A grouping system used to form teams full of thinking styles for highly debating

Dai-Yi Wang, Yen-Chun Liu, Chuen-Tsai Sun
Dept. of Computer and Information Science
National Chiao Tung University
1001 Ta Hsueh Rd., HsinChu 300
Taiwan

Abstract: - Cooperation is an instrument popularly used in schools and society. Computer-supported cooperative learning also promises innovations to improve teaching. Among these is a branch devoted to form effective teams. However, sometimes we do not have access to relevant students’ learning information to build the learning models. Therefore, we designed a grouping system connected by on-line questionnaire, and then this system can group students by their thinking styles without constructing students’ modeling. Our grouping system used a Hill Climbing algorithm and a Simulated Annealing algorithm to form teams full of thinking styles for highly debating.

Key-Words: - Cooperative learning, small group learning, computer-supported cooperative learning, computer assisted group composing system, group composition

1 Introduction
Cooperation is an instrument popularly used in schools and society. It dose not only increase productivity, but also arouses people’s motivation and encourages interpersonal interaction. However, can groups of randomly picked individuals lead to successful cooperation? Educational research [1] [2] points out that effective cooperative learning need to be designed carefully.

Along with the improvement in information technologies, computer-supported cooperative learning also promises innovations to improve teaching [3-5]. At the same time, there are many adaptive/intelligent educational systems that facilitate web-based learning [6]. Among these is a branch devoted to form effective teams [7] [8].

No matter how the group formation is initiated, whether by system detection or by user requisition, grouped students by complementary concept, grouped students by browsing behaviors, and grouped students randomly through interned points, they all adopt dynamic grouping—shifting students from an individual mode to a collaborative learning mode.

However, sometimes we do not have access to relevant students’ learning information, such as when they first time to learn in this environment. Moreover, some project-oriented or problem-solving learning activities need to initially proceed by groups. Hence, designing a grouping system that can quickly group learners without building a learning model is worth investigating.

On the other hand, many researchers found that some psychological features (e.g., self efficacy [10] or thinking styles (TSs) [11]) have a strong effect on the outcomes of cooperative learning. Moreover, web-based psychological questionnaires can quickly facilitate the acquisition of this data. Therefore, we can design a grouping system connected by on-line questionnaire, and then this system can group students by psychological features. It is useful to form groups without constructing students’ modeling.

In this paper, we proposed a novel grouping system that can form highly agitated teams according to teammates’ thinking styles. Section 2 analyzes some of the problems in using TS to compose groups. Section 3 introduces our grouping system and the implementation algorithm. Finally, section 4 offers a simple conclusion and future perspectives of using this computer-supported group composition system.

2 Problem Analysis
In this study, we adopted thinking styles as our grouping features. Thinking style (TS) was proposed by [11]. It is the way that people prefer to use their ability. Sternberg introduced 13 kinds of thinking styles. All these TSs affect every person at the same time, but differing degrees. Hence, we cannot simply classify a person’s thinking style by his/her striking...
style, but rather need to simultaneously consider the degree of all styles in each person. To simplify our illustration, we only discuss executive, legislative and judicial TSs in this paper.

Sternberg suggested that an effective team should have high degrees of executive, legislative and judicial TS to make members both influence and agitate. Hence, before defining group TSs and setting out grouping goals, we define individual’s TS first.

We can regard each person’s TS as a point in geometric space if we draw each psychometric variable as a dimension. By operating the space vectors, we easily transform the abstractive problem into a structured, procedural problem. For instance, we can transform the problem of the diversity of two individuals’ TSs into the problem of the geometric distance of two points. This diversity can suitably be defined by Euclidean distance.

A team is composed of members whose TSs differ by degree. Assume we have three individuals whose TSs are listed in Table 1.

<table>
<thead>
<tr>
<th>Table 1. The TS of three individuals</th>
</tr>
</thead>
<tbody>
<tr>
<td>Member</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>Executive TS</td>
</tr>
<tr>
<td>Legislative TS</td>
</tr>
<tr>
<td>Judicial TS</td>
</tr>
</tbody>
</table>

**Definition:** If these three individuals are selected to form team 1, the TSs of team 1 are defined as:

$$G_1 \Rightarrow \begin{cases} 
A_1 = \text{Max}(A_{11}, A_{12}, A_{13}) \\
L_1 = \text{Max}(L_{11}, L_{12}, L_{13}) \\
J_1 = \text{Max}(J_{11}, J_{12}, J_{13}) 
\end{cases}$$

Teammates having different TS profiles reciprocally influence and agitate as the group’s TS begins to take shape. To enhance the reciprocation, the goal in forming teams is to ensure there is a high degree of executive TS, legislative TS and judicial TS in each team.

Prominent TSs may be contributed by different teammates, or one person may show prominence in more than one area. For this study, ideally the striking values would be supplied by different teammates. However, we cannot forecast or control the TS distribution of the experiment sample, so we adopted a priority mechanism. The highest priority is that the striking styles belong to different individuals. If a situation arises that priority one cannot be accomplished, then the second priority is that one person possesses two striking styles. The lowest priority is that one person supplies all styles.

We used $Ai$, $Li$ and $Ji$ to present the characteristic values of team $i$. And if we have N teams, then their characteristic value set is:

$$(G_1 \sim G_n) \Rightarrow \begin{cases} 
(A_1, A_2, A_3, \ldots, A_n) \\
(L_1, L_2, L_3, \ldots, L_n) \\
(J_1, J_2, J_3, \ldots, J_n)
\end{cases}$$

Our goal is to find a combination that makes the characteristic value of each team very large. That is, we want to find a grouping that causes maximum $A$, $L$ and $J$, which is the minimum characteristic value of all teams.

$$A = \text{Min}(A_1, A_2, A_3, \ldots, A_n)$$

$$L = \text{Min}(L_1, L_2, L_3, \ldots, L_n)$$

$$J = \text{Min}(J_1, J_2, J_3, \ldots, J_n)$$

$\Rightarrow$ Find a grouping that has maximum $A$, $L$ and $J$

After the above analysis, we know the grouping condition is to find $\text{Max} \ A(p), L(p), J(p)$ where $p$ is a grouping result.

### 3 Methodology: Two-phase grouping system

Following the above discussion, our first goal is to construct teams where teammates have different TSs. This means that points represented by a team must be apart from each other. The second goal is to ensure full representation of thinking styles within each team.

To achieve these goals, we referred to the two-phase grouping framework proposed by [12]. The first phase includes categorizing all persons to attain the first aim—teammates must have a distinct style inclination. The number of teammates decides the number of categories. Subsequently, we can choose one person from each category to form a group. As a result, teammates’ TSs are heterogeneous.

Assume we want to form N triad teams. The flow of the two-phase grouping framework is shown as Fig. 1.

In the first phase we adopted a K-means algorithm to control the teammate count as K-means can precisely set the cluster number. In this study, the cluster number was set as three to form triad teams. Users forming a different size of team need to adjust the K parameter accordingly.

Furthermore, we added a control mechanism to ensure the number in each category was equal. This mechanism breaks the heterogeneity within each category. However, we cannot control the sample
distribution. All individuals should be assigned into teams, and all team sizes should be equal. Therefore, this phase runs repeatedly to find better categorizations and uses the second phase to improve the grouping solution.

In section 2, we know that different grouping leads to different $A$, $L$, and $J$ values. Our goal is to find a grouping that leads to maximum $A$, $L$, and $J$. Fig. 2 illustrates this problem by the concept of searching. The computing complexity of this problem is $O(n!)$ for forming $n$ m-person teams.

We therefore used Hill Climbing and Simulated Annealing, iterative improvement algorithms, to find an optimal solution. Iterative improvement algorithms search toward the target from an initial state.

Hill Climbing algorithms always move to a state that is better than the current state. Fig. 3 shows the flow of a Hill Climbing algorithm used to find our grouping target.

To find the next best $p'$, in this study, we used the current grouping to find the near grouping. Near grouping is defined as exchanging teammates once. Exchanging only occurs when two members are categorized in the same category. The latter condition ensures that heterogeneity is maintained in phase 1.

Fig. 4 shows an example of exchanging. If there are $N$ teams, then there are $3 \times C_2^N$ possibilities of the near solution. We must find the best one from these near grouping.

The $A$, $L$, and $J$ value, analyzed in section 2, were compared to evaluate whether the near grouping was better than the current grouping.

If $$\begin{cases} A_1 \geq A_2 \\ L_1 \geq L_2 \\ J_1 \geq J_2 \end{cases}$$ and $(A_1-A_2) + (L_1-L_2) + (J_1-J_2) > 0$ 

$p1$ is better than $p2$.

When the $A$, $L$, and $J$ values equal, we adopt a more complicated comparison. Table 2 shows an example of two teams produced by two groupings with the TS values listed.

Table 2. The TS distributions of two teams

<table>
<thead>
<tr>
<th>Team 1</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>EXE</td>
<td>LEG</td>
<td>JUD</td>
<td></td>
</tr>
<tr>
<td>member1</td>
<td>28</td>
<td>15</td>
<td>17</td>
</tr>
<tr>
<td>member2</td>
<td>20</td>
<td>15</td>
<td>10</td>
</tr>
<tr>
<td>member3</td>
<td>20</td>
<td>15</td>
<td>17</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Team 2</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>EXE</td>
<td>LEG</td>
<td>JUD</td>
<td></td>
</tr>
<tr>
<td>member1</td>
<td>28</td>
<td>15</td>
<td>17</td>
</tr>
<tr>
<td>member2</td>
<td>10</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>member3</td>
<td>5</td>
<td>10</td>
<td>5</td>
</tr>
</tbody>
</table>
Both teams have a member with the same characteristic TS values, (28, 15, 17). However, considering the diversity between teammates, the effect on entirety, and the potentiality of finding a better solution, the formation of Team 2 is better than that of Team 1. Therefore, we introduce \textit{Evalgroup} to form teams with significant TS variation. The value of the \textit{Evalgroup} is the larger, the better.

\begin{align*}
\text{Evalgroup(Team } i \text{)} &= \\
&= (\left| A_{i1} - A_{i2} \right| + \left| A_{i1} - A_{i3} \right| + \left| A_{i2} - A_{i3} \right|) + \\
&+ (\left| L_{i1} - L_{i2} \right| + \left| L_{i1} - L_{i3} \right| + \left| L_{i2} - L_{i3} \right|) + \\
&+ (\left| J_{i1} - J_{i2} \right| + \left| J_{i1} - J_{i3} \right| + \left| J_{i2} - J_{i3} \right|)
\end{align*}

\begin{align*}
\text{Evalgroup(Team 1)} &= \\
&= (8 + 8 + 0) + (0 + 0 + 0) + (7 + 7 + 0) = 30
\end{align*}

\begin{align*}
\text{Evalgroup(Team 2)} &= \\
&= (18 + 23 + 5) + (10 + 5 + 5) + (12 + 0 + 12) = 90
\end{align*}

\text{Evalgroup(Team 2)} = 90 > \text{Evalgroup(Team 1)} = 30

\Rightarrow \text{The formation of Team 2 is better than that of Team 1.}

Extending this evaluation formula to \textit{n} teams, the formula is defined as:

\begin{align*}
\text{Eval2}(p) &= \\
&\sum_{i=1}^{n} \left( \left| A_{i1} - A_{i2} \right| + \left| A_{i1} - A_{i3} \right| + \left| A_{i2} - A_{i3} \right| \right) + \\
&\left( \left| L_{i1} - L_{i2} \right| + \left| L_{i1} - L_{i3} \right| + \left| L_{i2} - L_{i3} \right| \right) + \\
&\left( \left| J_{i1} - J_{i2} \right| + \left| J_{i1} - J_{i3} \right| + \left| J_{i2} - J_{i3} \right| \right)
\end{align*}

where \textit{p} is a grouping result.

If \text{Eval2}(p') > \text{Eval2}(p) then \textit{p'} is better than \textit{p}. Therefore, \text{Eval2}(p) is the second evaluation.

A Hill Climbing algorithm with a two-evaluation mechanism was repeatedly used to find a better grouping result. The algorithm stopped when it could not find a better solution. However, it could potentially get stuck in local maxima. For this reason we used Simulated Annealing algorithms to mend this occurrence. Simulated Annealing algorithms can accept the worse solution by chance to jump over local maxima. This probability decreases exponentially. The flow of our Simulated Annealing method is shown as Fig. 5.

In Fig. 5, the \textit{δ} (Perturbation Rule) is designed to randomly select the next status instead of selecting the next best status in Hill Climbing. Like with Hill Climbing, we swapped teammates to produce the next grouping. However, we limited the swap to one time; that is, only two of all the teams can exchange one member once.
In our Simulated Annealing algorithm, the evaluation function is similar to that of our Hill Climbing algorithm that combines two evaluation mechanisms. Its formula is defined as:

$$\Delta \text{Eval} = [A'(p') - A(p)] + [L(p') - L(p)] + [J(p') - J(p)] + 0.1 \times [\text{Eval2}(p') - \text{Eval2}(p)]$$

If $\Delta \text{Eval} > 0$, meaning $p'$ is better than $p$, then $p$ would be replaced by $p'$, else keeping the current status. But in order to avoid falling into a local optimal, we allowed $p'$ to replace $p$ with $e^{\Delta \text{Eval} / T}$ probability when $\Delta \text{Eval} \not\geq 0$. Along with the iteration increasing, $T$ decreased gradually, reducing the probability of accepting the worse solution. This searching problem therefore converged. Moreover, we set a threshold (0.0000000000001) to stop this algorithm.

In this study, we integrated two iterative improvement algorithms to achieve the grouping goal thus providing each team with strong thinking styles. Hill Climbing is used to find an approximate solution quickly, and Simulated Annealing is used to avoid falling into local optimal.

### 4 Conclusion

In this paper, we propose a novel grouping system in accordance with Sternberg’s theory—groups filled with all three thinking styles enhance brainstorming. In order to create teams composed of teammates with strikingly different TSs, we adopted a two-phase framework. The first phase is used to build team heterogeneity by an amended K-means algorithm; the second phase is designed to form teams composed full of thinking styles using Hill Climbing and Simulated Annealing.

Although we used Sternberg’s TSs as the grouping target in this study, it is easy to expand using other psychometric variables. In general, the higher the psychometric degree implies the more conspicuous the behavior, and multiple unusual stimulants lead to additional reciprocal interaction and brainstorming. Therefore, we created a model to form highly heterogeneous teams.

### References:


