validity of the overall ranking which is created by aggregating individual evaluations of the referees. To make Cook‘s statement clear, assume that referees A and B agree on $W > X$ and referee C returns $Z > Y$. Since the two subsets (W, X) and (Y, Z) have no common proposal, it is not possible to combine the two separate rankings into an overall ranking. Therefore, this approach is often criticized for disregarding the problem of creating an overall ranking.

In 2005, Cook *et al* [Cook et al., 2005] proposed a set covering integer programming model to maximize the different number of proposal pairs that will be evaluated by one or more reviewers. This new approach helps to facilitate meaningful aggregation of individual rankings of multiple reviewers. In this methodology, a 0-1 proficiency matrix and referee capacities are the input parameters and the objective is to maximize the number of proposal pairs that are evaluated by at least one referee. In Table 2.3, the adapted version of the proficiency matrix in Table 2.2 is given. One should note that the proficiency coefficients are either 0 or 1 in this case. Similarly, each referee is willing to review at most two proposals.

Reviewers/Proposals	***W***	***X***	***Y***	***Z***
A	1	1	1	0
B	1	1	0	1
C	0	1	1	1

Table 2.3: Sample $\{0, 1\}$ Proficiency Coefficient Matrix

In the optimal solution of this example, referee A reviews proposals X and Y, referee B reviews proposals W and X, and referee C reviews proposals Y and Z. As it is clear, this solution provides overlaps in the subsets of the proposals assigned to referees and thus it constitutes a good basis for the aggregation stage.

Cook *et al* [Cook et al., 2005] also proposed a heuristic method for large problem instances. In this thesis, we propose two alternative optimization models and a heuristic method using the same objective function. We will present our solution approaches and compare the results in the following chapters.

2.3 Studies on Aggregation

Aggregation of individual evaluations is the last stage of the peer review process. This stage is shaped according to the methods used in the ranking stage.

In cardinal ranking setting, the final ranking is obtained by sorting all proposals according to the average of the grades given by the referees. Although this aggregation scheme is very straightforward, the results may be biased because of the generosity of the referees, which is explained in Section 2.1.

In ordinal ranking or pairwise comparison methods, aggregation schemes are much more complicated. The evaluation of a set of proposals by a single referee can be seen as a directed acyclic graph. Hence, the evaluation of a group of proposals by a group of referees can be modeled as the union of a set of directed acyclic graphs. If the union of all directed acyclic graphs does not constitute a cycle, then it means the evaluations of referees are consistent. Otherwise, creating a consensus ranking becomes a problem of obtaining an acyclic directed graph eliminating the minimum number of arcs. For instance, assume that a proposal pair (X, Y) is reviewed by three referees A, B, and C. Referees A and B may prefer X over Y while referee C prefers Y over X. For this small example, since two referees agree on that X is greater than Y, we will disregard the preference of referee C rather than disregarding the reviews of A, B.

Cook *et al* 2005 [Cook et al., 2005] summarized the studies on this problem. It was first examined by Kemeny and Snell in 1962 [Kemeny and Snell, 1962] and later by Bogart in 1975 [Bogart, 1975]. The problem of creating consensus ranking has been studied by many researchers including Cook and Seiford [Cook and Seiford, 1978], Kirkwood and Sarin [Kirkwood and Sarin, 1985], and Cook and Kress [Cook and Kress, 1991]. Since our connection to this problem is very limited, we will not get into the details of these studies in this thesis. However, we remark that our objective function in the assignment stage is chosen to enhance the aggregation stage.

2.4 Focus and Contribution

In this thesis, we propose alternative optimization models for the problem modeled as a set-covering integer optimization problem by Cook *et al* [Cook et al., 2005]. The main difference between Cook *et al*'s approach and proposed alternatives is that we eliminated some of the decision variables in Cook *et al*'s formulation and defined new ones. Hence the number of decision variables defined in our models is less than that of Cook *et al*'s. However, the number of constraints increases. With the help of this change, a solution to the problem is expressed with a smaller set of decision variables.

We also developed a heuristic algorithm for large problems. This algorithm is based on the neighborhood search.

We conduct computational experiments on randomly generated instances in an attempt to assess the performances of our exact and heuristic approaches. In comparison with previously proposed exact and heuristic approaches, our computational experiments reveal the promising performance of our solution approaches.

Chapter 3

PROBLEM DEFINITION AND OPTIMIZATION MODELS

In the previous chapter, we gave a short description of the problem studied in this thesis. In this chapter, we formally define the problem in Section 3.1. We present our solution approaches in Section 3.2.

3.1 Problem Definition

The reader should recall that after the assignment, each referee is asked to return a partial ranking on the proposals she is assigned to. There are several ranking methods for expressing preferences among proposals. In Chapter 2, we gave brief definitions of three different ranking methods: cardinal ranking, ordinal ranking, and pairwise comparison. We argued that pairwise comparison is more practical than cardinal and ordinal rankings in most of the situations. Therefore, we build our solution methodology based on pairwise comparisons in which, a referee makes a direct choice over each pair of proposals she is assigned to. For example, if a referee is assigned to review four proposals, she will make $\binom{4}{2}$ comparisons and indicate her preferences for each pair.

Assume that there are N proposals, which are going to be reviewed by K referees. Each proposal is associated with some research areas and could be reviewed by referees who are knowledgeable in these areas. Moreover, referees are willing to review only a limited number of proposals. We refer to this number as the capacity of a referee and denote by u. The objective is to assign a subset of N proposals of size u to each referee in order to cover the maximum number of different proposal pairs.

The reader should also recall that since each referee reviews a subset of N proposals and returns her pairwise comparisons, an aggregation scheme is applied by the funding organization to create an overall ranking of all proposals. In most cases, there are two types of inconsistencies among the partial rankings of different referees. The first type may occur when a proposal pair is evaluated by more than one referee. For example, assume that proposal pair (X, Y) is reviewed by three different referees, labeled as A, B, and C. This type of inconsistency occurs in situations where referees A and B return $X > Y$ and referee C returns the opposite. The second type of inconsistency is based on the transitivity property of inequality. For example, if referee A returns $X > Y$ and referee B returns $Y > Z$, transitivity property suggests that $X > Z$. However, a third referee, say referee C, may return $Z > X$. There are various types of aggregation methods. The most common methods are based on creating a consensus ranking which is explained in Section 2.3. The main idea is to disregard the minimum number of reviews which cause inconsistencies.

In this study, we focus on the assignment problem considering the integrity of the whole process. Therefore, our objective is to cover the maximum number of proposal pairs. Our goal is to have an allocation where, maximum number of pair of proposals is evaluated by at least one referee. If we can do this without exhausting all capacity of the referees, we use the remaining capacity so as to maximize the number of proposal pairs which are reviewed by more than one referee.

A simple calculation would be helpful to express our objective clearly. Since we have N proposals, there are a maximum of $\binom{N}{2}$ proposal pairs. In addition, each referee is willing to review u proposals, which turns out that we have a total capacity of reviewing $K \times \binom{u}{2}$ proposal pairs. Assume there is a problem instance where $K \times \binom{u}{2} >> \binom{N}{2}$. For such a problem instance, we will put priority to obtain a solution where $\binom{N}{2}$ different pair of proposals is reviewed by at least one referee. If we can do this without using all capacities of the referees, the question of how we will use the rest of the capacities rises up.

Cook *et al* formalizes this objective with a vector of pair allocations $E^* = (e_1, e_2, ...)$, where e_h, h=$1, 2, \ldots$ denotes the number of proposal pairs evaluated by h referees. First, we prefer, whenever possible, that every pair of proposal be evaluated by at least one referee. Furthermore, for $h = 1, 2, ...$, the objective is to obtain allocations that are as balanced as possible. For example, assume that there are three proposals labeled as X, Y, and Z. Assume there are three referees A, B, and C who are willing to review two proposals. If referees A, B, and C are assigned to review (X, Y), (X, Y), and (Y, Z), respectively, then we will have a pair-allocations vector $E^* = (2, 1, 0, 0, ...)$. On the other hand, if we have an allocation (X, Y), (X, Z), and (Y, Z), then the vector will be $E^* = (3, 0, 0, 0, ...)$, which is better than the previous solution since a larger number of different proposal pairs is reviewed. Therefore, our main goal is to allocate pairs of proposals in such a way that the coordinates of the vector E^* are increased in a lexicographic order.

3.2 Solution Approaches

In this section, we first present the mathematical formulation proposed by Cook *et al* [Cook et al., 2005] and then present two alternative formulations.

3.2.1 Formulation Proposed by Cook et al

Cook *et al* proposed a set-covering integer-programming formulation to solve the problem introduced in Section 3.1. We can summarize all of the parameters used for this formulation as follows:

K Number of referees

N Number of proposals

u Number of proposals a referee is willing to review

F_k A collection of subsets of proposals, which contains u proposals that referee k is qualified to review. $k = 1, ..., K$

C^I_{kp} An indicator whose value is 1 if the combination of referee k and proposal p satisfy $p \in I$ for $I \in F_k$, and 0 otherwise. $k = 1, ..., K, \quad I \in F_k, \quad p \in I$

W^h A weight associated with "level" h. $h = 1, ..., K$

H The collection of all ordered pairs of proposals $\{p, q\}$, for which there exist at least one referee who is qualified to review both proposals, $p, q \in \{1, ..., N\}$, $(p < q)$

T_{pq} The number of referees capable of reviewing the pair of proposals $\{p, q\}$. $\{p, q\} \in H$

We next define the decision variables:

x_k^I A binary variable whose value is 1 if referee k reviews the subset I of proposals, and 0 otherwise. $k = 1, ..., K, \quad I \in F_k$

t_{pq}^h A binary variable whose value is 1 if the number of referees who review the pair of proposals $\{p, q\}$ is exactly h, and 0 otherwise. $\{p, q\} \in H, \quad h = 1, ..., T_{pq}$

Using the parameters and decision variables defined above, Cook *et al* formulate the problem as follows:

$$max \qquad \sum_{\{p,q\} \in H} \sum_{h=1}^{T_{pq}} W^h.t_{pq}^h, \tag{3.1}$$

subject to

$$\sum_{k=1}^{K} \sum_{I \in F_k} C_{kp}^I.C_{kq}^I.x_k^I \geq \sum_{h=1}^{T_{pq}} h.t_{pq}^h, \quad \forall \{p, q\} \in H, \tag{3.2}$$

$$\sum_{h=1}^{T_{pq}} t_{pq}^h \leq 1, \quad \forall \{p, q\} \in H, \tag{3.3}$$

$$\sum_{I \in F_k} x_k^I = 1, \quad k = 1, ..., K, \tag{3.4}$$

$$x_k^I \in \{0, 1\}, \quad k = 1, ..., K, \quad \forall I \in F_k, \tag{3.5}$$

$$t_{pq}^h \in \{0, 1\}, \quad \forall \{p, q\} \in H, \quad h = 1, ..., T_{pq}. \tag{3.6}$$

The objective function and constraints are explained below.

1. **Objective Function**

$$max \quad \sum_{\{p,q\}\in H} \sum_{h=1}^{T_{pq}} W^h.t_{pq}^h \tag{3.1}$$

The objective function is a linear function denoting the overall weight associated with all pairs of proposals. The contribution of each proposal pair $\{p, q\}$ to the objective function is determined by a weight W^h associated with the number of times $h = 1, \ldots, K$, that the pair $\{p, q\}$ is reviewed.

After the allocation, a proposal pair may not be reviewed by any referee. In this situation, the contribution of this pair to the objective function value will be 0. On the other hand, a pair of proposals may be reviewed by one or more referees and even all of the referees. It is very straightforward that in order to have a better judgment on a pair of proposals, we want it to be reviewed by the maximum number of referees possible. However, we have a limited capacity of reviewing proposal pairs and we prefer to have the maximum number of different proposal pairs to be reviewed at least once rather than having some proposal pairs to be reviewed many times. If we can still make assignments after having all possible proposal pairs reviewed by at least one referee, then the objective is to have the maximum number of proposal pairs to be reviewed twice. This approach is the same until we totally distribute our capacity to distinct proposal pairs in a balanced way. For this reason, W^h values are chosen in order to obtain as balanced solutions as possible. The following example illustrates how W^h values helps us to make allocations such that we use our reviewing capacity in the desired way.

Assume that 6 proposals labeled as p, q, r, s, t, u and there are 5 referees labeled as A, B, C, D, and E. The proficiency matrix is given in Table 3.1. In this example, there are $\binom{6}{2}$= 15 different proposal pairs. However, proposal pair $\{s, t\}$ cannot be reviewed because there is no referee who is eligible to evaluate proposals s and t together. Hence, we focus on 14 distinct proposal pairs. Assume that each referee is willing to review three proposals. In this case, 5 referees will perform $\binom{3}{2}\times 5 = 15$ proposal pair comparisons.

Referees/Proposals	$\boldsymbol{p}$	$\boldsymbol{q}$	$\boldsymbol{r}$	$\boldsymbol{s}$	$\boldsymbol{t}$	$\boldsymbol{u}$
$\boldsymbol{A}$	1	0	1	1	0	0
$\boldsymbol{B}$	1	1	0	0	1	1
$\boldsymbol{C}$	0	1	1	1	0	1
$\boldsymbol{D}$	1	1	0	1	0	1
$\boldsymbol{E}$	1	1	1	0	1	1

Table 3.1: Sample Proficiency Coefficient Matrix

Consider the following possible solution:

- Referee A is assigned to proposals p, r, s
- Referee B is assigned to proposals t, u, q
- Referee C is assigned to proposals q, r, u
- Referee D is assigned to proposals p, q, s

Before we give the assignments of Referee E, we provide the number of reviews for each pair of proposals after the assignments of Referees A, B, C, and D in Table 3.2.

	$\boldsymbol{p}$	$\boldsymbol{q}$	$\boldsymbol{r}$	$\boldsymbol{s}$	$\boldsymbol{t}$	$\boldsymbol{u}$
$\boldsymbol{A}$	-	1	1	2	0	0
$\boldsymbol{B}$	-	-	1	1	1	2
$\boldsymbol{C}$	-	-	-	1	0	1
$\boldsymbol{D}$	-	-	-	-	0	0
$\boldsymbol{E}$	-	-	-	-	-	1
$\boldsymbol{F}$	-	-	-	-	-	-

Table 3.2: Number of Reviews for Each Pair of Proposals

As we can observe 8 proposal pairs are reviewed once and 2 proposal pairs are reviewed twice. 4 proposal pairs are not reviewed yet. Since Referee E is eligible to review 5 proposals and we will assign 3 proposals to her, there are $\binom{5}{3}$ =10 possible assignments. We will investigate 3 of these 10 possible assignments that summarize the speciality of the objective function:

- Assignment Alternative 1: Proposals q, r, u are assigned to Referee E

  With this assignment, Referee E is going to review proposal pairs $\{q,r\}, \{q,u\}$, and $\{r,u\}$ and the Table 3.2 will be updated as in Table 3.3.

	$\boldsymbol{p}$	$\boldsymbol{q}$	$\boldsymbol{r}$	$\boldsymbol{s}$	$\boldsymbol{t}$	$\boldsymbol{u}$
$\boldsymbol{A}$	-	1	1	2	0	0
$\boldsymbol{B}$	-	-	**2**	1	1	**3**
$\boldsymbol{C}$	-	-	-	1	0	**2**
$\boldsymbol{D}$	-	-	-	-	0	0
$\boldsymbol{E}$	-	-	-	-	-	1
$\boldsymbol{F}$	-	-	-	-	-	-

Table 3.3: Number of Reviews for Each Pair of Proposals after Assignment Alternative 1

  As we can observe, this assignment returns two once-reviewed proposal pairs into twice reviewed pairs and one twice-reviewed proposal pair into thrice-reviewed pair. The number of non-reviewed proposal pairs is still 4. The pair-allocations vector of this solution is $E^* = (6,3,1,0,0)$ which means there are 6 once-reviewed, 3 twice-reviewed, 1 thrice-reviewed proposal pairs and no pair of proposal is reviewed 4 or 5 times.

- Assignment Alternative 2: Referee E is assigned to proposals r, t, u

  Referee E is going to review proposal pairs $\{r,t\}$, $\{r,u\}$, and $\{t,u\}$.

Table 3.2 will now be updated as in Table 3.4.

	p	***q***	***r***	***s***	***t***	***u***
A	-	1	1	2	0	0
B	-	-	1	1	1	2
C	-	-	-	1	**1**	**2**
D	-	-	-	-	0	0
E	-	-	-	-	-	**2**
F	-	-	-	-	-	-

Table 3.4: Number of Reviews for Each Pair of Proposals after Assignment Alternative 2

In this assignment, 1 non-reviewed proposal pair is turned into once-reviewed, and 2 once-reviewed proposal pairs are turned into twice-reviewed. The pair-allocations vector of this solution is $E^* = (7, 4, 0, 0, 0)$. This assignment is preferable to the assignment given in Alternative 1 since it reduces the number of non-reviewed proposals from 4 to 3. The reader should note that there is no thrice-reviewed proposal pair in this solution. However, the number of once-reviewed proposal pairs is increased from 6 to 7 and the number of twice-reviewed proposal pairs is increased from 3 to 4.

- Alternative 3: Referee E is assigned to proposals p, r, t.

  With this assignment, Referee E evaluates proposal pairs $\{p, r\}$, $\{p, t\}$, and $\{r, t\}$ and the number of reviews for each pair of proposals will be as in Table 3.5:

  In this alternative, the pair-allocations vector will be $E^* = (9, 3, 0, 0, 0)$. This solution is the best alternative for this example, where the number of once-reviewed proposal pairs is 9 and the number of twice-reviewed proposal pairs is 3.

	p	q	r	s	t	u
A	-	1	**2**	2	**1**	0
B	-	-	1	1	1	2
C	-	-	-	1	**1**	1
D	-	-	-	-	0	0
E	-	-	-	-	-	1
F	-	-	-	-	-	-

Table 3.5: Number of Reviews for Each Pair of Proposals after Assignment Alternative 3

In order to favor the balanced solutions, in the objective function, the contribution of a review for a non-reviewed proposal pair must be greater than the contribution of an additional review for a once-reviewed proposal pair. For the example given above, assume that $W^1 = 25$, $W^2 = 45$, $W^3 = 60$, $W^4 = 70$, and $W^5 = 75$. In such a parameter setting, the marginal contributions of a review for a non-reviewed, once-reviewed, and twice-reviewed proposal pairs will be 25, 20, and 15, respectively. To make it clear, assume that proposal pair $\{p, q\}$ is non-reviewed. If we can have it reviewed by one referee, the contribution of $\{p, q\}$ to the objective function value will increase from 0 to 25. Instead if we make a once reviewed proposal pair, say $\{q, r\}$, to be twice-reviewed, then its contribution will increase from 25 to 45, where the marginal contribution is less than the former case.

For the above selection of W^h values, the objective function values of the 3 alternative assignments can be calculated by multiplication of pair allocations vector and the vector of W^h values as given below:

- Alternative 1: $(6, 3, 1, 0, 0) \times (25, 45, 60, 70, 75)^T = 345$
- Alternative 2: $(7, 4, 0, 0, 0) \times (25, 45, 60, 70, 75)^T = 355$
- Alternative 3: $(9, 3, 0, 0, 0) \times (25, 45, 60, 70, 75)^T = 360$

Since the marginal contribution of a review for a proposal pair that has been previously reviewed fewer number of times is more for each $h, h = 1, \ldots K$, the objective function value of a more balanced solution will always be greater than that of non-balanced solutions. Cook *et al* suggests that the values of W^h selected as an increasing sequence of positive numbers with decreasing increments, that is, $0 < W^1 < W^2 < \ldots$ and $W^h - W^{h-1} < W^{h-1} - W^{h-2}$ for all $h \geq 2$. They also prove that W^h values minimize the pair-allocations vector by lexicographic order, which means that their selection of W^h values ensures the optimal solution for every problem instance.

2. **Consistency Constraints**

$$\sum_{k=1}^{K} \sum_{I \in F_k} C_{kp}^I . C_{kq}^I . x_k^I \geq \sum_{h=1}^{T_{pq}} h . t_{pq}^h, \quad \forall \{p, q\} \in H, \tag{3.2}$$

$$\sum_{h=1}^{T_{pq}} t_{pq}^h \leq 1, \quad \forall \{p, q\} \in H, \tag{3.3}$$

Equation (3.2) ensures that x_k^I and t_{pq}^h variables are consistent for each pair of proposals. On the left hand side, if a subset I is assigned to referee k, then the corresponding x_k^I variable takes value 1. The product of parameters $C_{kp}^I . C_{kq}^I$ is 1 if both proposals p and q are in the subset I and 0 otherwise. Hence, the product $C_{kp}^I . C_{kq}^I . x_k^I$ takes value 1, if subset I is assigned to referee k and contains both proposals p and q. If we sum these products over all referees k, $k = 1, \ldots, K$ and all $I \in F_k$, we obtain the number of times the proposal pair $\{p, q\}$ is reviewed. The right hand side of the equation is to obtain only one non-zero t_{pq}^h variable for each pair $\{p, q\}$, where h is equal to the number of times that the pair is reviewed.

Equation (3.3) is of technical nature. Observe that, equation (3.2) fails to ensure that there is only one non-zero t_{pq}^h variable for each pair of proposals. In situations where a pair is reviewed three times or more, more than one t_{pq}^h

variable may be equal to 1. For this reason, at most one of the binary variables t_{pq}^h can be equal to 1, for $h = 1, \ldots, T_{pq}$.

3. **Capacity Contraints**

$$\sum_{I \in F_k} x_k^I = 1, \quad k = 1, ..., K, \tag{3.4}$$

Equation (3.4) avoids a referee to be assigned to more than one subset of proposals of size u. This constraint satisfies the capacity requirement for each referee k, $k = 1, \ldots, K$.

4. **Integrality Constraints**

$$x_k^I \in \{0, 1\}, \quad k = 1, ..., K, \quad \forall I \in F_k, \tag{3.5}$$

$$t_{pq}^h \in \{0, 1\}, \quad \forall \{p, q\} \in H, \quad h = 1, ..., T_{pq}. \tag{3.6}$$

Equations (3.5) and (3.6) define the range of values for the decision variables.

3.2.2 Formulation 1

The formulation proposed by Cook *et al* [Cook et al., 2005] is computationally effective. However, the number of variables defined in this model is growing too fast as the problem size increases. In Cook's formulation, they define a variable for each subset of proposals of size u that referee k is qualified to review. Assume that the average number of proposals a referee is qualified to review is Y in a specific problem. The number of variables defined in this problem is proportional to $K.\binom{Y}{2}$. As K and Y grow, the number of variables increase exponentially. For this reason, Cook's formulation does not work for large problems due to shortage of memory.

Formulation 1 is an alternative way to solve the same problem adressed by Cook *et al.* In this formulation, number of variables is growing in proportion to $K.Y$ rather than $K.\binom{Y}{2}$. Below we define parameters, decision variables, and constraints.

We first present the parameters used for this formulation as follows:

K Number of referees,

N Number of proposals,

u Number of proposals a referee is willing to review,

W^h A weight associated with "level" h, $h = 1, ..., K$,

H The collection of all ordered pairs of proposals $\{p, q\}$, for which there exists at least one referee who is qualified to review both proposals, $p, q \in \{1, ..., N\}$, $(p < q)$,

P_k The collection of all pairs of proposals $\{p, q\}$ that referee k is qualified to review, $\{p, q\} \in H$,

T_{pq} The number of referees capable of reviewing the pair of proposals $\{p, q\}$, $\{p, q\} \in H$,

I_k The collection of all proposals that referee k is qualified to review, $k = 1, ..., K$.

We next define the decision variables:

y_{kp} A binary variable whose value is 1 if referee k reviews proposal p and 0 otherwise, $k = 1, ..., K$, $p \in I_k$

z_{kpq} A binary variable whose value is 1 if referee k reviews pair of proposals $\{p, q\}$ and 0 otherwise, $k = 1, ..., K$, $\{p, q\} \in P_k$,

t^h_{pq} A binary variable whose value is 1 if the number of referees that review the pair of proposals $\{p, q\}$ is exactly h and 0 otherwise. $\{p, q\} \in H$, $h = 1, ..., T_{pq}$.

Using the parameters and decision variables defined above, we formulate the following integer programming model:

$$max \quad \sum_{\{p,q\} \in H} \sum_{h=1}^{T_{pq}} W^h . t^h_{pq},$$

subject to

$$\sum_{k=1}^{K} z_{kpq} = \sum_{h=1}^{T_{pq}} h . t^h_{pq}, \quad \forall \{p, q\} \in H, \tag{3.7}$$

$$\sum_{h=1}^{T_{pq}} t_{pq}^h \leq 1, \quad \forall \{p,q\} \in H, \tag{3.8}$$

$$\sum_{p \in I_k} y_{kp} \leq u, \quad k = 1, ..., K, \tag{3.9}$$

$$\sum_{\{p,r\} \in P_k} z_{kpr} + \sum_{\{o,p\} \in P_k} z_{kop} = y_{kp}.(u-1), \quad k = 1, ..., K, \quad p \in I_k, \tag{3.10}$$

$$y_{kp} \in \{0,1\}, \quad k = 1, ..., K, \quad p \in I_k, \tag{3.11}$$

$$z_{kpq} \in \{0,1\}, \quad k = 1, ..., K, \quad \forall \{p,q\} \in P_k, \tag{3.12}$$

$$t_{pq}^h \in \{0,1\}, \quad \forall \{p,q\} \in H, \quad h = 1, ..., T_{pq}. \tag{3.13}$$

The objective function is the same as that of Cook's formulation. Next, we describe the constraints:

1. **Consistency Constraints**

$$\sum_{k=1}^{K} z_{kpq} = \sum_{h=1}^{T_{pq}} h.t_{pq}^h, \quad \forall \{p,q\} \in H \tag{3.7}$$

$$\sum_{h=1}^{T_{pq}} t_{pq}^h \leq 1, \quad \forall \{p,q\} \in H \tag{3.8}$$

Equation (3.7) is the first constraint which ensures that the z_{kpq} and t_{pq}^h variables are consistent for each pair of proposals. On the left hand side, the number of times that a pair of proposal is reviewed is computed by counting the number of referees who reviewed that pair. The right-hand side of the equation is to obtain only one non-zero t_{pq}^h variable for each pair of proposal, where h is equal to the number of times that the pair is reviewed.

Equation (3.8) is of technical nature. Observe that Equation (3.7) fails to ensure that there is only one non-zero t^h_{pq} variable for each pair of proposal $\{p, q\}$ in situations where the number of reviews is at least 3. For this reason, the sum of all t^h_{pq} must be at most 1 for each pair of proposals $\{p, q\}$.

2. **Capacity Constraints**

$$\sum_{p \in I_k} y_{kp} \leq u \quad k = 1, ..., K, \tag{3.9}$$

Each referee is willing to review a certain number of proposals. We call this number as the capacity of a referee and make sure that the number of proposals assigned to a referee does not exceed her capacity.

3. **Pairwise Consistency Constraints**

$$\sum_{\{p,r\} \in P_k} z_{kpr} + \sum_{\{o,p\} \in P_k} z_{kop} = y_{kp}.(u-1), \quad k = 1, ..., K, \quad p \in I_k, \tag{3.10}$$

Although we make assignments based on individual proposals, the comparisons are made on pairs of proposals. This constraint relates the assignment of a single proposal to a referee with the number of reviewed proposal pairs by her. The equation implies that if proposal p is assigned to referee k, then she will review $u-1$ different proposal pairs where one of the proposals in each pair is proposal p. Any other pair that includes proposal p is not going to be reviewed by this referee. On the other hand, if proposal p is not assigned to referee k, than any proposal pair including proposal p will not be reviewed by her.

4. **Integrality Constraints**

Equations (3.11), (3.12), and (3.13) define the range of values that each decision variable can take.

3.2.3 Formulation 2

The main difference between the Formulation 1 and Formulation 2 is that we remove the decision variable y_{kp}, $k = 1, \ldots, K$, and $p = 1, \ldots, N$, in Formulation 2. For this reason, the capacity and pairwise consistency constraints are expressed using only z_{kpq}, $k = 1, \ldots, K$ and $\{p, q\} \in P_k$ variables. As a result, the number of decision variables decreased and the number of constraints increased.

Using the parameters and decision variables (excluding y_{kp}, $k = 1, \ldots, K$, and $p = 1, \ldots, N$) defined in Section 3.2.2, we formulate the following mixed integer programming model, which we refer to as Formulation 2:

$$max \quad \sum_{\{p,q\}\in H} \sum_{h=1}^{T_{pq}} W^h.t_{pq}^h,$$

subject to

$$\sum_{k=1}^{K} z_{kpq} = \sum_{h=1}^{T_{pq}} h.t_{pq}^h, \quad \forall \{p, q\} \in H,$$

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

$$\sum_{\{p,q\}\in P_k} z_{kpq} = \binom{u}{2}, \quad k = 1, ..., K, \tag{3.14}$$

$$-1 + z_{kpq} + z_{kqr} \leq z_{kpr}, \quad k = 1, ..., K, \quad \forall \{p, q\}, \{q, r\}, \{p, r\} \in P_k, \tag{3.15}$$

$$(2z_{kpq} - 1).(u - 1) \leq \sum_{\{o,p\}\in P_k} z_{kop} + \sum_{\{p,r\}\in P_k} z_{kpr}, \quad k = 1, ..., K, \quad \forall \{p, q\} \in P_k, \tag{3.16}$$

$$(2z_{kpq} - 1).(u - 1) \leq \sum_{\{o,q\}\in P_k} z_{koq} + \sum_{\{q,r\}\in P_k} z_{kqr}, \quad k = 1, ..., K, \quad \forall \{p, q\} \in P_k, \tag{3.17}$$

$$z_{kpq} \in \{0,1\}, \qquad k = 1,...,K, \quad \forall \{p,q\} \in P_k, \tag{3.18}$$

$$t^h_{pq} \in \{0,1\}, \qquad \forall \{p,q\} \in H, \quad h = 1,...,T_{pq}. \tag{3.19}$$

The objective function and the consistency constraints are the same as those of Formulation 1. Next, we describe the modified versions of capacity and pairwise consistency constraints:

1. **Capacity Constraints**

$$\sum_{\{p,q\} \in P_k} z_{kpq} = \binom{u}{2}, \qquad k = 1,...,K, \tag{3.14}$$

$$-1 + z_{kpq} + z_{kqr} \le z_{kpr}, \qquad k = 1,...,K, \quad \forall \{p,q\}, \{q,r\}, \{p,r\} \in P_k, \tag{3.15}$$

In equation (3.14), we sum the z_{kpq} variables for each referee $k, k = 1, \ldots, K$, in order to obtain the number of pairs a referee reviews. Since this is a maximization problem, u proposals are assigned to each referee in any optimal solution. Therefore, the number of proposal pairs each referee reviews will be equal to $\binom{u}{2}$. However, this constraint is not enough to make sure that not more than u proposals are assigned to each referee.

We note that every subset of proposals of size u implies $\binom{u}{2}$ proposal pairs, however every $\binom{u}{2}$ proposal pairs do not necessarily imply u proposals. For example, a set of proposals $\{p, q, r\}$ constitutes 3 proposal pairs as $\{p, q\}, \{p, r\}$, and $\{q, r\}$ for every $\{p, q, r\}$. However, three proposal pairs $\{p, q\}, \{r, t\}$, and $\{u, v\}$ correspond to 6 different proposals. At this point, equation (3.14) may not be sufficient to satisfy capacity limitations. Equation (3.15) restricts the arbitrary selection of $\binom{u}{2}$ proposal pairs for a referee. This constraint forces the selection of proposal pairs such that u proposals are assigned to each referee.

Equation (3.15) ensures that if a referee reviews proposal pair $\{p, q\}$ and $\{q, r\}$, then he must also review $\{p, r\}$. For instance, if z_{kpq} and z_{kqr} are both 1, then the inequality forces z_{kpr} to be equal to 1. Otherwise, the inequality becomes redundant.

2. **Pairwise Consistency Constraints**

$$(2z_{kpq} - 1).(u - 1) \leq \sum_{\{o,p\} \in P_k} z_{kop} + \sum_{\{p,r\} \in P_k} z_{kpr} \qquad k = 1, ..., K, \quad \forall \{p, q\} \in P_k, \tag{3.16}$$

$$(2z_{kpq} - 1).(u - 1) \leq \sum_{\{o,q\} \in P_k} z_{koq} + \sum_{\{q,r\} \in P_k} z_{kqr} \qquad k = 1, ..., K, \quad \forall \{p, q\} \in P_k, \tag{3.17}$$

Equations (3.16) and (3.17) are the modified version of the pairwise consistency constraint in Formulation 1. This time, we formulate our constraints based on proposal pairs. If referee $k, k = 1, , K$, reviews proposal pair $\{p, q\}$, then proposals p and q are assigned to her. Thus, referee k is going to review $u - 1$ different proposal pairs that include proposal p and $u - 1$ different proposal pairs that include proposal q. If z_{kpq} is 1, then the right hand side of the constraints 5 and 6 force this requirement. Otherwise, these constraints become redundant.

3. **Integrality Constraints**

Equations (3.18) and (3.19) define the range of values that each decision variable can take.

Chapter 4

HEURISTIC ALGORITHMS

Our numerical experiments indicate that our optimization models can be solved to optimality for small to medium sized problem instances. For larger problem instances, we propose a heuristic model. In Section 4.1, we present the heuristic algorithm by Cook *et al.* In Section 4.2, we will introduce our heuristic algorithm.

4.1 Heuristic Algorithm by Cook et al

Cook *et al* [Cook et al., 2005] proposed a greedy algorithm for the optimization problem defined in Section 3. They define a priority for each pair of proposal, and at each step the pair with the highest priorty is assigned to the referee who has the largest reviewing capacity. The initial priorities for each pair of proposals are determined by T_{pq} - the larger the number of reviewers that can review pair, the smaller its priority is. After each assignment, the priority of each pair is updated according to the number of referees who have already been assigned to review that pair.

Before we introduce the definition of the weight and priority of pairs, we should first introduce two outcome measures:

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

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

These outcome measures are used to calculate the weights of proposal pairs at the beginning of every step. The initial value for the weight of proposal pair $\{p, q\}$ is T_{pq} and pairs with larger T_{pq} value have larger weights, which means there is a lower priority to assign them to reviewers. After the first assignment, the weights are updated according to the following formula:

$$W_{pq} = T_{pq}.(1 + 2.n_{pq}.(1 + 2.\sqrt{n_{p-q}.n_{q-p}})) \tag{4.1}$$

If the proposals in a pair have not been assigned to a referee, n_{pq} value remains 0 and the weight is not updated. Otherwise, n_{pq} becomes positive and the weight is increased. Additionally, if n_{pq} is positive, Cook *et al* also account the potential of a pair to be assigned to other referees. The weight of a pair $\{p, q\}$ is increased by the square root of the product $n_{p-q}.n_{q-p}$. For example, suppose that we have not assigned any proposals to referees yet and 4 referees are eligible to review proposal pair $\{p, q\}$. Since $T_{pq} = 4$, the weight W_{pq} is also 4 at the beginning. Then, we assign proposal pair $\{p, q\}$ 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 $W_{pq} = 44$.

Before we present the pseudocode of the algorithm, we should define a variable to keep the remaining capacity of each referee:

u_k Remaining capacity for each referee k, $k = 1, \ldots, K$.

The steps of the heuristic procedure are given below:

1. Prioritization of proposals

   (a) Compute for each pair $\{p, q\}$ a weight W_{pq}.

   $$W_{pq} = T_{pq}.(1 + 2.n_{pq}.(1 + 2.\sqrt{n_{p-q}.n_{p-q}}))$$

   (b) Order the pairs in a nondecreasing order of W_{pq}. Break ties arbitrarily.

2. Assignment of reviewers

   (a) Choose the first pair in the ordered list. If you encounter a pair for which there is no available referee, skip it.

(b) 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

   (a) Decrease the number of proposals that the selected referee can read by either one or two (depending on whether both proposals were assigned to her for the first time).

   (b) If there are no more available referees (i. e. $u_k = 0 \ \forall k$), stop.

   (c) Otherwise, return to Step 1.

4.2 Proposed Heuristic

To solve this problem, we have developed a construction and improvement heuristic. In the construction part, we build an initial solution in a small amount of time. In the improvement part, we take the initial solution and for each referee, we replace assigned proposals with unassigned ones to obtain new solutions with higher objective function values.

4.2.1 Construction Phase

The construction heuristic obtains a feasible solution through modifying the partial solution obtained from the LP relaxation of Formulation 1. The basic idea is to obtain a feasible solution to the original problem by rounding y_{kp} values to integer values (0 or 1). Because of the capacity constraints, we need to round up y_{kp} values of u proposals for each referee. The pseudocode of the construction phase is given below:

1. Get the LP optimum solution of Formulation 1 for the problem.

2. If the LP optimum solution is a feasible solution to the original problem (i.e. all y_{kp} values are integral), then stop. The LP optimal solution is optimal for Formulation 1. Otherwise, proceed to Step 3.

3. For each referee, sort the proposals in I_k in descending order of y_{kp} values. Break ties arbitrarily.

4. For each referee, assign the first u proposals in the sorted list (i.e. round up the corresponding y_{kp} values to 1 for the first u proposals. The rest of the y_{kp} values will be equal to 0.)

After we obtain the starting solution, we apply the improvement stage to obtain better solutions if possible.

4.2.2 Improvement Phase

In the initial solution, each referee $k, k = 1, \ldots, K$, is assigned to a subset of proposals of size u. This subset is denoted by A_k. On the other hand, we can also define another subset of proposals for each referee k, say U_k, which consists of the proposals she is qualified to review but are not assigned to her yet. Reader should note that the union of A_k and U_k gives I_k for all referees $k, k = 1, \ldots, K$.

In the improvement phase, we pick a referee k, $k = 1, \ldots, K$, and keep an index l, $l = 1, \ldots, u$, and calculate the objective functions of the solutions where the l^{th} proposal from the list A_k is replaced with every single proposal in the list U_k. If any one of these replacements yields a better objective function value, we apply the most improving one. The most improving proposal in the list U_k is added to the list A_k in the l^{th} order and the replaced proposal is added to the list U_k. Otherwise, we do not perform replacement for the l^{th} proposal in A_k. This procedure ends for a referee, when we repeat the procedure for the entire index l. After, we perform the process for the next referee. The algorithm stops if there does not exist any improving replacements for any referees. The pseudocode for improvement phase is given below:

1. $k = 1$, $l = 1$, $numOfImprovements = 0$

2. Calculate the objective function values of the solutions where the l^{th} proposal of A_k is replaced with every single proposal in U_k.

3. If there are any improving replacements, then apply the most promising one and perform the corresponding replacement in the lists A_k, and U_k. Also increase $numOfImprovements$ by 1.

4. If $l = u$, then increase k by 1 and set $l = 1$, else increase l by 1.

5. If $k > K$ and $numOfImprovements$=0, stop. If $k > K$ and $numOfImprovem. > 0$, proceed to Step 1. Otherwise, proceed to Step 2.

In the following chapter, we perform experiments to test the efficiency of the previously proposed methods and methods we present in this thesis.

Chapter 5

COMPUTATIONAL STUDY

We conducted computational experiments in an attempt to assess the effectiveness of our exact and heuristic approaches. In this chapter, we present the results of our computational experiments. Furthermore, we compare our solution approaches with those proposed by Cook *et al* [Cook et al., 2005].

We used OPL 12.4 and Java in the implementation of the solution approaches. Moreover, we used CPLEX 10 as the integer linear programming solver. All computations were performed on a computer that has 4 GB DDR2 RAM, Intel Core i5-2520M 2.50 GHz processor, and 64-bit Windows 7 operating system.

In Section 5.1, we give the detailed explanation of the stages of computational study of mathematical models. Details of computational study of heuristic models is given in Section 5.2. Numerical results and discussions are given in Section 5.3.

5.1 Computational Study of Mathematical Models

In this thesis, we proposed two mathematical formulations and a heuristic approach and conducted computational experiments to test the effectiveness of our methods. We also compared our methods with the mathematical model and heuristic model approach proposed by Cook *et al.* In order to make a fair analysis, sufficient number of test problems are generated, solved, and reported. For convenience, we implemented an interface to handle all stages of the computational study. The flow chart for our computational experiments is given in Figure 5.1.

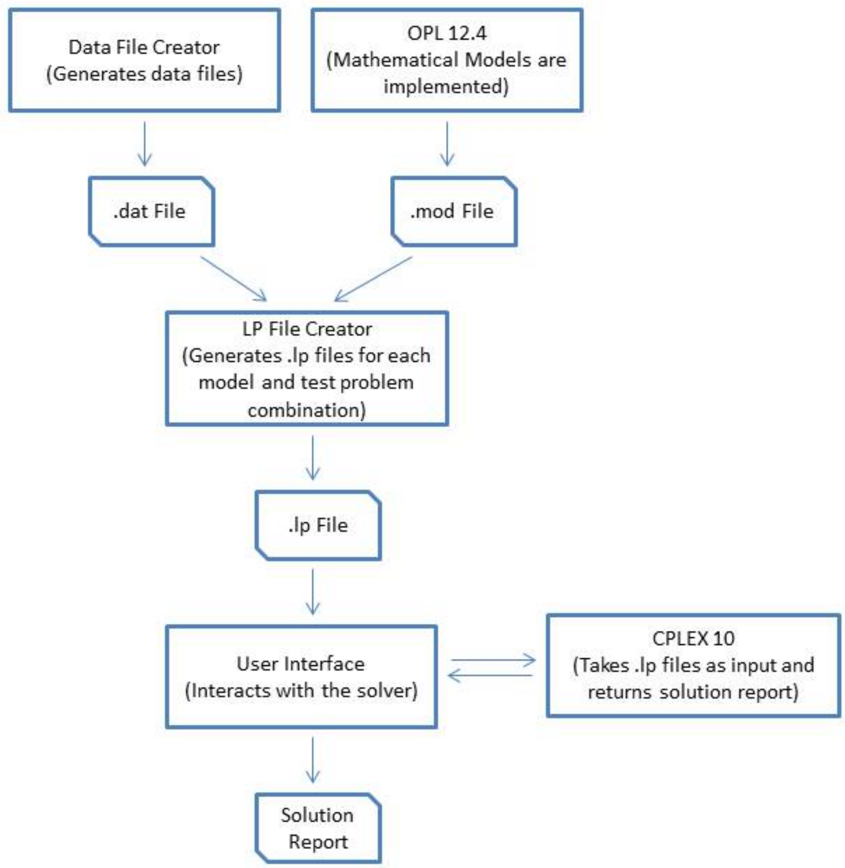

Figure 5.1: Flow Chart of Computational Setup

5.1.1 Generation of Data Files

We perform our computational experiments on ten classes of test problems given in Table 5.1. For each class, we randomly generated ten test problems. Each problem instance consists of number of referees K, number of proposals N, a fixed referee capacity u, and a proficiency matrix. Proficiency matrix consists of A_{kp} parameters for each referee k, $k = 1, \ldots, K$, and each proposal p, $p = 1, \ldots N$. If referee k is eligible to review proposal p, binary parameter A_{kp} gets value 1, and 0 otherwise. While randomly generating data files, a Bernoulli random variable with probability P is used to determine the value of A_{kp} for each referee k, $k = 1, \ldots, K$, and proposal

p, $p = 1, \ldots, N$.

Class	No. of Referees	No. of Proposals	k	P(A_{kp}=1)
A1	40	20	3	0.4
A2	40	20	4	0.4
B1	50	20	3	0.3
B2	50	20	4	0.3
C1	60	30	3	0.3
C2	60	30	4	0.3
D1	80	40	3	0.25
D2	80	40	4	0.25
E1	200	100	3	0.1
E2	200	100	4	0.1

Table 5.1: Ten Classes of Test Problems

A sample data file with 10 referees, and 15 proposals is given in Figure 5.2. In this problem instance, each referee has a capacity of reviewing 3 proposals. The proficiency matrix named A is generated by a Bernoulli random variable with probability 0.4 in this case.

```
m=10; n=15; k=3;
A=[ [1, 0, 1, 0, 1, 0, 1, 1, 0, 1, 1, 0, 0, 0, 1]
    [0, 0, 1, 0, 1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0]
    [1, 1, 0, 0, 0, 1, 1, 0, 0, 0, 1, 0, 0, 1, 0]
    [1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 1, 1]
    [1, 0, 0, 0, 1, 1, 0, 1, 0, 1, 1, 0, 1, 0, 0]
    [1, 1, 1, 0, 1, 0, 0, 1, 0, 0, 0, 1, 1, 0, 1]
    [0, 0, 0, 1, 1, 0, 1, 0, 1, 1, 1, 0, 0, 1, 1]
    [0, 0, 0, 1, 0, 0, 1, 0, 1, 1, 0, 0, 1, 0, 0]
    [1, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0]
    [0, 1, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 1] ];
```

Figure 5.2: Sample Data File

Data files are generated by a computer program implemented in Java. The program generated ten test problems (*.dat) for each class of problems. Math.random() method is used to simulate the Bernoulli random variable and a control flow is included in the code to make sure that every proposal can be reviewed by at least one referee in any problem instance.

5.1.2 Implementation of Mathematical Formulations

In this study, OPL 12.4 is used to implement three mathematical formulations including Formulation 1, Formulation 2, and Formulation of Cook *et al.* For each formulation, a model file (**.mod*) is generated, which is used to create LP files explained in Section 5.1.3.

During the implementation, a simple procedure helped us to reduce the number of decision variables defined in our model, if possible. This procedure is based on the observation that in any optimal solution, the capacities of all referees are exhausted. At this point, it is easy to see that if referee k is eligible to review a proposals and a is less than or equal to the capacity u, than there is no need to define a decision variable for referee k. It is obvious that all of the proposals referee k is eligible to review are assigned to her in any optimal solution.

5.1.3 Generation of LP Files

As we stated before, optimization problems are solved with CPLEX 10, which is one of the most powerful commercial solvers. There are two common file formats provided as an input for CPLEX solver: MPS and LP. In this study, OPL 12.4 is used in the implementation of mathematical models, and a Java program is coded to generate LP files for each problem instance using classes and methods of ilog.cplex package, which is a package including classes and methods for a Java project to interact with CPLEX solver. This code takes a .mod and a .dat file as input and generates an .lp file. In our case, there are three different mathematical models and 100 test problems from 10 different classes. Therefore, using this code, 300 .lp files are generated to be solved by CPLEX.

5.1.4 Interface

The last step of the computational study was to solve each .lp file using CPLEX. For convenience, we developed an interface to easily manage the computational experiments. Using this interface, we can randomly generate data files, and create .lp files for a specified data file ($*.dat$) and mathematical model ($*.mod$). Also, we can solve .lp files with this interface. After solving the problem, program outputs a solution report including details.

5.2 Implementation of Heuristic Algorithms

The main objective of this study is to get near-optimal solutions for large-scale problems in a reasonable amount of time. For this reason, we developed a heuristic algorithm, which is presented in Section 4.2.

We implemented proposed heuristic model in Java in order to conduct numerical experiments. The program takes a data file as input and at the first step, it calls OPL 12.4 to solve the problem with the linear programming relaxation of Formulation 1. Then, we construct a starting solution using the LP optimal solution following the procedure in the construction phase presented in Section 4.1.

After constructing a starting solution, we generate two lists: A_k and U_k for each referee k. A_k contains the proposals assigned to referee k in the starting solution, and U_k contains the proposals referee k is qualified to review but were not assigned to her yet. In the improvement phase, we pick a referee k, $k = 1, \ldots, K$, and keep an index $l, l = 1, \ldots, u$, and calculate the objective function value of the solutions where l^{th} proposal from the list A_k is replaced with every single proposal in the list U_k. If there are any replacements which yield a better objective function value, we apply the most promising one. Therefore, the most improving proposal in the list U_k is added to A_k in the l^{th} order and the removed proposal is added to list U_k. This procedure ends for a referee, when we repeat the procedure for entire index l. After, we perform the same operation for the next referee. The algorithm stops when there does not exist any improving replacements for any referees.

We also implemented the heuristic approach by Cook *et al.* The results of the computations are given in the following section.

5.3 Numerical Results and Discussions

In this section, we present some numerical experiments to demonstrate the performances of alternative solution approaches. As we stated before, we defined 10 classes of problems of different sizes. Ten test problems are created for each class and we solve these problems with three mathematical formulations and two heuristic algorithms. In total, we performed 500 runs in an attempt to compare the methods by Cook *et al* and the alternative methods proposed in this thesis. In Table 5.1, number of referees, number of proposals, capacities, and the probability of Bernoulli random variable that we use for generating the proficiency matrix is presented for each class of problems.

	Cook's Formulation		**Formulation 1**		**Formulation 2**	
Class	Time (sec.)	Gap (%)	Time (sec.)	Gap (%)	Time (sec.)	Gap (%)
A1	0,85	0,00	2,83	0,00	25,75	0,00
A2	223,27	0,00	1.778,92	0,04	1.801,41	0,47
B1	1,27	0,00	11,49	0,00	28,76	0,00
B2	7,29	0,00	1.278,93	0,02	1.273,63	0,01
C1	3,18	0,00	5,09	0,00	39,07	0,00
C2	1.825,07	0,17	1.930,01	0,11	1.800,04	1,16
D1	3,83	0,00	4,88	0,00	40,18	0,00
D2	81,93	0,00	1.804.19	0,01	1.698,14	0,28
E1	10,33	0,00	9,57	0,00	79,31	0,00
E2	32,81	0,00	468,32	0,00	1.800,57	0,12

Table 5.2: Average Solution Times and Optimality Gaps

For each class of problems, we solved 10 test problem instances and average solution times and gaps are presented in Table 5.2. The time limit for each run is 1.800 seconds.

Numerical results show that the formulation proposed by Cook *et al* is very applicable in solving small to medium size problems. However, as the problem size increases, because of the exponential growth of number of variables, we can not solve the problem with this approach due to shortage of memory. Assume that average number of proposals a referee is qualified to review is Y in a specific problem. For each referee, there are $\binom{Y}{u}$ different subsets of proposals and the number of variables is proportional to $K.\binom{Y}{u}$. As K and Y grow larger, the number of variables increase exponentially. However, in Formulation 1, the number of variables is growing proportional to $K.Y$. Table 5.3 presents the average number of variables and constraints for each class of problems and each formulation. As we can see, the number of variables in Cook's formulation is growing much faster than the proposed alternatives.

	Cook's Formulation		**Formulation 1**		**Formulation 2**	
Class	# of Var.	# of Cons.	# of Var.	# of Cons.	# of Var.	# of Cons.
A1	4.391	420	2.973	741	2.652	8.310
A2	6.287	419	2.904	731	2.613	8.084
B1	2.535	422	2.130	701	1.852	5.222
B2	2.922	414	2.044	666	1.806	5.008
C1	9.918	928	5.919	1.482	5.365	17.807
C2	13.538	927	5.389	1.453	4.963	16.190
D1	17.378	1.635	9.508	2.441	8.702	30.121
D2	35.859	1.629	9.548	2.429	8.967	31.382
E1	45.920	8.715	25.796	10.693	23.818	79.703
E2	89.231	8.725	25.962	10.746	23.987	80.408

Table 5.3: Average Number of Variables and Constraints

In order to test the performances of heuristic algorithms, we solved the same problem instances with Cook's and our heuristics. The average optimality gaps and solution times are provided in Table 5.4. The gaps are calculated taking the optimal value as upper bound for the problems solved to optimality. For the rest of the problems, we take the objective value of the lp optimal solution obtained by Formulation 1 as the upper bound. Results show that the proposed heuristic method outperforms Cook *et al*'s heuristis.

	Cook's Heuristic		**Proposed Heuristic**	
Class	Gap (%)	Time (sec.)	Gap (%)	Time (sec.)
A1	0,28	0,15	0,01	1,33
A2	0,45	0,08	0,11	0,36
B1	0,23	0,16	0,09	0,47
B2	0,24	0,08	0,10	0,28
C1	0,09	0,45	0,00	0,97
C2	0,36	0,39	0,11	0,88
D1	0,04	0,74	0,00	1,39
D2	0,20	0,58	0,02	1,46
E1	0,01	26,85	0,00	10,40
E2	0,03	38,87	0,00	8,57

Table 5.4: Average Gap and Solution Times

In the following chapter, we give the concluding remarks and research directions.

Chapter 6

CONCLUSION AND FUTURE RESEARCH

Peer review is the evaluation of a work or performance by a group of people in a related profession. It is used for many purposes such as scholarly evaluation of scientific publications, funding of research proposals, accreditation, promotion, and certification.

In this thesis, we focus on the application of peer review for funding of research proposals. Today, a large portion of the academic research is sponsored through funding agencies and organizations. The funding process starts with a call for proposals. Researchers who seek financial support submit their research proposals for evaluation. These proposals are collected by the funding organization and then distributed to a group of referees for peer review. Then, the individual evaluation of each referee is collected and aggregated into a single overall ranking.

In this project, we focus on the process of proposal evaluation by referees. The scope of this project is to develop solution methodologies to assign proposals to referees so as to maximize a certain set of criteria subject to a set of constraints.

In this thesis, we proposed two alternative integer linear programming models referred to as Formulation 1 and Formulation 2. For larger problems, we designed a heuristic approach based on the solution of the LP relaxation of Formulation 1.

Although mathematical formulations fall behind of that of Cook *et al* in terms of computational efficiency, for large problems, the proposed heuristic performs much better than the previously proposed heuristic approach.

A further research may be conducted on the panel based version of the studied problem. In this version, referees are divided into groups referred to as panels. Before assigning proposals to referees, we should first assign proposals to panels. After, the proposals are assigned to be reviewed by the referees in the related panel. In order to easily aggregate the final rankings obtained from each panel, we should assign proposals to panels such that maximum number of overlaps occur. Also we should try to assign the proposals to the referees in the related panel with the same objective with the problem studied in this thesis.

BIBLIOGRAPHY

[Benferhat and Lang, 2001] Benferhat, S. and Lang, J. (2001). Conference paper assignment. *International Journal of Intelligent Systems*, 16:1183 – 1192.

[Bogart, 1975] Bogart, K. P. (1975). Preference structures ii: Distances between asymmetric relations. *SIAM Journal on Applied Mathematics*, 29(2):254 – 262.

[Campanario, 1998] Campanario, J. M. (1998). Peer review for journals as it stands today. *Science Community*, 19:181 – 211.

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

[Cole et al., 1981] Cole, S., Cole, J. R., and Simon, G. A. (1981). Chance and consensus in peer review. *Science*, 214(4523):881 – 886.

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

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

[Cook et al., 2005] Cook, W. D., Golany, B., Kress, M., and Penn, M. (2005). Optimal allocation of proposals to reviewers to facilitate effective ranking. *Management Science*, 51(4):655 – 661.

[Hodgson, 1995] Hodgson, C. (1995). Evaluation of cardiovascular grant-in-aid applications by peer review-influence of internal and external reviewers and committees. *Canadian Journal of Cardiology*, 11(10):864 – 868.

[Janak et al., 2006] Janak, S. L., Taylor, M. S., Floudas, C., Burka, M., and Mountzizris, T. (2006). Novel and effective integer optimization approach for the NSF panel-assignment problem: A multiresource and preference-constrained generalized assignment problem. *Industrial Engineering Chemistry Research*, 45:258 – 265.

[Jayasinghe et al., 2001] Jayasinghe, U. W., Marsh, H. W., and Bond, N. (2001). Peer review in the funding of research in higher education: The Australian experience. *Educational Evaluation and Policy Analysis*, 23(4):343 – 364.

[Kemeny and Snell, 1962] Kemeny, J. G. and Snell, L. J. (1962). *Preference Rankings: An Axiomatic Approach*. Mathematical Models in the Social Sciences, Boston.

[Kirkwood and Sarin, 1985] Kirkwood, C. W. and Sarin, R. K. (1985). Ranking with partial information - a method and an application. *Operations Research*, 33(1):38 – 48.

[Merelo-Guervos et al., 2003] Merelo-Guervos, J., Garcia-Castellano, J., Castillo, P., and Arenas, M. (2003). How evolutionary computation and Perl saved my conference. *Algoritmos Evolutivos y Bioinspirados*, pages 93 – 99.

[National Science Board, 2012] National Science Board (2012). *Science and Engineering Indicators 2012*. National Science Foundation, Arlington VA.

[Schirrer et al., 2007] Schirrer, A., Doemer, K. F., and Hartl, R. (2007). Reviewer assignment for scientific article using memetic algorithms. *OR CS Interfaces Series*, 39:113 – 134.

[Spier, 2002] Spier, R. (2002). The history of the peer review process. *Trends in Biotechnology*, 20(8):357 – 358.

[Sun et al., 2007] Sun, Y. H., Ma, J., Fan, Z. P., and Wang, J. (2007). A hybrid knowledge and model approach for reviewer assignment. In 40^{th} *Annual Hawaii International Conference on System Sciences*.

[Tian et al., 2002] Tian, Q., Ma, J., and Liu, O. (2002). A hybrid knowledge and model system for research and development project selection. *Expert Systems with Applications*, 23:265 – 271.