Recomposing Small Learning Groups at Scale—A Data-driven Approach and a Simulation Experiment

Zhilin Zheng[1], Niels Pinkwart[2]

Abstract: Group re-composition has thus far been rarely studied. The recent emergence of large scale online learning contexts (e.g. MOOCs) might bring about an opportunity for its application due to the reported high drop-out rate. In this paper, we propose a novel data-driven approach to address the problem of group re-composition. Through a simulation experiment, we saw its capability in decreasing the drop-out rate in groups and bringing more cohesive groups when compared against a random grouping strategy.

Keywords: Group Formation; Group Re-composition; Group Dynamics; MOOC; Learning Analytics.

1 Introduction

Back in the 1950s, teachers and researchers began to experiment with within-class grouping, between-class grouping and even cross-grade grouping [Ga86, Sl87, Ku92]. They often composed small learning groups according to students' achievement, attainments and aptitudes. More recently, other grouping features such as learning styles, demographic characteristics or behavioural attributes have been tested as well. Although much research attempts to find the generalizable theories that can make efficient groups in many cases, a consensus has not yet been reached on which type of groups are more likely to yield great learning outcome. In recent years, large-scale learning contexts, typically Massive Open Online Courses (MOOCs), bring about opportunities to study this problem in a new direction. Data-driven methods have begun to be employed [We16]. Basically, the data-driven methods create a data model that can predict grouping outcomes based on analysis of students' relevant data. This data model can then be used to make groups that could be more potentially successful. These methods obviously do not reapply the generalised grouping theories anymore. But the higher requirement of data size challenges their practical implementation. Obviously, too few data can hardly train a relatively unbiased data model. Nevertheless, massive enrolments in recent MOOCs, in principle, secure the quantity of data. Another typical feature in recent MOOCs, the reported high dropout rates [Jo14], could however make many small learning groups dysfunctional, because very scarce human resource would be left behind in those groups. Especially when we set up group learning in a multi-task context, recomposing those dysfunctional groups for the next group tasks would be necessary.

[1] Humboldt University of Berlin, Department of Computer Science, Berlin, zhilin.zheng@hu-berlin.de
[2] Humboldt University of Berlin, Department of Computer Science, Berlin, niels.pinkwart@hu-berlin.de

Thus, in the paper, we propose a data-driven approach to address the group re-composition problem. The remainder of this paper is organized as follows. After a review of recent studies on relevant topics, we propose our approach and report on a test using a simulation platform. Next, we interpret the experimental results and finally conclude our findings.

2 State of the art

As digitalized learning or e-learning technologies advanced, group composition has been studied on the basis of computer-supported methods. Among such, Moreno et al. [MOV12] suggested a genetic algorithm. Graf et al. [GB06] proposed an ant colony optimization method. Hsu et al. [Hs14] recommended an artificial bee algorithm and Zheng et al. employed a discrete particle swarm optimization approach [ZP14]. The grouping attributes employed vary and include Belbin roles [YA12], learning performance [GB06] and background knowledge [Hs14]. With regard to the grouping results, most of the approaches produce either homogeneous groups or heterogeneous groups or a mixture of both in some cases [KBS14]. All approaches rely on static data collected before a group task starts and none of those methods accounts for data alterations caused by group dynamics. For example, individual's roles or learning performance could change over time.

Group re-composition sometimes is unavoidable during group operation for several reasons. First, teachers may find dysfunction in groups. Second, students themselves want to dissolve their groups [SA10, p.55]. In classrooms, manual redistribution of those students would probably not be a bad choice. However, in large-scale online learning contexts, it would not be affordable anymore. To the best of our knowledge, only one recent publication attempts to address this group re-composition problem [SB14]. This paper proposes a dynamic group formation method to improve Computer Supported Collaborative Learning (CSCL) groups. The authors dynamically retrieved group interaction data and iteratively made use of it to compose groups for each new task. This work pioneers the use of group dynamics to compose groups. Yet, it still leaves much room for improvement. First, the authors merely observed the dynamic change of each student and reapplied a grouping method (i.e. Group Technology) to recompose learning groups over and over again, that is, the subtle association between group dynamics and group success has not been taken into account. Second, the method cannot offer us the concrete rules to compose either successful or weak groups. Instead, they repeatedly used the same grouping criteria (which may be questionable for any specific group). Third, it is necessary to scale up the method for recent larger-scale online learning platforms.

3 Methods

3.1 Group re-composition approach

An overview of our proposed algorithmic re-grouping method is depicted in Fig. 1. First of all, we can start with an initial group formation. Random groups can be employed here unless useful participant data is available. The goal is to improve this initial group formation in the following task(s).

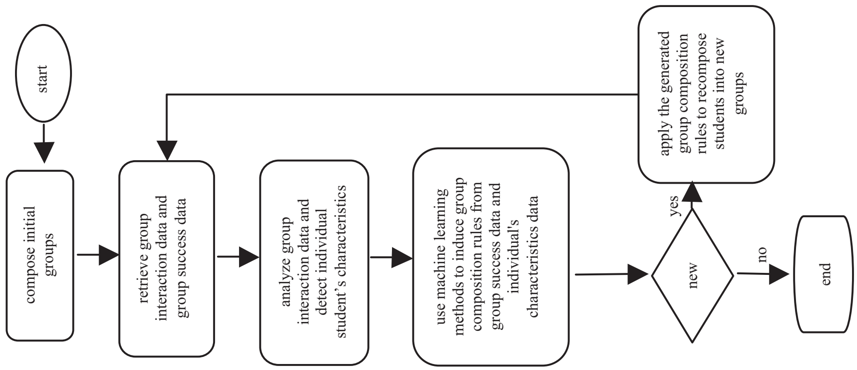

Fig. 1: Schema of the proposed group re-composition approach.

Next, we retrieve group data from the first group task. This data mainly comes from two sources: group interactions and group success (e.g. learning performance). Based on group interaction data, we can further detect individual student's roles in his/her group using Social Network Analysis (SNA). Merging with group success data, we can then apply machine leaning methods to induce group composition rules that indicate which group roles (combined together) make successful groups or weak groups. Those generated composition rules are employed to suggest new groups for the next task if necessary. Through this iterated process, we can learn group composition knowledge from the data and apply it to recompose groups task by task.

The proposed approach has at least five advantages. First, it does not rely on potentially overly general grouping theories (that frequently do not apply to the situation of concrete groups). It is totally data-driven and directly reflects the truth encoded in the data. Second, it also works with students of whom no initial information is known. Third, it accounts for group dynamics. For example, if somebody's role changes over time, the data would accordingly reflect such. Fourth, it makes a dynamic connection between grouping attributes and group outcome. It could easily be extended to grouping attributes other than roles. Fifth, this approach is able to generate concrete group composition rules that can be applied to suggest group assignment in some other similar learning contexts.

3.2 Simulation system

Due to the nature of a data-driven approach, data source certainly plays an important role in evaluating the proposed approach. As shown in Fig. 1, group interaction data and group success data are two main data sources. Empirical methods such as field studies would be a straightforward way to collect such data. However, a suitable MOOC test course has not yet been found, perhaps, due to the requirement of designing multiple group tasks in a course. Instead, computer simulation was chosen to validate the proposed approach. Thanks to Nygren's studies [Ny10, Ny11, Ny12], modelling of group discussion is computationally possible. We thus chose it to generate the group interaction data in this paper. Via simulation, the group success data, such as learning performance, is not realistic to be collected either. We thus had to choose group cohesion that can be calculated from each simulating group as an alternative.

Our simulation system as a whole functionally consists of group composer, group interaction simulator, roles detector, machine learning and regrouping components. Fig. 2 gives an overview on how these five main components work together.

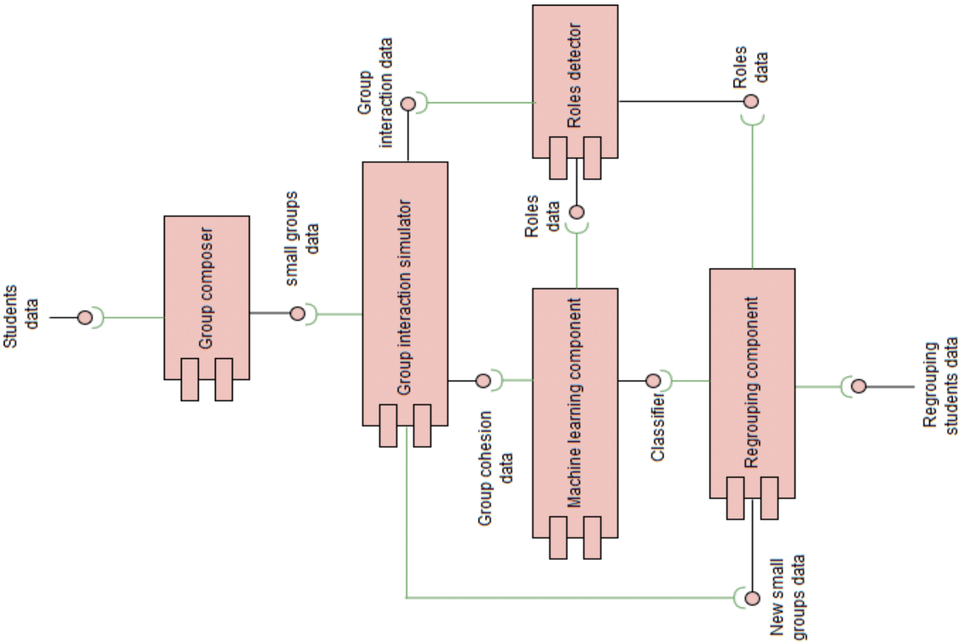

Fig. 2: Simulation system diagram

The Group composer implements the function of composing initial learning groups. It takes a set of students and group size as input. The output of this component is small learning groups of the given size. The initial groups can be composed at random. Note that in case of uneven split over all resulting groups (i.e. the last group could not have the size of the given number), the last composed group either stay alone or merge into the last second group, depending on the actual size of the group and the assigned group size.

The Group interaction simulator generates the interaction data that is crucial for the next steps of the whole simulation. The heart of this component stems from Nygren's work on simulation of user participation and interaction in online discussion groups [Ny12]. Aiming at showcasing the social structure of group members, as is intuitively known to all, such an interaction simulator principally has to answer the following questions. 1) Who will start a discussion post at the next moment? 2) Will this post mention other participants (replies, comments and mentions of any sort)? 3) If so, who exactly will be mentioned? Regarding the first question, everyone has an opportunity to make a new post. The probability to make a new post is proportional to one's accumulative number of posts that have been already made. To answer the second question, Nygren used a probability parameter which observed from an empirical study to decide on the attachment of a groom (a groom could be comments and mentions of any sort). To answer the third question, they resorted to two strategies. First, the more often the participants are groomed, the more likely they have a chance to receive a groom again during the following interaction. Second, the more Groom-balance is, the more probability one has. Groom-balance is the difference between the number of grooms a participant has given away and the number of grooms he/she has received. Fig. 3 depicts the inner mechanism of the group interaction simulator.

```
paramters:
  - P_nys //Probability to select a not-yet-spoken member as a new speaker
  - P_groom_sum //probability to select a speaker according to the sum of grooms received
already
  - P_groomed //probability to select a speaker out of groomed members:
  - g_rate //the grooming rate
  - P_groom_balance // probability to select the groomed person according to groom balance
  - num_posts // the total number of posts

for (i=0; i<num_posts; i++):
  //select a speaker (post maker) out of group members
  if rand() > P_nys:
    Select a not-yet-spoken member
  else: //select a speaker out of the yet-spoken members
    if rand() > P_groom_sum:
      Select the speaker with the max amount of grooms received
    else: //select a speak according to their grooming status
      if rand() > P_groomed:
          Randomly select one from the groomed group members
      else:
          select one from the ungroomed members
  // Decide if a post contains a groom
  if rand() > g_rate:
    Attach a groom to the post
    //Decide whom is groomed if the post contains a groom
    if rand() > P_groom_balance:
      Groom the member owning the max amount of groom balance
    else:
      Groom the member made the max amount of posts
```

Fig. 3: Group interaction simulator

The Roles detector detects individuals' roles in each group. Roles identification has been studied for years. Social Network Analysis (SNA) has been recently used to address this role identification problem [Su10, MMD15]. A critical question has to be answered when one applies SNA to detect group roles is how to map SN's metrics to specific roles. A wealth of studies has addressed this issue. Marcos et al. successfully identified isolated students, student-coordinators and teacher dependent students by observing SN's degrees, closeness and betweenness centrality [Ma08]. They recently extended it to detect more roles (e.g. teacher-facilitator) via applying their SNA tool, Role-AdaptIA [MMD15]. Rabbany et al. visualized the leaders and peripheral students using their Meerkat-ED tool [RTZ11]. Brokers as an important role in social networks were also studied by Stuetzer et al. using SNA and its SN characteristics have so far been uncovered [St13]. Referring to the aforementioned studies, the present work builds on six roles in total: *leader*, *disseminator*, *responder*, *broker*, *lurker* and *peripheral.* The mapping criteria can be seen in Tab. 1.

Roles	Degree	In-degree	Out-degree	Closeness-centrality	Betweenness-centrality
leader	H	M/H	M/H	H	
disseminator	M/H	M/H		M/H	
responder	M/H		M/H	M/H	
broker					M/H
lurker	N	N	N	N	N
peripheral	L	L/N	L/N	L	L/N

Note: H—High (bound: 0.7-1); M—Medium (bound: 0.2-0.7); L—Low (bound: 0-0.2); N—Null

Tab. 1: Criteria to Map SN Metrics into Group Roles

The Machine learning component trains a classifier that is able to predict the group outcome of any new form of group. Specifically, it sums up the number of students for each role in each group and regards group success as the target (in this paper, if the group cohesion of a group is larger than the median, it would be considered as a successful group). A classifier is a product of the machine learning component (i.e. a decision tree). It tells how good or bad the newly formed groups are. For example, a decision tree could indicate three group composition rules: 1) if the number of lurkers is not more than 2.5, the groups could be successful; 2) if the number of lurkers is more than 2.5 and the number of leaders is more than 1.5, the groups would then be good too; 3) if the number of lurkers is more than 2.5 and the number of leaders is not more than 1.5, we then would get unsuccessful groups. The total amount of predicted successful groups is a measure to evaluate the quality of the resulting group formations.

The Regrouping component functions as an optimizer. It creates new group formations for students who want to leave and iteratively optimizes grouping results with respect to the quality evaluated by the classifier. A discrete-PSO algorithm is applied to perform this optimization [ZP14].

3.3 Parameter settings

Regarding group interaction parameters, we almost borrowed all those parameter settings from the Nygren's work [Ny12], except for the following parameters. First of all, the number of posts sets up the total number of posts assumed to be made in each group in each task (70 in this work). The roles stereotyping rate is a probability to boost such active roles as leader and disseminator to make a new post. This is only applied to the tasks after the first one. Nygren did not observe the participants' behaviors in the subsequent tasks. The modeling is reported to be of Monte-Carlo type. If we run this modeling for the subsequent tasks the same as the first one, this would lead us to randomness. And this actually does not comply with our assumption that the group roles play an important part in group interaction. To avoid this, we applied an additional policy for regrouping that leaders and disseminators have some privileges to make a new post (determined by this stereotyping rate). It was set to 0.5 in this work.

Drop-out parameters define how many students drop out. Our simulator models the drop-out on a daily basis. Dropout students, by definition, are the most inactive students. Suppose a group task follows the pace of most MOOCs' weekly releasing mode, a weekly drop-out rate needs to be defined and the daily drop-out rate can just average it over 7 days of a week. Note that the vast drop out of MOOC students normally occurs in the first week (approx. 50%) followed by a comparatively smaller yet stable decline rate [Kl15, Ba13, MO13] in the following weeks. As such, placing a team task in the first week would not be a desirable thing to do because of the foreseeable group instability. Our parameters do not model the first week but assume a more or less stable decrease over time (as is realistic from week 2 on). From the second week onwards, we can estimate that the weekly drop-out rate could range from 0 to 50%. Still, we do not know how much exactly that weekly drop-out rate is. In practice, the answer should vary depending on different courses. With that in mind, there is no harm to set the weekly drop-out rate to a random float number ranging from 0 to 50%.

Regrouping parameters define that how many students from four different categories would leave the current groups for the newly composed ones. Categorizing students is based on their participation and group performance. A pair of active and inactive participation together with another pair of successful and unsuccessful group performance composes the four different categories in this set of parameters. The baseline of defining the leaving rates is threefold. First, active students should be more likely to leave for new groups than their inactive counterparts. Second, unsuccessful students should desire more chances to make a success via joining new groups. Hence, the active but unsuccessful students should more likely to leave than the others. Besides, the most conservative but 'most clever' could be the inactive but successful students. They, in fact, take full advantage of the group work without any substantial effort. They could continuously enjoy the benefits as a free rider so as not to leave the current groups unless any negative consequences are foreseeable. Based on the upper assumptions, we can simply randomize the leaving rates of those four categories of students (ranging from 0 to 1), but they should also follow the aforementioned three laws.

3.4 Simulation experiment

To examine the proposed group re-composition's impact via the simulator, two indicators were selected to be observed, namely group cohesion and drop-out. Note that we could not use learning performance as an indicator in our simulation (as no learning was modeled). Alternatively, group cohesion was selected as a substitute for the learning performance for the sake of their positive relationship to each other. Dating back to 1990s, Evans et al. found such positive relationship by the use of meta-analysis [ED91].

Group cohesion, as a structural measure of social network, in this work, is defined as the number of the actual inter-ties among group members divided by the total number of possible ties between any pair of members [Wi14]. By definition, group cohesion directly reflects inter-person ties in groups. The higher the group cohesion is, the stronger ties the groups have. Such strong ties, as inter-communication pipelines, undoubtedly address the salient problem of information-sharing and knowledge-sharing. Group members can thus better know each other and faster fulfil their common goals as a result [Ha99, LFF10]. Group cohesion, to some extent, mirrors group performance. It was thereby chosen as an indicator to learning performance in this simulation work.

Drop-out as another important indicator reflects students' engagement. In current MOOCs, students' engagement is reported to be associated with course content and personal motivations. In group work, what factors can explain such has never been studied thus far. The observation thereby, on the one hand, can hopefully imply some unseen hints to address the high drop-out problem. On the other hand, it can reveal the proposed group re-composition's impact.

Regarding the experimental procedure, first of all, we simulated those 10,000 students' (1,000 groups) group interactions in the first task. In the meantime, we regularly removed dropout students. After their interaction, we selected the students who had desired for a new group. The selection was based on the regrouping parameters mentioned in Section 3.3. We then composed them into new groups using the proposed approach. In order to highlight results in a comparable fashion, we also copied that number of students and composed them into random groups. The former is named algorithmic condition and the latter is named random condition. In the real world, certainly, this is not feasible. In simulation, it is however fairly easy and allows us to simulate what happens to the same students when re-grouped both algorithmically and randomly. We next simulated all those new groups' interactions in Task 2 and removed the dropout students again. When the second task was over, we counted the number of dropout students and the number of cohesive groups from both the algorithmic condition and the random condition. Note that a cohesive group is a group with a group cohesion that is higher than the median cohesion over all groups in both conditions. We ran the whole process 10 times in avoidance of any biased result probably generated by chance.

4 Results

4.1 Impact on group cohesion and dropout

Tab. 2 presents the group cohesion and drop-out. Recall that the whole simulation was repeatedly run 10 times intending to avoid by-chance results. As shown in the table, the amount of cohesive learning groups in the algorithmic condition is larger than in the random condition (algorithmic: 0.466 vs random: 0.421). A t-test was performed and the result indicates that the algorithmic condition produced significantly more cohesive groups than the random condition (p-value: 0.035), which shows its superiority on increasing group cohesion.

Runtimes	#Cohesive groups/total		#Drop-out/total	
	Algorithmic	Random	Algorithmic	Random
#1	0.447	0.335	0.428	0.571
#2	0.482	0.476	0.453	0.546
#3	0.420	0.411	0.445	0.554
#4	0.504	0.488	0.537	0.462
#5	0.440	0.36	0.503	0.496
#6	0.503	0.468	0.446	0.553
#7	0.486	0.464	0.465	0.534
#8	0.447	0.419	0.516	0.483
#9	0.480	0.360	0.416	0.583
#10	0.456	0.431	0.416	0.522
Ave.	0.466	0.421	0.468	0.530
SD	0.026	0.051	0.037	0.037
p-value	0.035		0.002	

Tab. 2: Number of cohesive Groups and Drop-out

Regarding drop-out, more dropout students came from the random groups than from the algorithmic ones (algorithmic: 0.468 vs random: 0.530). Likewise, a conducted t-test indicates a significant difference between both conditions (p-value: 0.002).

Although learning performance was not observable in this simulation experiment, it is not difficult to infer the positive impact on learning performance based on the reported positive ties between group cohesion and learning performance. Since we also saw a lower drop-out rate in the algorithmic groups, the impact on declining the drop-out rate appears positive too.

4.2 Impact of class size

Massive enrolment (thousands of participants) is one of the typical (and defining) MOOC features. Testing re-grouping methods on a data set of 10,000 students as we did is thus obviously necessary. However, one could ask whether the approach also works with a few thousand participants or with even smaller courses, such as an on-campus moodle course with a few hundred students typically. In an attempt to answer such a question, another two simulation experiments were run with three thousand students and one hundred students respectively.

When the participation is set to three thousand students, the observation on drop out and group cohesion reveals no difference to the case of ten thousand students. Similarly, more cohesive groups came from the algorithmic condition than the random one (average: 0.474 vs 0.428). Students tended to drop out more likely in the random condition (0.475 vs 0.524). A statistical test again confirmed significant differences (#cohesive groups: p-value = 0.011, dropout: p-value= 0.001).

In the case of one hundred students, the drop-out rate found in both conditions is almost same (average: 0.501 vs 0.498, p-value: 0.959). The reason for the deficits in this case has their roots in the very limited number of groups to be composed. The number of newly composed groups should be fewer than the total 10 groups (group size is set to 10). In such a small possibility scope, there is no need to challenge the algorithm's capability. In other words, the algorithmic method could perform no better than a random method in such a small case.

Varying the course size from one hundred to ten thousand over these three experiments, the observations tell us two points. First, the scale of participation does matter for the simulation results. Second, the proposed data-driven approach seems to make a positive effect beyond a certain level of participation. At least, for a hundred students, it does not reveal any of its superiorities.

5 Conclusions

This work proposed a novel data-driven approach to address the group re-composition problem that has not attracted much research attention thus far. The recent massive online open courses are arguably a fit to its application. A second contribution of the paper is the evaluation method via simulated classes. Through a simulation experiment, we saw its capability in decreasing the drop-out rate in groups and bringing more cohesive groups when compared against a random grouping strategy. Because of computer simulation's inherent drawbacks of stripping away realism to some extent, the findings are accordingly sensitive to the simulation settings and assumptions. We are certainly aware of this and therefore based the simulation on parameters and theories extracted from literature. Yet, certainly empirical evidence needs to be collected to back our findings.

Bibliography

[Ba13] Balakrishnan, G.: Predicting student retention in massive open online courses using hidden markov models. Master's thesis, EECS Department, University of California, Berkeley, 2013.

[ED91] Evans, C.R.; Dion, K.L.: Group Cohesion and Performance: A Meta-Analysis. Small Group Research 22, 175-186, 1991.

[Ga86] Gamoran, A.: Instructional and Institutional Effects of Ability Grouping. Sociology of Education 59, 185-198 , 1986.

[GB06] Graf, S.; Bekele, R.: Forming heterogeneous groups for intelligent collaborative learning systems with ant colony optimization. In: Proc. Int. Conf. on Intelligent Tutoring Systems, Springer-Verlag, 2171765, S. 217-226, 2006.

[Ha99] Hansen, M.T.: The Search-Transfer Problem: The Role of Weak Ties in Sharing Knowledge across Organization Subunits. Administrative Science Quarterly 44, S. 82-111, 1999.

[Hs14] Hsu, C.-C.; Chen, H.-C.; Huang, K.-H.; Huang, K.-K.; Huang, Y.-M.: The development of an adaptive group composition system on Facebook for collaborative learning using an artificial bee colony algorithm. Appl. Math 8, S. 157-164 , 2014.

[Jo14] Jordan, K.: MOOC Completion Rates: The Data. http://www.katyjordan.com/MOOCproject.html, Stand: 27.08.2014.

[Kl15] Kloft, M.; Stiehler, F.; Zheng, Z.; Pinkward, N.: Predicting MOOC Dropout over Weeks Using Machine Learning Methods. In: Proc. of the 2014 Empirical Methods in Natural Language Processing Workshop on Modeling Large Scale Social Interaction in Massively Open Online Courses, S. 60-65, 2015.

[KBS14] Konert, J.; Burlak, D.; Steinmetz, R.: The Groupf Formation Problem: An Algorithmic

Approach to Learning Group Formation. Open Learning and Teaching in Educational Communities, S. 221-234. Springer, 2014.

[Ku92] Kulik, J.A.: An Analysis of the Research on Ability Grouping: Historical and Contemporary Perspectives. Research-Based Decision Making Series, 1992.

[LFF10] Lechner, C.; Frankenberger, K.; Floyd, S.W.: Task Contingencies in the Curvilinear Relationships Between Intergroup Networks and Initiative Performance. Academy of Management Journal 53, S. 865-889 , 2010.

[Ma08] Marcos-Garcıa, J.A.; Martınez-Monés, A.; Dimitriadis, Y.; Rodrıguez-Triana, M.J.: Role-AdaptIA: A role-based adaptive tool for interaction analysis, 2008.

[MMD15]Marcos-García, J.-A.; Martínez-Monés, A.; Dimitriadis, Y.: DESPRO: A method based on roles to provide collaboration analysis support adapted to the participants in CSCL situations. Comput Educ 82, S. 335-353, 2015.

[MO13] MOOCs@Edinburgh-Group: MOOCs @ Edinburgh 2013 – Report #1, 2013.

[MOV12] Moreno, J.; Ovalle, D.A.; Vicari, R.M.: A genetic algorithm approach for group formation in collaborative learning considering multiple student characteristics. Comput Educ 58, S. 560-569 , 2012.

[Ny10] Nygren, E.: Modeling the social dynamics of online discussion sites. In: Proc. Int. Workshop on Modeling Social Media, ACM, 1835982, S. 1-8, 2010.

[Ny11] Nygren, E.: Grooming Analysis Modeling the Social Interactions of Online Discussion Groups. In (Atzmueller, M.; Hotho, A.; Strohmaier, M.; Chin, A., Hrsg.): Int. Workshops on Analysis of Social Media and Ubiquitous Data, S. 37-56, 2011.

[Ny12] Nygren, E.: Simulation of User Participation and Interaction in Online Discussion Groups. In (Atzmueller, M.; Chin, A.; Helic, D.; Hotho, A., Hrsg.): Int. Workshops on Modeling and Mining Ubiquitous Social Media, S. 138-157, 2012.

[RTZ11] Rabbany, R.; Takaffoli, M.; Zaïane, O.R.: Analyzing participation of students in online courses using social network analysis techniques. In: Proc. Int. Conf. on Educational Data Mining, 2011.

[SA10] Shimazoe, J.; Aldrich, H.: Group Work Can Be Gratifying: Understanding & Overcoming Resistance to Cooperative Learning. College Teaching 58, S. 52-57, 2010.

[SB14] Srba, I.; Bielikova, M.: Dynamic Group Formation as an Approach to Collaborative Learning Support. Learning Technologies, IEEE Transactions on, 2014.

[Sl87] Slavin, R.E.: Ability Grouping and Student Achievement in Elementary Schools: A Best-Evidence Synthesis. Review of Educational Research 57, S. 293-336 , 1987.

[St13] Stuetzer, C.M.; Koehler, T.; Carley, K.M.; Thiem, G.: “Brokering” Behavior in Collaborative Learning Systems. Procedia - Social and Behavioral Sciences 100, S. 94-107, 2013.

[Su10] Sundararajan, B.: Emergence of the most knowledgeable other (MKO): Social network analysis of chat and bulletin board conversations in a CSCL system. Electronic Journal of e-Learning 8, S. 191-208, 2010.

[We16] Wen, M.; Maki, K.; Wang, X.; Dow, S.P.; Herbsleb, J.; Rose, C.: Transactivity as a Predictor of Future Collaborative Knowledge Integration in Team-Based Learning in Online Courses. In: Proc. Int. Conf. on Educational Data Mining, 2016.

[Wi14] Wise, S.: Can a team have too much cohesion? The dark side to network density. European Management Journal 32, S. 703-711, 2014.

[YA12] Yannibelli, V.; Amandi, A.: A deterministic crowding evolutionary algorithm to form learning teams in a collaborative learning context. Expert Systems with Applications 39, S. 8584-8592, 2012.

[ZP14] Zheng, Z.; Pinkwart, N.: A Discrete Particle Swarm Optimization Approach to Compose Heterogeneous Learning Groups. In: Proc. Int. Conf. on Advanced Learning Technologies (ICALT), S. 49-51, 2014.