

Intelligent Group Formation in Computer Supported Collaborative Learning Scripts

Ishari Amarasinghe, Davinia Hernández-Leo, Anders Jonsson
Universitat Pompeu Fabra, Barcelona, Spain
{ishari.amarasinghe, davinia.hernandez, anders.jonsson}@upf.edu

Abstract—Well-structured collaborative learning groups scripted based on Collaborative Learning Flow Patterns (CLFPs) often result in successful collaborative learning outcomes. Formulation of such learner groups based on instructor defined criteria promises potentially effective performance of participating students. However, forming student groups manually based on multiple criteria often fails due to its complexity and the time limitations of practitioners. Hence, an intelligent assistance which supports adaptive collaboration scripting based on instructor defined criteria, while adhering to CLFPs is presented. Constraint Optimization techniques have been used for learner group formation and preliminary tests revealed that the proposed approach could be utilized when formulating student groups while satisfying team formation criteria.

Keywords—Computer Supported Collaborative Learning (CSCL); Collaborative Learning Flow Patterns (CLFPs); Jigsaw; Binary Integer Programming; Constraint Optimization

I. INTRODUCTION

Collaborative learning is a pedagogical approach in which learners collaborate amongst peers towards achieving learning goals while constructing shared knowledge and understanding. In such contexts, collaboration could occur between a pair of students or within a larger group. Collaborative learning triggers significant individual cognitive processes which may often result in “socio-cognitive” conflicts among individual learners [1]. Resolving such conflicts via discussions with peers, cause individuals to achieve improved competence levels and knowledge gains. However, it is difficult to ensure that learning via interactions may occur in any situation [2]. Realization of success in collaboration settings often require adequate scaffolds [3].

When considering scaffolding strategies in the context of collaborative learning, “scripts” plays a significant role [1]. Effective interactions among learners could be fostered by adapting to Collaborative Learning FLOW Patterns (CLFPs) which are derived from broadly accepted practice rather than from general learning theories [4]. This paper presents, a novel binary integer programming approach towards group formation in CSCL environments based on Jigsaw CLFP. Flexibility towards grouping based on instructor defined criteria is facilitated and the proposed approach was tested using real world datasets.

II. LEARNER GROUP FORMATION CRITERIA

During the work presented in this paper, Jigsaw CLFP was adhered when formulating learner groups. It consists of

TABLE I
INTRINSIC CONSTRAINTS APPLIED IN JIGSAW CLFP

Phase	Intrinsic or Hard Constraints
Phase 01	Each student is allocated to study one sub problem
	Each task is allocated to a minimum number of students
Phase 02	A student can work only in one Jigsaw group
	There should be at least one student for each task within the Jigsaw group
	Each Jigsaw group should have a minimum number of students

three major phases known as task allocation, expert group formation and Jigsaw group formation. During task allocation each individual student is assigned to study a particular task, while in expert phase students who studied the same task work collaboratively. Finally, students who have studied different tasks are grouped together forming Jigsaw groups [4]. In the work presented in this paper, suggestions for task allocation and expert group formation are computed simultaneously and presented in Phase 01 while Phase 02 depicts Jigsaw group allocations. Further, CLFPs inherit a set of conditions commonly known as constraints which shape up the desired collaboration [5]. Intrinsic constraints are mandatory to be satisfied (see Table I) while extrinsic constraints are induced by contextual factors or arbitrary decisions [3].

III. PROPOSED APPROACH

When considering the nature of the problem being addressed where grouping is done adhering to different constraints, it was seen that constraint optimization could be adapted when formulating learner groups based on Jigsaw CLFP. A scenario with intrinsic constraints mentioned in Table I can be modeled as a constraint optimization problem, using the following mathematical notations.

A. Problem Formulation

Given a total set of T tasks, N students the problem is to assign tasks for each pair of students with the goal of minimizing the cost incurred during task assignment. The Phase 01 of the problem can be modeled as follows:

$$\text{Minimize } \sum_{i=1}^N \sum_{j=1}^N \sum_{k=1}^T X_{ik} X_{jk} C_{ij} \quad (1)$$

subject to

$$\sum_{k=1}^T X_{ik} = 1 \quad \forall i \in \{1, \dots, N\} \quad (2)$$

$$\sum_{i=1}^N X_{ik} \geq G \quad \forall k \in \{1, \dots, T\} \quad (3)$$

where X_{ik} denotes assigning student i to task k , X_{jk} denotes assigning student j to task k . For each pair of students i and j , the cost C_{ij} is included as a term in the objective function precisely when i and j are assigned to the same task k . Cost C_{ij} could take on any value larger than or equal to 0 depending on the extrinsic constraints applied in each learning scenario. Further constraint (2) ensures that each individual i can be assigned to only one task k . Constraint (3) guarantees that each task k is assigned to a minimum number of students G based on the practitioners' input on task allocation.

Similarly in Phase 02, we can formulate M number of total Jigsaw groups with the goal of minimizing the cost incurred when assigning students to groups. However, during this phase an additional constraint (4) has been added to the model to make sure that at least one student from each task (from phase 01) is presented in each Jigsaw group.

$$\sum_{i=1}^N B_{im} X_{ik} \geq 1 \quad \forall k \in \{1, \dots, T\}, \forall m \in \{1, \dots, M\} \quad (4)$$

B_{im} denotes assigning student i to Jigsaw group m , X_{ik} denotes the previous task assignment (during phase 01) of student i to task k .

During problem modeling, extrinsic constraints applied during a particular phase have been incorporated into the objective function parameters. Hence if, and based on the extent that, the conditions on the variables are not satisfied (intrinsic constraints) extrinsic constraints which have some variable values in the objective function would be penalized. The following example demonstrates how objective function parameters could be encoded depending on the requirement of formulating homogeneous and heterogeneous student groups.

Example: To the extent possible participants who are allocated to the same task during phase 01 are required to have similar knowledge levels and they should belong to different gender categories.

In this scenario extrinsic constraints are related to both homogeneity and heterogeneity of student data since, *similar knowledge levels* and *different gender categories* are considered. Based on the extrinsic constraints specified, cost term C_{ij} associated with a pair of students i and j could be defined as follows:

$C_{ij} = 0$, if both students have similar knowledge levels and if they belong to different genders

$C_{ij} = 2$, if both students have different knowledge levels and if they belong to similar genders

$C_{ij} = 1$, otherwise (*i.e.* i and j differ in one parameter but not the other)

TABLE II
PHASE 01 RESULTS

Task 1	Task 2	Task 3	Task 4	Task 5
1	4	7	3	6
2	5	14	8	10
11	15	16	9	12
13	22	18	17	21
		19	20	

TABLE III
PHASE 02 RESULTS

Group 1	Group 2	Group 3	Group 4
2	1	3	4
5	7	10	6
8	12	13	11
9	14	18	17
16	15	22	19
21	20		

TABLE IV
PHASE 01 RESULTS

Task 1		Task 2		Task 3		Task 4	
LID	GD	LID	GD	LID	GD	LID	GD
8	1	3	2	1	2	6	2
9	2	4	1	2	1	11	2
10	1	5	1	7	1	13	1
12	1	16	1	17	2	14	1
19	1			18	1	15	1

IV. TESTING AND EVALUATION

The algorithm was deployed on a personal computer with Intel(R) Core(TM) i5-2430M CPU@ 2.40GHz X 4 having 4GB of RAM. Implementation was done using Python and SQLite database. Gurobi Optimizer [6] version 6.5 has been used to solve different problem instances using real world student data obtained from authors of [7].

A. Group Formation Design Analysis

Algorithm was evaluated in terms of intrinsic and extrinsic constraint satisfaction. Table II and Table III shows grouping results for 22 students (each number represents a student). Execution of the algorithm finished within a few seconds (0.005 sec. during Phase 01 and 0.010 sec. in Phase 02), providing optimal results for the given problem instance.

Table IV and Table V shows results of a sample scenario which considers extrinsic constraints. Heterogeneity of gender details (GD) was considered during phase 01 while homogeneity of language preferences for collaboration (LP) was considered in phase 02. Execution of the algorithm finished within a few seconds (1.222 sec. during Phase 01 and 0.275 sec. in Phase 02) providing optimal allocations.

It should be noted that based on the way that we have modelled the problem, the cost parameter which represents associated extrinsic constraints in a given problem instance is completely general, meaning that instructors could incorporate extrinsic constraints to the model depending on the learning context without any hard limits.

TABLE V
PHASE 02 RESULTS

Group 1		Group 2		Group 3		Group 4	
LID	LP	LID	LP	LID	LP	LID	LP
4	1	5	1	1	1	3	1
6	2	10	1	2	1	13	1
7	1	11	1	9	1	17	1
8	1	12	1	14	1	19	1
15	2	18	1	16	1		

B. Scalability and Performance Analysis

Based on a number of tests conducted, it was noted that the algorithm scale well, when extrinsic constraints were excluded. For instance, it took only 0.104 sec. during Phase 01 to allocate 5 tasks to 160 students (each task was allocated to 32 students) and 0.025 sec. during Phase 02 to allocate 160 students to 10 groups. However, obtaining optimal grouping results became harder with an increased number of learners and extrinsic constraints. Based on test results, it was concluded that when the problem is more constrained (i.e. three extrinsic constraints per phase, more learners) the algorithm takes more time to finish execution.

V. RELATED WORK

Different algorithms, frameworks, tools and techniques have been developed over time to address the learner group formation problem. However, most of the existing approaches model a fixed set of parameters [8] or are only able to handle a minor number of learner attributes when forming groups [9]. On the other hand work done by [10], [11] have adapted similar techniques to formulate learner groups. However, the problem modeling approach they have presented is different from our work and they have not adapted to CLFPs which result in formulating complex grouping structures. Nevertheless, many authors have evaluated the scalability of the suggested approaches considering fewer grouping parameters [12], [10], [13] while many have not provided test results although they argue that the suggested approaches scale well [14], [8].

VI. CONCLUSION

During the work presented in this paper a novel binary integer programming approach for group formation based on Jigsaw CLFP was proposed. The suggested approach could handle different grouping constraints defined with regard to a particular learning scenario hence it addresses the multiple criteria grouping problem. Cost function parameters could be effectively used when formulating groups incorporating a number of extrinsic constraints, without restricting grouping criteria using hard limits. Based on test results it was noticed that the algorithm formulates learner groups providing optimal grouping results within seconds based on intrinsic constraint specified. Further, it was determined obtaining near-optimal results via approximations (running algorithm as an any-time solution) would be advantageous in complex scenarios (i.e., different extrinsic constraints applied for grouping) due to limited computation time allowed in classroom scenarios.

We have already modeled and conducted several tests on regrouping learners, which would support educators when adapting to constantly changing learning environments. However it was determined that further work is needed to achieve learner regrouping on the fly. As for future work, it is of importance to investigate on heuristics which could optimally solve complex problem instances. Moreover, implementation of a group formation service which provides grouping and regrouping recommendations offered by the algorithm would facilitate its application and adaption in real practise.

ACKNOWLEDGMENT

This research is funded by Spanish Ministry of Economy and Competitiveness (TIN2014-53199-C3-3-R, MDM-2015-0502) and RecerCaixa (COT Project). Prof. H. Spoelstra from Open University of the Netherlands, Dr. C. Burt from The University of Melbourne and Prof. H. Ramalhinho from Universitat Pompeu Fabra are gratefully acknowledged.

REFERENCES

- [1] F. Fischer, I. Kollar, H. Mandl, and J. M. Haake, *Scripting computer-supported collaborative learning: Cognitive, computational and educational perspectives*. Springer Science & Business Media, 2007, vol. 6.
- [2] W. M. Cruz and S. Isotani, "Group formation algorithms in collaborative learning contexts: A systematic mapping of the literature," in *Lecture Notes in Computer Science Collaboration and Technology*, 2014, vol. 8658, pp. 199–214.
- [3] P. Dillenbourg and P. Tchounikine, "Flexibility in macro-scripts for computer-supported collaborative learning," *Journal of computer assisted learning*, vol. 23, no. 1, pp. 1–13, 2007.
- [4] D. Hernández-Leo, J. I. Asensio-Pérez, and Y. Dimitriadis, "Computational representation of collaborative learning flow patterns using ims learning design," *Educational Technology & Society*, vol. 8, no. 4, pp. 75–89, 2005.
- [5] K. Manathunga and D. Hernández-Leo, "A multiple constraints framework for collaborative learning flow orchestration," in *International Conference on Web-Based Learning*. Springer, 2016, pp. 225–235.
- [6] I. Gurobi Optimization, "Gurobi optimizer reference manual," 2016. [Online]. Available: <http://www.gurobi.com>
- [7] H. Spoelstra, P. Van Rosmalen, T. Houtmans, and P. Sloep, "Team formation instruments to enhance learner interactions in open learning environments," *Computers in Human Behavior*, vol. 45, pp. 11–20, 2015.
- [8] D.-Y. Wang, S. S. Lin, and C.-T. Sun, "Diana: A computer-supported heterogeneous grouping system for teachers to conduct successful small learning groups," *Computers in Human Behavior*, vol. 23, no. 4, pp. 1997–2010, 2007.
- [9] A. Ounnas, H. Davis, and D. Millard, "A framework for semantic group formation," in *Eighth IEEE International Conference on Advanced Learning Technologies*, 2008, pp. 34–38.
- [10] A. A. Kardan and H. Sadeghi, "Modeling the learner group formation problem in computer-supported collaborative learning using mathematical programming," in *The 8th National and the 5th International Conference on e-Learning and e-Teaching (ICeLeT 2014)*, 2014, pp. 1–5.
- [11] J. M. Balmaceda, S. N. Schiaffino, and J. A. D. Pace, "Using constraint satisfaction to aid group formation in csl," *Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial*, vol. 17, no. 53, pp. 35–45, 2014.
- [12] C. E. Christodoulopoulos and K. A. Papanikolaou, "A group formation tool in an e-learning context," in *19th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2007)*, 2007, vol. 2, pp. 117–123.
- [13] L.-K. Soh, N. Khandaker, and H. Jiang, "Multiagent coalition formation for computer-supported cooperative learning," 2006, vol. 2, pp. 1844–1851.
- [14] R. Cavanaugh, M. Ellis, R. Layton, and M. Ardis, "Automating the process of assigning students to cooperative-learning teams," in *Proceedings of the 2004 ASEE Annual Conference*, 2004.