The 13th Mathematical Contest in Modeling

The Well-Mixed Assignments

Advisor: *Liu Baoxiang*

Students: *Lu Zhenyu* *Zhang Hongcheng* *Lu Huaijing*

Hebei Institute of Technology
1997.2.10

Summary

An Tostal Corporation wants a list of meeting assignments which should mix the board member as well as possible. The ideal assignments should satisfy the following criteria:

●Common membership of groups for the different sessions should be minimized.

●Each board member should have the same number of times with each other board member in a discussion group.

●For the morning sessions, no board member should be in the same senior officer's discussion group twice.

●No discussion group should contain a disproportionate number of in-house members.

In order to make each board member mixed well, we use the compound satisfaction of four criteria as the basis of judging. Through analogy, we introduce "**Exclusive Force**" of physics into the model. Combining all the criteria together, we describe the "**Exclusive Forces**" through **Multiple Criteria Function**. And then establish the algorithm that the secretary could use to adjust the assignments.

According to our model, we make a computer program and get the satisfactory assignment, which meet the criteria as follows:

●The maximum of common memberships is 3.

●The times of any two board members assigned in a discussion group is no more than 3, and mostly one or two.

●No board member is assigned to the same senior officer's discussion group twice.

●The proportion of in-house members in each group are all close to the ideal proportion: 9/29. Among the total 34 groups in 7 sessions, 20 times the proportion of in-house member is 33.33%, and 10 times of 28.57%, and each for 2 time of 40% and 25%.

●Besides, our assignment can also meet one criterion not given,

that is the board members led by each senior officer are almost the same in number.

●To those that some board members will cancel at the last minute or some not scheduled will show up, we only need to make some small changes to our computer program, and then can get the satisfactory solution.

The Well-Mixed Assignments

Abstract

To make the board members mixed well , we use the compound satisfaction of four criteria as the basis of judging. Through analogy , we introduce "**Exclusive Force**" of physics into the model. Combining all the criteria together ,we describe the "**Exclusive Force**" through **Multiple Criteria Function** . And then establish an algorithm that the secretary could use to adjust the assignments. The results of our computer program and the analysis of the stability will further prove the algorithm reasonable。

In accordance with those cases that "some board members will cancel at the last minute or that some not scheduled will show up" and " meetings involving different levels of participation for each type of attendees", we have revised the computer program, and obtained a satisfactory solution.

Key words

Exclusive Force **Multiple Criteria Decision**
Weighting Coefficient **Current State**

Introduction

It is believed that large groups discourage productive discussion and that a dominant personality will usually control and direct the discussion. To avoid this phenomenon it is common to schedule several sessions with a different mix of people in each group. Thus, attendees should be mixed as well as possible.

A meeting of An Tostal Corporation will be attended by 29 Board Members of which nine are In-house members. The meeting is to be an all-day affair with three sessions scheduled for the morning and four for the afternoon. Each morning session will consist of six discussion groups with each discussion group led by one of the corporation's six senior officers. None of these officers are board members. The senior officers will not be involved in the afternoon sessions and each of these sessions will consist of only four different discussion groups.

The president of the corporation wants a list of board-member assignments to discussion groups for each of the seven sessions. The assignments should achieve as much of a mix of the members as possible. The ideal assignment should satisfy the following criteria:

●Common membership of groups for the different sessions should be minimized.

●Each board member should have the same number of times with each other board member in a discussion group.

●For the morning sessions, no board member should be in the same senior officer’s discussion group twice.

●No discussion group should contain a disproportionate number of in-house members.

Give a list of assignments for members 1-9 and 10-29 and officers 1-6. Indicate how well the criteria in the previous paragraphs are met. Since it is possible that some board members will cancel at the last minute or that some not scheduled will show up, an algorithm that the secretary could use to adjust the assignments with an hour’s notice would be appreciated. It would be ideal if the algorithm could also be used to make assignments for future meetings involving different levels of participation for each type of attendees.

Assumptions

●Whether a discussion would be controlled by a dominant personality depends only on how well the attendees are mixed.

●When drafting the assignments, we need only to consider how to mix well and need not to consider the board members’ willingness.

●Types of attendees may be different, but the attendees of the same type should have the same working ability.

●To ensure the discussions be fruitful, every group should contain each type of attendees.

Analysis of the Problem

1. Quantification of the Criteria

The original problem statement gives four criteria to judge whether the assignment is suitable. Before solving the problem, we quantify the criteria first.

●Common Membership of Groups refers to the number of common members of two groups for different sessions. For a certain group, the common membership is different when compared with different groups. Here, we call the largest common membership as “**common membership of the group**”; and among all the groups in different sessions, the largest “common membership of group” as “**maximum of common memberships**”. The maximum of common memberships and the times it appears can be used to describe how well this criterion is met.

●If $x_i^{[j]}$ is defined as the number of board members in group *i* of session *j*. When the criterion: “Each board member should have the same number of times with each other board member in a discussion group” is met, we have the following formula:

$$\sum_{j=1}^{3}\sum_{i=1}^{6}(x_i^{[j]})^2+\sum_{j=4}^{7}\sum_{i=1}^{4}(x_i^{[j]})^2=29(28s+7) \qquad 〖1〗$$

(where s stands for the number of times any two board members assigned in a discussion group, which we call “ **meeting times** ”)

This is the necessary condition of this criterion .

Proof :

$\because$ In group *i* of session j , the membership anyone meets in the group is: $(x_i^{[j]}-1)$, so the total meeting times for $x_i^{[j]}$ board members in group i is : $x_i^{[j]}(x_i^{[j]}-1)$.

$\therefore$ In an all-day affair, the total meeting times for all the members is:

$$\sum_{j=1}^{3}\sum_{i=1}^{6}x_i^{[j]}(x_i^{[j]}-1)+\sum_{j=4}^{7}\sum_{i=1}^{4}x_i^{[j]}(x_i^{[j]}-1)$$
$$=\sum_{j=1}^{3}\sum_{i=1}^{6}(x_i^{[j]})^2+\sum_{j=4}^{7}\sum_{i=1}^{4}(x_i^{[j]})^2-29\times 7$$

If any two members are assigned in the same group for s times, the total meeting times can also be: $29\times 28s$.

Thus we obtain:

$$\sum_{j=1}^{3}\sum_{i=1}^{6}(x_i^{[j]})^2+\sum_{j=4}^{7}\sum_{i=1}^{4}(x_i^{[j]})^2-29\times7=29\times28s$$

which reduces to :

$$\sum_{j=1}^{3}\sum_{i=1}^{6}(x_i^{[j]})^2+\sum_{j=4}^{7}\sum_{i=1}^{4}(x_i^{[j]})^2=29(28s+7)$$

●To " no board member should be in the same senior officer's discussion group twice", we quantify the criterion as following:

Let R_{ij} be the number of times that member i appears in the senior officer j 's discussion group, we have:

$$\max\{R_{i1},R_{i2},R_{i3},R_{i4},R_{i5},R_{i6}\}=1 \quad , \quad \sum_{j=1}^{6}R_{ij}=3$$

$$(i=1,2,\cdots\cdots29)$$

●The proportion of in-house members in group i of session j is:

$$\frac{I_i^{[j]}}{x_i^{[j]}}=\frac{\textit{the number of in-house members in group } i \textit{ of session } j}{\textit{the total number of board members in group } i \textit{ of session } j}$$

$$\left(\textit{where.} I_i^{[j]} \textit{ represants the number of in-house members in group } i \textit{ of session } j\right)$$

For An Tostal Corporation, the most ideal proportion is 9/29. But it is impossible to reach. So the best proportion we can get should be make:

$$\min\sum_i\left(\frac{I_i^{[j]}}{x_i^{[j]}}-\frac{9}{29}\right)^2 \qquad 〖2〗$$

satisfied . Here, we should make sure that each group has at least one in-house member: this is the essential requirement for "proper proportion" (See **Assumption**) .The following **Table 1** gives us all the possible cases in which the in-house members can be combined . It also shows us the best number in each group according to the formula 〖2〗.

Table 1.

	Assignment of in-house members	$\min\sum_i\left(\frac{I_i^{[j]}}{x_i^{[j]}}-\frac{9}{29}\right)^2$

Morning	1 1 1 2 2 2 1 1 1 1 2 3 1 1 1 1 1 4	3 3 3 7 7 6 3 3 3 3 7 10 4 3 3 3 3 13
Afternoon	1 1 1 6 1 1 2 5 1 1 3 4 1 2 2 4 1 3 2 3 2 2 2 3	3 3 3 20 3 3 7 16 3 3 10 13 3 6 7 13 3 10 6 10 7 6 6 10

2. Does An Absolutely Ideal Assignment Exist?

We consider that an absolutely ideal assignment should strictly satisfy four criteria at the same time. Thus, from the quantified criteria, it can be seen that formula 〖1〗 and 〖2〗 should be tenable at the same time. Using *Cauchy inequality* ,after derivation, we found that when formula 〖1〗 and 〖2〗 are strictly satisfied, the maximum of common memberships is at least 11 【See **Appendix A**】.Thus, the "absolutely ideal" assignment is not the ideal assignment that people want to get. So:

The absolutely ideal assignment that strictly satisfies the four criteria does not exist.

3. Further Analysis of the Problem

According to the previous analysis, it is impossible to strictly satisfied the four criteria at the same time. Then , can we only manage to strictly satisfy one criterion? During the argument of whether an absolutely ideal assignment exists, it is found that there are some internal and delicate relationships among the four criteria. If only one criterion used exclusively, the other criteria may be badly met. Therefore, we should synthetically consider the four criteria, looking for an algorithm satisfying the four criteria as well as possible . From the above analysis, we design the model as follows.

Design of the Model

1. Thoughts of the Design

It is known from the above analysis that the absolutely ideal assignment does not exist, and only strictly satisfy one criterion is also unreasonable. So, to find an algorithm which can synthetically meet the needs of all the four criteria is vital.

Generally speaking, when people arrange a certain session of the meeting, they always assign the board member according to the previous sessions. So we design our model according to the following two aspects:

●Assuming that $j-1$ sessions have already been assigned, when arranging session j, we should first queue all the members according to how many times they met each other during the previous sessions. So we can assign the members according to this sequence.

This is because if we assign those who meet others less, after they have been assigned to the groups , it is very difficult to assign those who meet others more. Wherever they are assigned , they will meet the people they have already met many times. So , to avoid this , we should consider those first who meet the others for more times, We call it “**priority queue of members**”.

●After the priority queue, the members can then be assigned in turn. Assuming that there are already $i-1$ members have been assigned, because the result is different when members i is assigned into different groups. And whether the result is good depends on how the four criteria are met. So, comparing how well the four criteria are met, we can decide which group member i should be assigned into. Because all these four criteria are limiting criteria, analogized from “**exclusive force**” in physics, each criterion can be seen as an exclusive effect. The strength of this exclusive effect can be regarded as the strength of the exclusive force and the compound satisfaction can be seen as the compound force, if the compound satisfaction of the four criteria is used to describe how well the mix is . Thus , we can arrange the member according to the exclusive from each groups . To a certain member, the group that he is arranged into must be the group with the minimum exclusive force.

These two aspects are described in mathematical formula as shown in following:

2. Realization of the Thoughts

（1）A Rule for Priority queue of members

Since there are only two types of members: in-house and non-in-house members, It is necessary to assign the in-house members preferentially (which we will address the reason later). To those members with the same type, we should consider those who have met others for many times first. Usually we can describe this through the sum of “meeting times” that one board member meets other board members.

Assuming that *m* is the total number of board members. If we use M_{ij} to describe the meeting times between member *i* and member *j* in the previous sessions, P_i to describe the meeting times between member *i* and other $m-1$ members.

Thus: $$P_i = \sum_{\substack{j=1 \\ j\neq i}}^{m} M_{ij} \qquad (M_{ij} = M_{ji},\ i, j = 1,2,\cdots\cdots,m.)$$

But there may be cases: for example , if A and B represent two members with the same type, their meeting times with other four members are:

A: (1,1,1,1)

B: (3,1,0,0)

If we use the formula $P_i = \sum_{\substack{j=1 \\ j\neq i}}^{m} M_{ij}$ to queue A and B , they have the same priority. But to reach the criterion that “each board member should meet each other board member the same times”, member B should be prior to member A. This is because member B has less desirable choice on which group can be assigned to than A. The different results caused by using different methods (above formula and our prospect) is because :

To members of the same type , sorting them according to the value of the above P_i can only reflect the difference in the meeting times of the total, but can not reflect the difference in the meeting times of the single. So we modify the representation of P_i as following:

$$P_i = \sum_{\substack{j=1 \\ j\neq i}}^{m} (M_{ij})^k \qquad (k\in N) \qquad 〖3〗$$

Here, the $(M_{ij})^k$ can indicate the difference in both the total number and the single. The difference in meeting times of the single is now magnified by $(M_{ij})^k$ 。 The larger value *k* is , the more the difference in single meeting times is magnified.

At the same time, $\sum_{\substack{j=1 \\ j \neq i}}^{m} (M_{ij})^k$ can also reflect the difference in total meeting times. Thus , after this modify, representation P_i can reflect both the sensitivity on different meeting times and the over-all aspect of total meeting times. Generally speaking, if $2 \le k \le 4$, the difference in the single can be fully reflected . Here, let k = 2. Based on the previous paragraphs, we get the rules for priority queue of members as:

Firstly, calculate value of $P_i = \sum_{\substack{j=1 \\ j \neq i}}^{m} (M_{ij})^k$ (i=1,2,⋯,m), and then, queue all the members according to their P_i value. The larger one member's P_i value is , the prior he is arranged.

（2）Formula of Exclusive Force:

Now there is a priority queue of members.Before member i is assigned, each component exclusive force against him from each group must be caculated first. In the following, we developed four formulas of component exclusive force for each criterion.

●component of exclusive force from the meeting times

Here , the component of exclusive force is caused by the meeting times between member i and those members in group j . Similar to Eq.〖3〗, we develop a formula of $F_{j1}^{(i)}$, which describes the component of exclusive force from the meeting times:

$F_{j1}^{[i]} = \sum_{l} (M_{il})^k$ (where l represents each member's code number in group j .k∈N) ....〖4〗

Differing from priority, single meeting times should be attached greater importance here. That is , we should magnify disparity between two component exclusive forces caused by two different meeting times. Thus we can reach the criterion "meet each other the same times" as possible as we can. So k of Eq. 〖4〗

should be larger than k of Eq. 〖3〗,where k=4.

●component of exclusive force from the common membership of group

Let $C_j^{[i]}$ be the common membership of group j . Assuming member i is assigned to group j , we develop a formula of $F_{j2}^{[i]}$ to describe the component of exclusive force from the common membership of group j.

$$F_{j2}^{[i]} = (C_j^{[i]})^k \qquad (k = 1,2,3,\cdots\cdots) \qquad 〖5〗$$

Considering the common membership of group j should be as small as possible, we select $\left(C_j^{[i]}\right)^k$ to improve the formula's sensitivity. Thus , a single common member's increase can cause the exclusive force rise remarkably. Through many times of calculation, we consider that k=2 is desirable.

●component of exclusive force from the number of officers

Assuming that the number of groups led by officer j always numbered j , we use $F_{j3}^{[i]}$ to describe the component of exclusive force given by member i in group j , thus:

$$F_{j3}^{[i]} = \begin{cases} 0 & \textit{member i never led by officer j} \\ 1 & \textit{member i was once led by officer j} \end{cases} \qquad 〖6〗$$

●component of exclusive force from the proportion of in house members.

To make sure that the proportion of in-house members be suitable, we decide to assign the in-house members first when giving the assignment. Since the number of in-house members is only a few, the alteration of a single in-house members of a group will make the proportion alter a lot. So, we'd better arrange in-house members first.

When the meeting is arranged , if all the participants are of the same type, we needn't consider the suitability of the proportion, we just need to make the assignment as equal as we can. Thus, members in the group will give an exclusive effect to the new member, which can be described with a component exclusive force as:

$$F_{j4}^{[i]} = N_j^{[i]}$$

(where $N_j^{[i]}$ stands for the number of board members in group *j* before member *i* is added in)

Under this component of exclusive force, member *i* is apt to be added into the group with less members. When we assign in-house members, we just use this formula.

After the in-house members being assigned in using the above method, the non-in-house members can then be assigned according to the number of in-house members in each group. Assuming that there are I_j in-house members in group *j*. thus $\frac{I_j}{N_j^{[i]}+1}$ is the proportion of in-house members after member *i* is assigned into group *j* , to make the proportion suitable, we should let this value as close as $\frac{9}{29}$. Here, we use $\left(\frac{9}{29}\middle/\frac{I_j}{N_j^{[i]}+1}\right)$ to describe the suitable degree of the in-house members' proportion of group *j* , The larger this value is , the more difficult it is to assign in the member *i* . So this value can be used to describe the component exclusive force from the proportion of in-house members.

$$F_{j4}^{[i]} = \frac{9(N_j^{[i]}+1)}{29I_j}$$

Sum up:

$$F_{j4}^{[i]} = \begin{cases} N_j^{[i]} & \textit{for in-house member} \\ \dfrac{9(N_j^{[i]}+1)}{29I_j} & \textit{for non-in-house member} \end{cases} \qquad 〖7〗$$

Having described all the four component exclusive forces separately. We are then going to compound these forces. Because these component forces all have obstructive factors to member *i* being assigned, analogizing the composition of forces in one line, we can consider to add them together . But the dimensions of these component exclusive forces are different, so we should normalize them first.

Assuming that there are altogether *n* groups , thus , after normalization , the numbered l component exclusive force $f_{jl}^{[i]}$ to member i from group j is:

$$f_{jl}^{[i]} = \frac{F_{jl}^{[i]}}{\sum_{k=1}^{n} F_{kl}^{[i]}} \qquad (l = 1,2,3,4) \qquad 〖8〗$$

In addition, when we arrange the assignment, since the demands of different criteria are different , this requires to give different **weighting coefficients** to each component exclusive forces in order to fit different criteria. Using "**linear weight sum method** " of "**Multiple Criteria Decision**", we get the **exclusive force after composing** $f_j^{[i]}$ as :

$$f_j^{[i]} = \lambda_1 \cdot f_{j1}^{[i]} + \lambda_2 \cdot f_{j2}^{[i]} + \lambda_3 \cdot f_{j3}^{[i]} + \lambda_4 \cdot f_{j4}^{[i]} = \sum_{l=1}^{4} \lambda_l \cdot f_{jl}^{[i]} \qquad 〖9〗$$

(Where λ_l stands for the weighting coefficient of component exclusive force numbered l .

$\sum_{l=1}^{4} \lambda_l = 1$)

From the inquiry of the problem " for the morning sessions, no board member should be in the same senior officer's discussion group twice.", the weight λ_3 is given a very large number to make this criterion satisfied strictly. In the solution , let $\lambda_3 = 0.8$. But in practice, it is not always necessary to satisfy this criterion strictly , so , the representation of $F_{j3}^{[i]}$ can be altered to make it fit various circumstances . For example: $F_{j3}^{[i]} = D_j^{[i]}$ （Where $D_j^{[i]}$ stands for the number of times that member i led by officer j.) This can make the assignment more flexible.

After the formula of exclusive force obtained, the condition of assigning member *i* into group *j* is :

$$f_j^{[i]} = \min\left\{ f_1^{[i]}, f_2^{[i]}, \cdots\cdots, f_n^{[i]} \right\}$$

Solution of the Model

1. Algorithm

（1）Definition of State Variable

In some session, which group of a certain member should be assigned to depends on the assignments of the previous sessions. So the key to the problem solution lies in how to store and use the information of the assignments in the previous sessions. Here, the assignments of the previous sessions is defined as "**current state**", which includes several essential parts as following.

●**"meeting times" of two individual board members in the previous sessions**

This state variable is an determinative factor of ascertaining members' "**priority queue of members**" and "**exclusive force**" , which caused by "**meeting times**". In the program, array m[i][j] represents the times that member *i* with member *j* are assigned in the same group in the previous sessions. (*i* , *j* = 1,2,........,29)

●**Members of a certain group in certain session**

The calculation of "**common membership of group**" is determined by this value. we use array comm[7][6][29] to store this information.

comm[i][j][k]=1 represents member *k* is assigned to group *i* of session *j* ;
comm[i][j][k]=0 represents member *k* is not assigned to group *i* of session *j* ;

●**The number of the group that some board member is assigned to in each session**

This state variable can be used to calculate whether some member appears in the same officer's group twice. In the program, array desk[29][7] is used to store this message, desk[i][j] represents the number of the group which member *i* is assigned to in session *j* .

While arranging the assignment , whether "**priority queue of members**" or "**exclusive force**", we need to use some intermediate variables such as $C_j^{[i]}, M_{il}$ and so on. All of these intermediate variables' value can be obtained from the calculation of the above three arrays describing the "**current state**".

（2）Flow Diagram of the Program

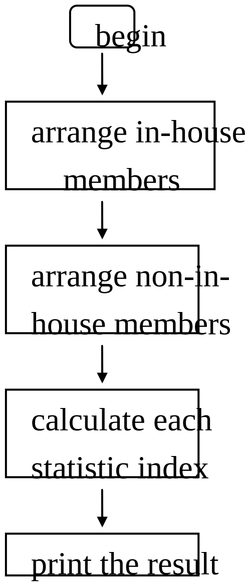


Figure 1.

Figure 1 is the flow diagram of the general algorithm, which includes : arranging the in-house, non-in-house members and printing the final result and statistic index. We programmed the algorithm with *Turbo C++ 1.0* 【see **Appendix C**】.

The procedure of arranging in-house members and that of non-in-house members are almost the same, except the formulas for **component of exclusive force** are different. When arranging the in-house members , they should be made as equiponderant as possible, while arranging the non-in-house-members, the proportion should be made as proper as possible.

The concrete flow diagram of how to arrange the in-house and non-in-house members is given in **Figure 2**:

The Flow Diagram of Arranging the In-house or Non-in-house Members

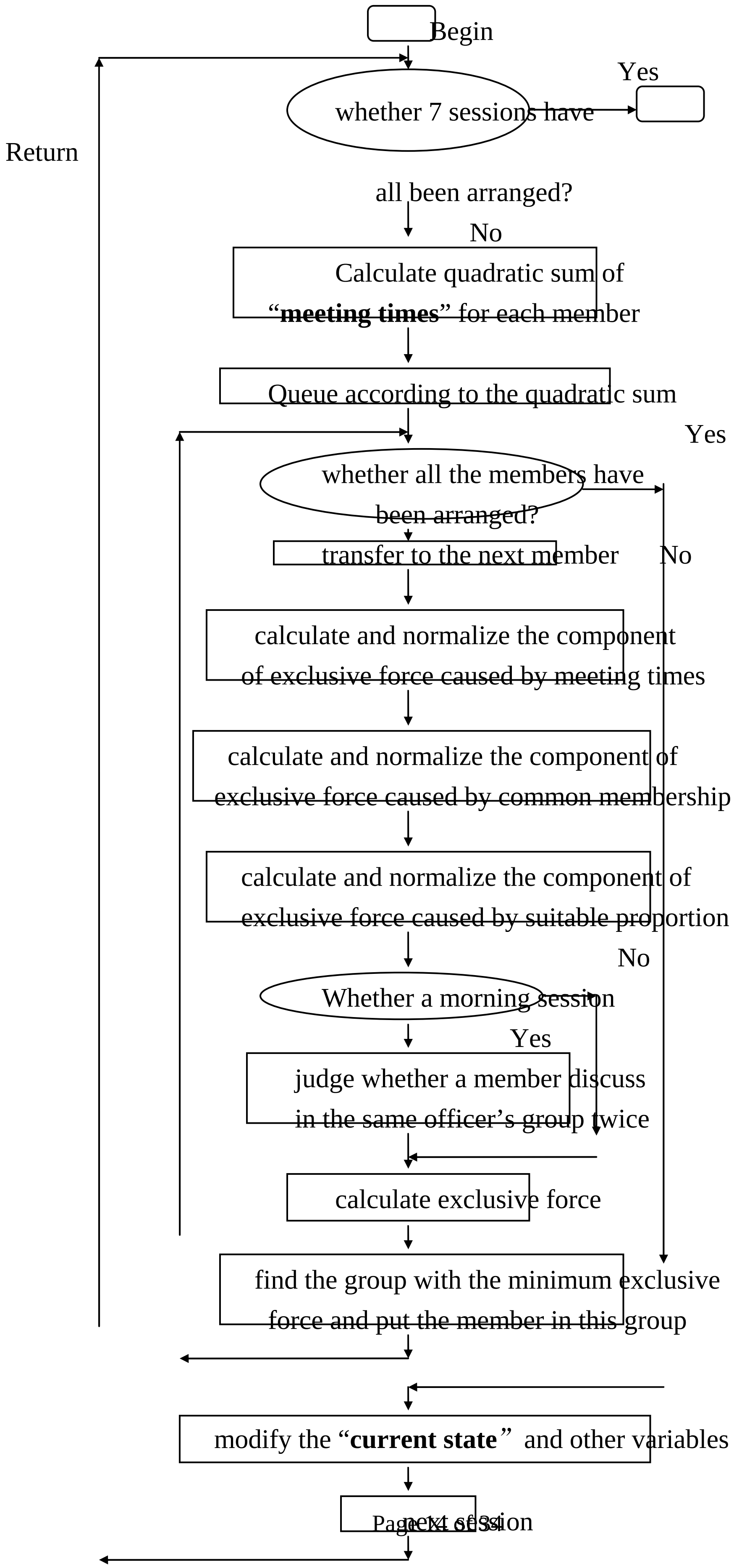


Figure 2.

2. Results of the Model

The results that take the four criteria into consideration at the same time:

(**Note:** During the morning session , team number that is officer's number.)

Here, each weighting coefficient is :

$$\lambda_1 = 0.033 \qquad \lambda_2 = 0.033 \qquad \lambda_3 = 0.8 \qquad \lambda_4 = 0.133$$

The maximum of common memberships is 3, appearing for 14 times.

Common Membership of Each Group **Table 2.**

	group 1	group 2	group 3	group 4	group 5	group 6
	2	1	2	2	3	3
Morning	3	2	2	2	2	2
	2	2	2	3	2	3
	3	3	3	2		
Afternoon	3	3	2	2		
	2	2	3	2		
	3	2	3	3		

Meeting times for each board member **Table 3**.

Person No	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	(No.)
Meet 0 times:	6	3	4	1	4	3	5	5	4	2	6	4	6	9	7	
Meet 1 times:	12	12	16	16	12	14	15	13	12	15	7	9	11	10	9	

Meet 2 times:	10	12	5	9	11	11	7	8	11	11	15	13	10	10	12
Meet 3 times:	1	2	4	3	2	1	2	3	2	1	1	3	2	0	1
Meet 4 times:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Person No.	16	17	18	19	20	21	22	23	24	25	26	27	28	29	(No.)
Meet 0 times:	5	5	5	3	4	7	6	6	7	5	2	5	2	8	
Meet 1 times:	12	10	12	14	5	10	11	10	13	12	12	10	13	3	
Meet 2 times:	12	11	11	9	17	9	11	12	6	9	13	12	13	16	
Meet 3 times:	0	3	1	3	3	3	1	1	3	3	2	2	1	2	
Meet 4 times:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

The Assignments (which with ( ) refers to the No. of in-house member)

```
***************** Session 1 ***************
Group 1: (1) ,13  ,22  ,
Group 2: (2) ,14  ,23  ,
Group 3: (3) ,15  ,24  ,
Group 4: (4) ,(5) ,10  ,16  ,19  ,25  ,28  ,
Group 5: (6) ,(7) ,11  ,17  ,20  ,26  ,29  ,
Group 6: (8) ,(9) ,12  ,18  ,21  ,27  ,

***************** Session 2 ***************
Group 1: (4) ,(9) ,11  ,20  ,28  ,12  ,15  ,23  ,
Group 2: (5) ,(3) ,17  ,25  ,18  ,13  ,
Group 3: (6) ,(2) ,10  ,26  ,27  ,22  ,
Group 4: (7) ,29  ,21  ,
Group 5: (8) ,16  ,24  ,
Group 6: (1) ,19  ,14  ,

***************** Session 3 ***************
Group 1: (5) ,(8) ,26  ,17  ,19  ,21  ,
Group 2: (4) ,(1) ,11  ,27  ,29  ,16  ,
Group 3: (9) ,25  ,14  ,
Group 4: (6) ,(3) ,20  ,12  ,13  ,22  ,
Group 5: (2) ,28  ,18  ,15  ,
```

```
Group 6: (7) ,10  ,23  ,24  ,

***************** Session 4 ***************
Group 1: (4) ,(8) ,(2) ,20  ,26  ,13  ,25  ,15  ,24  ,
Group 2: (5) ,(9) ,11  ,22  ,27  ,23  ,
Group 3: (6) ,(1) ,12  ,29  ,19  ,10  ,18  ,
Group 4: (3) ,(7) ,17  ,28  ,16  ,21  ,14  ,

***************** Session 5 ***************
Group 1: (4) ,(3) ,26  ,28  ,29  ,22  ,18  ,
Group 2: (5) ,(2) ,(7) ,20  ,12  ,27  ,19  ,23  ,24  ,
Group 3: (6) ,(9) ,17  ,15  ,25  ,16  ,
Group 4: (8) ,(1) ,11  ,13  ,21  ,10  ,14  ,

***************** Session 6 ***************
Group 1: (4) ,(7) ,12  ,25  ,13  ,21  ,
Group 2: (5) ,(1) ,26  ,23  ,15  ,18  ,16  ,
Group 3: (2) ,(3) ,(9) ,20  ,17  ,19  ,29  ,22  ,10  ,
Group 4: (6) ,(8) ,11  ,27  ,28  ,24  ,14  ,

***************** Session 7 ***************
Group 1: (2) ,(9) ,(1) ,12  ,29  ,26  ,25  ,28  ,21  ,
Group 2: (4) ,(6) ,17  ,19  ,23  ,13  ,14  ,
Group 3: (5) ,(7) ,11  ,22  ,15  ,24  ,
Group 4: (3) ,(8) ,20  ,27  ,10  ,18  ,16  ,
```

Analysis of the Results

（1）The maximum of common memberships

From the above results it can be seen that the maximum of common memberships is 3, and appears for 14 times. Comparing the result with that ignoring proportionate number of in-house members and board members in one officer’s discussion group less than twice, the maximum of common memberships did not alter, only appears for 4 times more. So the result of common membership is satisfactory.

（2）Meeting times

Though the final result does not make each board member with each other board member in a discussion group the same number of times, it can be seen from **Table 3** that different people's meeting times are balanced, and no board members meet each other for many times. Most of the board members can meet each other for one or two times.

（3）Senior officer

Because the senior officers' weighting coefficient is very large, so the final result strictly ensures that no board member stays in the same senior officer's discussion group twice.

（4）proportion of in-house members

According to the result, we calculate the proportion of in-house members of 34 groups in the all-day affair, as shown in the following **Table 4**:

Table 4.

group No.	1	2	3	4	5	6
morning	33.33% 33.33% 33.33%	33.33% 40.00% 33.33%	33.33% 40.00% 33.33%	28.57% 33.33% 33.33%	28.57% 33.33% 25.00%	33.33% 33.33% 25.00%
afternoon	33.33% 28.57% 33.33% 33.33%	33.33% 33.33% 28.57% 28.57%	28.57% 33.33% 33.33% 33.33%	28.57% 28.57% 28.57% 28.57%		

From the Table it can be found that for 20 times the proportion of in-house member is 33.33%, and 10 times of 28.57%, and each for 2 times of 40% and 25%. So the proportion of in-house member is satisfying.

（5）Each senior officer's workload

From the analysis of the problem, it is found that besides the four given criteria, there is still another criterion always considered, that is the board members led by each senior officer should about the same in quantity, which we call "making each senior officer's workload equipoise". According to the results, the six senior officers' workloads are : 16, 15, 12, 16, 14, 13. The maximal difference between two officers' workloads is 4. So the workload of each senior officer is satisfactory.

The Analysis of Stability

In order to test the stability of the model, we vary the Weighting Coefficients and examine the effect on the final result (See **Appendix B** for the testing results). From the results, the stability of the model is satisfactory.

Generalization of the Model

In an all-day sessions, it is possible that some board members cancel the meeting at the last minute or some not scheduled shows up. The total number of attendees and the criteria of mix are always changing too. So we generalize the model as:

●In case of some board members cancel the meeting

The algorithm is still adaptable when some members cancel the meeting . But the in-house members proportion and “current state” should be altered correspondingly. The alteration of the in-house members proportion is very easy: for example, if a non-in-house member cancel the meeting , just alter 9/29 to 9/30 in formula 〖7〗. According to whether the board member who cancel meetings is in-house member or not , “current state” should be altered as follows:

A non-in-house member cancels the meeting.

According to the algorithm, this does not effect the in-house members' assignments. Only array m[i][j]， which represents meeting times need to be altered. Put zero to all array elements relative to the new member. The other two arrays about “current state” do not need to be altered. Then all non-in-house members should be assigned again based on the altered “current state.”

An in-house member cancels the meeting .

The “current state ”should be altered first (as above), then all in-house and non-in-house members should be assigned again according to the altered “current state”.

●In case of some board members who have not been scheduled show up

In this case, the proportion of in-house members should also be altered first, then the “current state”.

Non-in-house members who have not been scheduled show up.

Since new non-in-house members have never shown up in the previous sessions, array m[i][j] should be enlarged and the new members be added in. Then the meeting times of the new members in meeting with others equal zero. The others two arrays should still remain the original values. Then according to the altered "current state", rearrange all the non-in-house members.

In-house members who have not been scheduled show up.

In this case , the "current state" should be altered in the same way as above, then all in-house and non-in-house members should be altered again according to the altered "current state". So that the members could still be mixed as well as possible.

●In case of the total of attendees is altered

Since this case has already been considered in the modeling process, alteration of the algorithm is no need, but some parameters of the program and the arrays of the "current state" should be altered correspondingly.

Supposing now there are 11 in-house members, 25 non-in-house members, and 6 senior officers, the other aspects are the same as those in the original problem. Assignments have been made in this case, and the result is satisfactory. (Limited by the space, the result is not given here.)

●In case of several new criteria are added.

While arranging the assignment, whether the assignment is good or not is based on how well the four criteria could be met. If some new criteria are added, only some component exclusive forces are added correspondingly.

The formula with many criteria is:

Supposing *n* criteria are given, the weighting coefficient of criterion *k* is λ_k ,the component exclusive forces of criterion *k* is $f_{jk}^{[i]}, (k = 1,2,\cdots\cdots,n)$, then:

$$f_j^{[i]} = (\lambda_1, \lambda_2, \cdots\cdots, \lambda_n) \cdot \begin{pmatrix} f_{j1}^{[i]} \\ f_{j2}^{[i]} \\ \vdots \\ \vdots \\ f_{jn}^{[i]} \end{pmatrix} = \sum_{k=1}^{n} \lambda_k \cdot f_{jk}^{[i]}$$

$$\text{....................................}\left(\sum_{k=1}^{n} \lambda_k = 1 \right)$$

Strengths and Weaknesses

The model is practical to make assignments for all kinds of meetings involving different levels of participation for each type of attendees, and is easy to modify when some origin criteria are canceled or some new criteria are added; what need to do is just to cancel some original formula of **exclusive force** or to add some new ones.

The weakness of this model is that the **weighting coefficient** has some subjective elements inside. If mathematical approach can be applied to make the **weight** certain, it will be more ideal. On the other hand, the subjective in valuing weighting coefficients gives an easy way to adapt different cases. When the degree of the criteria's satisfaction alters, it just need to adjust the **weight** accordingly.

References

【1】Japanese Mathematical Academy, (1984) *Mathematical Encyclopedia* Science Publication House

【2】Proposer's Commentary, The Outstanding Emergency Power Restoration Pares, *The UMAP Journal*, v. 13 (1992), No. 3,273-274.

【3】Qian Songdi, (1990) *Operational Research.* Tsinghua University Publication House

Appendix A

〖**The proof of Non Absolutely Ideal Assignments**〗

When Eq. 〖1〗 is strictly satisfied, there has:

$$\sum_{j=1}^{3}\sum_{i=1}^{6}(x_i^{[j]})^2+\sum_{j=4}^{7}\sum_{i=1}^{4}(x_i^{[j]})^2=29(28s+7)$$

from **Cauchy inequality**: $\frac{x_1^2+x_2^2+\cdots+x_n^2}{n}\geq\left(\frac{x_1+x_2+\cdots x_n}{n}\right)^2$,

($x_1,x_2,\cdots\cdots,x_n\geq 0$; only when $x_1=x_2=\cdots=x_n$, “$=$” is tenable。)

we can see:

To a certain session, if the member in different groups are balanced, the value of

$$\sum_{j=1}^{3}\sum_{i=1}^{6}(x_i^{[j]})^2+\sum_{j=4}^{7}\sum_{i=1}^{4}(x_i^{[j]})^2$$

will be smaller. And because:

$$\sum_{j=1}^{3}\sum_{i=1}^{6}(x_i^{[j]})^2+\sum_{j=4}^{7}\sum_{i=1}^{4}(x_i^{[j]})^2=29(28s+7)$$

so s will be smaller.

whereas the more difference , the greater s is.

To An Tostal Corporation:

①The assignment which makes value of s minimum should satisfy:

$$\begin{cases}\textit{number of members of each group in the morning should respectively be}: 5\,5\,5\,5\,5\,4\\ \textit{number of members of each group in the afternoon should respectively be}: 7\,7\,7\,8\end{cases}$$

②The assignment which makes value of s maximum should satisfy:

$$\begin{cases}\textit{number of members of each group in the morning should respectively be}: 2\,2\,2\,2\,2\,19\\ \textit{number of members of each group in the afternoon should respectively be}: 2\,2\,2\,23\end{cases}$$

Apply ①② to Eq. 〖1〗, we obtain:

$$\begin{cases} \min s = 1.3 \\ \max s = 3.8 \end{cases} \quad \text{considering } s \in N \text{ in Eq. 〖1〗, so} \quad \begin{cases} \min s = 2 \\ \max s = 3 \end{cases}$$

For s=2 , we use computer searching for the solution to Eq. 〖1〗 in range of the assignments in **Table 1**. The result is listed in the following **Table 5**:

Table 5.

	Result 1	Result 2	Result 3
number of members in each session in the morning.	3 3 3 7 7 6 3 3 3 3 7 10 3 3 3 3 7 10	3 3 3 7 7 6 3 3 3 7 7 6 3 3 3 3 7 10	3 3 3 7 7 6 3 3 3 7 7 6 3 3 3 7 7 6
numberof members in each session in the afternoon.	3 3 3 20 3 3 3 20 7 6 6 10 7 6 6 10	3 3 3 20 3 3 3 20 7 6 6 10 3 10 6 10	3 3 3 20 3 3 3 20 3 10 6 10 3 10 6 10

Analyzing the three solutions, all of them have assignment (3，3，3，20) appears twice in the afternoon. So, to the assignment in the afternoon , even if assign 9 members of the group contain 20 members to the other groups in the next assignment , there is still 11 members left when (3, 3, 3, 20) appears the second time. Thus the maximum of common memberships is 11 at least. Obviously, it is not well mixed.

Further calculation indicates , when s=3, the maximum of common memberships is even greater than 11. Accordingly:

The absolutely ideal assignment that strictly satisfies the four criteria does not exist.

Appendix B

〖The Analysis of Stability〗

Each weighting coefficient is :

$\lambda_1 = 0.029$ $\lambda_2 = 0.014$ $\lambda_3 = 0.9$ $\lambda_4 = 0.057$

The maximum of common memberships is 3, appearing for 15 times.

Common Membership of Each Group

2	2	2	2	3	2
2	3	2	2	2	2
2	2	2	2	2	3

3	3	3	2
2	3	3	2
3	3	3	3
3	3	3	2

Meeting times for each board member

(No.)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
meet 0 times:	6	4	7	2	2	3	6	4	4	2	4	2	5	6	7
meet 1 times:	12	12	10	13	15	12	10	14	12	13	9	13	10	13	11
meet 2 times:	11	11	11	12	11	13	13	10	11	12	14	12	13	10	10
meet 3 times:	0	2	1	2	1	1	0	1	2	2	2	2	1	0	1
meet 4 times:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

(No.)	16	17	18	19	20	21	22	23	24	25	26	27	28	29
meet 0 times:	7	7	6	4	5	4	4	5	4	5	5	3	4	4
meet 1 times:	12	8	8	12	9	14	14	11	15	12	8	14	13	11
meet 2 times:	9	14	14	13	15	10	11	12	9	10	12	11	10	12
meet 3 times:	1	0	1	0	0	1	0	1	1	2	4	1	2	2
meet 4 times:	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Each weighting coefficient is :

$$\lambda_1 = 0.009 \qquad \lambda_2 = 0.018 \qquad \lambda_3 = 0.9 \qquad \lambda_4 = 0.073$$

The maximum of common memberships is 4, appearing for 3 times.

Common Membership of Each Group

2	3	2	3	3	2
3	3	2	3	2	2
3	2	2	3	2	2

3	3	3	3
3	4	3	2
3	2	4	3
4	3	3	3

Meeting times for each board member

(No.)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
meet 0 times:	6	6	6	3	3	2	7	3	6	6	4	4	8	6	4
meet 1 times:	14	12	15	12	15	15	10	16	9	11	12	11	10	9	15
meet 2 times:	7	4	2	10	8	11	10	9	11	10	10	12	10	12	8
meet 3 times:	2	7	6	4	3	1	2	1	3	2	3	2	1	2	2
meet 4 times:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

(No.)	16	17	18	19	20	21	22	23	24	25	26	27	28	29
meet 0 times:	4	3	6	5	4	7	6	6	5	2	2	3	7	5
meet 1 times:	15	15	11	13	5	7	10	10	18	17	16	11	10	10
meet 2 times:	7	9	8	6	16	10	12	10	5	9	7	14	8	8
meet 3 times:	3	2	4	5	4	5	1	3	1	1	4	1	4	6
meet 4 times:	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Each weighting coefficient is :

$\lambda_1 = 0.018$ $\lambda_2 = 0.09$ $\lambda_3 = 0.9$ $\lambda_4 = 0.073$

The maximum of common memberships is 4, appearing for 2 times.

Common Membership of Each Group

2	3	2	3	3	3
3	3	3	2	2	2
3	4	2	3	2	2

3	3	3	3
3	3	3	3
2	3	3	2
4	3	3	3

Meeting times for each board member

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	(No.)
meet 0 times:	7	3	4	3	3	4	7	4	4	6	5	4	10	7	4	
meet 1 times:	13	13	16	14	12	14	11	15	11	10	9	10	7	8	11	
meet 2 times:	5	10	5	7	12	8	9	9	11	11	11	13	11	13	12	
meet 3 times:	4	3	4	5	2	3	2	1	3	2	4	2	1	1	2	
meet 4 times:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

	16	17	18	19	20	21	22	23	24	25	26	27	28	29	(No.)
meet 0 times:	6	4	5	4	5	7	6	7	7	4	3	4	5	5	
meet 1 times:	11	9	11	14	3	6	11	10	11	12	13	12	14	11	
meet 2 times:	10	11	11	9	17	13	8	8	11	12	10	12	7	10	
meet 3 times:	2	5	2	2	4	3	4	4	0	1	3	1	3	3	
meet 4 times:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

Appendix C

〖The program used to arrange the assignments〗

```
/*-----------------------------------------------------------------
-------
m[45][45]     *** m[i][j] --- meeting times of member i and member j
mem[6][45]    *** mem[i][j] --- who is in group i
aga[6]        *** aga[i] --- the exclusive force of group k
imp[45][2]    *** imp[i][j] --- the priority queue of members
                               imp[i][0] contain the No. of the
member
                               imp[i][1] contain Weighting
Coefficient
comm[20][6][45]  *** comm[i][j][k] --- the member in group j , session
i
desk[45][20]  *** desk[i][j] --- the group number that member i stay
                                in session j
amount        *** the num of board members (the maximum is 45 )
prop[20][6]   *** prop[i][j] --- the numbers of inhouse members in
group
                                j session i
-----------------------------------------------------------------
------*/

#include <stdio.h>
#include <stdlib.h>
#include <math.h>

int inhouse=9,amount,am=29,ci=6;            //ci: number of
officers
int m[45][45],desk[45][20];
int mem[6][45],prop[20][6],i,h;
int imp[45][2],comm[20][6][45];
int in[6],zhuo;
int temp[6][14],result[45][7];
int session,group,begin;
```

```
    float mi[6];
    float aga[6],gong[6],ming[6],num[6],max=0;
    float lamda3=96,lamda1=2,lamda4=8,lamda2=2; // Weighting
Coefficient
    main()
    {
      float g=0;
      clrscr();
      amount=inhouse;
      ini_all();
      g=lamda3+lamda1+lamda4+lamda2;
      lamda3=lamda3/g;lamda1=lamda1/g;
      lamda4=lamda4/g;lamda2=lamda2/g;
      printf("officer:%4.3f  meeting times:%4.3f \n",lamda3,lamda1);
      printf("proportion:%4.3f  common member:%4.3f\n",lamda4,lamda2);

      begin=0;
      amount=inhouse;
      find(1,3,6,1);
      find(3,7,4,0);
      out_put();
      cacu_prop();

      begin=inhouse;
      amount=am;
      find(0,3,6,1);
      find(3,7,4,0);
      out_put();
      common_times();

      return;
    }
    find(int r,int p,int q,int noon)
    {
      int j,k,l,a,b,c,d;

      session=p; group=q;
```

```
    for(h=r;h<session;h++)
    {
      for(j=0;j<group;j++)num[j]=0;
      cacu_imp();
      queue_imp();

      for(j=0;j<group;j++)
          for(k=0;k<45;k++)
             mem[j][k]=0;
      for(j=0;j<group;j++)
        {
          k=0;
          while(comm[h][j][k]!=0)
           {
             mem[j][comm[h][j][k]-1]=1;
             k=k+1;
           }
         }
    for(i=begin;i<amount;i++)
     {
       for(j=0;j<group;j++)
          aga[j]=0;

       aga_meet();
       if(noon==1)aga_officer();
       aga_prop();
       aga_common();

       find_min_aga();
       put_in();
     }
    }
    printf("****************** To session %u
******************\n\n",h);
    times_of_meet();
    printf("\nThe maximum common memberships: %2.0f\n\n",max+1);
    times_of_everyone();
```

```
return;
}

ini_all()
{
  int l,j,k,a;

  for(l=0;l<20;l++)
    for(j=0;j<6;j++)
     prop[l][j]=1;

  j=amount%ci;
  k=(amount-j)/ci;
  for(l=0;l<ci;l++)
    in[l]=k;
  for(l=ci;l>=ci-j;l--)
    in[l]=in[l]+1;
  for(l=0;l<45;l++)
    for(j=0;j<20;j++)
     desk[l][j]=0;
  a=0;
  for(l=0;l<6;l++)
   {
    for(j=0;j<in[l];j++)
      desk[j+a][0]=l+1;
    a=a+in[l];
   }

  for(l=0;l<45;l++)
    for(j=0;j<45;j++)
     m[l][j]=0;
  a=0;
  for(l=0;l<6;l++)
   {
    for(j=0;j<in[l];j++)
      for(k=0;k<in[l];k++)
        m[j+a][k+a]=1;
```

```
    a=a+in[l];
   }
  for(j=0;j<45;j++)
    m[j][j]=0;

  for(l=0;l<6;l++)
    for(j=0;j<14;j++)
      temp[l][j]=0;
  a=1;
  for(l=0;l<6;l++)
   for(j=0;j<in[l];j++)
     {
      temp[l][j]=a;
      a=a+1;
     }

  for(l=0;l<20;l++)
   for(j=0;j<6;j++)
    for(k=0;k<45;k++)
       comm[l][j][k]=0;

  for(l=0;l<6;l++)
   for(j=0;j<14;j++)
     comm[0][l][j]=temp[l][j];

  for(l=0;l<45;l++)
   {
    imp[l][0]=l;
    imp[l][1]=0;
   }
  for(j=0;j<group;j++)
      for(k=0;k<45;k++)
        mem[j][k]=0;

  return(1);
}
```

```
cacu_imp()
{
  int l,j,k;

  for(l=begin;l<amount;l++)
   { k=0;
     for(j=0;j<amount;j++)
       k=k+pow(m[l][j],2);
     imp[l][0]=l;imp[l][1]=k;
   }
  return(1);
}

queue_imp()
{
  int l,j,k,f;

  for(l=begin;l<amount;l++)
     for(j=l;j<amount;j++)
      { if(imp[l][1]<imp[j][1])
         {
          k=imp[l][0];f=imp[l][1];
          imp[l][0]=imp[j][0];imp[l][1]=imp[j][1];
          imp[j][0]=k;imp[j][1]=f;
         }
      }
  return(1);
}

aga_meet()
{
   int j,k;
   float f=0;

   for(j=0;j<group;j++)
     {
       for(k=0;k<amount;k++)
```

```
        {
          if(mem[j][k]==1&&m[imp[i][0]][k]!=0)
          aga[j]=aga[j]+pow(m[imp[i][0]][k],4);
        }
      }
   for(j=0;j<group;j++)
      f=f+aga[j];

   for(j=0;j<group;j++)
      aga[j]=f==0?0:aga[j]/f;

   for(j=0;j<group;j++)
     aga[j]=lamda1*aga[j];

   return;
}

aga_prop()
{
   int j;
   float g=0,num1[6];

   for(j=0;j<group;j++)
      num1[j]=(num[j]+1)/prop[h][j];

   for(j=0;j<group;j++)
      g=g+pow(num1[j],1);

   for(j=0;j<group;j++)
      num1[j]=g==0?0:pow(num1[j],1)/g;

   for(j=0;j<group;j++)
     aga[j]=aga[j]+lamda4*num1[j];

   return;
}
```

```
aga_officer()
{
  int a,j;
    for(j=0;j<h;j++)
     {
       a=desk[imp[i][0]][j]-1;
       aga[a]=aga[a]+1*lamda3;
     }
  return;
}

aga_common()
{
  int j,k,a,b,c,d;
  float g=0;

    for(j=0;j<group;j++)
      { ming[j]=0,mi[j]=0;}

    for(j=0;j<h;j++)
       for(k=0;k<6;k++)
       { a=0;
         while(comm[j][k][a]!=0)
          {
            if(comm[j][k][a]==(imp[i][0]+1))
               {
                for(d=0;d<group;d++)gong[d]=0;
                for(b=0;b<group;b++)
                 {
                  c=0;
                  while(comm[j][k][c]!=0)
                  {
                    if(mem[b][comm[j][k][c]-1]==1)
                       gong[b]=gong[b]+1;
                    c=c+1;
                   }
                 }
```

```
                for(d=0;d<group;d++)
                 {
                  if(ming[d]<gong[d])
                     ming[d]=gong[d];
                 }
               }
            a=a+1;
          }
       }

for(d=0;d<group;d++)
 {
  mi[d]=ming[d];
  g=g+pow(ming[d],2);
 }
for(d=0;d<group;d++)
  ming[d]=g==0?0:pow(ming[d],2)/g;

for(d=0;d<group;d++)
  aga[d]=aga[d]+lamda2*ming[d];

return;
}

find_min_aga()
{
  int j;
  float min=10000;

   for(j=0;j<group;j++)
    if(aga[j]<min)
      { min=aga[j];zhuo=j;}
   if(max<mi[zhuo])max=mi[zhuo];

  return;
}
```

```
put_in()
{
   int k,d;

   for(k=0;k<amount;k++)
    {
      if(mem[zhuo][k]==1)
      {
        m[imp[i][0]][k]=m[imp[i][0]][k]+1;
        m[k][imp[i][0]]=m[k][imp[i][0]]+1;
      }
    }
   mem[zhuo][imp[i][0]]=1;
   num[zhuo]=num[zhuo]+1;
   desk[imp[i][0]][h]=zhuo+1;
   d=0;
   while(comm[h][zhuo][d]!=0)d++;
   comm[h][zhuo][d]=imp[i][0]+1;
   return;
}
times_of_meet()
{
 int j,k,w[8];

 for(j=0;j<8;j++)
   w[j]=0;

 for(k=0;k<45;k++)
   for(j=0;j<7;j++)
    result[k][j]=0;

 printf("           The meeting times  \n\n");
 for(k=0;k<amount;k++)
  { for(j=0;j<amount;j++)
    {
     result[k][m[k][j]]=result[k][m[k][j]]+1;
     if(m[k][j]==0)w[0]=w[0]+1;
```

```
        if(m[k][j]==1)w[1]=w[1]+1;
        if(m[k][j]==2)w[2]=w[2]+1;
        if(m[k][j]==3)w[3]=w[3]+1;
        if(m[k][j]==4)w[4]=w[4]+1;
        if(m[k][j]==5)w[5]=w[5]+1;
        if(m[k][j]==6)w[6]=w[6]+1;
        if(m[k][j]==7)w[7]=w[7]+1;
       }
     }

    for(k=0;k<8;k++)
      printf(" %u total appears for %u times\n",k,w[k]/2);
    return;
   }

   times_of_everyone()
   {
    int j,k;

    printf("             Meeting Times for Each Board Member \n\n");
    printf("          ");
    for(k=0;k<15;k++)
      printf("%2u ",k+1);
    printf("  (No.)\n");

printf("-------------------------------------------------------------
\n");
    for(k=0;k<7;k++)
     { printf("meet %2u times:",k);
       for(j=0;j<15;j++)
         printf("%2u ",result[j][k]);
       printf("\n");
     }

printf("=============================================================
\n");
    if(amount>15)
```

```
      {
       printf("          ");
       for(k=15;k<amount;k++)
         printf("%2u ",k+1);
       printf("  (No.)\n");

printf("---------------------------------------------------------------
\n");
       for(k=0;k<7;k++)
        { printf("meet %2u times:",k);
          for(j=15;j<amount;j++)
            printf("%2u ",result[j][k]);
          printf("\n");
        }
    printf("==========================================================
===\n\n");
      }
     return;
    }
    out_put()
    {
      int j,k,l;
      j=0;
      while(comm[j][0][0]!=0&&j<20)
       {
        k=0;
        printf(" **************** session %u  ***************\n",j+1);
        while(comm[j][k][0]!=0&&k<6)
         {
          l=0;
          printf(" Group %u : ",k+1);
          while(comm[j][k][l]!=0&&l<amount)
           {
            if(comm[j][k][l]<=inhouse)
               printf("(%-1u) ,",comm[j][k][l]);
            else
               printf("%-4u,",comm[j][k][l]);
```

```
          l=l+1;
         }
        printf("\n");
        k=k+1;
       }
      printf("\n");
      j=j+1;
     }
     return;
}
cacu_prop()
{
   int d,j,k;

     for(j=0;j<20;j++)
       for(k=0;k<6;k++)
        {
          d=0;
          while(comm[j][k][d]!=0)d++;
          prop[j][k]=d;
        }
     return;
}
common_times()
{
   int j,k,l,m,u,maa,ww[45][2];
   int a,maxx,eee[10][6];

   for(j=0;j<10;j++)
     for(k=0;k<6;k++)
       eee[j][k]=0;

   for(j=0;j<am;j++)
     for(k=0;k<2;k++)
       ww[j][k]=0;

   for(j=0;j<7;j++)
```

```
{
for(k=0;k<6;k++)
{
    u=0;
    while(comm[j][k][u]!=0)
    { ww[comm[j][k][u]-1][1]=1;
       u=u+1;
    }
    maxx=0;
    for(l=0;l<7;l++)
    {
    for(m=0;m<6;m++)
    {
      maa=0;
      if(l!=j&&m!=k)
      {
        u=0;
        while(comm[l][m][u]!=0)
        {
          ww[comm[l][m][u]-1][0]=1;
          u=u+1;
        }
       for(a=0;a<am;a++)
        {
         if(ww[a][0]==1&&ww[a][1]==1)
          maa=maa+1;
        }
       for(a=0;a<am;a++)
         ww[a][0]=0;
      }
      if(maxx<maa)maxx=maa;
    }
    }
    eee[j][k]=maxx;
    for(a=0;a<am;a++)
     {ww[a][1]=0;ww[a][0]=0;}
}
```

```
}
for(j=0;j<10;j++)
 {
   for(k=0;k<6;k++)
    if(eee[j][k]==0)
      printf("    ",eee[j][k]);
    else
      printf("%2u  ",eee[j][k]);
   printf("\n");
 }

printf("\nThe maximum of common memberships is: %2.0f\n\n",max+1);
maxx=0;
for(j=0;j<10;j++)
   for(k=0;k<6;k++)
    if(eee[j][k]==(max+1))
      maxx=maxx+1;

printf("It appears for %u times\n\n",maxx);
return;
}
```

```
主管:0.900 次数:0.029 比例:0.057 公共成员:0.014

最大公共成员数为: 3     最大公共成员出现次数为: 15

=============================
 2  2  2  2  3  2
 2  3  2  2  2  2
 2  2  2  2  2  3
------------------------
 3  3  3  2
 2  3  3  2
 3  3  3  3
 3  3  3  2
=============================

        每人同其他人见面次数统计值

      1  2  3  4  5  6  7  8  9 10 11 12 13 14 15   (号)
----------------------------------------------------------
见 0 次面: 6  4  7  2  2  3  6  4  4  2  4  2  5  6  7
见 1 次面:12 12 10 13 15 12 10 14 12 13  9 13 10 13 11
见 2 次面:11 11 11 12 11 13 13 10 11 12 14 12 13 10 10
见 3 次面: 0  2  1  2  1  1  0  1  2  2  2  2  1  0  1
见 4 次面: 0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
====================================================================
     16 17 18 19 20 21 22 23 24 25 26 27 28 29   (号)
----------------------------------------------------------
见 0 次面: 7  7  6  4  5  4  4  5  4  5  5  3  4  4
见 1 次面:12  8  8 12  9 14 14 11 15 12  8 14 13 11
见 2 次面: 9 14 14 13 15 10 11 12  9 10 12 11 10 12
见 3 次面: 1  0  1  0  0  1  0  1  1  2  4  1  2  2
见 4 次面: 0  0  0  0  0  0  0  0  0  0  0  0  0  0
====================================================================

主管:0.900 次数:0.009 比例:0.073 公共成员:0.018

最大公共成员数为: 4     最大公共成员出现次数为: 3

==============================
 2  3  2  3  3  3
 3  3  2  3  2  2
 3  2  2  3  2  2
```

```
--------------------------
3  3  3  3
3  4  3  2
3  2  4  3
4  3  3  3
============================

        每人同其他人见面次数统计值

      1  2  3  4  5  6  7  8  9 10 11 12 13 14 15  (号)
--------------------------------------------------------------
见 0 次面: 6  6  6  3  3  2  7  3  6  6  4  4  8  6  4
见 1 次面:14 12 15 12 15 15 10 16  9 11 12 11 10  9 15
见 2 次面: 7  4  2 10  8 11 10  9 11 10 10 12 10 12  8
见 3 次面: 2  7  6  4  3  1  2  1  3  2  3  2  1  2  2
见 4 次面: 0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
=====================================================================
     16 17 18 19 20 21 22 23 24 25 26 27 28 29  (号)
--------------------------------------------------------------
见 0 次面: 4  3  6  5  4  7  6  6  5  2  2  3  7  5
见 1 次面:15 15 11 13  5  7 10 10 18 17 16 11 10 10
见 2 次面: 7  9  8  6 16 10 12 10  5  9  7 14  8  8
见 3 次面: 3  2  4  5  4  5  1  3  1  1  4  1  4  6
见 4 次面: 0  0  0  0  0  0  0  0  0  0  0  0  0  0
=====================================================================

主管:0.900 次数:0.018 比例:0.073 公共成员:0.009

最大公共成员数为: 4      最大公共成员出现次数为: 2

============================
2  3  2  3  3  3
3  3  3  2  2  2
3  4  2  3  2  2
-------------------------
3  3  3  3
3  3  3  3
2  3  3  2
4  3  3  3
============================

        每人同其他人见面次数统计值
```

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	(号)
见 0 次面:	7	3	4	3	3	4	7	4	4	6	5	4	10	7	4	
见 1 次面:	13	13	16	14	12	14	11	15	11	10	9	10	7	8	11	
见 2 次面:	5	10	5	7	12	8	9	9	11	11	11	13	11	13	12	
见 3 次面:	4	3	4	5	2	3	2	1	3	2	4	2	1	1	2	
见 4 次面:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

	16	17	18	19	20	21	22	23	24	25	26	27	28	29	(号)
见 0 次面:	6	4	5	4	5	7	6	7	7	4	3	4	5	5	
见 1 次面:	11	9	11	14	3	6	11	10	11	12	13	12	14	11	
见 2 次面:	10	11	11	9	17	13	8	8	11	12	10	12	7	10	
见 3 次面:	2	5	2	2	4	3	4	4	0	1	3	1	3	3	
见 4 次面:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

&&&&&&&&&&&&&&&&&& inhouse 成员人数为 11,总人数为36 &&&&&&&&&&&&&&&&&&&
考虑主管及比例的情况

主管:0.370 次数:0.148 比例:0.296 公共成员:0.185

****************** 到第 7 届会议为止 ******************

见面次数总和统计

0 共出现 107 次
1 共出现 252 次
2 共出现 254 次
3 共出现 35 次
4 共出现 0 次

最大公共成员数为: 3

每人同其他人见面次数统计值

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18 (号)
见 0 次面:	5	6	3	6	6	5	5	4	6	6	5	8	7	7	7	8	8	6
见 1 次面:	18	14	16	13	16	16	18	19	14	14	13	9	9	10	10	11	12	15
见 2 次面:	11	12	14	15	12	14	10	10	14	14	18	16	19	17	18	16	14	12
见 3 次面:	2	4	3	2	2	1	3	3	2	2	0	3	1	2	1	1	2	3
见 4 次面:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36 (号)
见 0 次面:	7	5	6	7	7	7	4	8	4	10	4	4	5	4	3	7	7	7
见 1 次面:	12	14	16	11	12	12	18	13	21	10	16	18	18	17	14	14	10	11
见 2 次面:	12	14	12	16	15	16	12	13	10	14	15	12	11	13	18	15	18	16
见 3 次面:	5	3	2	2	2	1	2	2	1	2	1	2	2	2	1	0	1	2
见 4 次面:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

**************** 第 1 届会议 ***************
第 1 组: (1) ,17 ,28 ,
第 2 组: (2) ,(3) ,12 ,18 ,23 ,29 ,34 ,
第 3 组: (4) ,(5) ,13 ,19 ,24 ,30 ,35 ,
第 4 组: (6) ,(7) ,14 ,20 ,25 ,31 ,36 ,
第 5 组: (8) ,(9) ,15 ,21 ,26 ,32 ,

第 6 组: (10) ,(11) ,16 ,22 ,27 ,33 ,

***************** 第 2 届会议 ***************
第 1 组: (2) ,(8) ,13 ,25 ,31 ,16 ,
第 2 组: (4) ,(9) ,14 ,30 ,36 ,33 ,17 ,
第 3 组: (3) ,(10) ,20 ,29 ,15 ,26 ,28 ,
第 4 组: (5) ,(11) ,12 ,34 ,35 ,32 ,
第 5 组: (6) ,(1) ,18 ,19 ,22 ,27 ,
第 6 组: (7) ,23 ,24 ,21 ,

***************** 第 3 届会议 ***************
第 1 组: (3) ,(9) ,36 ,35 ,20 ,27 ,23 ,
第 2 组: (5) ,(8) ,15 ,31 ,22 ,24 ,
第 3 组: (2) ,(11) ,14 ,25 ,18 ,32 ,
第 4 组: (4) ,(1) ,29 ,33 ,19 ,16 ,21 ,
第 5 组: (10) ,(7) ,30 ,12 ,13 ,28 ,
第 6 组: (6) ,26 ,34 ,17 ,

***************** 第 4 届会议 ***************
第 1 组: (2) ,(9) ,(7) ,(1) ,20 ,33 ,31 ,12 ,22 ,32 ,
第 2 组: (3) ,(8) ,(11) ,36 ,19 ,13 ,18 ,24 ,26 ,17 ,
第 3 组: (4) ,(10) ,25 ,15 ,35 ,27 ,34 ,21 ,
第 4 组: (5) ,(6) ,30 ,14 ,29 ,16 ,23 ,28 ,

***************** 第 5 届会议 ***************
第 1 组: (2) ,(4) ,(10) ,20 ,18 ,22 ,24 ,26 ,23 ,
第 2 组: (9) ,(11) ,(6) ,31 ,19 ,29 ,35 ,21 ,17 ,
第 3 组: (3) ,(1) ,(5) ,33 ,12 ,13 ,14 ,15 ,25 ,28 ,
第 4 组: (7) ,(8) ,36 ,30 ,32 ,27 ,16 ,34 ,

***************** 第 6 届会议 ***************
第 1 组: (2) ,(11) ,(6) ,20 ,33 ,13 ,15 ,30 ,21 ,34 ,
第 2 组: (4) ,(3) ,(7) ,19 ,14 ,22 ,32 ,23 ,16 ,17 ,
第 3 组: (9) ,(5) ,18 ,31 ,28 ,26 ,27 ,
第 4 组: (10) ,(1) ,(8) ,12 ,36 ,24 ,35 ,29 ,25 ,

***************** 第 7 届会议 ***************
第 1 组: (11) ,(9) ,(7) ,12 ,14 ,15 ,24 ,29 ,27 ,
第 2 组: (2) ,(3) ,(5) ,19 ,22 ,30 ,25 ,21 ,26 ,
第 3 组: (10) ,(6) ,(8) ,33 ,35 ,18 ,23 ,32 ,28 ,17 ,
第 4 组: (4) ,(1) ,20 ,13 ,36 ,16 ,31 ,34 ,

2 3 3 3 2 2

3 2 3 2 2 2
2 2 2 3 3 2
3 3 3 3
3 3 3 3
3 3 2 3
3 3 3 3

最大公共成员数为: 3

最大公共成员出现次数为: 22

```
/*------------------------------------------------------------------------

        ************************************************
        * 该程序用于解决 : 考虑 inhouse 成员分配问题, *
        * 人数 1 -- 45 人任意, 是否考虑主管任意, 比例  *
        * 尽量按 inhouse 成员均衡的情况.             *
        ************************************************

m[45][45]    *** m[i][j] --- 第 i 与第 j 人见面的次数.***
mem[6][45]   *** mem[i][j] --- 第 i 桌上有谁 ***
aga[6]       *** aga[i] --- 第 k 桌的排斥力 ***
imp[45][2]   *** imp[i][j] --- 第 i 人的重要性 其中:imp[i][0] 为人的代码,
                         imp[i][1] 为人的权重. ***
num[6]       *** num[i] --- 第 j 桌的现有人数
comm[20][6][45]  *** comm[i][j][k] --- 第 i 次会议第 j 桌的人员. ***
desk[45][20]  *** desk[i][j] --- 第 i 个人在第 j 次会议上所处的桌号.
result[45][7] *** result[i][j] --- 用于存放见面次数均衡情况表
amount       *** 人数 (amount) 最大值.(现为 45 人)
prop[20][6]   *** prop[i][j] --- 第 i 次会议第 j 桌有几个 inhouse.
------------------------------------------------------------------------*/

#include <stdio.h>
#include <stdlib.h>
#include <math.h>

int inhouse=9,amount,am=29,ci=6;    // amount 人数; ci 上午小组数
int m[45][45],desk[45][20];

int mem[6][45],prop[20][6],i,h;
int imp[45][2],comm[20][6][45];
int in[6],zhuo;                     // in[6] 中存放初始各桌人数比例
int temp[6][14],result[45][7];      // result 用于存放见面次数均衡情况表
int section,group,begin;            // section 为开的会议的届数

float mi[6];                        // mi[6] 中间变量,无实际意义
float aga[6],gong[6],ming[6],num[6];
float u1=10,u2=4,u3=8,u4=5,max=0;    // u1:主管 u2:次数 u3:比例 u4:公共成员

main()
{
  float g=0;

  clrscr();
```

```
 amount=inhouse;            //先排 inhouse 成员,令最大人数为 inhouse.
 ini_all();                 //初始化各个变量

 g=u1+u2+u3+u4;             //将权重归一
 u1=u1/g;u2=u2/g;u3=u3/g;u4=u4/g;

 printf("主管:%4.3f 次数:%4.3f 比例:%4.3f 公共成员:%4.3f\n\n",u1,u2,u3,u4);

 begin=0;                   //先排 inhouse 成员,令初始人号为 0 号.
 amount=inhouse;            //安排 inhouse 个成员.
 find(1,3,6,1);             //find(开始届数,结束届数,组数,是否有主管)
 find(3,7,4,0);
 out_put();
 cacu_prop();               //计算应有非 inhouse 成员比例.

 begin=inhouse;             //再排非 inhouse 成员,初始人号为 inhouse 号.
 amount=am;                 //安排剩余成员.
 find(0,3,6,1);             //find(开始届数,结束届数,组数,是否有主管)
 find(3,7,4,0);
 out_put();

 common_times();

 return;
}
find(int r,int p,int q,int noon)
{
 int j,k,l,a,b,c,d;

 section=p;
 group=q;

//====================== 各届会议排序主循环开始 ============================
for(h=r;h<section;h++)
{
 for(j=0;j<group;j++)
   num[j]=0;
 cacu_imp();                //计算各个成员的重要性
 queue_imp();               //各个成员权重排序

 for(j=0;j<group;j++)       //将第 j 桌上有谁清零,令排斥力均为零
  {
   for(k=0;k<45;k++)
```

```
      mem[j][k]=0;
    }
  for(j=0;j<group;j++)          //填入 in house 成员.
    {
     k=0;
     while(comm[h][j][k]!=0)
      {
       mem[j][comm[h][j][k]-1]=1;
       k=k+1;
      }
     }

//--------------------- 每届会议各桌人员安排循环开始 ----------------------

for(i=begin;i<amount;i++)     //重要性依次为 i 的人
 {
  for(j=0;j<group;j++)       //令排斥力均为零
     aga[j]=0;

  aga_meet();              //计算第 i 人由见面次数引起的排斥力
  if(noon==1)aga_officer(); //在同一主管下不能出现两次
  aga_prop();              //计算第 i 人由比例失衡引起的排斥力
  aga_common();            //计算第 i 人去各桌出现公共成员数引起的排斥力

  find_min_aga();          //找到阻力最小的一桌,将桌号附给 "zhuo"
  put_in();                //将重要性为 i 的人放入 "zhuo" 中
 }
//--------------------- 每届会议各桌人员安排循环结束 ----------------------
}
//====================== 各届会议排序主循环结束 ==========================

printf("****************** 到第 %u 届会议为止 ******************\n\n",h);

times_of_meet();           //输出见面次数总和
printf("\n最大公共成员数为: %2.0f\n\n",max+1);   //输出最大 common member 员数
times_of_everyone();        //输出每人同其他人见面次数统计值
return;
}

//---------------------------- 初始化各个变量 -----------------------------
ini_all()
{
 int l,j,k,a;
```

```
for(l=0;l<20;l++)          //初始化 prop[20][6]
  for(j=0;j<6;j++)
   prop[l][j]=1;

j=amount%ci;               //初始化 in[6]
k=(amount-j)/ci;
for(l=0;l<ci;l++)
  in[l]=k;
for(l=ci;l>=ci-j;l--)
  in[l]=in[l]+1;
for(l=0;l<45;l++)          //初始化 desk[45][20]
  for(j=0;j<20;j++)
   desk[l][j]=0;
a=0;
for(l=0;l<6;l++)
 {
  for(j=0;j<in[l];j++)
    desk[j+a][0]=l+1;
  a=a+in[l];
 }

for(l=0;l<45;l++)          //初始化 m[45][45]
  for(j=0;j<45;j++)
   m[l][j]=0;
a=0;
for(l=0;l<6;l++)
 {
  for(j=0;j<in[l];j++)
    for(k=0;k<in[l];k++)
      m[j+a][k+a]=1;
  a=a+in[l];
 }
for(j=0;j<45;j++)
  m[j][j]=0;

for(l=0;l<6;l++)           //初始化 temp[6][14]
  for(j=0;j<14;j++)
    temp[l][j]=0;
a=1;
for(l=0;l<6;l++)
 for(j=0;j<in[l];j++)
  {
   temp[l][j]=a;
   a=a+1;
```

```
  }

 for(l=0;l<20;l++)              //初始化 comm[i][j][k]
  for(j=0;j<6;j++)
   for(k=0;k<45;k++)
     comm[l][j][k]=0;

 for(l=0;l<6;l++)
  for(j=0;j<14;j++)
    comm[0][l][j]=temp[l][j];

 for(l=0;l<45;l++)              //初始化 imp[45][2]
  {
   imp[l][0]=l;
   imp[l][1]=0;
  }
 for(j=0;j<group;j++)           //初始化 mem[6][45]
  {
    for(k=0;k<45;k++)
      mem[j][k]=0;
  }
 return(1);
}

//---------------------- 计算各个成员的重要性 -------------------------
cacu_imp()
{
 int l,j,k;

 for(l=begin;l<amount;l++)
  { k=0;
    for(j=0;j<amount;j++)
      k=k+pow(m[l][j],2);
    imp[l][0]=l;imp[l][1]=k;
  }
 return(1);
}
//------------------------ 各个成员权重排序 ---------------------------
queue_imp()
{
 int l,j,k,f;

 for(l=begin;l<amount;l++)
    for(j=l;j<amount;j++)
```

```
    { if(imp[l][1]<imp[j][1])
       {
        k=imp[l][0];f=imp[l][1];
        imp[l][0]=imp[j][0];imp[l][1]=imp[j][1];
        imp[j][0]=k;imp[j][1]=f;
       }
    }
  return(1);
}
//---------- 计算第 i 人在各桌受到的由见面次数引起的排斥力 ------------
aga_meet()
{
  int j,k;
  float f=0;

  for(j=0;j<group;j++)              //第 j 桌
   {
    for(k=0;k<amount;k++)          //第 k 个人
     {
      if(mem[j][k]==1&&m[imp[i][0]][k]!=0)
      aga[j]=aga[j]+pow(m[imp[i][0]][k],4);
     }
   }
  for(j=0;j<group;j++)              //计算∑f(j)
    f=f+aga[j];

  for(j=0;j<group;j++)              //归一化
    aga[j]=f==0?0:aga[j]/f;

  for(j=0;j<group;j++)              //乘以系数 u2
   aga[j]=u2*aga[j];

  return;
}
//--------------- 计算第 i 人由比例失衡引起的排斥力 -----------------
aga_prop()
{
  int j;
  float g=0,num1[6];

  for(j=0;j<group;j++)
    num1[j]=(num[j]+1)/prop[h][j];

  for(j=0;j<group;j++)              //计算∑num(j)
```

```
   g=g+pow(num1[j],1);

  for(j=0;j<group;j++)          //归一化
   num1[j]=g==0?0:pow(num1[j],1)/g;

  for(j=0;j<group;j++)          //归一化
   aga[j]=aga[j]+u3*num1[j];

  return;
}
//------------ 在同一主管下不能出现两次,否则排斥力增加 ---------------
aga_officer()
{
 int a,j;
  for(j=0;j<h;j++)              //第 j 次会议
   {
    a=desk[imp[i][0]][j]-1;      //第 i 个人所在的桌号
    aga[a]=aga[a]+1*u1;          //阻力增加
   }
 return;
}
//---------------- 计算第 i 人去各桌会出现的公共成员数 ------------------
aga_common()
{
 int j,k,a,b,c,d;
 float g=0;

  for(j=0;j<group;j++)
   { ming[j]=0,mi[j]=0;}

  for(j=0;j<h;j++)                  //第 j 次会议
    for(k=0;k<6;k++)                 //第 k 桌
    { a=0;
     while(comm[j][k][a]!=0)          //第 a 个人
      {
       if(comm[j][k][a]==(imp[i][0]+1))
        {
         for(d=0;d<group;d++)gong[d]=0;
         for(b=0;b<group;b++)        //对应现在的 b 个桌子
          {
           c=0;
           while(comm[j][k][c]!=0)
           {
            if(mem[b][comm[j][k][c]-1]==1)
```

```
                gong[b]=gong[b]+1;
              c=c+1;
             }
           }
          for(d=0;d<group;d++)
           {
            if(ming[d]<gong[d])
              ming[d]=gong[d];
           }
         }
       a=a+1;
      }
    }

for(d=0;d<group;d++)              //计算∑ming(i)
 {
  mi[d]=ming[d];
  g=g+pow(ming[d],2);
 }
for(d=0;d<group;d++)              //归一化
  ming[d]=g==0?0:pow(ming[d],2)/g;

for(d=0;d<group;d++)              //公共成员的加权影响
  aga[d]=aga[d]+u4*ming[d];

return;
}
//-------------- 找到阻力最小的一桌,将桌号附给 "zhuo" ---------------
find_min_aga()
{
  int j;
  float min=10000;

   for(j=0;j<group;j++)
    if(aga[j]<min)
      { min=aga[j];zhuo=j;}
   if(max<mi[zhuo])max=mi[zhuo];    //找出 common member 最大者

  return;
}
//----------------- 将重要性为 i 的人放入 "zhuo" 中 -----------------
put_in()
{
   int k,d;
```

```
  for(k=0;k<amount;k++)           //第 k 个人
   {
    if(mem[zhuo][k]==1)
    {
      m[imp[i][0]][k]=m[imp[i][0]][k]+1;
      m[k][imp[i][0]]=m[k][imp[i][0]]+1;
    }
   }
  mem[zhuo][imp[i][0]]=1;
  num[zhuo]=num[zhuo]+1;
  desk[imp[i][0]][h]=zhuo+1;
  d=0;
  while(comm[h][zhuo][d]!=0)d++;
  comm[h][zhuo][d]=imp[i][0]+1;
  return;
}
//------------------------ 输出见面次数总和 ------------------------
times_of_meet()
{
 int j,k,w[8];

 for(j=0;j<8;j++)             //初始化 w[8]
  w[j]=0;

 for(k=0;k<45;k++)             //初始化 result[45][7]
  for(j=0;j<7;j++)
   result[k][j]=0;

 printf("                  见面次数总和统计 \n\n");
 for(k=0;k<amount;k++)
 { for(j=0;j<amount;j++)
   {
    result[k][m[k][j]]=result[k][m[k][j]]+1;
    if(m[k][j]==0)w[0]=w[0]+1;
    if(m[k][j]==1)w[1]=w[1]+1;
    if(m[k][j]==2)w[2]=w[2]+1;
    if(m[k][j]==3)w[3]=w[3]+1;
    if(m[k][j]==4)w[4]=w[4]+1;
    if(m[k][j]==5)w[5]=w[5]+1;
    if(m[k][j]==6)w[6]=w[6]+1;
    if(m[k][j]==7)w[7]=w[7]+1;
   }
 }
```

```
 for(k=0;k<8;k++)                    //输出最终见面总次数.
   printf(" %u 共出现 %u 次\n",k,w[k]/2);
 return;
}
//------------------- 输出每人同其他人见面次数统计值 -----------------
times_of_everyone()
{
 int j,k;

 printf("              每人同其他人见面次数统计值  \n\n");
 printf("        ");
 for(k=0;k<15;k++)
   printf("%2u ",k+1);
 printf("  (号)\n");
 printf("------------------------------------------------------------\n");
 for(k=0;k<7;k++)
  { printf("见%2u 次面:",k);
    for(j=0;j<15;j++)
      printf("%2u ",result[j][k]);
    printf("\n");
  }
 printf("==========================================================================\n");
 if(amount>15)
  {
   printf("        ");               //输出每人同其他人见面次数统计值
   for(k=15;k<amount;k++)
     printf("%2u ",k+1);
   printf("  (号)\n");
   printf("------------------------------------------------------------\n");
   for(k=0;k<7;k++)
    { printf("见%2u 次面:",k);
      for(j=15;j<amount;j++)
        printf("%2u ",result[j][k]);
      printf("\n");
    }
   printf("==========================================================================\n\n");
  }
 return;
}
out_put()
{
  int j,k,l;
  j=0;
```

```
 while(comm[j][0][0]!=0&&j<20)
  {
   k=0;
   printf(" **************** 第 %u 届会议 **************\n",j+1);
   while(comm[j][k][0]!=0&&k<6)
    {
     l=0;
     printf(" 第 %u 组: ",k+1);
     while(comm[j][k][l]!=0&&l<amount)
      {
       if(comm[j][k][l]<=inhouse)
          printf("(%-1u) ,",comm[j][k][l]);
       else
          printf("%-4u,",comm[j][k][l]);
       l=l+1;
      }
     printf("\n");
     k=k+1;
    }
   printf("\n");
   j=j+1;
  }
  return;
}
//------------------ 计算应有非 inhouse 成员比例 ----------------------
cacu_prop()
{
 int d,j,k;

  for(j=0;j<20;j++)
    for(k=0;k<6;k++)
     {
       d=0;
       while(comm[j][k][d]!=0)d++;
       prop[j][k]=d;
     }
  return;
}
//---------------- 计算各小组公共成员数及最大公共成员数 ----------------
common_times()
{
 int j,k,l,m,u,maa,ww[45][2];
 int a,maxx,eee[10][6];
```

```
for(j=0;j<10;j++)
  for(k=0;k<6;k++)
    eee[j][k]=0;

for(j=0;j<am;j++)
  for(k=0;k<2;k++)
    ww[j][k]=0;

for(j=0;j<7;j++)
{
for(k=0;k<6;k++)
{
    u=0;
    while(comm[j][k][u]!=0)
    { ww[comm[j][k][u]-1][1]=1;
      u=u+1;
    }
    maxx=0;
    for(l=0;l<7;l++)
    {
    for(m=0;m<6;m++)
    {
      maa=0;
      if(l!=j&&m!=k)
      {
        u=0;
        while(comm[l][m][u]!=0)
        {
          ww[comm[l][m][u]-1][0]=1;
          u=u+1;
        }
       for(a=0;a<am;a++)
        {
         if(ww[a][0]==1&&ww[a][1]==1)
          maa=maa+1;
        }
       for(a=0;a<am;a++)
         ww[a][0]=0;
      }
      if(maxx<maa)maxx=maa;
    }
    }
    eee[j][k]=maxx;
    for(a=0;a<am;a++)
```

```
    {ww[a][1]=0;ww[a][0]=0;}
 }
 }
 for(j=0;j<10;j++)
  {
   for(k=0;k<6;k++)
    if(eee[j][k]==0)
     printf("   ",eee[j][k]);
    else
     printf("%2u  ",eee[j][k]);
   printf("\n");
  }

//------------ 输出最大 common member 员数,及其次数. -----------------
 printf("\n最大公共成员数为: %2.0f\n\n",max+1);
 maxx=0;
 for(j=0;j<10;j++)
   for(k=0;k<6;k++)
    if(eee[j][k]==(max+1))
     maxx=maxx+1;

 printf("最大公共成员出现次数为: %u\n\n",maxx);
 return;
 }
```

最终结果

主管:0.455 次数:0.091 比例:0.364 公共成员:0.091

****************** 到第 7 届会议为止 ******************

见面次数总和统计

0 共出现 69 次
1 共出现 165 次
2 共出现 158 次
3 共出现 28 次
4 共出现 0 次

最大公共成员数为: 3

每人同其他人见面次数统计值

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15 (号)
见 0 次面:	6	3	4	1	4	3	5	5	4	2	6	4	6	9	7
见 1 次面:	12	12	16	16	12	14	15	13	12	15	7	9	11	10	9
见 2 次面:	10	12	5	9	11	11	7	8	11	11	15	13	10	10	12
见 3 次面:	1	2	4	3	2	1	2	3	2	1	1	3	2	0	1
见 4 次面:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	16	17	18	19	20	21	22	23	24	25	26	27	28	29 (号)
见 0 次面:	5	5	5	3	4	7	6	6	7	5	2	5	2	8
见 1 次面:	12	10	12	14	5	10	11	10	13	12	12	10	13	3
见 2 次面:	12	11	11	9	17	9	11	12	6	9	13	12	13	16
见 3 次面:	0	3	1	3	3	3	1	1	3	3	2	2	1	2
见 4 次面:	0	0	0	0	0	0	0	0	0	0	0	0	0	0

**************** 第 1 届会议 ***************

第 1 组: (1) ,13 ,22 ,
第 2 组: (2) ,14 ,23 ,
第 3 组: (3) ,15 ,24 ,
第 4 组: (4) ,(5) ,10 ,16 ,19 ,25 ,28 ,
第 5 组: (6) ,(7) ,11 ,17 ,20 ,26 ,29 ,
第 6 组: (8) ,(9) ,12 ,18 ,21 ,27 ,

**************** 第 2 届会议 **************
第 1 组: (4) ,(9) ,11 ,20 ,28 ,12 ,15 ,23 ,
第 2 组: (5) ,(3) ,17 ,25 ,18 ,13 ,
第 3 组: (6) ,(2) ,10 ,26 ,27 ,22 ,
第 4 组: (7) ,29 ,21 ,
第 5 组: (8) ,16 ,24 ,
第 6 组: (1) ,19 ,14 ,

**************** 第 3 届会议 **************
第 1 组: (5) ,(8) ,26 ,17 ,19 ,21 ,
第 2 组: (4) ,(1) ,11 ,27 ,29 ,16 ,
第 3 组: (9) ,25 ,14 ,
第 4 组: (6) ,(3) ,20 ,12 ,13 ,22 ,
第 5 组: (2) ,28 ,18 ,15 ,
第 6 组: (7) ,10 ,23 ,24 ,

**************** 第 4 届会议 **************
第 1 组: (4) ,(8) ,(2) ,20 ,26 ,13 ,25 ,15 ,24 ,
第 2 组: (5) ,(9) ,11 ,22 ,27 ,23 ,
第 3 组: (6) ,(1) ,12 ,29 ,19 ,10 ,18 ,
第 4 组: (3) ,(7) ,17 ,28 ,16 ,21 ,14 ,

**************** 第 5 届会议 **************
第 1 组: (4) ,(3) ,26 ,28 ,29 ,22 ,18 ,
第 2 组: (5) ,(2) ,(7) ,20 ,12 ,27 ,19 ,23 ,24 ,
第 3 组: (6) ,(9) ,17 ,15 ,25 ,16 ,
第 4 组: (8) ,(1) ,11 ,13 ,21 ,10 ,14 ,

**************** 第 6 届会议 **************
第 1 组: (4) ,(7) ,12 ,25 ,13 ,21 ,
第 2 组: (5) ,(1) ,26 ,23 ,15 ,18 ,16 ,
第 3 组: (2) ,(3) ,(9) ,20 ,17 ,19 ,29 ,22 ,10 ,
第 4 组: (6) ,(8) ,11 ,27 ,28 ,24 ,14 ,

**************** 第 7 届会议 **************
第 1 组: (2) ,(9) ,(1) ,12 ,29 ,26 ,25 ,28 ,21 ,
第 2 组: (4) ,(6) ,17 ,19 ,23 ,13 ,14 ,
第 3 组: (5) ,(7) ,11 ,22 ,15 ,24 ,
第 4 组: (3) ,(8) ,20 ,27 ,10 ,18 ,16 ,

2 1 2 2 3 3
3 2 2 2 2 2

2 2 2 3 2 3
3 3 3 2
3 3 2 2
2 2 3 2
3 2 3 3

最大公共成员数为: 3

最大公共成员出现次数为: 14

无 inhouse 成员,无主管

主管:0.750 次数:0.100 比例:0.050 公共成员:0.100

****************** 到第 7 届会议为止 ******************

见面次数总和统计

0 共出现 63 次
1 共出现 189 次
2 共出现 160 次
3 共出现 8 次
4 共出现 0 次

最大公共成员数为: 3

每人同其他人见面次数统计值

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15 (号)
见 0 次面:	5	5	4	5	2	4	4	4	4	5	5	4	6	4	5
见 1 次面:	14	11	14	13	16	14	16	14	13	13	12	14	10	12	9
见 2 次面:	10	13	10	10	10	11	8	10	11	10	11	10	13	13	15
见 3 次面:	0	0	1	1	1	0	1	1	1	1	1	1	0	0	0
见 4 次面:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

===

	16	17	18	19	20	21	22	23	24	25	26	27	28	29 (号)
见 0 次面:	4	5	4	3	3	6	5	3	4	4	5	5	5	5
见 1 次面:	13	13	13	15	16	10	14	16	13	11	13	12	13	11
见 2 次面:	12	10	12	11	9	13	9	9	12	14	10	11	10	13

见 3 次面: 0 1 0 0 1 0 1 1 0 0 1 1 1 0
见 4 次面: 0 0 0 0 0 0 0 0 0 0 0 0 0 0
==

**************** 第 1 届会议 ***************
第 1 组: 1 ,2 ,3 ,4 ,
第 2 组: 5 ,6 ,7 ,8 ,9 ,
第 3 组: 10 ,11 ,12 ,13 ,14 ,
第 4 组: 15 ,16 ,17 ,18 ,19 ,
第 5 组: 20 ,21 ,22 ,23 ,24 ,
第 6 组: 25 ,26 ,27 ,28 ,29 ,

**************** 第 2 届会议 ***************
第 1 组: 5 ,11 ,17 ,23 ,29 ,
第 2 组: 6 ,12 ,18 ,24 ,2 ,
第 3 组: 7 ,13 ,19 ,25 ,3 ,
第 4 组: 8 ,14 ,20 ,26 ,4 ,
第 5 组: 9 ,15 ,21 ,27 ,1 ,
第 6 组: 10 ,16 ,22 ,28 ,

**************** 第 3 届会议 ***************
第 1 组: 5 ,12 ,19 ,26 ,1 ,
第 2 组: 6 ,13 ,20 ,27 ,16 ,
第 3 组: 7 ,14 ,21 ,29 ,
第 4 组: 8 ,15 ,23 ,2 ,28 ,
第 5 组: 9 ,17 ,24 ,3 ,10 ,
第 6 组: 11 ,18 ,25 ,22 ,4 ,

**************** 第 4 届会议 ***************
第 1 组: 5 ,13 ,15 ,24 ,22 ,14 ,4 ,29 ,
第 2 组: 6 ,11 ,19 ,23 ,27 ,3 ,28 ,
第 3 组: 8 ,12 ,17 ,25 ,1 ,21 ,16 ,
第 4 组: 9 ,18 ,20 ,26 ,10 ,2 ,7 ,

**************** 第 5 届会议 ***************
第 1 组: 14 ,23 ,5 ,18 ,16 ,3 ,10 ,8 ,
第 2 组: 9 ,29 ,28 ,20 ,12 ,19 ,15 ,25 ,
第 3 组: 4 ,27 ,24 ,11 ,2 ,7 ,21 ,
第 4 组: 22 ,13 ,1 ,6 ,26 ,17 ,

**************** 第 6 届会议 ***************
第 1 组: 14 ,15 ,25 ,27 ,2 ,3 ,17 ,26 ,
第 2 组: 28 ,5 ,24 ,12 ,18 ,11 ,20 ,

第 3 组: 23 ,29 ,9 ,16 ,1 ,13 ,7 ,
第 4 组: 19 ,10 ,4 ,8 ,21 ,22 ,6 ,

**************** 第 7 届会议 ***************
第 1 组: 15 ,12 ,4 ,3 ,23 ,13 ,20 ,
第 2 组: 14 ,27 ,28 ,18 ,22 ,7 ,1 ,
第 3 组: 25 ,5 ,9 ,11 ,2 ,16 ,21 ,6 ,
第 4 组: 24 ,29 ,8 ,26 ,10 ,19 ,17 ,

2 3 2 2 2 3
2 2 2 2 2 2
2 2 2 2 3 2
2 2 2 2
2 3 2 2
3 2 2 2
3 2 3 3

最大公共成员数为: 3

最大公共成员出现次数为: 8

无 inhouse 成员,有主管

主管:0.750 次数:0.100 比例:0.050 公共成员:0.100

****************** 到第 7 届会议为止 ******************

见面次数总和统计

0 共出现 66 次
1 共出现 184 次
2 共出现 160 次
3 共出现 10 次
4 共出现 0 次

最大公共成员数为: 3

每人同其他人见面次数统计值

```
        1  2  3  4  5  6  7  8  9 10 11 12 13 14 15  (号)
------------------------------------------------------------
见 0 次面: 8  7  7  4  4  3  3  4  6  4  5  4  4  4  3
见 1 次面: 7 10 10 16 15 16 14 11 10 17 13 13 13 15 16
见 2 次面:13 11 11  8  9 10 12 13 13  7  9 12 11  9  8
见 3 次面: 1  1  1  1  1  0  0  1  0  1  2  0  1  1  2
见 4 次面: 0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
见 5 次面: 0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
见 6 次面: 0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
============================================================
       16 17 18 19 20 21 22 23 24 25 26 27 28 29  (号)
------------------------------------------------------------
见 0 次面: 5  4  3  5  6  3  3  5  6  5  5  4  5  4
见 1 次面:10 14 15 13 10 14 13 12 12 10 13 13 11 12
见 2 次面:14 11 10 11 13 12 13 11 10 14 10 11 12 12
见 3 次面: 0  0  1  0  0  0  0  1  1  0  1  1  1  1
见 4 次面: 0  0  0  0  0  0  0  0  0  0  0  0  0  0
见 5 次面: 0  0  0  0  0  0  0  0  0  0  0  0  0  0
见 6 次面: 0  0  0  0  0  0  0  0  0  0  0  0  0  0
============================================================


**************** 第 1 届会议 ***************
第 1 组: 1   ,2   ,3   ,4   ,
第 2 组: 5   ,6   ,7   ,8   ,9   ,
第 3 组: 10  ,11  ,12  ,13  ,14  ,
第 4 组: 15  ,16  ,17  ,18  ,19  ,
第 5 组: 20  ,21  ,22  ,23  ,24  ,
第 6 组: 25  ,26  ,27  ,28  ,29  ,

**************** 第 2 届会议 ***************
第 1 组: 5   ,11  ,17  ,23  ,29  ,
第 2 组: 10  ,15  ,20  ,25  ,2   ,
第 3 组: 6   ,16  ,21  ,26  ,3   ,
第 4 组: 7   ,12  ,22  ,27  ,4   ,
第 5 组: 8   ,13  ,18  ,28  ,1   ,
第 6 组: 9   ,14  ,19  ,24  ,



**************** 第 3 届会议 ***************
第 1 组: 6   ,12  ,18  ,9   ,24  ,
第 2 组: 11  ,13  ,21  ,27  ,19  ,
```

第 3 组: 5 ,15 ,22 ,28 ,2 ,
第 4 组: 8 ,20 ,23 ,26 ,14 ,
第 5 组: 7 ,16 ,25 ,29 ,4 ,
第 6 组: 10 ,17 ,1 ,3 ,

**************** 第 4 届会议 ***************
第 1 组: 9 ,13 ,23 ,2 ,16 ,27 ,10 ,
第 2 组: 7 ,15 ,24 ,21 ,3 ,28 ,8 ,17 ,
第 3 组: 6 ,20 ,29 ,22 ,12 ,1 ,19 ,
第 4 组: 11 ,25 ,4 ,18 ,5 ,26 ,14 ,

**************** 第 5 届会议 ***************
第 1 组: 15 ,3 ,6 ,14 ,4 ,27 ,23 ,
第 2 组: 13 ,7 ,28 ,20 ,26 ,9 ,19 ,
第 3 组: 11 ,8 ,22 ,24 ,10 ,29 ,18 ,16 ,
第 4 组: 25 ,21 ,12 ,17 ,2 ,5 ,1 ,

**************** 第 6 届会议 ***************
第 1 组: 11 ,28 ,2 ,9 ,16 ,14 ,21 ,
第 2 组: 13 ,8 ,15 ,29 ,26 ,5 ,12 ,23 ,
第 3 组: 3 ,25 ,18 ,22 ,7 ,1 ,27 ,19 ,
第 4 组: 24 ,17 ,4 ,20 ,10 ,6 ,

**************** 第 7 届会议 ***************
第 1 组: 8 ,29 ,1 ,9 ,27 ,21 ,20 ,4 ,
第 2 组: 5 ,3 ,26 ,24 ,19 ,10 ,
第 3 组: 11 ,7 ,23 ,18 ,2 ,15 ,12 ,
第 4 组: 13 ,22 ,25 ,28 ,16 ,17 ,14 ,6 ,

2 2 2 2 2 2
3 2 2 3 2 2
2 2 2 3 2 2
2 2 3 2
2 2 2 2
3 3 3 2
3 2 3 3

最大公共成员数为: 3

最大公共成员出现次数为: 10

会 议 最 优 分 派 方 案

【摘要】

本文从保证会议成员尽量交叉混合的目的出发，将四条标准的综合满足程度作为成员分派方案的衡量依据，通过类比，引入物理学中“排斥力”的概念，将四条标准的综合作用通过一个多目标函数用排斥力形象地刻划出来。并根据此公式，制定了成员的分派规则和相应的算法。计算机求解出的结果和模型稳定性分析进一步证明了公式的构造是合理的。

在模型推广中，针对那些“某些成员在最后一分钟放弃参加会议或某些没有计划在内的成员要来参加会议”和“各种类型参加者数目任意”的会议分派情况，我们将程序加以适当调整，使这些问题得以解决。

【关键词】

排斥力　　多目标决策　　权系数　　当前状态

一、问题重述

为避免出现会议讨论被权威人士控制的现象，通常安排数届会议分组进行讨论，各届会议中小组成员不同，以使与会人员尽量交叉混合。

An Tosal 公司的某一会议将由 29 个董事会成员参加，其中 9 人是内部成员。该会议是全天的，上午安排 3 届，下午安排 4 届。上午各届会议均由 6 个讨论小组组成，每个小组由公司的 6 名上级主管之一主持（这些主管都不是董事会成员）。主管们不参加下午的会议，下午的会议每届由 4 个不同的讨论小组组成。

公司总裁想要一份分派表，分派表应尽可能地使成员交叉混合。理想的分派应满足以下几个标准：

●不同届会议中各组公共成员的数目最少；

●每一成员和其他任一成员分在同一小组的次数相等；

●在上午会议中，董事会成员不得在同一主管带领的讨论小组中出现两次；

●讨论小组中的内部成员比例要适当。

给出一个成员 1～9 和 10～29 以及主管 1～6 的分派表。评估上述标准满足的如何。因为可能某些成员在最后一分钟放弃参加会议或某些没有计划在内的成员要来参加会议，所以希望给出一份规则，使秘书可以据此按一小时内的通知重新分派。如果这份规则还可以用于那些各类参加者数目任意的会议分派问题将更为理想。

二、模型假设

●会议是否会出现被权威人士控制的现象，完全取决于交叉混合的好坏。
●制定分派方案时，只考虑尽量使成员交叉混合，而不考虑成员的主观要求。
●参加会议的成员只存在身份的不同，同一身份的成员的工作能力是相同的。
●为了保证会议的效果，每组必须包含各种身份的成员。

三、问题的分析

1、对衡量标准的数量化

原问题中给出了四条衡量分派方法好坏的标准，为使问题更加明确，我们对这些衡量标准进行数量化分析。

●公共成员数目是指不同届会议的两小组间相同成员个数。然而，对任一确定的小组来说，与不同的小组比较会有不同的公共成员数。在这里，我们定义其中最大的公共成员数作为该小组的公共成员数（以下简称公共成员数），并将所有小组对应的公共成员数中的最大值及其出现次数作为衡量这条关于公共成员的标准满足情况的依据。

●若设 $x_i^{[j]}$ 为第 j 届会议第 i 个小组的人数，当满足“每个董事会成员与其他任一董事会成员分在同一讨论小组的次数相等”时，则有以下公式成立：

$$\sum_{j=1}^{3}\sum_{i=1}^{6}(x_i^{[j]})^2+\sum_{j=4}^{7}\sum_{i=1}^{4}(x_i^{[j]})^2=29(28s+7) \qquad 〖1〗$$

(其中 s 为每一董事会成员同其他任一董事会成员分在同一小组的次数，以下简称“**见面次数**”)

上式即为“每个董事会成员和其他任一董事会成员分在同一讨论小组的次数相等”成立的必要条件。

证明如下：

$\because$ 第 j 届会议第 i 个小组中任意一个在该小组所见人数为：($x_i^{[j]}-1$)，从而该组 $x_i^{(j)}$ 个人所见人数之和为：$x_i^{[j]}(x_i^{[j]}-1)$.

$\therefore$ 对全天会议来说，所有成员所见人数总和为：

$$\sum_{j=1}^{3}\sum_{i=1}^{6}x_i^{[j]}(x_i^{[j]}-1)+\sum_{j=4}^{7}\sum_{i=1}^{4}x_i^{[j]}(x_i^{[j]}-1)$$

$$=\sum_{j=1}^{3}\sum_{i=1}^{6}(x_i^{[j]})^2+\sum_{j=4}^{7}\sum_{i=1}^{4}(x_i^{[j]})^2-29\times 7$$

若每一董事会成员和其他任一董事会成员分在同一讨论小组的次数均为 s 次，则全天内所有成员所见人数总和又可写为: $29\times 28s$.

即：

$$\sum_{j=1}^{3}\sum_{i=1}^{6}(x_i^{[j]})^2+\sum_{j=4}^{7}\sum_{i=1}^{4}(x_i^{[j]})^2-29\times 7=29\times 28s$$

整理得：

$$\sum_{j=1}^{3}\sum_{i=1}^{6}(x_i^{[j]})^2+\sum_{j=4}^{7}\sum_{i=1}^{4}(x_i^{[j]})^2=29(28s+7)$$

●对于“董事会成员不能在同一主管带领的讨论小组中出现两次”，我们用以下方式加以描述：

设 R_{ij} 为编号为 i 的成员在编号为 j 的主管带领的讨论小组中出现的次数，则有：

$$\max\{R_{i1},R_{i2},R_{i3},R_{i4},R_{i5},R_{i6}\}=1\quad,\qquad \sum_{j=1}^{6}R_{ij}=3$$

$$(i=1,2,\cdots\cdots 29)$$

●在这里，我们定义第 j 届会议第 i 个小组的内部成员比例为：

$$\frac{I_i^{[j]}}{x_i^{[j]}}=\frac{\text{第j届会议第i个小组内部成员数}}{\text{第j届会议第i个小组总成员数}}$$

$$\left(I_i^{[j]}\text{为第}j\text{届会议第}i\text{个小组内部成员数}\right)$$

对于 An Tosal 公司而言，最理想的内部成员比例为 9/29，显然该比例是不可能严格达到的，那么所能达到的最适当比例应使

$$\sum_i\left(\frac{I_i^{[j]}}{x_i^{[j]}}-\frac{9}{29}\right)^2 \qquad 〖2〗$$

达到最小。在这里，保证每一小组至少有一名内部成员是“比例适当”最基本的要求（见假设）。下表给出了 9 名内部成员上、下午所有可能的分派方案及对应的使〖2〗式达到最小的董事会成员人数分派：

	内部成员人数分派	$\min\sum_i\left(\frac{I_i^{[j]}}{x_i^{[j]}}-\frac{9}{29}\right)^2$
上午	1 1 1 2 2 2	3 3 3 7 7 6
		3 3 3 3 7 10
	1 1 1 1 2	4 3 3 3 3 13

	3 1　1　1　1　1 4	
下午	1　1　1　6 1　1　2　5 1　1　3　4 1　2　2　4 1　3　2　3 2　2　2　3	3　3　3　20 3　3　7　16 3　3　10　13 3　6　7　13 3　10　6　10 7　6　6　10

表　1。

2、绝对理想的分派存在吗?

我们认为绝对理想的分派应当是同时严格满足题中四条标准的分派。此时由上文中衡量标准的数量化可知，〖1〗〖2〗两式应当同时成立。利用 Cauchy 不等式，通过数学推导，我们发现：当严格满足〖1〗〖2〗式时，全天会议中公共成员数至少为 11【推导见**附录 1**】。显然，这样的分派根本不是人们所期望的交叉混合。故：

同时严格满足四条标准的绝对理想分派是不存在的。

3、对问题的进一步分析

基于上述分析，可知四条标准不能同时严格满足。那么，若仅以其中某一标准是否严格满足作为衡量成员交叉混合好坏的唯一依据，可行吗？我们在论证绝对理想的分派是否存在的过程中发现：这四条标准间存在着某些内在的微妙的联系，仅以一条标准作为衡量依据，一味追求单一标准的严格满足，将会使其他标准的满足程度发生相应的改变，这种改变通常是向着不理想的方向发展的。因此，我们应该将四条标准综合考虑，寻找一种使它们都尽可能满足的规则。基于上述分析，我们设计了如下模型。

四、模型设计

1、模型的思路设计

由前文，绝对理想的分派是不存在的，仅仅追求单一标准严格满足的作法也是不可行的，那么，寻找一种使四条标准都尽可能满足的规则是当前的关键。

通常，人们在安排某届会议时，总是根据前几届会议成员的安排情况来决定本届会议成员的分派。因此，我们从以下两个方面出发进行思路设计：

●假设目前已安排了 $j-1$ 届会议，在安排第 j 届会议时，首先应该根据每个

成员在前 $j-1$ 届会议中与其他成员见面次数由多到少的顺序，对成员进行排序。在制订分派计划时，应以此顺序对成员加以考虑。

因为，如果先安排那些同其他成员见面次数少的人，一旦他们进入各组后，再安排那些与其他成员见面次数较多的人时，就会出现无论这些人被分派到哪一组，都有可能同已见过多次面的成员分在同一组。为避免以上情况发生，我们应优先考虑分派那些同其他成员见面次数较多的人。我们称这种排序为**成员的优先排序**。

●在成员的优先排序方案确定之后，就可以依次对各个成员的分派加以考虑了。假定前 $i-1$ 个成员已被编入各个小组内，由于第 i 名成员被编入不同的小组产生的效果是不同的，而效果的好坏又是以四条标准的满足程度来衡量的。因此，通过对不同小组四条标准满足程度的对比，可确定究竟将其编入哪一小组。由于这四条标准对某个成员来说都是限制性的。类比于物理学中“**排斥力**”的概念，可将每条标准的限制都看作是对各成员的一种排斥作用，而限制的大小可形象地刻划为排斥力的大小。如果我们希望以四条标准的综合满足程度来描述交叉混合好坏的话，则这种综合满足程度可看作是各排斥力的一个合力。于是我们可以通过比较各小组对第 i 名成员排斥合力（以下简称**排斥力**）的大小来决定该成员应编入哪个小组。对某个成员来说，他最终被编入的小组必是对他排斥力最小的小组。

下面我们将通过数学公式来分别刻划以上两个方面。

2、思路实现

（1）成员的优先排序准则

由于存在两种类型的成员：内部成员和非内部成员，为使内部成员比例适当，应首先分派九名内部成员（原因见后），而对同一种身份的成员来说可按见面次数多少排序。通常情况下，可用某一成员与其他成员见面次数之和来描述。

设 m 为成员总数,若以 M_{ij} 表示此届会议前编号为 i 的成员(以下简称成员 i)与编号为 j 的成员(以下简称成员 j)的见面次数，P_i 表示成员 i 与其他 $m-1$ 个成员见面次数之和。

则： $P_i = \sum_{\substack{j=1 \\ j\neq i}}^{m} M_{ij}$ （$M_{ij} = M_{ji}$ ，$i, j = 1,2,\cdots\cdots,m.$）

但是会出现这样一种情况：例如，若以 A、B 表示两个不同的成员，他们与相同的四个人见面次数如下：

A: (1,1,1,1)

B: (3,1,0,0)

若用式：$P_i = \sum_{\substack{j=1 \\ j\neq i}}^{m} M_{ij}$ 来衡量 A、B 的优先考虑次序，则二者的考虑次序是相同的，但是，如果从尽量使每一成员与其他成员见面次数相等的角度出发，由

于 B 可选择余地相对 A 来说要小，故应优先考虑 B。这种由 P_i 得到的结果和我们期望得到的结果的矛盾是由于以下原因造成的：

对具有可比性的同一群人来说，仅以上面 P_i 表达式作为优先排序准则只注重了成员间见面次数总和的大小，而无法体现不同的两个人在与其他在人见面次数上的“单个差异”。为了消除这种影响，我们对上式进行修正，得出了能够同时考虑见面的总和及单个差异的影响的 P_i 表达式如下：

$$P_i=\sum_{\substack{j=1\\j\neq i}}^{m}(M_{ij})^k \qquad \text{（k 为自然数）} \qquad 〖3〗$$

在这里，我们通过 $(M_{ij})^k$ 来放大不同的两个人在与其他人见面次数上的单个差异，k 的取值越大，这种单个差异的放大程度也就越大。同时，$\sum_{\substack{j=1\\j\neq i}}^{m}(M_{ij})^k$ 还能充分体现出见面次数总和的影响。这样，P_i 就综合考虑了差异的**灵敏性**和次数的**全面性**两方面的因素。通常，k 的取值为 $2\le k\le 4$ 就能充分反映出“单个差异”的特点，本题中我们取 k=2。由以上分析得到，成员的优先排序方法如下：

首先，由公式〖3〗计算出编号为 i 的成员的 P_i （i=1,2,…,m)值，然后，按照 P_i 值大小对所有成员依次排序，P_i 值越大考虑时越优先。

（2）排斥力公式的确定：

在成员的优先排序确定之后，考虑成员 i 编入哪个小组时，应首先计算各小组对他的排斥力大小。下面我们分别给出四条标准对应的排斥分力公式：

●见面次数因素对应的排斥分力：

在这里，这个排斥分力是由编号为 j 的小组（以下简称“小组 j”）内的成员与成员 i 的见面次数产生的。类似于〖3〗式，我们构造见面次数因素下的排斥分力 $F_{j1}^{(i)}$ 如下：

$$F_{j1}^{[i]}=\sum_{l}(M_{il})^k \qquad \text{（}l\text{ 为小组 }j\text{ 内成员编号, k 为自然数）}$$

〖4〗

但不同于成员优先排序准则的是，这里更侧重了见面次数的灵敏性，因为为了使某一成员与其他成员的见面次数尽量均衡，我们应尽量加大不同见面次数对应的排斥分力之间的差距。所以〖4〗式中 k 的取值应比〖3〗式中大一些，本题中我们取 k=4。

●公共成员因素对应的排斥分力

以 $C_j^{[i]}$ 表示将成员 i 编入小组 j 后，该小组与前几届会议中的各小组间公共成员数的最大值。我们构造排斥分力 $F_{j2}^{[i]}$ 为：

$$F_{j2}^{[i]}=(C_j^{[i]})^k \qquad (k=1,2,3,\cdots\cdots) \qquad 〖5〗$$

出于使公共成员数尽量小的考虑，我们取 $C_j^{[i]}$ 的 k 次幂以提高公式的灵敏性，

使得公共成员稍有增加，就引起排斥力的显著变化。经过反复计算验证，对本题来说，k=2 时效果最好。

●主管因素对应的排斥分力

不妨设主管 j 带领的小组组号始终为 j，以 $F_{j3}^{[i]}$ 表示此标准下小组 j 对成员 i 的排斥分力。则：

$$F_{j3}^{[i]} = \begin{cases} 0 \\ 1 \end{cases}$$ （若成员 i 未在主管 j 带领的小组中出现过，取 0；否则取 1。）〖6〗

●比例因素对应的排斥分力

为了便于保证本题中内部成员比例适当，在进行成员分派时，我们决定先安排内部成员。这是因为内部成员的数量较少，某小组内单个内部成员的变动会造成内部成员比例变动很大，所以应尽量先安排内部成员。

在会议安排时，如果成员均为同一种身份，就不存在比例的影响，为了使成员更好地交叉混合，应尽量使各组人数均衡，这样各组人数的多少就会对成员的进入产生一种排斥分力，可用如下公式刻划：

$F_{j4}^{[i]} = N_j^{[i]}$ （$N_j^{[i]}$ 为成员 i 编入前小组 j 中已有成员数）

在这个排斥分力作用下，成员 i 会倾向于进入人数最少的一组。在安排内部成员分派时，就属于这种情况。

在内部成员按上述法则安排就绪后，我们就可以按各组中内部成员的人数，来安排非内部成员。设小组 j 中内部成员数为 I_j，则 $\dfrac{I_j}{N_j^{[i]}+1}$ 为非内部成员 i 进入小组 j 后该组的内部成员比例，为使比例适当，应尽量使其比值趋近 $\dfrac{9}{29}$，我们以 $\left(\dfrac{9}{29}\middle/\dfrac{I_j}{N_j^{[i]}+1}\right)$ 描述内部成员比例适当的满足程度，其比值越大，成员 i 进入该组的困难越大。所以我们用其刻划内部成员比例因素对应的排斥分力 $F_{j4}^{[i]}$，即：

$$F_{j4}^{[i]} = \frac{9(N_j^{[i]}+1)}{29I_j}$$

综上所述

$$F_{j4}^{[i]} = \begin{cases} N_j^{[i]} & for.in-house.member \\ \dfrac{9(N_j^{[i]}+1)}{29I_j} & for.non-in-house.member \end{cases} \qquad 〖7〗$$

至此，我们已将四个标准对应的排斥分力分别刻划出来。下面对它们进行合成。由于这些分力都对成员 i 的进入起一种阻碍作用，类比于物理学中同一直线上多个力的合成，可考虑将其进行代数相加。但是，由于各标准对应的排斥分力的量纲不同，故需先对其进行归一化处理。

设共有 n 个小组，则归一化后小组 j 对成员 i 的第 l 个排斥分力 $f_{jl}^{[i]}$ 为：

$$f_{jl}^{[i]}=\frac{F_{jl}^{[i]}}{\sum_{k=1}^{n}F_{kl}^{[i]}} \qquad (l=1,2,3,4) \tag{〖8〗}$$

另外，由于在制定方案时,对各条标准满足程度的要求不同，这就要求我们对不同标准下的排斥分力加以不同的权系数，从而可通过改变权系数来满足要求。应用多目标决策中的线性加权和法，我们确定合成后的排斥力 $f_j^{[i]}$ 如下：

$$f_j^{[i]}=\lambda_1\cdot f_{j1}^{[i]}+\lambda_2\cdot f_{j2}^{[i]}+\lambda_3\cdot f_{j3}^{[i]}+\lambda_4\cdot f_{j4}^{[i]}=\sum_{l=1}^{4}\lambda_l\cdot f_{jl}^{[i]} \tag{〖9〗}$$

（λ_l 为第 l 个排斥分力的权系数，且 $\sum_{l=1}^{4}\lambda_l=1$）

由于题中要求“在上午会议中，董事会成员不得在同一主管带领的讨论小组中出现两次”，我们应将权系数 λ_3 取得足够大，使得该条标准得以严格满足，本题中取 $\lambda_3=0.8$。但实际中并非总是要求这条标准得以严格满足，对此，我们可改变 $F_{j3}^{[i]}$ 的表达式使之适用于不同情况，例如可使 $F_{j3}^{[i]}=D_j^{[i]}$ （$D_j^{[i]}$ 为成员 i 在主管 j 带领的小组下出现的次数)，这样分派策略更加灵活。

在排斥力公式确定后，将成员 i 编入小组 j 的条件为：

$$f_j^{[i]}=\min\left\{f_1^{[i]},f_2^{[i]},\cdots\cdots,f_n^{[i]}\right\}$$

五、模型求解

1、算法

（1）状态变量的定义

由于该模型在求解过程中，将某成员编入哪一组取决于前几届会议的安排情况，故模型求解的关键就在于如何存储和利用前几届会议安排的信息。在这里，我们定义“当前状态”为前几届会议的安排情况。主要包括以下几个方面。

●前几届会议中，任意两成员之间的见面次数。

它是确定成员优先排序准则及由见面次数引起的排斥分力的决定因素。程序中，通过数组 m[29][29]来存储。m[i][j]表示成员 i 与成员 j 在前几届会议中的见面次数(i , j = 1,2,⋯⋯,29)。

●某届会议中处于某个小组中的成员

它决定着公共成员数目的计算。由数组 comm[7][6][29]存储。

comm[i][j][k]=1 表示第 i 届会议中成员 k 处于小组 j；

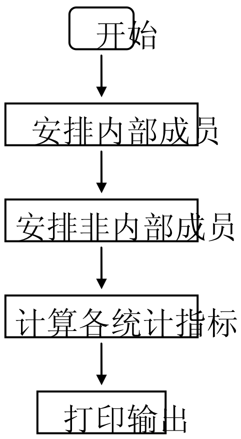
comm[i][j][k]=0 表示第 i 届会议中成员 k 不在小组 j。

●某一编号的成员在某届会议中所处的小组号

它用来帮助我们计算并判断某一成员是否在同一主管下出现了两次。程序中通过数组 desk[29][7]存储，desk[i][j]表示成员 i 在第 j 届会议中所处的小组组号。

在安排某届会议时，无论是成员的优先排序，还是排斥力大小的确定，都需要用到一些中间变量，如 $C_j^{[i]}, M_{il}$ 等等，所有这些变量我们都可通过以上三个存储“当前状态”的数组间接计算求得。

（2）程序框图

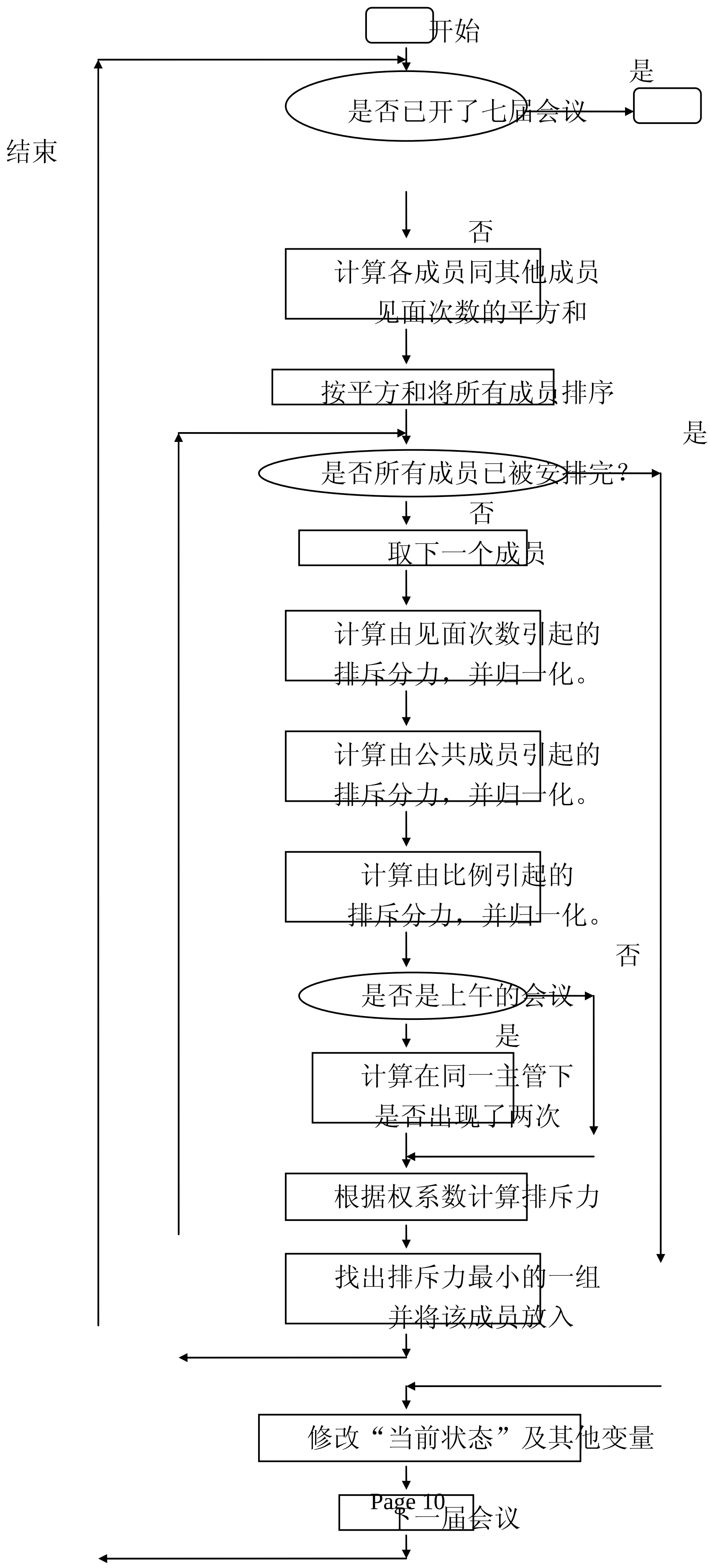


左面是整个算法的框图，它包括了对内部成员与非内部成员的安排以及各种统计指标的计算输出。根据算法，我们用 Turbo C++ 1.0 编制了程序，见**【附录 3】**。

安排内部成员与非内部成员的过程大体相同，只是在成员比例方面计算公式有所不同，安排内部成员时应以各组人数尽量均衡为准则，而安排非内部成员时应以内部成员比例适当为主要原则。

下面我们给出了其具体的计算框图：

安排内部成员或非内部成员的计算机框图



2、模型结果

同时考虑四条标准时的最终结果（注：上午会议中，小组号即为主管号）

此时，各权系数为： $\lambda_1 = 0.033$ $\lambda_2 = 0.033$ $\lambda_3 = 0.8$ $\lambda_4 = 0.133$

公共成员数为: 3 人 公共成员为 3 的小组出现的次数为: 14 次

各届会议中各小组的公共成员数

	小组 1	小组 2	小组 3	小组 4	小组 5	小组 6	
	2	1	2	2	3	3	
上午		3	2	2	2	2	2
	2	2	2	3	2	3	
	3	3	3	2			
下午		3	3	2	2		
	2	2	3	2			
	3	2	3	3			

每人同其他人见面次数统计值

成员编号	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	（号）
见 0 次面:	6	3	4	1	4	3	5	5	4	2	6	4	6	9	7	
见 1 次面:	12	12	16	16	12	14	15	13	12	15	7	9	11	10	9	
见 2 次面:	10	12	5	9	11	11	7	8	11	11	15	13	10	10	12	
见 3 次面:	1	2	4	3	2	1	2	3	2	1	1	3	2	0	1	
见 4 次面:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

```
 成员编号 16 17 18 19 20 21 22 23 24 25 26 27 28 29        (号)
---------------------------------------------------------------
见 0 次面: 5  5  5  3  4  7  6  6  7  5  2  5  2  8
见 1 次面:12 10 12 14  5 10 11 10 13 12 12 10 13  3
见 2 次面:12 11 11  9 17  9 11 12  6  9 13 12 13 16
见 3 次面: 0  3  1  3  3  3  1  1  3  3  2  2  1  2
见 4 次面: 0  0  0  0  0  0  0  0  0  0  0  0  0  0
===============================================================
```

各届会议成员分派如下（带（）者表示内部成员）

```
 ***************** 第 1 届会议 ***************
 第 1 组: (1) ,13  ,22  ,
 第 2 组: (2) ,14  ,23  ,
 第 3 组: (3) ,15  ,24  ,
 第 4 组: (4) ,(5) ,10  ,16  ,19  ,25  ,28  ,
 第 5 组: (6) ,(7) ,11  ,17  ,20  ,26  ,29  ,
 第 6 组: (8) ,(9) ,12  ,18  ,21  ,27  ,

 ***************** 第 2 届会议 ***************
 第 1 组: (4) ,(9) ,11  ,20  ,28  ,12  ,15  ,23  ,
 第 2 组: (5) ,(3) ,17  ,25  ,18  ,13  ,
 第 3 组: (6) ,(2) ,10  ,26  ,27  ,22  ,
 第 4 组: (7) ,29  ,21  ,
 第 5 组: (8) ,16  ,24  ,
 第 6 组: (1) ,19  ,14  ,

 ***************** 第 3 届会议 ***************
 第 1 组: (5) ,(8) ,26  ,17  ,19  ,21  ,
 第 2 组: (4) ,(1) ,11  ,27  ,29  ,16  ,
 第 3 组: (9) ,25  ,14  ,
 第 4 组: (6) ,(3) ,20  ,12  ,13  ,22  ,
 第 5 组: (2) ,28  ,18  ,15  ,
 第 6 组: (7) ,10  ,23  ,24  ,

 ***************** 第 4 届会议 ***************
 第 1 组: (4) ,(8) ,(2) ,20  ,26  ,13  ,25  ,15  ,24  ,
 第 2 组: (5) ,(9) ,11  ,22  ,27  ,23  ,
```

第 3 组: (6) , (1) , 12 , 29 , 19 , 10 , 18 ,
第 4 组: (3) , (7) , 17 , 28 , 16 , 21 , 14 ,

***************** 第 5 届会议 ***************
第 1 组: (4) , (3) , 26 , 28 , 29 , 22 , 18 ,
第 2 组: (5) , (2) , (7) , 20 , 12 , 27 , 19 , 23 , 24 ,
第 3 组: (6) , (9) , 17 , 15 , 25 , 16 ,
第 4 组: (8) , (1) , 11 , 13 , 21 , 10 , 14 ,

***************** 第 6 届会议 ***************
第 1 组: (4) , (7) , 12 , 25 , 13 , 21 ,
第 2 组: (5) , (1) , 26 , 23 , 15 , 18 , 16 ,
第 3 组: (2) , (3) , (9) , 20 , 17 , 19 , 29 , 22 , 10 ,
第 4 组: (6) , (8) , 11 , 27 , 28 , 24 , 14 ,

***************** 第 7 届会议 ***************
第 1 组: (2) , (9) , (1) , 12 , 29 , 26 , 25 , 28 , 21 ,
第 2 组: (4) , (6) , 17 , 19 , 23 , 13 , 14 ,
第 3 组: (5) , (7) , 11 , 22 , 15 , 24 ,
第 4 组: (3) , (8) , 20 , 27 , 10 , 18 , 16 ,

六、模型的结果分析

（1）公共成员数的评估

从以上结果我们可以看出，最大公共成员数目为 3，次数为 14，所以公共成员数是很小的。将此结果与忽略内部成员及主管因素时所得的结果相比较，最大公共成员数目没有发生改变，只是出现的次数多了 4 次，可见公共成员数是令人满意的。

（2）见面次数的评估

本模型最终结果中，虽然没能保证每个董事会成员同其它董事会成员见面次数相等，但从见面次数结果统计表来看，各成员之间的见面次数是比较均衡的，没有出现两成员见面次数过多的情况。每个董事会成员都能保证同其他绝大多数成员见过一次或两次面。

（3）主管因素

由于在求解过程中，给来自主管的排斥分力加以很大的权系数，所以在结果中严格保证了上午的会议中，任一董事会成员均不在同一主管带领的小组中出现两次。

（4）内部成员比例因素

根据计算结果，我们求得全天 34 个小组中内部成员比例如下表：

小组号	1	2	3	4	5	6
上　午	33.33% 33.33% 33.33%	33.33% 40.00% 33.33%	33.33% 40.00% 33.33%	28.57% 33.33% 33.33%	28.57% 33.33% 25.00%	33.33% 33.33% 25.00%
下　午	33.33% 28.57% 33.33% 33.33%	33.33% 33.33% 28.57% 28.57%	28.57% 33.33% 33.33% 33.33%	28.57% 28.57% 28.57% 28.57%		

从表中我们可以看出，内部成员比例为 33.33%的情况出现了 20 次，比例为 28.57%的情况出现了 10 次，比例为 40%和 25%的情况各出现了 2 次。内部成员比例是相当令人满意的。

（5）主管工作量

在对题目的分析中，我们发现，除原来的四条标准外，实际中还经常要考虑一条标准，即：要求在上午会议中每个主管所带领过的成员总数应尽可能均衡，也就是主管的工作量尽量均衡。从我们的计算结果中可以看出，每个主管所带领过的成员总数分别为：16 , 15 , 12 , 16 , 14 , 13 ,不同两个主管工作量最大之差不超过 4，所以主管工作量是比较均衡的。

七、稳定性分析

为检验权系数取值不同对结果造成的影响，我们选取了几组不同的权系数进行检验，发现模型具有很高的稳定性。检验结果见**附录【2】**

八、模型推广

在会议召开过程中，可能某些成员会放弃参加会议或某些没有计划在内的成员要来参加会议，同时，与会人员的数目和交叉混合的标准也都不是固定的。针对这些情况，我们对该模型作如下推广。

●某些成员放弃参加会议的情况

前面，我们在模型设计中制定的规则在这里并不会由于成员放弃参加会议而发生改变，成员对会议的放弃只涉及到“内部成员所占比例”以及“当前状态”的改变。对于比例的变化，我们可通过将〖7〗式中 9/29 相应修改即可。而“当前状态”是通过三个数组来描述的，对于放弃参加会议的成员是内部成员或非内部成员两种情况，我们将数组分别作出如下调整：

（1）若非内部成员 i 在某届会议后放弃参加会议，对内部成员的分派不会产生影响，则我们只需调整程序中描述任意两个成员见面次数的数组 m[i][j]，将其

他所有成员同成员 i 的见面次数在数组 m[i][j]中清为零，其余两个数组不用改变，这样就可以保证在以后的成员分派过程中，根据调整后的当前状态即可重新计算分派所有非内部成员。

（2）若内部成员 i 在某届会议后放弃参加会议，则首先应根据上述规则调整当前状态，然后对内部成员和非内部成员按调整后的当前状态依次分派。

●某些没有计划在内的成员前来参加会议的情况

对于这种情况，也首先对〖7〗式中 9/29 的内部成员比例加以修改，然后再对当前状态进行调整。

（1）前来参加会议的成员是非内部成员的情况

由于新成员在前几届会议中从未出现过，故我们可将数组 m[i][j]加以扩充，添加入新成员，并将该成员同其他成员的已见面次数记为 0。描述状态的其它两个数组仍可保持不变。然后根据“当前状态”，重新按我们的算法分派以后各届的非内部成员。

（2）前来参加会议的成员是非内部成员

对于这种情况，除了仍需将“当前状态”数组按上述法则加以改变外，还要根据调整后的状态依次对以后各届的内部成员和非内部成员重新加以分派，以保证仍能使得成员尽量交叉混合。

●与会人员数目任意时的情况

由于我们在制订分派法则的过程中，已经包括了各种类型参加者数目任意的情况，故无需改变分派法则，而只将程序及描述当前状态的各数组稍加改变，即可求解计算。

我们利用程序对内部成员数为 11，董事会成员总数为 36，主管为 6 人的情况重新进行了分派。限于篇幅，这里就不再给出。

●影响交叉混合效果的因素数目较多时的情况

模型建立时，我们是根据四条标准的满足程度来衡量交叉混合好坏的。对于那些影响交叉混合效果的因素数目较多的情况，我们只需相应地增加排斥分力即可。

引入多个衡量标准后的排斥力公式如下:

设有 n 个标准，第 k 个标准对应的权系数为 $\lambda_k \quad (k=1,2,\cdots\cdots,n)$，则：

$$f_j^{[i]} = (\lambda_1, \lambda_2, \cdots\cdots, \lambda_n) \cdot \begin{pmatrix} f_{j1}^{[i]} \\ f_{j2}^{[i]} \\ \vdots \\ \vdots \\ f_{jn}^{[i]} \end{pmatrix} = \sum_{k=1}^{n} \lambda_k \cdot f_{jk}^{[i]}$$

$$\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (\quad \sum_{k=1}^{n} \lambda_k = 1 \quad)$$

九、模型优缺点

本模型最主要的优点是它的实用性强，对于那些参加会议成员类型不同，数目任意，以及衡量交叉混合程度的标准有所增减的情况，均可应用我们的算法。

模型的缺点是权系数的取值带有一定的主观性。若能通过某些数学方法来确定权系数的取值，将更为理想。但是从这里又可以得出模型的另外一个主要优点，就是：模型有较大的灵活性，当对衡量交叉混合的标准侧重不同时，可通过调整权系数来满足要求。

【参考文献】

钱颂迪 ：《运筹学》 清华大学出版社 1990 年

The UMAP Journal, v. 13(1992),No. 3.

日本数学会：《数学百科词典》 科学出版社 1984 年

附录【1】

〖对绝对理想的分派不存在的证明〗

严格满足〖1〗式时，应有：

$$\sum_{j=1}^{3}\sum_{i=1}^{6}(x_i^{[j]})^2+\sum_{j=4}^{7}\sum_{i=1}^{4}(x_i^{[j]})^2=29(28s+7)$$

由 Cauchy 不等式：$\dfrac{x_1^2+x_2^2+\cdots+x_n^2}{n}\ge\left(\dfrac{x_1+x_2+\cdots x_n}{n}\right)^2$ ，

（ $x_1,x_2,\cdots\cdots,x_n\ge 0$；当且仅当 $x_1=x_2=\cdots=x_n$ 时，“＝”成立。）

可知：

对任一会议不同小组之间成员数取值相差越小，则

$$\sum_{j=1}^{3}\sum_{i=1}^{6}(x_i^{[j]})^2+\sum_{j=4}^{7}\sum_{i=1}^{4}(x_i^{[j]})^2=29(28s+7)$$ 越小，即 s 越小。

反之，相差越大，s 越大。对 An Tosal 公司而言：

①使得 s 取值最小的全天会议成员分派应满足：

$$\begin{cases}\text{上午均为}\quad 5\ 5\ 5\ 5\ 5\ 4\\ \text{下午均为}\quad 7\ 7\ 7\ 8\end{cases}$$

②使得 s 取值最大的全天会议成员分派应满足：

$$\begin{cases}\text{上午均为}\quad 2\ 2\ 2\ 2\ 2\ 19\\ \text{下午均为}\quad 2\ 2\ 2\ 23\end{cases}$$ （若每组最小成员数允许为 2）

将①②中分派情况代入〖1〗式，

求得：$\begin{cases}\min s=1.3\\ \max s=3.8\end{cases}$ 由于〖1〗式中 s 只能取整数，故 $\begin{cases}\min s=2\\ \max s=3\end{cases}$

对于 s=2 的情况，我们将由〖2〗式得出的**表 1** 中的董事会成员分配，通过

计算机搜索 s=2 时〖1〗式的解，结果如下表：

	结果 1	结果 2	结果 3
上午各届成员数目	3 3 3 7 7 6 3 3 3 3 7 10 3 3 3 3 7 10	3 3 3 7 7 6 3 3 3 7 7 6 3 3 3 3 7 10	3 3 3 7 7 6 3 3 3 7 7 6 3 3 3 7 7 6
下午各届成员数目	3 3 3 20 3 3 3 20 7 6 6 10 7 6 6 10	3 3 3 20 3 3 3 20 7 6 6 10 3 10 6 10	3 3 3 20 3 3 3 20 3 10 6 10 3 10 6 10

对应于以上三组解，下午成员人数均有两届出现（3，3，3，20）的情况，所以，即使成员数为 20 的小组下一次将其中的 9 人全部分派入其余 3 组，仍有 11 人留在同一组内，即公共成员数为 11。显然，远未达到董事会成员尽量交叉混合的效果。

进一步计算表明，当 s=3 时公共成员数已经超过了 11。综上所述，我们存在如下结论：

同时满足四条标准的绝对理想的分派是不存在的。

附录【2】

〖稳定性分析〗

主管:0.900 次数:0.029 比例:0.057 公共成员:0.014

最大公共成员数为: 3　　　最大公共成员出现次数为: 15

=========================

```
2  2  2  2  3  2
2  3  2  2  2  2
2  2  2  2  2  3
```

```
3  3  3  2
2  3  3  2
3  3  3  3
```

```
 3    3    3    2
==========================
```

每人同其他人见面次数统计值

```
              1   2   3   4   5   6   7   8   9  10  11  12  13  14  15    (号)
-------------------------------------------------------------
见 0 次面: 6   4   7   2   2   3   6   4   4   2   4   2   5   6   7
见 1 次面:12  12  10  13  15  12  10  14  12  13   9  13  10  13  11
见 2 次面:11  11  11  12  11  13  13  10  11  12  14  12  13  10  10
见 3 次面: 0   2   1   2   1   1   0   1   2   2   2   2   1   0   1
见 4 次面: 0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
=============================================================
             16  17  18  19  20  21  22  23  24  25  26  27  28  29    (号)
-------------------------------------------------------------
见 0 次面: 7   7   6   4   5   4   4   5   4   5   5   3   4   4
见 1 次面:12   8   8  12   9  14  14  11  15  12   8  14  13  11
见 2 次面: 9  14  14  13  15  10  11  12   9  10  12  11  10  12
见 3 次面: 1   0   1   0   0   1   0   1   1   2   4   1   2   2
见 4 次面: 0   0   0   0   0   0   0   0   0   0   0   0   0   0
=============================================================
```

主管:0.900 次数:0.009 比例:0.073 公共成员:0.018

最大公共成员数为: 4   最大公共成员出现次数为: 3

```
===========================
 2    3    2    3    3    3
 3    3    2    3    2    2
 3    2    2    3    2    2
 --------------------------
 3    3    3    3
 3    4    3    2
 3    2    4    3
 4    3    3    3
 ==========================
```

每人同其他人见面次数统计值

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	(号)
见 0 次面:	6	6	6	3	3	2	7	3	6	6	4	4	8	6	4	
见 1 次面:	14	12	15	12	15	15	10	16	9	11	12	11	10	9	15	
见 2 次面:	7	4	2	10	8	11	10	9	11	10	10	12	10	12	8	
见 3 次面:	2	7	6	4	3	1	2	1	3	2	3	2	1	2	2	
见 4 次面:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

	16	17	18	19	20	21	22	23	24	25	26	27	28	29	(号)
见 0 次面:	4	3	6	5	4	7	6	6	5	2	2	3	7	5	
见 1 次面:	15	15	11	13	5	7	10	10	18	17	16	11	10	10	
见 2 次面:	7	9	8	6	16	10	12	10	5	9	7	14	8	8	
见 3 次面:	3	2	4	5	4	5	1	3	1	1	4	1	4	6	
见 4 次面:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

主管:0.900 次数:0.018 比例:0.073 公共成员:0.009

最大公共成员数为: 4　　　　最大公共成员出现次数为: 2

2	3	2	3	3	3
3	3	3	2	2	2
3	4	2	3	2	2

3	3	3	3
3	3	3	3
2	3	3	2
4	3	3	3

每人同其他人见面次数统计值

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	(号)
见 0 次面:	7	3	4	3	3	4	7	4	4	6	5	4	10	7	4	

```
见 1 次面:13 13 16 14 12 14 11 15 11 10  9 10  7  8 11
见 2 次面: 5 10  5  7 12  8  9  9 11 11 11 13 11 13 12
见 3 次面: 4  3  4  5  2  3  2  1  3  2  4  2  1  1  2
见 4 次面: 0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
=============================================================
          16 17 18 19 20 21 22 23 24 25 26 27 28 29   （号）
-------------------------------------------------------------
见 0 次面: 6  4  5  4  5  7  6  7  7  4  3  4  5  5
见 1 次面:11  9 11 14  3  6 11 10 11 12 13 12 14 11
见 2 次面:10 11 11  9 17 13  8  8 11 12 10 12  7 10
见 3 次面: 2  5  2  2  4  3  4  4  0  1  3  1  3  3
见 4 次面: 0  0  0  0  0  0  0  0  0  0  0  0  0  0
=============================================================
```