# Forming and Scaffolding Human Coalitions: A Framework and An Implementation For Computer-Supported Collaborative Learning Environment

Nobel Khandaker Leen-Kiat Soh

Computer Science and Engineering University of Nebraska Lincoln, NE, USA (402) 472-6738 Email: {knobel, lksoh}@cse.unl.edu

Abstract: Computer-supported collaborative learning (CSCL) environments are used today as a platform for delivering distance education and as a tool to improve student understanding via collaborative learning methods. The success of a CSCL environment in improving the knowledge of a student depends on the quality of group work of its participants. However, forming human user groups that allow all the users collaborate effectively is difficult because of the dynamic nature of the human users and the complex interplay of human factors (e.g., comfort level, proficiency, etc.). Furthermore, human behaviors change over time due to their ability to learn new skills. Thus, a framework that accommodates the unique nature of human behavior and uses it to improve the outcome of the coalitions is needed. In this paper, we present *i*HUCOFS - a multiagent framework for forming and scaffolding human coalitions. We also discuss an implementation of the *i*HUCOFS framework (VALCAM) in a CSCL environment called I-MINDS. Preliminary results indicate that VALCAM can make a positive impact on the learner coalitions formed in I-MINDS.

**Keywords**: Computer-supported collaborative learning, multiagent system, human coalition formation, scaffolding.

# **1. Introduction**

Computer-supported collaborative learning (CSCL) environments have become a popular platform for delivering distance education or supplementing traditional classrooms with outside-the-class group activities. A typical CSCL environment consists of a set of tools to facilitate communication and collaboration of the students. However, a better equipped CSCL tool could also contain provisions for the instructor to form and support student coalitions. However, forming human coalitions in a CSCL environment poses a variety of challenges. The lack of familiarity among the users, their decreased social presence, and their varying levels of knowledge and expertise all add up to the difficulty of formation and support of human learner coalitions. Furthermore, because individual human behaviors change and inter-person relationships evolve over time, a group of peers who did not work well together initially could end up working well to-

gether in the end due to increased familiarity and comfort level. Therefore, due to the dynamic nature of the human users, a fixed scripted coalition formation algorithm may not provide the best solution. This also implies that it is possible for a coalition formation algorithm to form a group of lesser expected utility for the current task with the hope of a better reward in the future as the group members improve the quality of their group work over time. Thus, a human coalition formation framework that forms human coalitions in general should also facilitate the betterment of individual human users, i.e., support the formed coalitions, over time as group members work together. However, this support could be explicit or implicit. In the case of explicit support, the framework would help the coalition members directly by providing hints, clues, recommendations, etc. In the case of implicit support, the framework would create a working environment which would facilitate changes in the members' behaviors that benefit future coalitions. We denote the combination of implicit and explicit coalition support provided by the framework as scaffolding.

Although the *formation* and the *scaffolding* of human user coalitions is an integral part of a CSCL environment, the typical CSCL environments do not address them. For example, Constantino-González [6] proposed a web-based environment called Collaborative Learning Environment for Entity-Relationship Modeling (COLER) in which student can solve Entity-Relationship (ER) problems while working synchronously in small groups at a distance. Barros and Verdejo [1] used activity theory to design the DEGREE environment that monitors and mediates group activity. Ogata and Yano [18] developed a collaborative learning environment using knowledge awareness and information filtering. Grave et al. [9] created a multi-layer architecture on a multiagent framework that is able to initiate and manage student training. Although the typical CSCL systems do not automate the general coalition formation process, there have been some research approaches to form two-member human user groups to provide peer-support to learners. For example, Li et al. [14] used agent technology with fuzzy set theory to find matching peers for human users based on similar preferences or expertise. Bull et al. [2] combines a 1-to-1 peer help network and a discussion forum to provide offline peer help to learners in I-HELP. However, in these peer-help systems, a peer group is built based on 1-to-1 experience instead of taking account how a group would work together as a team. Furthermore, noise, uncertainty and incomplete information inherent in the human group formation environment are also not addressed. Finally, there have also been approaches to provide scaffolding (or support) to human coalitions. For example, Constantino-González et al. [6] provides support by advising a student to improve his or her collaborative skills (e.g., participation, communication, etc.) in COLER. Vizcaíno [27] described a virtual student architecture that improves collaboration by detecting and avoiding situations (e.g., off-topic conversations) that decrease the benefits of learning in collaboration. However, these research approaches for realizing scaffolding use only short term approaches (solving the task at hand) for scaffolding and do not try to improve the behavior of the human users and the coalitions in the long term.

In this paper, we describe the Integrated Human Coalition Formation and Scaffolding (iHUCOFS) framework, which has been previously proposed in [23]. The *i*HUCOFS framework is designed to form and scaffold coalitions, trading off expected utility of solving the current task and the potential utility of better coalitions in the future. This paper formalizes the *i*HUCOFS and details its representational and characteristic assumptions, and elaborates the different types of human learning that occurs in group work. Further, this paper describes an algorithm called VALCAM [23] [24] that implements a portion of the *i*HUCOFS framework in a CSCL environment called I-MINDS [24] [25]. VALCAM is an auction-based multiagent learning algorithm that forms human coalitions through an iterative auction. I-MINDS, which stands for Intelligence Multiagent Infrastructure for Distributed Systems in Education, is a CSCL environment for learners in synchronous learning and a classroom management applications for instructors for large classroom or distance education situations. We have previously evaluated the usefulness of I-MINDS as a CSCL environment in [12] [23]. [24]. We further present more comprehensive results of using the *i*HUCOFS framework to form and scaffold human coalitions in this paper.

This paper is organized as follows: Section 2 describes the *i*HUCOFS framework: assumptions, problem characteristics, and design principles. Section 3 briefly presents the VALCAM algorithm, an implementation of the *i*HUCOFS framework. Section 4 describes the basic architecture of I-MINDS and outlines our implementation of VALCAM in I-MINDS. Section 5 presents the results of our two-semester long experiment of using VALCAM. Section 6 discusses the research work related to the collaborative learning systems and research work related to human coalition formations in collaborative learning scenarios. Finally, Section 7 concludes and touches upon some ongoing and future work.

# 2. iHUCOFS Framework

Here we describe a framework called the Integrated Human Coalition Formation and Scaffolding (*i*HUCOFS) framework. As alluded to earlier, a multiagent system handling human coalitions has to consider both coalition formation and coali-

Furthermore, scaffolding coalitions intion scaffolding. volves two types of support: explicit and implicit. There exists also a tradeoff between forming and scaffolding coalitions. For example, if we are forming a coalition where all group members are good at what they do and are good at working with each other in a group, then scaffolding is not as important. On the other hand, if a coalition consists of group members who are not familiar with each other and where some members do not have sufficient expertise or knowledge to contribute to the group work, then scaffolding plays an important role. Further, putting different members in a coalition could lead to different types of learning among the members. For example, by putting a poor-performing student in a group of better-performing students, it is possible that the poor-performing student might learn by observation from other members of the group, while those members might learn by teaching the poor-performing students. Thus, a system needs to determine on which part to focus its computational resources: coalition formation or coalition scaffolding. Driven by this tradeoff, an agent in such a system must also deal with two different roles: as a representative for and as an advisor to its human user.

In the following, we first propose a set of assumptions defining the environment for the *i*HUCOFS framework and how the tradeoffs take place in the multiagent environment. We then describe a set of design principles addressing specific characteristics of the problem.

# 2.1 Assumptions

Here we propose a set of assumptions about the problem and the *i*HUCOFS framework. These assumptions are divided into two categories: *representational assumptions* and *characteristic assumptions*. The representational assumptions are designed to describe the multiagent environment in which *i*HUCOFS resides. The characteristic assumptions describe the characteristics and behaviors of the various actors and the environment itself. While we present formal description of the representational assumptions, due to space restrictions we only briefly describe the characteristic assumptions. In this framework, each *human user* has a dedicated *user agent*, and they communicate or work together through the user agents. This is akin to computer-supported collaborative problem solving. For describing the assumptions, we define the following functions:

- 1. *Execute*(*x*, *y*, *t*) states that the human user *x* executes task *y* at time *t*
- 2. MemberOf(x, y) states that the human user x is a member of the coalition y. Notice that there is no time factor included in this function. That is because we assume that the coalitions change over time and the definition of a coalition contains a time index in it
- 3. *IsRepresentedBy*(*x*, *y*, *t*) states that the human user *x* is represented by the user agent *y* at time *t*

**Representational Assumption 1.** There is a set of autonomous *agents* in the multiagent system environment *E*, specified as  $A = \{U, G, S\}$ . Here,  $U = \{u_i | i \in 1 ... n_u\}$  is a set of user agents,  $G = \{g_i | i \in 1 ... n_g\}$  is a set of group agents, and *S* is a system agent. We also assume that the user agents and the group agents operate in the system temporarily.

**Representational Assumption 2.** There is a set of autonomous human users in the multiagent system specified by  $H = \{h_i | i \in 1..n_h\}.$ 

**Representational Assumption 3.** There is a set of independent, real-time *tasks* in the problem domain specified as  $T = \{T_i | j = 1 \dots n\}.$ 

The tasks in our environment are events that the human users need to handle. Specifically, we define each task  $T_j$  as a 8-tuple:

$$T_j = (ty_j, ta_j, tl_j, ts_j, tc_j, tr_j, tq_j, tw_j)$$
(1)

(2)

Where,

- 1.  $ty_j$  refers to the type of the *j*th task
- 2.  $ta_j$  refers to the starting time of the *j*th task
- 3. *tl<sub>j</sub>* refers to the time limit within which the *j*th task must be solved
- 4.  $ts_j$  denotes the set of subtasks that constitute the task  $T_j$ . Furthermore,  $ts_j = \{T_j^l | l = 1, ..., |ts_j|\}$ , where  $|ts_j|$  refers to the number of subtasks in  $ts_j$ . We specify the *l*th subtask of the *j*th task  $T_j^l$  as:

$$T_{j}^{l} = (ty_{j}^{k}, ta_{j}^{k}, tl_{j}^{k}, ts_{j}^{k}, tc_{j}^{k}, tr_{j}^{k}, tq_{j}^{k})$$
Where,

- a.  $ty_i^k$  denotes the type of the *k*th subtask of the *j*th task
- b.  $ta_j^k$  denotes the starting time of the execution of the *k*th subtask of the *j*th task
- c.  $tl_j^k$  denotes the time length of the execution of *k*th subtask of the *j*th task
- d.  $ts_j^k$  denotes the set of subtasks that constitute the kth subtask of the *j*th task
- e.  $tc_j^k$  denotes the constraints among the subtasks that belong to the *k*th subtask of the *j*th task
- f.  $tr_j^k$  denotes the resources required for executing the *k*th subtask of the *j*th task
- g.  $tq_j^k = \{tq_{m,j}^k | m = 1 \dots | ts_j^k |\}$ , where  $tq_j^k \in [0,1]$ , denotes the required qualities of the completed subtasks  $ts_j^k$  that belongs to the *k*th subtask of the *j*th task
- h.  $tw_j^k = \{tw_{m,j}^k | m = 1 \dots | ts_j^k |\}$ , where  $tw_j^k \in [0,1]$ , denotes the reward that can be earned by completing the *m*th subtask  $ts_j^k$  according to the quality specification  $tq_j^k$ . Here,  $ts_j^k$  is the *k*th subtask of the *j*th task
- 5.  $tc_j$  denotes the constraints among the subtasks of the *j*th task. For example,  $tc_j$  may contain constraints that restrict the order in which the subtasks  $ts_i$  may be executed
- 6.  $tr_j$  denotes the resource requirements for the *j*th task. An example of the resource requirement could be the expertise or capability of the human users who will execute this task
- 7.  $tq_j = \{tq_{m,j} | m = 1 \dots | ts_j |\}$ , where  $tq_{m,j} \in [0,1]$  specifies the final required quality of the *m*th completed subtask of the *j*th task
- 8.  $tw_j = \{tw_{m,j} | m = 1 \dots | ts_j |\}$ , where  $tw_{m,j} \in [0,1]$  specifies the reward that can be earned by completing the *m*th subtask of the *j*th task according to the quality  $tq_{m,j}$

**Representational Assumption 4.** A human user executes only one task at any given time.

 $\forall h_i \in H, T_i, T_j \in T \ Execute(h_i, T_i, t) \sim Execute(h_i, T_j, t)$  if  $T_i \neq T_j$  (3)

**Representational Assumption 5.** Each human user  $h_i$  is assigned a user agent  $u_i$ . This user agent helps the human user form coalitions with other human users and solve tasks.

 $\forall h_i \in H \exists u_i \in U \text{ s.t. } Is Represented By (h_i, u_i, t)$  (4) where  $i = 1 \dots |H|$ . Furthermore, this assignment is one-toone and  $n_u = n_h$ .

**Representational Assumption 6.** To help the human users *H* accomplish a task  $T_j \in T$ , the system agent may initiate a set of activities so that the human users can form coalitions. The user agents  $u_i \in U$  assigned to the human users  $h_i \in H$  participate in these coalition formation activities to form coalitions for their respective human users.

A coalition contains a set of human users who have agreed to cooperate with each other to solve an assigned task. The set of all the human coalitions working in the multiagent system at time t is denoted by  $C_t = \{C_{k,t} | k = 1 \dots | C_t]\}$ . Once the coalitions are formed, the system agent assigns a group agent  $g_k$  to each coalition  $C_{k,t}$ . Once assigned to a coalition, the group agent acts as a representative of the system agent and monitors and communicates the progress of the group as a whole to the system agent.

**Representational Assumption 7.** Due to his or her interaction with the environment E, a human user acquires new knowledge, and learns new capabilities and behaviors. For *i*HUCOFS, we define two categories of human learning: Derivative (DL) and Communicative (CL). In Derivative Learning, the human users are able to learn new capabilities, concepts, and behaviors by interacting with the environment. An example of Derivative Learning would be when a human user learns something by watching the behavior of the members of his or her group. In Communicative Learning, the human users are able to learn new capabilities, concepts and behaviors from some explicit communication with someone else. An example of Communicative learning would be when an instructor teaches something to a human user.

**Representational Assumption 8.** Each user agent  $u_i$  constructs a model  $hm_{i,t}$  of its assigned human user  $h_i$  by observing his or her behavior in E at time t. This model  $hm_{i,t}$  at time t is represented by a 6-tuple:

$$hm_{i,t} = \langle K_{i,t}, B_{i,t}, DLC_{i,t}, CLC_{i,t}, CA_{i,t}, EU_{i,t} \rangle$$
(5)  
Here,  $K_{i,t}$  represents the human user's *knowledge base* and

$$K_{i,t} = \{ (ct_{ty}, ex_{i,ty,t}) \}$$
(6)

Where  $ct_{ty}$  denotes the capabilities that are necessary to solve tasks of type ty and  $ex_{i,ty,t} \in [0, \zeta_{ty}], \zeta_{ty} \in \mathbb{R}$  denotes  $h_i$ 's expertise level for capability  $ct_{ty}$  at time t. In brief, the human user's knowledge base contains the capabilities that he or she uses to execute various tasks while working in a coalition. We define the operator knowledge base update operator  $\Vdash_k$  for the knowledge base  $K_{i,t}$  as:

 $K_{i,t} \Vdash_k (ct_{ty}) \text{ if } (ct_{ty}, \delta_{ty}^k) \in K_{i,t} \text{ for some } \delta_{ty}^k \in [0, \zeta_{ty}] (7)$ Moreover,  $K_{i,t}$  changes over time as the human user interacts with the states of the environment E. So,  $K_{i,t}$  $\xrightarrow{Interaction \ with \ E} K_{i,t'} \text{ . Here, } t' = t + \Delta t \text{ and the } \cup_{update}^k$ operation is defined as:

$$K_{i,t'} = K_{i,t} \cup_{update}^{k} \left( ct'_{ty} \right)$$
(8)

Where

$$K_{i,t'} = \begin{cases} K_{i,t} \cup (ct'_{ty}, \delta^{k0}_{ty}) \text{ if } K_{i,t} \Vdash_k (ct'_{ty}) \\ K_{i,t} \cup (ct'_{ty}, ex'_{i,ty,t} \pm \delta^{ku}_{i,ty}) \text{ otherwise} \end{cases}$$
(9)

Where  $\delta_{i,ty}^{ku} \in \mathbb{R}$  is a variable that represents  $h_i$ 's ability to update his or her knowledge base  $K_{i,t}$ .

In *i*HUCOFS, we represent a human user's knowledge about *what* to do in an environment state with the *behavior* base  $B_{i,t}$  where

$$B_{i,t} = \{ (es_{ty,t}, ac_{ty,t}, ut_{i,ty,t}) \}$$
(10)

where  $es_{ty,t}$  denotes an environment state that could be encountered by the human user  $h_i$  while solving a task of type ty at time t,  $ac_{ty,t}$  denotes the  $h_i$ 's expected action in the state  $es_{ty,t}$ , and  $ut_{i,ty,t}$  is the expected utility for  $h_i$  when he or she applies  $ac_{ty,t}$  on environment state  $es_{ty,t}$ . Again, we define the *behavior base update operator*  $\Vdash_b$  as:

 $B_{i,t} \Vdash_{b} (es_{ty,t}, ac_{ty,t}) \text{ if } (es_{ty,t}, ac_{ty,t}, \delta_{ty}^{b}) \in B_{i,t} (11)$ for some  $\delta_{ty}^{b} \in \mathbb{R}$ . Furthermore,

 $ut_{i,ty,t} = fnc(ct_{ty}, es_{ty,t}, ac_{ty,t})$ (12) where *fnc* is some function that depends on the state-action pair (*es*<sub>ty,t</sub>, *ac*<sub>ty,t</sub>) of environment *E* and tasks of type *ty*.

 $B_{i,t}$  also gets updated as the human user interacts with the environment states. So,  $B_{i,t} \xrightarrow{Interaction \ with E} B_{i,t'}$  where  $t' = t + \Delta t$  and

 $B_{i,t'} = B_{i,t} \cup_{update}^{b} (es'_{ty,t}, ac'_{ty,t}, ut'_{i,ty,t})$ 

where

$$B_{i,t'} = \begin{cases} B_{i,t} \cup (es'_{ty,t}, ac'_{ty,t}, ut'_{i,ty,t}) \\ if B_{i,t} \Vdash_{b} (es'_{ty,t}, ac'_{ty,t}) \\ B_{i,t} - (es'_{ty,t}, ac'_{ty,t}, ut'_{i,ty,t}) \cup \\ (es'_{ty,t}, ac'_{ty,t}, ut'_{i,ty,t} \pm \delta^{bu}_{ty}) otherwise \end{cases}$$
(1)

here  $\delta_{ty}^{bu} \in \mathbb{R}$ .

Not all human users are able to learn new behaviors at the same rate. We define the abilities of a human user  $h_i$  to learn something (capability or behavior) about a task of type ty using derivative and communicative learning by  $DLC_{i,t}$ ,  $CLC_{i,t}$  respectively. Here,  $DLC_{i,t}$  is a set defined as:

$$DLC_{i,t} = \left\{ \left( dlk_{i,ty,t}, dlb_{i,ty,t} \right) | ty \in T_j \right\}$$
(15)

where  $dlk_{i,ty,t} \in [0,1]$  and  $dlb_{i,ty,t} \in [0,1]$ . Further,  $CLC_{i,t}$  is a set defined as:

$$CLC_{i,t} = \{ (clk_{i,ty,t}, clb_{i,ty,t}) | ty \in T_j \}$$

$$(16)$$

where,  $clk_{i,ty,t} \in \{0,1\}$  and  $clb_{i,ty,t} \in \{0,1\}$ .

Finally, we denote the combined autonomy of the human user and his or her assigned user agent while working on a task of type ty at time t by

 $CA_{i,t} = \{ (ha_{i,ty,t}, ua_{i,ty,t}) | ty \in 1 \dots | ty_{j,k} | \}$ (17) Here, the autonomy of the human user while working in coalition  $C_{k,t}$  at time *t* executing a task of type *ty* is defined by

$$ha_{i,k,ty,t} = \frac{|DS_{i,t}^{n}|}{|DS_{i,t}|}$$
(18)

where

$$DS_{i,t} = \left\{ \left( es_{n,ty,t}, ac_{n,ty,t} \right) | n \in \mathbb{Z} \right\}$$
(19)

 $DS_{i,t}$  is a set of state-action pairs generated by the human user  $h_i$  and the user agent  $u_i$  at time t and for working on tasks of type ty. Further,  $DS_{i,t}^h \subseteq DS_{i,t}$  and

$$DS_{i,t}^{h} = \left\{ \left( es_{n,ty,t}, ac_{n,ty,t}^{h} \right) | n \le \left| DS_{i,t} \right| \right\}$$
(20)

Here,  $ac_{n,ty,t}^{h}$  is the action generated by the human user. Notice that,  $ha_{i,ty,t} \in [0,1]$ . Then we define the user agent's autonomy while the human user  $h_i$  is working in coalition  $C_{k,t}$  executing a task of type ty as

$$ua_{i,k,ty,t} = 1 - ha_{i,k,ty,t}$$
 (21)

 $EU_{i,t}$  is a set of real values that represents the estimated utility that can be gained by  $h_i$  by joining a coalition at time *t* measured from the perspective of  $u_i$ .  $EU_{i,t}$  is defined as

$$EU_{i,t} = \{ eu_{i,j,k,t} | k = 1 \dots | C_t | \}$$
(22)

So,  $eu_{i,k,t}$  is the estimated amount of utility that can be gained by be gained by  $h_i$  measured from the perspective of  $u_i$  by joining a coalition  $C_{k,t}$  and executing a task  $T_j$  at time t.

**Representational Assumption 9.** A coalition  $C_{k,t} \in C$  at time *t* can be specified as a 12-tuple,

$$C_{k,t} = \begin{pmatrix} U_{i,k}, H_{i,k}, g_k, o_{S_j,k,t}, su_{j,k,t}, T_j, R_{i,j,k,t}, Y_{i,j,k,t}, OM_{i,j,k,t} \\ TQA_{j,k,t}, PCN_{j,k,t}, TCN_{j,k,t} \end{pmatrix}$$
(23)

where  $U_{i,k} \subseteq U, H_{i,k} \subseteq H$ ,  $g_k \in G, T_j \in T, os_{j,k,t} \in su_{j,k,t} \in [0,1]$ 

$$i_{i,j,k,t} = \{r_{i,j,k,t} | i = 1 \dots | H_{ik} | \}$$
 (24)

$$y_{i,k,t} = \{y_{i,j,k,t} | i = 1 \dots | H_{ik} | \}, y_{i,j,k,t} \in [0,1]$$
 (25)

$$OM_{i,j,k,t} = \{ (ou_{i,j,k,t}, oh_{i,j,k,t}) | i = 1 \dots |H_{ik}| \}$$
(26)

$$TQA_{j,k,t} = \{tqa_{m,j,k,t} | m = 1 \dots | ts_j | \}$$
(27)

$$PCN_{j,k,t} = \{pcn_{i,m,j,k,t} | t = 1 \dots | H_{i,k} |, m = 1 \dots | ts_j | \} (28)$$
$$TCN_{j,k,t} = \{tcn_{i,m,j,k,t} | i = 1 \dots | H_{i,k} |, m = 1 \dots | ts_j | \} (29)$$

Here,

 $os_{j,k,t}$  is the amount of resources spent by the system agent to form coalition  $C_{k,t}$  to solve task  $T_j$  at time tmeasured from the perspective of the system agent. Examples of this cost could be communication bandwidth, computational time, deliberation time, etc.

- $su_{j,k,t}$  is the expected utility that can be gained by S by forming coalition  $C_{k,t}$  to solve task  $T_j$  at time t
- *r<sub>i,j,k,t</sub>* denotes the expected reward the human user *h<sub>i</sub>* can earn by working in the coalition *C<sub>k,t</sub>* calculated from the perspective of the user agent *u<sub>i</sub>*
- $y_{i,j,k,t}$  denotes the expected utility the human user  $h_i$  can gain by joining the coalition  $C_{k,t}$  and solving task  $T_j$  cooperatively at time t with the members of  $C_{k,t}$  calculated from the perspective of the user agent  $u_i$  assigned to  $h_i$ . Although,  $r_{i,j,k,t}$ , and  $y_{i,j,k,t}$  are estimates, when the assigned tasks are completed at time  $t = ta_{j,k} + tl_{j,k}$ , these estimated values become actual values.
- $tqa_{j,k,t}$  denotes the quality of the completed subtasks in  $ts_{j,k} \in T_j$  achieved by the coalition  $C_{k,t}$  at time t. We also assume that  $tqa_{m,j,k,t} \le tq_{m,j,k} \forall m = 1 \dots |ts_{j,k}|, t$ .
- *pcn<sub>i,m,j,k,t</sub>* denotes human user *hi*'s *estimated potential* contribution for completing the *m*th subtask in *ts<sub>j,k</sub>* ∈ *T<sub>j,k</sub>* in coalition *C<sub>k,t</sub>* at time *t* measured from the perspective of *u<sub>i,k</sub>*.
- $tcn_{i,m,j,k,t}$  denotes human user hi's actual contribution for completing the *m*th subtask in  $ts_j \in T_j$  that was achieved by the coalition  $C_{k,t}$  at time *t* measured from the perspective of  $g_k$ .

•  $ou_{i,j,k,t}$  and  $oh_{i,j,k,t}$  are the costs of forming coalition  $C_{k,t}$  incurred by the user agent and the human user respectively. Examples of the cost incurred by the user agent while forming the coalition can be communication bandwidth, deliberation time, etc. Examples of costs incurred by the human user can be time, misconceptions, misunderstanding of their human counterparts, communication with the assigned user agent, and communication with other human users, etc.

**Representational Assumption 10.** The formed coalitions are non-overlapping. So, at any time t,

$$\forall h_i \in H, C_{k,t}, C_{k't} \in C$$
  
MemberOf( $h_i, C_{k,t}$ )  $\rightarrow \sim$ MemberOf( $h_i, C_{k',t}$ ) if  $k \neq k'$ (30)

**Representational Assumption 11.** Each group agent  $g_k \in G$  is assigned to a coalition  $C_{k,t}$ . This assignment is one-to-one and  $n_g = |C_t|$ .

**Representational Assumption 12.** The *effectiveness* of a coalition  $C_{k,t}$  (as defined in Eq. (23)) working on task  $T_j$  (as defined in Eq. (1)) is defined as,

where

$$\xi_{C_{k,t}} = \langle \xi_{m,j,k,t} \rangle \tag{31}$$
$$m = 1 \dots |TOA_{i,k,t}| \text{ and }$$

$$\xi_{m,j,k,t} = \left[1 - \left(tq_{m,j,k} - tqa_{m,j,k,t}\right)\right]$$
(32)

**Representational Assumption 13.** A coalition is said to be *efficient from the perspective of a user agent* if it generates more reward for the participating human users by solving the assigned tasks than the total cost incurred by those human users and their assigned user agents while forming and working in the coalition. So, the efficiency of coalition  $C_{k,t}$  (as defined in Eq. (23)) measured from the perspective of the user agent  $u_i$  is denoted by

$$\eta^{u}_{\mathcal{C}_{k,t}} = r_{i,j,k,t} - \left[ o u_{i,j,k,t} + o h_{i,j,k,t} \right]$$
(33)

Here, the efficiency of the coalition from the user agent's point of view is determined by the reward it can earn by solving task  $T_j$  and the cost of forming and maintaining coalition  $C_{k,t}$ . Furthermore,  $t = ta_j + tl_j$ ,  $ta_j$  and  $tl_j$  are defined in Eq. (1). Further,  $ou_{i,j,k,t}$  and  $oh_{i,j,k,t}$  are defined in Eq. (23). So, according to our definition, the coalition  $C_{k,t}$  is efficient when (assuming  $\eta_{C_{k,t}}^u > 0$ )

$$r_{i,j,k,t} > ou_{i,j,k,t} + oh_{i,j,k,t}$$
 (34)

Similarly, a coalition is said to be *efficient from the perspective of a system agent* if it generates more reward for the system agent by solving the assigned tasks than the total cost incurred by the system agent while forming and maintaining the coalition. So, the *efficiency* of coalition  $C_{k,t}$  (Eq. (23)) measured from the perspective of the system agent is

$$\eta_{C_{k,t}}^{s} = \sum_{i=1}^{|ts_{j}|} tw_{m,j} - \left[\sum_{i=1}^{|H_{i,k}|} r_{i,j,k,t} + os_{j,k,t}\right]$$
(35)

Here, the efficiency of the coalition from the system agent's point of view is determined by the reward it can earn by solving task  $T_j$  and the cost of forming and maintaining coalition  $C_{k,t}$  and the rewards it has to distribute to the user agents. Furthermore,  $t = ta_j + tl_j$ ,  $ta_j$  and  $tl_j$  are defined in Eq. (1). Further,  $r_{i,j,k,t}$  and  $os_{j,k,t}$  are defined in Eq. (23). So, according to our definition, the coalition  $C_{k,t}$  is efficient from the perspective of the system agent when

$$\sum_{i=1}^{|ts_j|} tw_{m,j} > \left[ \sum_{i=1}^{|H_{i,k}|} r_{i,j,k,t} + os_{j,k,t} \right]$$
(36)  
i.e.,  $\eta_{C_{k,t}}^s > 0$ .

**Representational Assumption 14.** To capture the change in a human user's behavior due to the Derivative and Communicative learning, we define

$$L = (DL, CL) \tag{37}$$

Here, the tuple L represents the learning of the human user and it contains DL and CL which represent derivative and communicative learning respectively. We also define,

$$DL = (dk, db) \tag{38}$$

where dk is a function that updates the knowledge base of the human user as:

$$dk(K_{i,t}, es_{ty,t}) = \begin{cases} K_{i,t}, \text{ if } dlk_{i,ty,t} = 1\\ K_{i,t}, \text{ if } dlk_{i,ty,t} = 0 \end{cases}$$
(39)

Here  $K_{i,t'}$  and  $dlk_{i,ty,t}$  are defined in Eq. (9) and Eq. (15) respectively. Similarly, db is a function that updates the behavior base of the human user as:

$$b(B_{i,t}, es_{ty,t}) = \begin{cases} B_{i,t}, if \ dlb_{i,ty,t} = 1\\ B_{i,t}, if \ dlb_{i,ty,t} = 0 \end{cases}$$
(40)

where  $B_{i,t}$  and  $dlb_{i,ty,t}$  are defined in Eq. (14) and Eq. (15) respectively. Furthermore, communicative learning *CL* is defined as:

$$CL = (ck, cb) \tag{41}$$

where ck is function that updates the knowledge base of a human user as:

$$k(K_{i,t}, ct_{ty}) = \begin{cases} K_{i,t'}, & \text{if } clk_{i,ty,t} = 1\\ K_{i,t}, & \text{if } clk_{i,ty,t} = 0 \end{cases}$$
(42)

Here,  $K_{i,t'}$  and  $clk_{i,ty,t}$  are defined in Eq. (9) and Eq. (16) respectively. Further, *cb* is a function that updates the behavior base of the human user as:

$$cb(B_{i,t}, (es_{ty,t}, ac_{ty,t})) = \begin{cases} B_{i,t}', & if \ clb_{i,ty,t} = 1\\ B_{i,t}, & if \ clb_{i,ty,t} = 0 \end{cases}$$
(43)

Here,  $B_{i,t'}$  and  $clb_{i,ty,t}$  and defined in Eq. (14) and Eq. (16) respectively.

**Representational Assumption 15.** Due to the different types of learning described in Eq. (38) and Eq. (41), the knowledge base and the behavior base of the human user changes. As a result, the performance of a human user as an individual and as a coalition member would change. For example, due to learning new capabilities, a human user becomes able to solve the tasks encountered in future coalitions more efficiently (lower cost  $oh_{i,j,k,t}$  (Eq. (26)). To capture this change, we define the *performance change* of a human user  $h_i$  while working in a coalition  $C_{k,t}$  solving task  $T_j$  at time t as

$$PC_{i,j,k,t} = \sum_{K_{i,t} \vdash k} (ct_{ty,ex_{ty,t}}) |ex_{ty,t} - ex_{ty,t'}| + \sum_{K_{i,t'} \vdash k} (ct_{ty,ex_{ty,t'}}) ex_{ty,t'} + \sum_{B_{i,t} \vdash b} (es_{ty,t,ac_{ty,t},ut_{i,ty,t}}) |ut_{i,ty,t} - ut_{i,ty,t'}| + \sum_{B_{i,t'} \vdash b} (es_{ty,t,ac_{ty,t},ut_{i,ty,t'}}) ut_{i,ty,t'}$$
(44)

where  $K_{i,t}$ ,  $B_{i,t}$ ,  $K_{i,t'}$ , and  $B_{i,t'}$  are defined in Eq. (8), Eq. (10), Eq. (9), and Eq. (14) respectively.

**Representational Assumption 16.** The utility  $y_{i,j,k,t}$  gained by the human user  $h_{i,k}$  while working in a coalition  $C_{k,t}$  (Eq. (23)) is defined by

$$y_{i,j,k,t} = y_{i,j,k,t}^{ct} + y_{i,j,k,t}^{ft}$$
(45)

where  $y_{i,j,k,t}^{ct}$  is the utility gained for executing the current task  $T_j$  assigned to  $C_{k,t}$  and  $y_{i,j,k,t}^{ft}$  is the *estimated increase* of utility gains for the future tasks at  $t' = t + \Delta t$ .

The utility gained from the current task is rewarded to the human user due to his or her contribution in solving the subtasks of  $T_j$ . So,

$$y_{i,j,k,t}^{ct} \propto r_{i,j,k,t} - \left[ou_{i,j,k,t} + oh_{i,j,k,t}\right]$$
(46)

where  $r_{i,j,k,t}$ ,  $ou_{i,j,k,t}$ , and  $oh_{i,j,k,t}$  are defined in Eq. (24) and Eq. (26) respectively.

The estimated increase of utility for the future tasks arises from the fact that the human users learn from working in the coalitions. While working in a coalition, a human user interacts with the environment and executes tasks. Due his or her interaction with the environment, especially with other coalition members, a human user may be able to learn new capabilities and behaviors. These interactions may also allow a human user to improve his or her knowledge and expertise on the capabilities he or she already knows. This improved behavior and knowledge would increase the utility a human user earns by solving tasks in future coalitions. So, the expected increase of the future utility gained by working in future coalitions for a human user is proportional to his or her improvement in performance that resulted from working in that coalition. That means,

$$y_{i,j,k,t}^{ft} \propto PC_{i,j,k,t} \tag{47}$$

where  $PC_{i,j,k,t}$  is defined in Eq. (44).

**Representational Assumption 17.** The utility  $su_{j,k,t}$  gained by the system agent *S* by forming and maintaining a coalition  $C_{k,t}$  (Eq. (23)) is defined by

$$su_{j,k,t} = su_{j,k,t}^{ct} + su_{j,k,t}^{ft}$$
 (48)

where  $su_{j,k,t}^{ct}$  is the utility gained for executing the current task  $T_j$  assigned to  $C_{k,t}$  and  $su_{j,k,t}^{ft}$  is the *estimated increase* of utility gains for the future tasks at  $t' = t + \Delta t$ .

The utility gained from the current task is gained by the system user for solving the subtasks of  $T_j$ . So,

$$su_{j,k,t}^{ct} \propto \sum_{i=1}^{[ts_j]} tw_{m,j} - \left[\sum_{i=1}^{[H_{i,k}]} r_{i,j,k,t} + os_{j,k,t}\right]$$
(49)

where  $tw_{m,j}$ , is the total reward earned by the *S* for solving task  $T_j$  by forming coalition  $C_k$ ,  $r_{i,j,k,t}$  is the reward the system agent provides the human user  $h_i$  by *S*, and  $os_{j,k,t}$  is the cost incurred by *S* for forming the coalition  $C_{k,t}$ .

The estimated increase of utility for the future tasks arises from the fact that the human users learn from working in the coalitions and their performance changes over time (Eq. (44). This improvement in human user's behavior may then improve the utility  $su_{j,k,t}^{ct}$  for  $t = t + \Delta t$ . If the human users' performances improve, the better performing human users will be able execute the assigned task more efficiently (less cost  $os_{j,k,t}$ ). As a result, the system agent's utility for solving the future tasks ( $su_{j,k,t}^{ct}$ ) would improve too. So, the expected increase of the system agent's utility for future tasks generated by forming and maintaining coalition  $C_{k,t}$  is proportional to the sum of the potential improvements of performance of all the members of  $C_{k,t}$ . That means,

$$su_{j,k,t}^{ft} \propto \sum_{i \in H_{i,k}} PC_{i,j,k,t}$$
(50)

Here,  $PC_{i,j,k,t}$  is defined in Eq. (44).

**Representational Assumption 18.** We define the modeling accuracy of the human user model  $hm_{i,t}$  as:

$$MA_{i,t} = \left\{ ma_{i,j,k,t} \mid k \in \mathbb{Z} \right\}$$
(51)

Here,

$$ma_{i,k,t} = |eu_{i,j,k,t} - y_{i,j,k,t}|$$
(52)

where  $y_{i,j,k,t}$  is the actual utility achieved by the human user  $h_i$  by working in a coalition  $C_{k,t}$  and  $eu_{i,j,k,t}$  and is the utility that can be gained by the human user  $h_i$  by joining the coalition  $C_{k,t}$  as estimated by the human user model  $hm_{i,t}$ .

**Characteristic Assumption 1: Coalition Scaffolding.** According to socio-cultural theory [28], learning involves social interaction and dialogue, negotiation and collaboration and that 'scaffolded' or assisted learning can increase cognitive growth and understanding. In educational research, Scaffolding is referred to as a form of assistance provided to a learner by a more capable teacher or peer that helps the learners perform a task that would normally not be possible to accomplish by working independently.

Similar to the idea of scaffolding in a classroom, Scaffolding a human coalition is to support a group of humans to help them work together when solving a problem. In other words, the system agent and the user agents try to guide the human users to change their behaviors to improve their performance as individuals and as coalition members. This improved behavior is then observed by the user agents assigned to the human users when they are interacting with the environment E. As a result, the human users' models constructed by the user agents get updated. In other words, in *i*HUCOFS, the system agent S and the user agents  $u_i$ s scaffold the human user  $h_i$ s to see improvements in  $hm_{i,t}$ .

The change in a human user's behavior due to the scaffolding improves his or her performance for current and future tasks. In other words, scaffolding enables the human user *learn* new capabilities and behaviors which increase the utility that human user can earn from and contribute to the current and future coalitions he or she works in.

Scaffolding can be of two types: **I**. A human user is guided by the assigned user agent *explicitly* to help him learn how to change his or her behavior for the current task in the current coalition; and **II**. The system agent or the assigned user agent constructs environment states (*implicit help*) that allow the human user learn how to improve his or her behavior in the future coalitions.

Say, a human user  $h_i$  is working in a coalition  $C_{k,t}$  to execute a task  $T_j$  of type ty. Then, an example of Type I scaffolding could be hints or guidance related to tasks of type ty provided to  $h_i$  by the assigned user agent  $u_i$ . On the other hand, say a human user  $h_i$  is deciding which coalition  $C_{k,t} \in C_t$  to join to earn rewards by executing a task  $T_j$  of type ty which the human user does not know much about. In that case, an example of Type II scaffolding provided by the user agent  $u_i$  could be the advice to join the coalition that contains a set of users  $H_s \in H$  whose models  $hm_s$  indicate that they are able to execute the tasks of type ty. Notice that  $u_i$  is able to find out the most suitable coalition for  $h_i$  by communicating with the other user agent wants to get rewards by solving a set of tasks  $T = \{T_i | j = 1 \dots n\}$  of type

 $ty_j \in T_j$  by forming various coalitions of human users H and  $H_s \in H$ . Also, the system agent knows from the models of  $hm_{s,t}$ s that the human users  $H_s$  are not able to solve a subset of tasks  $T'_j \subseteq T$  of types  $ty'_j$ . The system agent also knows that, user models  $hm_{s',t}$  indicate the human users  $H_{s'} \in H - H_s$  are able to solve the tasks tasks  $T'_j \subseteq T$  are of types  $ty'_j$ . Then, the system agent S may provide some initiative (e.g., reward) to motivate the human users  $H_{s'}$  to form coalitions with human users  $H_s$ . Such a coalition may enable the human users  $H_{s'}$  and improve their performances. As a result, all the human users in set  $H_s$  and  $H_{s'}$  will be able to solve tasks of type  $ty'_j$  in future.

While using Type I scaffolding, the user agent provides the information about capabilities (e.g.,  $ct_{i,ty}$  related to a type of task ty) and information about the environment states and the optimal actions (e.g.,  $(es_{ty,t}, ac_{i,ty,t})$  related to tasks of type ty) to the human user in the hope that these information may invoke explicit human learning *EL*. If the human user is able to use his or her explicit learning, the information or guidance provided by the assigned user agent will improve the performance of the human user. In other words, Type I scaffolding can be defined as:

$$sc1(hm_{i,t}, es_{ty,t}) = \left(hm_{i,t'}, \left(es_{ty,t}, ac_{i,ty,t}\right)\right)$$
(53)

where  $ac_{i,ty,t}$  is the user agent's action on the environment state  $es_{ty,t}$  and  $hm_{i,t'}$  is an improved model of the human user  $h_i$  at  $t = t + \Delta t$ . An example of the improvement of the human user model could be if the human user is able to increase his or her expertise level for some capability in his or her knowledge base (Eq. (9)) or if the user could increase his or her utility for some state-action pair in the behavior base (Eq. (14)), etc.

On the other hand, while using Type II scaffolding techniques, the user agent generates a set of environment states  $es_{ty,t}$  related to tasks ty and those generated environment states improve the human user's model. So, Type II scaffolding can be defined as:

$$sc2(hm_{i,t}, es_{ty,t}) = (hm_{i,t}', (es_{ty,t}, ac_{i,ty,t}))$$
 (54)

Where  $ac_{i,ty,t}$  is the user agent's action on environment state  $es_{ty,t}$  to generate a set of states to improve  $hm_{i,t'}$  at some time  $t = t + \Delta t$ .

Finally, while using Type II scaffolding techniques, the system agent generates a set of environment state  $es_{ty,t}$  related to tasks ty and those generated environment states improve the model of a set of human users. So, Type II scaffolding can be defined as:

$$sc2(hm_{s,t}, es_{ty,t}) = (hm_{s,t'}, (es_{ty,t}, ac_{i,ty,t})) \forall h_s \subseteq H$$
(55)

where  $ac_{i,ty,t}$  is  $h_i$ 's action on state  $es_{ty,t}$  to generate a set of states to improve  $hm_{s,t}$  at some time  $t = t + \Delta t$ .

The reason behind using scaffolding is to invoke human learning. Human learning in a collaborative setting can come in various shapes and forms [11]. Next, we discuss the different types of learning and explain how they are related to the scaffolding in *i*HUCOFS.

For the following discussions, we assume that there is a coalition  $C_{k,t}$  (as defined in Eq. (22)) in the environment *E*. We also assume,  $H_{s,k} \subseteq H_{i,k}$  where

$$H_{s,k} = \left\{ h_{s_p,k} | p = 1 \dots | H_{i,k} - 1 | \right\}$$
(56)

To describe the learning, we assume the following:

- a.  $H_{s,k} \subseteq H_{i,k}$  where  $H_{s,k} = \{h_{s_p,k} | p = 1 \dots | H_{i,k} 1 | \}$
- b. The human user  $h_i$  is trying to learn capability  $ct_{i,ty}$  and state, action pair  $(es_{ty,t}, ac_{i,ty,t})$  through various learning processes
- c. Human user  $h_i$  and  $h_{s_p}$  have models  $hm_{i,t}$  and  $hm_{s_p,t}$  respectively. The human user model is defined in Eq. (5)

d. 
$$t = t + \Delta t$$
 and  $t = t + \Delta t$ 

- e. ObserveProcess(x, y, t) states that human user x is observing a process y at time t. Notice that, observing process z could mean observing a sequence of stateaction pairs for the human user x.
- f. Teach(x, y, z, t) states that human user x teaches human user y how to execute tasks of type z at time t
- g. Reflect(x, y, z, t) states that human user x explains his or her execution of a task of type z to human user y at time t

ObserveBehavior (x, y, (z, w)) states that human user x observes the action w executed by the user agent y at an environment state z

- i. Perform(x, (y, z)) states that human user x takes the action z while in an environment state y
  - Communicate(x, y, z, t) states that human user x communicates with human user y about a task of type z at time t

**a. Learning by Observation** – The users learn indirectly by observing other learners' learning process. This type of learning can be facilitated by the user agent by putting the human user in a group that contains users with similar deficiency of knowledge about a certain task. When such a group is working together to execute a task they do not know much about, the user agent can provide interactive targeted learning materials so that at least some of the users can learn from it. Then the other users will be able to observe their learning process and will learn from it. This learning can be described from the perspective of the user agents in *i*HUCOFS as:

 $\forall p \ Observe Process\left(h_{i,k}, dk\left(K_{s_{p},t}, es_{ty,t}\right), t\right) \land$ 

$$K_{s_{p},t} \not\models_{k} ct_{ty} \land (K_{i,t} \not\models_{k} ct_{ty}) \stackrel{\text{an}}{\to} K_{i,t'}$$
(57)

where user  $h_{i,k}$  observes the derivative learning process of user  $h_{s_p,k}$  while working together to execute task of type *ty*. Further,  $h_{i,k}$  and  $h_{s_p,k}$  do not have capability  $ct_{ty}$  in their knowledge bases  $K_{s_p,t}$  and  $K_{i,t}$  respectively. Furthermore,  $K_{i,t'}$  is defined in Eq. (8) and  $\mathbb{H}_k$  operator is defined in Eq. (7).

$$\forall p \ ObserveProcess\left(h_{i,k}, db\left(B_{s_{p},t}, es_{ty,t}\right), t\right) \land \\ \left(B_{s_{p},t} \Vdash_{b} \left(es_{ty,t}, ac_{ty,t}\right)\right) \land \left(B_{i,t} \Vdash_{b} \left(es_{ty,t}, ac_{ty,t}\right)\right) \\ \xrightarrow{db}{\rightarrow} B_{i,t'} (58)$$

where user  $h_{i,k}$  observes the derivative learning process of user  $h_{s_p,k}$  while working together to execute task of type ty. As a result, the behavior base of  $h_{i,k}$  changes. Further,  $h_{i,k}$  and  $h_{s_p,k}$  do not have state-action pair  $(es_{ty,t}, ac_{ty,t})$  in their behavior bases  $B_{s_p,t}$  and  $B_{i,t}$  respectively. Finally,  $B_{i,t'}$  is defined in Eq. (8).

**b.** Learning by Teaching/Guiding – Learning by teaching occurs when a human user learns or refines his or her own knowledge by teaching other group members. This type of learning is particularly useful in CSCL settings where the students learn by teaching each other. Again a human coalition formation framework can provide an environment for this type of learning by putting a human user in a group that would allow him or her to learn by teaching others. However, this type of learning requires that the user teaching others is knowledgeable about the assigned problem and is able to express his or her ideas and is comfortable about teaching others.

$$\forall p \, Teach\left(h_{s_{p,k}}, h_{i,k}, ty, t\right) \land \left(K_{s_{p,t}} \Vdash_{k} ct_{ty}\right) \land \\ \left(K_{i,t} \Vdash_{k} ct_{ty}\right) \xrightarrow{dk} K_{s_{p,t}'}$$
(59)

where user  $h_{s_p,k}$  teaches user  $h_{i,k}$ . and the knowledge base  $K_{s_p,t}$  changes to  $K_{s_p,t'}$  ( $i = s_p$  in Eq. (8)). Also,

$$\forall p \, Teach\left(h_{s_{p},k}, h_{i,k}, ty, t\right) \land \left(B_{s_{p},t} \Vdash_{b} \left(es_{ty,t}, ac_{ty,t}\right)\right) \land \left(B_{i,t} \Vdash_{b} \left(es_{ty,t}, ac_{ty,t}\right)\right) \xrightarrow{db}{\rightarrow} B_{s_{p},t}$$
(60)

where user  $h_{s_p,k}$  teaches user  $h_{i,k}$  at time t. and the behavior base  $B_{s_p,t}$  of user  $h_{s_p,k}$  is changed. Furthermore,  $B_{s_p,t'}$  can be found by substituting  $i = s_p$  in Eq. (14).

**c.** Learning by being Taught – This is the simplest type of learning where a human user learns when he or she is being taught by someone else. Therefore, we see that learning by teaching and learning by being taught may complement each other. When a human user is learning by teaching other group members, those group members could learn by being taught.

$$\forall p \, Teach\left(h_{s_{p},k}, h_{i,k}, ty, t\right) \land \left(K_{s_{p},t} \Vdash_{k} ct_{ty}\right) \land \left(K_{i,t} \not \Vdash_{k} ct_{ty}\right) \stackrel{ck}{\rightarrow} K_{i,t'}$$
(61)

where user  $h_{s_{p,k}}$  teaches user  $h_{i,k}$ . As a result, the knowledge base  $K_{i,t}$  of user  $h_{i,k}$  is changed. Furthermore,  $K_{i,t'}$  can be found in Eq. (8). Also,

$$\forall p \ Teach\left(h_{s_{p},k}, h_{i,k}, ty, t\right) \land \left(B_{s_{p},t} \Vdash_{b} \left(es_{ty,t}, ac_{ty,t}\right)\right) \land \left(B_{i,t} \Vdash_{b} \left(es_{ty,t}, ac_{ty,t}\right)\right) \overset{cb}{\rightarrow} B_{i,t'} (62)$$

where user  $h_{s_p,k}$  teaches user  $h_{i,k}$  at time *t*. As a result, the behavior base  $B_{i,t}$  of user  $h_{i,k}$  is changed. Furthermore,  $B_{i,t'}$  can be found in Eq. (14).

**d. Learning by Reflection/Self-Expression** – This type of learning occurs when a human user rethinks his or her own solution and analyses his or her self-thinking process. Schön p. 28 [21] describes the reflection

process as: "We think critically about the thinking that got us into this fix or this opportunity; and we may, in the process, restructure strategies of action, understanding of phenomena, or ways of framing problems." Learning by reflection could occur when a group of users have completed a problem and are analyzing their solution process. This type of learning can also be achieved by using Type I scaffolding in combination of a structured collaborative process. For example, after each problem is solved by the human users, the collaborative process could involve a stage where each human user would discuss why his or her solution worked or did not work. If a human user is reluctant to discuss his or her solution process, the user agent may prompt him or her and engage that user to reflect on his or her own solution or thinking process. Notice that learning by reflection

$$\forall p \ Explain\left(h_{i,k}, h_{s_{p},k}, ty, t\right) \land \left(K_{i,t} \Vdash_{k} ct_{ty}\right) \\ \stackrel{dk}{\rightarrow} K_{i,t}'(63)$$

where user  $h_{i,k}$  explains his or her execution of a task of type ty to user  $h_{s_p,k}$ . and the knowledge base  $K_{i,t}$  changes to  $K_{i,t'}$  can be found in (Eq. (8)). Also,

$$\forall p \ Explain\left(h_{i,k}, h_{s_{p},k}, ty, t\right) \land$$
$$\left(B_{i,t} \Vdash_{b} \left(es_{ty,t}, ac_{ty,t}\right)\right) \xrightarrow{db} B_{i,t'} (64)$$

where user  $h_{i,k}$  explains his or her execution of a task of type ty to user  $h_{s_p,k}$ . and the behavior base  $B_{i,t}$  is changed to  $B_{i,t'}$  (Eq. (14)).

e. Learning by Apprenticeship – In traditional apprenticeship, the expert shows the apprentice how to do a task, watches as the apprentice practices portions of the task, and then turns over more and more responsibility until the apprentice is proficient enough to accomplish the task independently [5]. This type of learning can be implemented by Type I scaffolding. When a group of users are working together, the user agent may guide the group members so that when the most knowledgeable member explains or teaches something to the other group members, it can prompt some other group member to reexplain and re-do the example or problem. This way, when that human user solves the problem again, he or she will learn by apprenticeship. Note that learning by being taught improves the knowledge or skill of the human user who is being taught by someone else. On the contrary, learning by apprenticeship improves the knowledge of the human user who is *observing and mimicking* someone else's behavior.

$$\forall p$$

$$ObserveBehavior\left(h_{i,k}, h_{s_{p},k}, \left(es_{ty,t}, ac_{ty,t}\right)\right) \land$$

$$Perform\left(h_{i,k}, \left(es_{ty,t'}, ac_{ty,t'}\right)\right) \land$$

$$\left(B_{s_{p},t} \Vdash_{b} \left(es_{ty,t}, ac_{ty,t}\right)\right) \land \left(K_{s_{p},t} \Vdash_{k} ct_{ty}\right) \land$$

$$\left(K_{i,t} \nvDash_{k} ct_{ty}\right) \land \left(B_{i,t} \nvDash_{b} \left(es_{ty,t}, ac_{ty,t}\right)\right) \stackrel{db}{\rightarrow} K_{i,t''} \land$$

$$B_{i,t''} (65)$$

where user  $h_{i,k}$  observes some behavior of user  $h_{s_p,k}$  at time *t* and then mimicks that same behavior at time *t*'. As a result, the knowledge and behavior bases  $K_{i,t}$  and

 $B_{i,t}$  of user  $h_{i,k}$  is changed to  $B_{i,t''}$  and  $K_{i,t''}$  can be (for t' = t'' in Eq. (8) and Eq. (14) respectively).

f. Learning by Practice – This type of learning occurs when a human user applies his or her existing knowledge to solve an assigned problem. This type of learning is very common in situations where each human user contributes to the solution of the assigned problem by working on it. However, there may be human users who are free-riding i.e., depending on the competent and the knowledgeable users to solve the assigned problem. As a result, these users do not learn by practice. The user agent can provide reinforce the human users to learn by practice using Type II scaffolding. If the user agent detects that one of the human user is free-riding and is not contributing to the solution of the problem, it may put that human user in a group which contains human users who are not so proficient or knowledgeable about that assigned problem. Then, the free-riding human user would be forced to step up his or her effort and work on the assigned problem to avoid failing and getting penalized as a group. Notice that while learning by practice, the human user improves his or her expertise of a capability which he or she already knows. However, while learning by apprenticeship, the human user learns something he or she does not know.

$$\forall p \ Perform\left(h_{i,k}, \left(es_{ty,t}, ac_{ty,t}\right)\right) \land \\ \left(B_{i,t} \Vdash_{b} \left(es_{ty,t}, ac_{ty,t}\right)\right) \land \left(K_{i,t} \Vdash_{k} ct_{ty}\right) \xrightarrow{db} B_{i,t'} \land K_{i,t'}$$
(66)

where user  $h_{i,k}$  executes some action on the environment that is required for execution for a task of type ty. As a result, the knowledge base  $K_{i,t}$  and the behavior base  $B_{i,t}$ of user  $h_{i,k}$  is changed. Furthermore,  $K_{i,t}$  and  $B_{i,t}$  can be found in Eq. (8) and in Eq. (14) respectively.

g. Learning by Discussion – This type of learning occurs when the human users discuss a topic with each other. The human users can be made involved in this type of learning by using both Type I and Type II scaffolding. Using Type II scaffolding, a human user can be put into a group which contains users who he or she is comfortable with. This higher level of comfort would increase the probability that they would discuss the assigned problem or the approach to solution. On the other hand, if the users in a group are not discussing the assigned problem with his or her group members, the user agent can ask him or her to join the ongoing class discussion or ask leading questions that would engage that reluctant user. Notice that this type of learning is basically a sequence of Learning by Observation, Learning by Teaching, Learning by being Taught, Learning by Reflection/Self-Expression, Learning by Practice with except that the roles of the human users are dynamic in Learning by Discussion. Furthermore, Learning by Discussion is different from Learning by Apprenticeship since there are actions that are observed or mimicked by the human users.

**Characteristic Assumption 2: Tradeoff between Formation and Scaffolding**. Say the system agent *S* is forming a coalition  $C_{k,t}$  (Eq. (23)) to solve a task  $T_j$ . When  $T_j$  is completed, the system agent is able to collect the rewards and as a result, its utility  $su_{j,k,t}$  increases. However,  $su_{j,k,t}$  consists of:  $su_{j,k,t}^{ct}$  and  $su_{j,k,t}^{ft}$  (Eq. (48)).  $su_{j,k,t}^{ct}$  comes from the rewards earned by executing task  $T_j$ , and  $su_{j,k,t}^{ft}$  comes from the improvement of the behavior of the human users in the coalition, i.e.,  $\sum_{i \in H_{i,k}} PC_{i,j,k,t}$  (Eq. (44)). Also, to get the task  $T_j$ solved, the system agent incurs cost  $os_{j,k,t}$ . This cost can be broken down as,

$$os_{j,k,t} = os_{j,k,t}^{cf} + os_{j,k,t}^{sc}$$
 (67)

Here,  $os_{j,k,t}^{sc}$  is the cost associated with forming the coalition,  $os_{j,k,t}^{sc}$  is the cost associated with scaffolding the coalition. If the system agent is able to earn a reward  $tw_j$  by solving a task  $T_j$ , then, its utility gain is inversely proportional to the cost of forming and scaffolding the coalition  $C_{k,t}$  and proportional to the reward  $tw_j$  and the improvement in the coalition members' performance. So,

$$su_{j,k,t} \propto \underbrace{tw_j \cdot \sum_{i \in H_{i,k}} PC_{i,j,k,t}}_{os_{j,k,t}}$$
(68)

To maximize its utility, the system agent could decide to spend more for forming the coalition (higher  $os_{j,k,t}^{cf}$ ), i.e., try to find the best possible set of people who can execute the assigned tasks without any further cost for maintaining the coalition. This would increase its utility for the current task, i.e.,  $su_{j,k,t}^{ct}$ . However, this choice requires that the system agent's knowledge about the human users (i.e., their models) is accurate and noise free. On the other hand, the system agent may choose to spend more for scaffolding the formed coalition in hope that the human performances are improved. As a result of this improvement, the system agent's utility for the future tasks (i.e.,  $su_{j,k,t}^{ft}$ ) would increase.

We also assume that the set of human users H in *i*HU-COFS changes over time. When new users join the system, their assigned user agents do not have accurate knowledge about them. As a result, the  $MA_{i,ty,t}$  values are low. Over time, after the user agents have observed the behaviors of their assigned human users for some time, the accuracy values  $MA_{i,ty,t}$  increase. At some time t = t', these models would be accurate enough to: (1) the system agent can form efficient and effective coalitions (2) the system agent is able to provide scaffolding to the human users to improve their behavior. So, when the system has a lot of new users, the system agent needs to emphasize the scaffolding process more. This is because the user agents' models of their human users are not accurate enough to form effective and efficient coalitions anyway. Therefore, spending resources to form the best possible coalition may not necessarily yield the maximum utility  $su_{j,k,t}$ . Over time, when the human users have been trained by the scaffolding process and the user modeling has become more accurate, it will be rational for the system agent to emphasize the coalition formation process more and spend more resources for forming the coalitions. In this situation, finding the right mix of people to work together is more important than scaffolding them after forming the coalition. So, over time, the system agent's emphasis crosses over to coalition formation from scaffolding.

**Characteristic Assumption 3: Dual Roles.** In *i*HU-COFS, the user agents assume two different roles: advisor and representative. When a user agent is acting as an advi-

sor, it takes decisions on behalf of the human user and has more autonomy than the human user, i.e.,  $ua_{i,k,t} > ha_{i,k,t}$  in the environment E. As an advisor, the user agent also tries to improve the behavior of the human user for the current tasks and future tasks by providing Type I and Type II scaffolding respectively. For example, a user agent may act as an advisor for a human user who is new to the system environment or who does not possess the necessary capabilities to execute tasks  $T_{i,k}$  while working in a coalition  $C_{k,t}$ . For such a user, the user agent may decide the coalition that would yield him or her the highest utility  $y_{i,j,k,t}$ . Furthermore, the user agent may also provide scaffolding to the human user while working in a coalition to improve his or her behavior in the future coalition. As a representative, the user agent follows the human user's advice and does not provide much scaffolding. The user agent may act as a representative for a human user who possesses the necessary capabilities to execute tasks  $T_{i,k}$ while working in a coalition  $C_{k,t}$ .

A human user's potential for contribution in a coalition  $C_{k,t}$  is denoted by  $pcn_{i,ty,j,k,t}$ . The value of  $pcn_{i,ty,j,k,t}$  tells the user agent  $u_i$  how much the human user  $h_i$  may be able to contribute while working in  $C_{k,t}$ . Since the utility gained by that human user is proportional to  $pcn_{i,ty,j,k,t}$ , a low  $pcn_{i,ty,j,k,t}$  value would mean smaller amount of earned utility for the human user. So, based on the potential contribution of a human user, the user agent can detect whether the human user is capable enough to work on his or her own in a coalition to execute the assigned tasks. Upon detecting such deficiency in the human user's capability, the user agent can assume the role of an advisor to guide the human user while he or she is working in the coalition. In that case, the user agent will have more autonomy than the human user in the environment E. On the other hand, if the user agent detects that the human user is able to work in the coalition on his or her own, it can assume a passive role as a representative. In that case, the human user will have more autonomy than the user agent in the environment E. Therefore, the user agent  $u_i$ 's autonomy  $ua_{i,ty,t}$  is a function of the human user's potential contribution  $pcn_{i,ty,j,k,t}$ , i.e.,

$$ua_{i,ty,t} = fnc(pcn_{i,ty,j,k,t})$$

(69)

**Characteristic Assumption 4: Tradeoff between Advi**sor and Representative. Say, the human user  $h_{i,k}$  is working in the coalition  $C_{k,t}$  (as defined in Eq. (23)) and the user agent incurs cost  $ou_{i,i,k,t}$ . This can be written as,

$$ou_{i,j,k,t} = ou_{i,j,k,t}^{ma} + ou_{i,j,k,t}^{cf} + ou_{i,j,k,t}^{sc}$$
(70)

where  $ou_{i,j,k,t}^{ma}$  is the cost of modeling the human user,  $ou_{i,j,k,t}^{cf}$  is the cost for forming the coalition, and  $ou_{i,j,k,t}^{sc}$  is the cost of scaffolding the human user. However, the value of  $ou_{i,j,k,t}^{sc}$  is a function of the user agent's autonomy  $ua_{i,k,ty,t}$ . If  $ua_{i,k,ty,t} = 0$ , the user agent is working as a representative of the human user following his or her every command without providing any scaffolding. On the other hand, if  $ua_{i,k,ty,t} = 1$ , the user agent is working as an advisor of the human user and taking all the decisions for him or her and providing scaffolding. So, as a mere representative, a user agent does not have any autonomy; as a mere advisor, a user agent has full autonomy. In brief, the value  $ou_{i,j,k,t}^{sc}$  is a function of  $ua_{i,k,t}$  where

$$ua_{i,k,t} = \sum_{ty_j \in T_j} ua_{i,k,ty_j,t}$$
(71)

So, we can write

$$ou_{i\,i\,k\,t}^{sc} = fnc(ua_{i,k,ty,t}) \tag{72}$$

Therefore, the optimum value of the user agent's autonomy that yields the lowest cost of scaffolding for a given task ty can be found by solving the equation

$$\frac{dou_{i,j,k,t}^{sc}}{dua_{i,k,t}} = 0 \tag{73}$$

Furthermore, the role of the user agent depends on what a user agent knows about the human user. Based on its model of the human user, the user agent may decide to be a representative or an advisor. As an advisor, the user agent has more autonomy than the human user and takes decisions on behalf of its assigned human user and provides scaffolding to its human user. Examples of advisory decisions can be *which coalition to join, how to execute a task, etc.* On the other hand, as a representative, the user agent becomes an assistant of the human user following his or her directions. In this case, the human user takes all the decisions and does not require any scaffolding from the user agent.

Since a human user's behavior changes over time, the user agent's role (advisor or representative) is dynamic. If the human user does not have the capabilities to solve the assigned task or if the human user not familiar with the existing human users, the user agent can assume the role of an advisor. As an advisor, the user agent can help the human user execute the assigned task or help him join the coalition that will yield the highest utility. Over time, that human user becomes familiar with other human users in the system due to his or her participation in the collaborative activities. Furthermore, due to the scaffolding provided by the user agent, the human user also learns how to solve tasks of type ty. At this point, the human user is able to form coalitions and is not in need of any scaffolding from the user agent. That means  $pcn_{i,ty,j,k,t}$  – the value of the potential contribution for a human user for tasks of type ty has become high. This high value then increases the potential reward achieved by the human user. Detecting this high value of potential contribution, a user agent may switch its role from being an advisor to being a representative and save resources (computation, deliberation time, etc.). However, in future, the user agent may detect that the human user is facing a task that he is not capable of executing or needs to form coalition with a set of human users who he or she is not familiar with (e.g., joins a new coalition formation environment). That means the user agent detects a low potential contribution  $(pcn_{i,ty,j,k,t})$  value for that human user. Then, the user agent will again assume the role of an advisor. So, depending on the human user's potential contribution, the user agent's autonomy will change and the user agent will switch its role from advisor and representative.

#### **2.2 Problem Characteristics**

In this section, we identify characteristics of human coalitions and describe design principles that address those characteristics. **Characteristic 1: Diversity**. Human users have different motivations, utility functions, and valuation of rewards.

**Characteristic 2: Inconsistency/Irrationality**. Human users can behave inconsistently and/or irrationally. Also, human users may learn and change their behaviors over time. This underlies the scaffolding component of the *i*HUCOFS framework.

**Characteristic 3: Incomplete Information/Noise**. It is close to impossible to completely model human reasoning and actions as there are always external factors (or noise) influencing how they behave in a coalition.

**Characteristic 4: Uncertain Outcomes**. Even with perfect information and accurate modeling, given the same problem, it is possible that the same coalition may not yield the same outcome.

**Characteristic 5: Characteristic Assumptions 3-5.** Human users can benefit from a well-formed coalition in the first place and good scaffolding after the coalition is formed.

**Characteristic 6: Characteristic Assumptions 3, 7.** A human user can co-exist in a symbiotic relationship with its user agents. A human user can instruct how its user agent should behave and can also rely on its user agent providing timely and useful advice.

#### **2.3 Design Principles**

**Design Principle 1: System and User Perspectives.** There should be a system agent and a set of user agents. A system agent is needed to evaluate and take decisions regarding a coalition, while a user agent is needed to be a representative of and an advisor to its human user. Also, the goal of a system agent and the goal of the user agent can also be different. However, the system agent does not impose any specific rules on the user agent. Instead, it wants the emergent behavior that results from the user agents' own goal: forming a beneficial group for its human user and scaffolding the coalition of its human user to complete the assigned task. This design principle addresses Characteristics 5 and 6.

**Design Principle 2:** User Modeling. The user agents must be able to model different user motivations, behaviors, and utilities and should be able to consider inconsistency or irrationality in their human users' actions or reasoning. This design principle addresses Characteristics 1 and 2.

In brief, there are two ways to model the behavior and performance of a human user. First, information about the human user can be collected from his or her interaction with the user agent, with the other human users and other group member. Since the user agent acts as a communication medium for the human users', they can closely monitor every action of him or her. The group agent can monitor the human user's actions with the other group members. With these three types of information, the entire interaction history of a human user with his or her group members can be constructed.

Second, information about the human users can also be collected from the evaluation scores of the human user in various individual and group activities. While a user model can be constructed by using the raw information about the interaction of human user with others, the evaluation scores collected by administering surveys can be used to crosscheck that model. For example, if the user interaction history indicates that a human user has been an active group member and that user's group members' evaluation of him or her is low, it may mean that the user is doing off topic discussions. Then the system and or group agent may provide him or her guidance and or hint to focus more on the assigned task.

**Design Principle 3:** Satisficing Solution. The system agent and the user agents must be able to take decisions with incomplete information or noise. Further, since outcomes are uncertain, it could be costly for the agents to devise an optimal solution only to find out that it does not lead to the expected outcome. Thus, this motivates the agents to make do with what they know, and sub-optimal but satisficing solutions may be preferable. This design principle addresses Characteristics 3 and 4.

**Design Principle 4: Learning Mechanism.** To overcome the noisy environment and incompleteness of the available information, the user agents should use a learning mechanism to filter out the necessary information to achieve the required level of accuracy. The learning mechanism could include typical *agent learning* (e.g., reinforcement learning) and also the *multiagent learning* where the user agents learn from each other's experience (e.g., learning by discussion and learning by observing). This design principle addresses characteristics 3 and 4.

**Design Principle 5: Scaffolding.** The proposed *i*HU-COFS environment is noisy and has incomplete information and uncertain outcomes. These characteristics imply that the user agents may not be able to collect accurate data to form the most suitable coalition. However, we know that human users may learn and improve their behavior when scaffolding is provided. Therefore, the user agents should spend more time and computational resources for scaffolding. Since the user agents' beliefs about the environment may contain inaccuracies, spending resources for forming the perfect coalition may not yield the best outcome in terms of utilities for the human users. On the other hand, spending more resources for scaffolding would mean that the human users would be able to improve their behavior and in turn improve the outcome for the current and future coalitions.

# 3. Implementation of *i*HUCOFS

With the assumptions, characteristics and design principles in hand, we have designed an iterative coalition formation algorithm called VALCAM where each user agent bids for joining the most compatible coalition with the virtual currency that it has earned from participating in previous coalitions. VALCAM environment consists of a system agent, a set of user agents assigned to the human users and a group agent assigned to each user group. The system agent hosts an iterative auction to form coalitions, where each user agent bids to join the most compatible coalition with the virtual currency that it has earned from participating in previous coalitions. In VALCAM, virtual currency is used as a reward to the user agents for solving the assigned task and for collaborating with the group members. A user agent's reward for solving the task is given by the system agent for executing the assigned task. Furthermore, the reward for collaboration is provided by the system agent as an incentive for the user agents for collaboration (Type II scaffolding). This incentive is provided because, by encouraging collaboration, the system agent encourages the human users to learn and improve their performances over time.

### **3.1 VALCAM Algorithm**

The details of VALCAM can be found in [24]. However, a brief description is as follows: suppose that *A* is the set of *user agents*, *m* is the number of non-overlapping coalitions that will be formed, and |A| > m, *j* is the current task assigned, *p* is the selected auction protocol e.g., Vickrey [20].

#### VALCAM-S (for system agent)

- 1. Initialize (create a set of *m* groups *G* and assign a group agent to each group)
- 2. Choose first members for each group g in G (select *better-performing* users as first members)
- 3. Start the auction according to *p* for users in *A*. For each group *g* in *G*, do,
  - a. Accept bids from the unassigned users
  - b. Assign the highest bidder to g
- 4. After completing *j*, assign individual and group payoffs to *A* based on the human user's individual performance and group performance

### VALCAM-U (for user agent)

- 1. Initialize (estimate and announce the human user's competence for the upcoming task)
- 2. For each round of bidding for group g, bid with an amount proportional to the average of *compatibility* and *performance* of the users in g. Compatibility measures the human users' view of one another, and *performance* measures the average performance of a human user

The performance measure denotes the performance of a human user measured from the perspective of the group agent and the user agent. Each time a user group completes a task, the individual and group performance is evaluated by the student agent and the group agent and a certain amount of virtual currency is assigned to that user. The amount of virtual currency assigned is proportional to the performance of the human user as an individual and as a group member (i.e., helpfulness in achieving the common group goal). Then, using the earned virtual currency, the user agents are able to form groups for the human users. Although this use of virtual currency rewards the user more who has performed well than the user who has not, the design of VALCAM prevents this assignment from becoming a rich- get-richer model by rewarding altruistic behavior during the group formation. Next, we discuss the details of designing VALCAM based on the design principles described in Section 3.

### **3.2 System and User Perspectives in VALCAM**

Based on Design Principle 1, VALCAM has two parts: VALCAM-S for the system agent and VALCAM-U for the user agent. The system agent and the user agents have different goals. The goal of the system agent is to form coalitions that can solve the assigned task at hand and also improve the quality of coalitions that will be formed in the future. On the other hand, the user agent tries to form groups that will improve the human user's learning and group work experience (i.e., to increase the utility that will be earned by the human user in future Eq. (42)). To achieve its goal, the system agent forms coalitions that are heterogeneous with respect to user performances by initiating the human user groups with users who are modeled as competent in solving the assigned task (step 2 of VALCAM-S). In these groups, the betterperforming knowledgeable users are able to help the not-soknowledgeable users solve the assigned task and train the latter to solve similar tasks in future coalitions. On the other hand, the user agent tries to join a group that contains users who are competent and are compatible with its assigned human users (step 2 of VALCAM-U). Such groups may encourage the poor-performing students to learn from the better-performing students. As a result, the performance of the poor-performing users would increase.

# 3.3 User Modeling in VALCAM

Based on Desigu Principle 2, VALCAM relies on the modeling of user competence (i.e., the knowledge base Eq. (6)) and their compatibility (step 2 of VALCAM-U). Accurate modeling of the above two attributes allows the system to better form and scaffold coalitions. Competence defines a human user's capability of solving a particular subtask of a problem. That means a competent user is able to execute the assigned task using his or her knowledge base and behavior base. Modeling the competence of the human users will allow the algorithm to create coalitions with members who are heterogeneous with respect to their performances for solving the task. Mixing high- and low-caliber human users in a coalition can help low-caliber human users learn to improve their performance over time due to Learning by Observation and Learning by being Taught ((Eq. (57) and Eq. (61)).

On the other hand, compatibility refers to the behavior (i.e., an element in the behavior base Eq. (10)) of a human user that allows him or her to use his or her knowledge base to execute the task in a collaborative setting. In terms of compatibility, if the coalition members do not get along with one another, they will work in a team instead of as a team [3]. That means a group of human users who do not get along well or do not like each other's working style, discussion, etc., would work towards achieving their individual goals instead of working with others to achieve the common goal of the group. As a result, the outcome of the coalition would suffer even when the members are highly competent at what they do. Compatibility between two human users denotes their working experience with each other. Furthermore, if past behavior can predict the future, it can be expected that the human users who have worked well with each other in the past, will be able to work well with each other in future. Therefore, by recording the working experience of a human user in a coalition, the user agent will be able to estimate the expected compatibility of this user with the members of a future coalition. Finally, using compatibility in the coalition formation process is an example of implicit scaffolding (Eq. (54)). Putting a human user in his or her favorite group

would mean that he or she will be more involved in the collaborative activities.

# **3.4 Satisficing Solution of VALCAM**

Based on Design Principle 3, we use a soon-enough, goodenough strategy and use an iterative auction to create human user groups. This iterative auction method does not involve any global decision making process to form user groups that yield optimum outcomes. Instead, VALCAM creates an environment that encourages the participating agents to make local decisions that they think are best for their assigned human users. Using those local decisions, VALCAM aims to form human user groups that can solve the assigned tasks and also train the human users to solve the future tasks better.

# 3.5 Learning Mechanism of VALCAM

Table 1 summarizes the learning mechanisms used in VALCAM.

Table 1: Learning Mechanism in VALCAM.

Learning Topic	Mechanism
User Compe- tence	Uses information retrieval with the
	evaluation history of a user to estimate
	the competence of a user on a topic
User Compati-	Uses reinforcement learning and user
bility of a group	modeling to estimate the compatibility
of users	of a set of users for an upcoming task
User Inconsis- tency	Uses the competence, compatibility
	and learning to calculate the expected
	outcome of a user's participation and
	calculates the inconsistency factor by
	finding the difference between the
	expected performance and the actual
	performance

# 3.6 Scaffolding in VALCAM

We have only implemented Type II (implicit) scaffolding in VALCAM. One way for the user agents to achieve Type II scaffolding is by assigning each student to a group where he or she is able to learn from others and improve his or her behavior. The system agent can achieve Type II scaffolding by rewarding the better-performing students to form coalitions with the poor-performing students. That way, the poorperforming students will be able to improve their performances for future tasks. So, to provide Type II scaffolding, VALCAM tries to create the best possible group for each member where he or she is able to engage in various types of learning as described in Section 2.1 (Eq. (57)-(66)). While forming a group that encourages learning, VALCAM's system agent encourages heterogeneity with respect to user performance by rewarding the better-performing students to form groups with poor-performing students. Creating a group with users of mixed performance level is important since not all types of learning occur in groups that are homogeneous in terms of user performance level. For example, a user group that contains only better-performing users would not encourage Learning by Observation (Eq. (57), (58)), Learning by Teaching/Guiding (Eq. (59), (60)), Learning by being Taught (Eq. (61), Eq. (62)), and Learning by Apprenticeship (Eq. (65)). VALCAM's user agents also provide Type II scaffolding by joining a group that contains users who are compatible with each other. Compatibility among the group members is important since not all human users work well with each other [3]. Therefore, to find a suitable group for a human user, VALCAM aims to find a balanced mix of better- and poor-performing human users who are compatible with each other.

# 4. I-MINDS

I-MINDS (Intelligent Multiagent Infrastructure for Distributed Systems in Education) employs a number of interacting intelligent software agents, representing individual students and the instructor, to create a CSCL environment. The rationale behind using multiagent intelligence is the agent's persistence in tracking and monitoring its environment (student and instructor activities), autonomy in decision making, and responsiveness in providing services to both students and instructors. The details of the I-MINDS system can be found in [12] [23] [24]. Briefly, in I-MINDS, each student has a personal agent (a student agent), each instructor has a personal agent (a teacher agent), and when students form a group, they are also assigned a group agent. Figure 1 shows the main components of a typical 1-MINDS classroom.



Fig. 1. I-MINDS classroom structure.

The agents in I-MINDS provide the four important services in a computer-supported learning environment [7]: (1) knowledge construction, (2) context for learning, (3) communication, and (4) collaboration. The teacher agent and the student agent works together to deliver the learning material to the participants for knowledge construction. The teacher agent in I-MINDS also provides the context for learning by structured learning scenarios (e.g., Jigsaw). The student agents in I-MINDS allow the students to communicate with each other using various tools e.g., chat, collaborative whiteboard, etc.

# 4.1 Teacher Agent

In I-MINDS, the teacher agent is designed to help the instructor carry out the CSCL sessions. The teacher agent allows the instructor to interact with students, send slides, manage Q&A sessions, administer quizzes, post evaluations, and form structured collaborative learning groups, and monitor individual and group performances. The teacher agent also allows the instructor to manage Q&A sessions in a large classroom by ranking the incoming questions. Furthermore, the teacher agent also helps the instructor by grouping similar questions together using the *utterance classification* approach like the AutoTutor [8] [15] [19].

The teacher agent also contains the Jigsaw module to carry out structured Jigsaw collaborative work. The Jigsaw procedure works as follows. First, the instructor divides the students into groups. Second, the instructor divides a problem into different parts (or sections). Third, the instructor assigns a part/section for every student such that members of the same group will have different sections to solve. The students who are responsible for the same section then work together to come up with solutions to the section to which they have been assigned and develop a strategy for teaching the solutions to their respective group members. Clarke [4] defined the Jigsaw structure into the following stages:

- 1. **Introduction:** the instructor introduces the topic to the whole classroom. Depending on the type of instruction, this stage may involve Learning by being Taught (Eq. (61), Eq. (62)), Learning by Apprenticeship (Eq (65)), Learning by Reflection/Self-Expression, and Learning by Practice for the students.
- 2. Focused Exploration: The focus groups *explore* issues pertinent to the section that they have been assigned. In this stage, the students usually learn new topics by collaborating with each other. This phase especially encourages Learning by Observation (Eq. (57), (58)), and Learning by Discussion.
- 3. **Reporting and Reshaping**: The students return to their original groups and *instruct* their teammates based on their findings from the focus groups. In this stage, the students actually assume the role of a *teacher* and *teach* their team members what they have learned during the Focused Exploration stage. This stage encourages Learning by Teaching/Guiding (Eq. (59), (60)), Learning by being Taught (Eq. (61), Eq. (62)), and Learning by Apprenticeship (Eq. (65)).
- 4. **Integration and Evaluation**: The team *connects* the various pieces of the solution generated by the individual members, *address* new problems posed by the instructor, or *evaluates* the group product. Due to the students' discussion of the proposed solution, this stage especially encourage Learning by Reflection/Self-Expression (Eq. (63), (64)), and Learning by Discussion.

# 4.2 Student Agent

The student agent serves a unique student in I-MINDS. It interacts with the student and exchanges information with the teacher agent and the group agents. The capabilities of a student agent includes a forum to exchange online and offline messages, a quiz module for testing the students' knowledge, a survey module to collect data from the students, a collaborative whiteboard, and a collaborative flowchart module. The student agent also maintains a dynamic profile of its student user and a dynamic profile of the peers that the student has interacted with through I-MINDS. Furthermore, a student agent is able to form buddy groups designed around the model described in [10] for its student user. Finally, the student agent also allows a student to form structured collaborative groups using the VALCAM algorithm.

### 4.3 Group Agent

In I-MINDS, a group agent is activated when there are *struc-tured* cooperative learning activities. Structured cooperative learning models explicitly specify how group activities are to be carried out in a sequence of steps to solve a joint task. Activities instrumented or tracked in during these steps include the number and type of messages sent among group members for each step, self-reported teamwork capabilities, peer-based evaluations of each team member, and evaluation of each team. Note that a group agent works entirely behind-the-scenes and thus does not have a GUI frontend.

# 4.4 Group Formation Using VALCAM

I-MINDS agents use the VALCAM algorithm (Section 3.6) to form groups for structured collaborative learning. To implement VALCAM, the teacher agent in I-MINDS acts as the system agent in VALCAM, the student agents in I-MINDS act as the user agents in VALCAM and the I-MINDS group agents act as the group agents in VALCAM. Furthermore, the students in I-MINDS classroom become the human users who are forming coalitions using VALCAM and the instructor becomes the person who controls and coordinates the group formation activities using the teacher agent (system agent) interface. In brief, I-MINDS agents use the following steps to implement VALCAM:

- 1. The instructor starts up the I-MINDS teacher agent and loads up a classroom session.
- 2. The students start their I-MINDS student agent clients and join the classroom session.
- 3. Once the students have joined the classroom session, the instructor delivers the instruction on the session topic. During this instruction, the students can ask questions or communicate with the other students through I-MINDS student agent GUI.
- 4. After delivering the instruction, the instructor starts the VALCAM group formation process using I-MINDS teacher agent GUI.
- 5. Once all the groups are formed, teacher agent assigns a task (e.g., solving a problem) to the students, then the teacher agent assigns a group agent to each student group using I-MINDS teacher agent GUI. Finally, the students collaborate to solve the assigned task.
- 6. At the end of the classroom session, the instructor conducts a quiz to evaluate the students' understanding of the assigned task after the collaborative work.
- 7. Finally, the students evaluate the performance of their teams and the performances of their group members by responding to surveys posted in I-MINDS.

Table 2 describes how the *i*HUCOFS design principles are implemented in VALCAM.

Table 2: Summary of the Implementation of the Design Principles of *i*HUCOFS in VALCAM.

Design Principle	Implementation in VALCAM
System and User Pers- pective	Teacher agent's goal is to form groups that would allow all the students learn the sub- ject topic by hosting iterative auction. The student agent's goal is to join a group that holds maximum potential in terms of colla- borative learning for its assigned student, i.e., a group with competent and compatible users.
User Mod- eling	A student agent's model of its assigned stu- dent include: <i>competence</i> (Eq. 6) (ability to solve assigned tasks), and <i>compatibility</i> (i.e., his or her likeness/evaluation of others students) (Eq. 10).
Satisficing Solution	Teacher agent and student agents use an iterative auction algorithm that sacrifices optimality at present (Eq. 48) to improve quality of coalitions in future by improving the behavior of students through learning.
Learning Mechanism	Student agent learns the assigned student's: <i>competence</i> (Eq. 6) from the evaluation score given by the instructor, and <i>compati- bility</i> (Eq. 10) by recording his or her evalu- ation of the other group members for each session to join groups that contain students who are competent and compatible with its assigned student. Since, such a group holds high potential for the assigned student in terms of collaborative learning; the student agent's learning improves the individual performance of the assigned student.
Scaffolding	Teacher agent and student agents provide Type II scaffolding (Eq. 54) by forming coalitions that balance the competence and compatibility of the students in hope that they will learn from each other and improve their behavior over time.

# 4.5 Implementation of I-MINDS

Fig. 2 and Fig. 3 show the current I-MINDS teacher agent and student agent interfaces respectively. For our research prototype and evaluations, the I-MINDS system was implemented in Java (SDK 1.4.2). We have used Java's socket functionalities to establish communication among agents, Java's Swing classes to create interfaces, and Java's JDBC technologies to connect to our MySQL database to store and retrieve all data. For implementing our whiteboard server, we have used the Java Media Framework. Finally, to implement the collaborative flowchart module (JFlowchart) of the student agents, we have used JHotDraw - an open source Java GUI framework for technical and structured graphics. Presently, we continue to develop our research prototype in Java. In parallel, we have also ported most of the I-MINDS features to Microsoft's ConferenceXP platform where the audio/video streaming, networking, archiving, tracking, and communication infrastructures are readily available. This porting has allowed us to deploy our system in wired and wireless environments and with more robust communication modes and data storage.



Fig. 3. I-MINDS student agent GUI.

# 5. Experiments and Results

We have evaluated I-MINDS in classrooms, previously reported in [12] [23] [24]. In this paper, we discuss the feasibility and the impact studies of VALCAM that show the validity of using *i*HUCOFS for human coalition formation.

# **5.1 Experiment Setting**

To evaluate VALCAM in a real world scenario, we deployed I-MINDS in CSCE 155 for two semesters, the first core course of computer science and computer engineering majors (i.e., CS1). The course has three 1-hour weekly lectures and one 2-hour weekly laboratory session. In each lab session, students were given specific lab activities to experiment with Java and practice hands-on to solve programming problems. In our experiment, there were 2-3 lab sections where each section had about 15-25 students. Our study utilized a control-treatment protocol. In the *control* section, students worked in cooperative learning groups without using I-MINDS. Students were allowed to move around in the room to join their groups to carry out face-to-face discussions. In the treatment section, students worked in cooperative learning groups using I-MINDS. Students were told to stay at their computers and were *only* allowed to communicate via I-MINDS. With this setup, we essentially simulated a distance classroom environment. After the group activities, all the students filled out surveys and took a post-test. This post-test score was graded by the instructor and used to measure student performance in terms of understanding the topic.

#### 5.2 Results

## 5.2.1 Feasibility Study 1

In this analysis, our objective was to see whether and how VALCAM provided Type II scaffolding. Fig. 4 and Fig. 5 show the average normalized post test scores for the control and treatment sections for Fall 2005 and Spring 2005.



Fig. 4. Average normalized post-test scores of Spring 2005.

As indicated in Fig. 4 and Fig. 5, the students in the treatment section were able to achieve post test scores that were comparable to that of the students in the control section. We also observe that the average normalized post test scores of the students in the treatment section improved over time for both Fall 2005 and Spring 2005 semesters. This could be an indication that VALCAM, due to its learning mechanism, might have been effective in forming better and better coalitions over time, and *achieving the goal of Type II scaffolding*. However, more semesters of data is needed to obtain enough significance for our observations.

To compare the performances of the students in the control and the treatment group, we have also calculated the slopes of the linear trend lines for the average normalized post-test scores (Fig. 4, Fig. 5) for the Fall 2005 and Spring 2005 experiments. The results show that in Fall 2005 and Spring 2005 experiments, the slopes of the trend lines for the treatment group were higher than those of the control group (Spring 2005: control group's slope=0.021, treatment group's slope=0.029, Fall 2005: control group's slope=0.010, treatment group's slope=0.014). Although the results are not conclusive, they hint that the students in the treatment group were able to improve their performances at a slightly higher rate than the students in the control group. Although the results are not conclusive, they hint that the students in the treatment group were able to improve their performances at a slightly higher rate than the students in the control group.



Fig. 5. Average normalized post-test scores for Fall 2005.

# 5.2.2 Feasibility Study 2

In this study, our objective was to measure how closely the payoff (in terms of virtual currency), a succinct representation of our user modeling, correlated with the actual performance of the students. We used the final lab (all 14 labs) and final exam scores as the actual performance indicators. In the beginning, every student started out with the same virtual currency since the agents assigned to the students had no prior background knowledge about them. Then as they formed coalitions and worked on different tasks, their virtual currency account was updated. As a result, the correlation improved (from ~0.10 to ~0.50 over four lab activities). Thus, as the students worked more with each other in the coalitions, our virtual currency model was able to capture their performance better. This indicates that the VALCAM design using the *i*HUCOFS framework is viable to learn the student models with sufficient accuracy.

### 5.2.3 Feasibility Study 3

In our Spring and Fall 2005 experiments, the main mode of communication for the students was text messages. In this study, our objective was to check whether it is possible for the students to communicate with their group members using the limited text chat capabilities of I-MINDS. Fig. 6 and Fig. 7 show the average count length of messages exchanged during each session for Spring and Fall 2005 sessions.



Fig. 6. Average message count and length in Spring 2005.



Fig. 7. Average message count and length for Fall 2005.

Even though the number of sessions in our experiments is not enough to draw any conclusions, a common trend is observed in both semesters. During both Spring and Fall 2005 semesters, the count of messages decreased and the average length of messages increased. This may indicate that as the students work in coalitions formed by VALCAM in I-MINDS, they sent fewer and lengthier (more explanatory) messages. This indicates that, as the students worked in their groups using I-MINDS, their need to explain things in detail to each other grew. Therefore, tools (e.g., whiteboard) that could aid the students to explain a concept in detail to each other could be helpful in this scenario. However, more data and experiments are needed to validate this claim.

#### 5.2.4 Impact Study 1

In this study, our objective was to measure the impact of VALCAM, through I-MINDS, on student's perception of their own competence, based on the results of the Self-Efficacy Questionnaire (SEQ) survey. The SEQ survey was conducted before the group activities started. Students entered their competency of completing a particular task. This contributes to step 2 of the VALCAM-U algorithm. We observe that for both semesters, students in the treatment section were on average less confident than the students in the control section about their ability to solve the assigned task before the lab activities started (30.98 vs. 33.53 out of 40). This is interesting. As discussed earlier in Feasibility Study 1, the students in the treatment sections performed comparably and eventually overtook those students in the control sections in terms of their post-test scores. This indicates that even though, VALCAM seemed to be able to provide useful Type II scaffolding, it did not improve students' perception of their own competence.

## 5.2.5 Impact Study 2

Similar to the previous study, here we wanted to measure the impact of VALCAM, through I-MINDS, on student's perception of their peers. The Peer Rating Questionnaire (PRQ) surveys were conducted in both control and treatment sections *after* each lab session was completed. The PRQ is designed to rate the helpfulness of the group members after they have gone through the group activities. This constitutes the

compatibility measure in step 2 of VALCAM-U. We find that students in the control section rated their peers better (higher means (35.95 vs. 35.78)) and more consistently (lower standard deviation values (3.54 vs. 6.42)) than the students in the treatment section. This is likely due to the students' discomfort due to heterogeneous groups (students of different calibers and levels of familiarities). On the other hand, we see indications that students in the treatment section seemed to rate their peers better over time (from 33.71 to 35.80 to 36.37 and 37.25). This might be due to the ability of VALCAM in forming more compatible groups over time—trading off between forming and scaffolding, the key to the *i*HUCOFS framework.

#### 5.2.6 Study of User Agent's Utility

The goal of VALCAM is to form and scaffold the human coalitions. However, an individual agent achieves that goal by trying to join a group that would provide the highest yield of virtual currency for the human users. That means, for an individual agent, the virtual currency earned by joining a group is a measure of its utility. Also, meaningful coalition formation and good scaffolding translates to high yield of virtual currency for the individual agents. So, to measure the utility of the whole multiagent system, the average amount of virtual currency accumulated after each day by the individual user agents was calculated. Fig. 8 shows that after each classroom the student agents (i.e., the user agents) were able to increase their virtual currency account balance on average.



Fig. 8. Average Virtual Currency Accumulated.

That means, after every session, the student agents were able to earn more virtual currency than it had spent during the coalition formation session. According to our policy of rewarding virtual currency, this also means that the human users were performing well on average in the groups and were allowing their user agents to accumulate virtual currency.

On the whole, the results of our experiments are not significant enough to claim any conclusion about the effectiveness of VALCAM in forming or scaffolding human coalitions due to insufficient human subjects and short duration of our study. However, our results hint: (1) the students in the control section were more confident about their own efficacy than those in the treatment section (Impact Study 1), (2) the students in the treatment section were able to learn better (higher learning rate and better individual scores) during their collaborative work than the students in the control section (Feasibility Study 1), and (3) the peer rating posted by the students in the treatment section improved over time as opposed to the students in the students in the control section. Although not conclusive, these three observations hint at the fact that VALCAM may have been improving the individual performance of the students in the treatment section and help-ing the users learn how to work as a team better over time.

# 6. Related Work

Here we discuss research work related to collaborative learning systems for human users and research efforts that are focused on forming and scaffolding human coalitions.

Constantino-González [5] proposed a web-based environment called Collaborative Learning Environment for Entity-Relationship Modeling (COLER) in which student can solve Entity-Relationship (ER) problems while working synchronously in small groups at a distance. The research evaluated the feasibility of generating advice based primarily on comparing students' individual and group solutions and tracking student participation (contributions to the group diagram). Their approach monitors individual work in private and shared workspaces to identify conflicts. COLER was designed for sessions in which students first solve problems individually and then join into small groups to develop group solutions. When all of the students have indicated readiness to work in the group, the shared workspace is activated, and they can begin to place components of their solutions in the workspace. COLER's coach is a personal, pedagogical agent that facilitates collaboration by encouraging students to discuss and participate during collaborative problem solving. Given personal and teammates' actions in the learning environment as input, the coach detects learning and participation opportunities, and then gives a message to the student to encourage discussion, participation, self-reflection, ER reviewing, or assign control to a teammate. To monitor participation, COLER detects time-triggered events, such as inactivity in the group area or the coached student having control of the group area for a long time (pencil handling). For our I-MINDS framework, the student agents correspond to COLER's coaches. Currently, each student agent is only capable of monitoring a student's activities and refining the buddy group of the student and reporting the student's profile to the teacher agent, and each student agent is designed to work behind-the-scenes non-intrusively.

Barros and Verdejo [1] defined a process-oriented qualitative description of a mediated group activity on three perspectives: (1) a group performance in reference to other groups, (2) each member in reference to other members of the group, and (3) the group by itself. The collaboration application is conversation-based, and thus the method to compute these attributes automatically is based on semistructured messages. The architecture of their proposed system, Distance Environment for Group ExperiencEs (DEGREE) is organized into four levels: configuration, performance, analysis and organization. At the configuration level, once the teachers have planned an experience at the collaborative level, they configure and install automatically the environment needed to support the activities of groups of students working together. At the performance level, a group of students can carry out collaborative activities with the support of the system. All the events related to each group and experiences are recorded. At the analysis level, the educator or instructor analyzes the user's interaction with tools for quantitative and qualitative analysis and make interventions in order to improve them. At the organization level, the instructor gathers, selects, and stores the results of collaborative learning experiences and the processes. The information is structured and valued for searching and reusing purposes, and stored as cases forming an organizational learning memory. I-MINDS' monitoring and recording of peer-to-peer activities are very similar to DEGREE's. In addition, Barros and Verdejo [1] globally described the activities supported by each of the above levels by means of the Activity Theory. Basically, the DEGREE system uses cases to store the expected collaborative learning experience (outcome), which is configured by the instructors. This experience also includes the decomposition of the task at hand into sub-tasks, to be carried out by the students jointly. DEGREE then provides graphical tools and interface methods for the instructor to monitor and observe the group activities. I-MINDS, though not explicitly following the Activity Theory, is similar to DEGREE is several aspects. I-MINDS has both structured and unstructured cooperative learning features. When the structured cooperative learning mode is invoked, the I-MINDS teacher agent outlines the task, subtasks, and the various activity phases as configured by the instructor. When the students carry out the subtasks going through the various phases, the activities are recorded to be analyzed later. In I-MINDS, the experience and expected outcomes are not stored as cases; instead, group agents are invoked to reward or penalize the students based on several performance metrics that we see as intrinsic to collaborative activities. Further, according to resultant virtual currencies that these students earn, I-MINDS assigns roles to the students in the next round of activities.

Ogata and Yano [18] used knowledge awareness and information filtering in an open-ended collaborative learning environment. Basically, an individual user's agent, called KA-Agent, autonomously informs the learner of the up-tothe-minute activities of other learners by comparing the learner's actions with the other learners' actions. The messages sent by the KA-Agent makes the learner aware of someone who has the same problem or knowledge as the learner, who has a different view about the problem or knowledge, and who has potential to assist solving the problem. The knowledge awareness filtering aims to sift out unacceptable KA messages that disturb learning, and give adequate priority and order KA messages according to individualized The KA-Agent is similar to I-MINDS student priority. agents, especially in the process of selecting buddies suitable for a particular student. The KA-Agent is also similar to I-MINDS teacher agent in the process of forming focus groups during the Jigsaw learning procedure.

Grave et al. [9] is another interesting research where a multiagent framework is used to build a multi-layer architecture that is able to initiate and manage student training. In this article, the authors present a multiagent architecture allowing the implementation of a dynamic CBR for the evaluation for the potential evolution of an observed situation. This architecture is designed on three layers of agents with a pyramidal relation. The bottom layer is used to build a representation of the target case (i.e., the current situation). The second layer is used to implement a dynamic elaboration of the target case and the upper layer implements a dynamic process of source cases. Although this multiagent layered approach can result in a flexible and adaptive learning or training environment, there are a few issues not addressed. The authors discuss that they are analyzing file tracks produced by a tool of self-training to build the ontology of the domain and specify the low layer by identifying the semantic features. If such domain specific approach is used, the resulting multiagent system may not be generic enough to be used in a typical student learning scenario. Therefore, a generic framework could be more helpful. Furthermore, in their layered multiagent framework the issues relating to collaborative work among learners have not been addressed.

On the whole, these collaborative learning systems do not provide any mechanism for forming human user groups that addresses the unique characteristics (Section 2.3) of human coalitions. However, using the *i*HUCOFS framework, I-MINDS tries to address the unique characteristics of human behavior to build meaningful and helpful learner coalitions. Furthermore, these research approaches do not take into account the changes in the human user's behavior that occurs due to learning. However, I-MINDS' user agents try to capture that change in the human user's behavior through modeling and use it to form better groups over time.

There have also been some approaches to form human user groups in the form of 1-to-1 peer groups. Li et al. [14] used agent technology with fuzzy set theory to find matching peers for human users based on similar preferences or expertise. Each agent, representing a user, communicates with others and exchanges information about specific knowledge questions. The responses of these agents are then judged based on response time and the response quality. Then using Zadeh's fuzzy set theory, their framework finds the most suitable set of peers for their users.

Another such peer help system is I-HELP [2]. I-HELP combines a 1-to-1 peer help network and a discussion forum to provide offline peer help to learners. In I-HELP, each human user is assigned a user agent which builds a model for its owner and also builds partial models of all the other user agents (representing other human users) that it comes into contact. This peer help system has some similarities with how I-MINDS' coalition formation module works. For example, in both I-HELP and I-MINDS, the previous user experiences are considered when forming groups. However, in these systems, agents locate peer help for their human users, but a peer group is built based on 1-to-1 experience without taking account how a group would work together as a team. Furthermore, noise, uncertainty and incomplete information in the environment are also not addressed.

The *scaffolding* of human coalitions has been researched in the application domain of the coalition formation after coalitions have been formed. For example, in COLER [6], students work synchronously in small groups at a distance. COLER assigns an agent to coach each learner to support and facilitate collaborative learning. The agent monitors the individual student's activities, detects the differences between the student's and his or her group's solutions, and advise the students on their collaborative skills, e.g., encouraging the students to participate, encouraging them to compare solutions with their other group members. In another research,

Vizcaíno [27] described a virtual student architecture that detected and avoided three situations that decrease the benefits of learning in collaboration: off-topic (off-task) conversations, students with passive behaviors, and problems related to students' learning. I-MINDS has the potential to identify off-topic conversations through its message scoring and grouping, and has the ability to detect and discourage passive behavior through its constant monitoring. Further, the I-MINDS teacher agent groups students into compatible peer groups in order to encourage active participations. An I-MINDS group agent, on the other hand, rewards and penalizes group activities and individual students' participation, taking into account how a group has performed and how the students perceived each other's contribution to the teamwork.

These research approaches for realizing scaffolding use only short term approaches (solving the task at hand) for scaffolding human coalitions. However, our notion of scaffolding in I-MINDS includes both short term (solving the task at hand) and long term improvement (improved performance due to learning) of user behavior.

# 7. Conclusion

We have introduced *i*HUCOFS – a framework for forming and scaffolding human coalitions. We have also described VALCAM – a preliminary implementation of the *i*HUCOFS framework for forming and scaffolding learner coalitions in I-MINDS-a CSCL environment. Finally, have we discussed the feasibility and the impact studies to demonstrate the validity of using *i*HUCOFS as a framework for forming and scaffolding human coalitions. Preliminary results hint that by using *i*HUCOFS framework, I-MINDS was able to form and impact the learner coalitions in the CSCL environment.

Future work includes continued development of the *i*HU-COFS framework to make it more precise and comprehensive. We are also working to improve the VALCAM algorithm by developing better modeling and tracking capabilities and by incorporating Type I scaffolding. We are also improving the system agent's reasoning capability in VALCAM so that it is able to take into account the various costs of forming and scaffolding the coalition and is able to choose the optimal seed selection policy while forming coalitions. Furthermore, we are also working to make the user agent's role (advisor or representative) dynamic. Finally, we are also improving the functionalities in I-MINDS (tracking, GUI) to perform longer experiments using VALCAM.

# Acknowledgement

This research was partially funded by the National Center for Information Technology in Education (NCITE) and the National Science Foundation (NSF SBIR# DMI-0441249). We would also like to thank Xuli Liu for his help in running the experiments, and Hong Jiang for the design of I-MINDS.

# References

- [1] B Barros and M F Verdejo, Analysing student interaction processes in order to improve collaboration. The DEGREE approach. International Journal of Artificial Intelligence in Education, 2000, Vol 11, pp. 221-241.
- [2] S Bull, J Greer, G McCella, L Kettel, and J Bows, User modeling in I-HELP: What, Why, When and How. Proceedings of ICUM, 2001, pp. 117 – 126.
- [3] C Chalmers and R Nason, Group Metacognition in a Computer-Supported Collaborative Learning Environment. Proceedings of ICCE, 2005, pp. 35-41.
- [4] J Clarke, Pieces of the Puzzle: The Jigsaw Method. In S. Sharan (ed.) Handbook of Cooperative Learning Methods, Greenwood Press, Westport, CT, 1994.
- [5] A Collins, J Brown, and A Holum. Cognitive apprenticeship: Making thinking visible. American Educator, 1991, Vol 6(11), pp. 38-46.
- [6] M A Constantino-González and D D Suthers, Coaching Collaboration by Comparing Solutions and Tracking Participating. Proceeding of EURO-CSCL, 2001, pp. 704-705.
- [7] D H Jonassen, Thinking Technology: Toward a constructivist design model. Educational Technology, 1994, Vol 34(3), pp. 34-37.
- [8] A C Graesser, K Wiemer-Hastings, P Wiemer-Hastings, R Kreuz, and Tutoring Research Group, Autotutor: a simulation of a human tutor. Journal of Cognitive Systems Research, 1999, vol 1, pp. 35-51.
- [9] P Grave, H Boukachour, and M Ennaji, From three multiagent layers to one teaching agent. In 25th European Annual Conference on Human Decision-Making and Manual Control, 2006, http://www.univvalenciennes.fr/congres/EAM06/PDF\_Papers\_author/Ses sion1\_Grave.pdf.
- [10] U Hoppe, Use of multiple student modeling to parametrize group learning. Proceedings of International Conference on Artificial Intelligence in Education, 1995, pp. 234-249.
- [11] A Inaba, T Supnithi, M Ikeda, R Mizoguchi, and J Toyoda, How can we form effective collaborative learning groups? Proceedings of ITS, 2000, pp. 282-291.
- [12]N Khandaker, L-K Soh, and H Jiang, Student learning and team formation in a structured CSCL environment. Proceedings of ICCE, 2006, pp. 185-192.
- [13]X Li and L-K Soh, Learning-based multi-phase coalition formation. Proceedings of AAMAS-Workshop on Coalition and Teams: Formation and Activity, 2004, pp. 9-16.
- [14]X Li, A R Montazemin, and Y Yuan, Agent based buddy-finding methodology for knowledge sharing. Information and Management, Vol. 43(3), 2006, pp. 283-296.
- [15]X Liu, X Zhang, J Al-Jaroodi, P Vemuri, H Jiang, and L-K Soh, I-MINDS: An Application of Multiagent System Intelligence to On-Line Education. Proceedings of IEEE-SMC, 2003, pp. 4864-4871.
- [16]X Liu, X Zhang, L-K Soh, J Al-Jaroodi, and H Jiang, A distributed, multiagent infrastructure for real-time, virtual classrooms. Proceedings of ICCE, 2003, pp. 640-647.
- [17]S Namala, An intelligent module for I-MINDS, M.S. Project Report, Computer Science and Engineering, University of Nebraska, Lincoln, NE.

- [18]H Ogata and Y Yano, Combining knowledge awareness and information filtering in an open-ended collaborative learning environment. International Journal of Artificial Intelligence in Education, Vol. 11, 2000, pp. 33-46.
- [19] A Olney, M Louwerse, E Mathews, J Marineau, H Hite-Mitchell, and A Graesser, Utterance classification in Autotutor. Proceedings of HLT-NAACL Workshop, 1995, pp. 1-8.
- [20]T Sandholm. Auctions, in Weiss, G. (Eds.) Multiagent Systems: A Modern approach to Distributed Artificial Intelligence. MIT Press, 2000, pp. 211-219.
- [21]D A Schön, D. A. Educating the reflective practitioner. 1987. San Francisco: Jossey-Bass Publishers.
- [22]S Sekine, and R Grisman, A corpus-based probabilistic grammar with only two non-terminals. Proceedings of 4<sup>th</sup> International Workshop on Parsing Technologies, 1995, pp. 216-223.
- [23]L-K Soh and N Khandaker, Forming and scaffolding human coalitions with a multiagent framework. Proceedings of AAMAS, 2007, pp. 394-396.
- [24]L-K Soh, N Khandaker, and H Jiang, Multiagent Coalition Formation for Computer-supported Cooperative Learning. Proceedings of IAAI, 2006a, pp. 1844-1851.
- [25]L-K Soh, N Khandaker, X Liu, and H Jiang, Computer-Supported Cooperative Learning System with Multiagent Intelligence. Proceedings of AAMAS, 2006b, pp. 1556-1563.
- [26]L-K Soh, N Khandaker, X Liu, and H Jiang, Computer-Supported structured cooperative learning. Proceedings of ICCE, 2005, pp. 428-435.
- [27]A Vizcaino and B Du Boulay, Using a Simulated Student to Repair Difficulties in Collaborative Learning. Proceedings of ICCE, 2002, pp. 349-353.
- [28]L Vygotsky. 1978. Mind in society: The development of higher psychological processes. Cambridge MA, Harvard University Press.

# **Author Bios**

**Nobel Khandaker** received his B.S. with Honors in Physics from the University of Dhaka, Bangladesh. He then completed his M.S. in Computer Science from the University of Nebraska Lincoln. He is now a Doctoral Candidate at the Department of Computer Science and Engineering at the University of Nebraska Lincoln. His primary research interests include teamwork and coalition formation for human participants, multiagent coalition formation, multiagent learning, computer-supported collaborative learning systems, and agent-based simulation.

Leen-Kiat Soh received his B.S. with Highest Distinction, M.S., and Ph.D. with Honors in Electrical Engineering from of University He the Kansas. is now an Associate Professor at the Department of Computer Science and Engineering at the University of Nebraska. His primary research interests are in multiagent systems and intelligent agents, especially in coalition formation and multiagent learning. He has applied his research to computer-aided education, intelligent decision support, and distributed GIS.