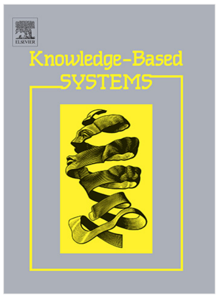
Accepted Manuscript



Synergistic team composition: A computational approach to foster diversity in teams

Ewa Andrejczuk, Filippo Bistaffa, Christian Blum, Juan A. Rodriguez-Aguilar, Carles Sierra

PII: S0950-7051(19)30274-6
DOI: https://doi.org/10.1016/j.knosys.2019.06.007
Reference: KNOSYS 4799

To appear in: *Knowledge-Based Systems*

Received date : 27 February 2019
Revised date : 4 June 2019
Accepted date : 5 June 2019

Please cite this article as: E. Andrejczuk, F. Bistaffa, C. Blum et al., Synergistic team composition: A computational approach to foster diversity in teams, *Knowledge-Based Systems* (2019), https://doi.org/10.1016/j.knosys.2019.06.007

Synergistic Team Composition: A Computational Approach to Foster Diversity in Teams

Ewa Andrejczuk[a,b], Filippo Bistaffa[a], Christian Blum[a,*], Juan A. Rodriguez-Aguilar[a], Carles Sierra[a]

[a]Artificial Intelligence Research Institute (IIIA-CSIC), Campus of the UAB, Bellaterra, Catalonia, Spain
[b]ST Engineering - NTU Corporate Laboratory, Nanyang Technological University, Singapore

Abstract

Co-operative learning in heterogeneous teams refers to learning methods in which teams are organised both to accomplish academic tasks and for individuals to gain knowledge. Competencies, personality and the gender of team members are key factors that influence team performance. Here, we introduce a team composition problem, the so-called *synergistic team composition problem* (STCP), which incorporates such key factors when arranging teams. Thus, the goal of the STCP is to partition a set of individuals into a set of *synergistic teams*: teams that are diverse in personality and gender and whose members cover all required competencies to complete a task. Furthermore, the STCP requires that all teams are balanced in that they are expected to exhibit similar performances when completing the task. We propose two efficient algorithms to solve the STCP. Our first algorithm is based on a linear programming formulation and is appropriate to solve small instances of the problem. Our second algorithm is an anytime heuristic that is effective for large instances of the STCP. Finally, we thoroughly study the computational properties of both algorithms in an educational context when grouping students in a classroom into teams using actual-world data.

Keywords: team composition, exact algorithms, heuristic algorithms, optimisation, coalition formation

1. Introduction

Active learning refers to a broad range of teaching techniques that engage students to participate in all learning activities in the classes. Typically, active learning strategies involve a substantial amount of students working together within teams. Research shows that students learn better when using active learning compared to the traditional

*Corresponding author
Email addresses: ewaa@ntu.edu.sg (Ewa Andrejczuk), filippo.bistaffa@iiia.csic.es (Filippo Bistaffa), christian.blum@iiia.csic.es (Christian Blum), jar@iiia.csic.es (Juan A. Rodriguez-Aguilar), sierra@iiia.csic.es (Carles Sierra)

schooling methods [1]. They do not only acquire and retain the information better but also are more content with their classes [2].

Nevertheless, not all teams facilitate learning. For team-based learning to be effective, every team composed in the classroom needs to be heterogeneous, i.e. diverse in individuals' characteristics. Furthermore, having some significantly weaker teams and some significantly stronger teams is undesirable. Hence, the distribution of teams in a classroom must be *balanced* in the sense that all teams are more or less equally strong.

Even though much research in the industrial, organisational, and educational psychology fields investigated what are the predictors of team success, to the best of our knowledge, there are no computational models to build teams for a given task that are broadly used in the classrooms. Frequently studied individual characteristics that influence team performance are competencies, personality traits, and gender [3, 4, 5, 6]. [3, 6] show a positive correlation between certain personality traits and team composition. [4, 5] show that in order to increase team performance, team members should be heterogeneous in their individual characteristics.

Some of those characteristics were also acknowledged by multiagent systems (MAS) research. The most studied characteristic in MAS research are competencies [7, 8, 9, 10, 11]. However, in these works agents' competencies are generaly represented as True/False characteristics, that is, an agent has or does not have a required competence. This is an oversimplified approach to model agents' competencies since it disregards any competence grade. In reality, competencies are non-binary because individuals are characterized by different grades of competencies. Unfortunately, MAS research has typically ignored significant psychology findings (with the exception of recent, preliminary works such as [7] and [9]).

To the best of our knowledge, neither the current MAS literature nor the current psychology literature has considered team composition based on competencies, personality and gender of individuals at the same time.

In this paper, we focus on the following team composition problem that is commonly encountered in education. We consider a *complex task that needs to be performed by multiple student teams of even size* [12]. The task requires each team to have at least one student with a minimum level of competence for each competence from a given set of competencies. There is a pool of students with varying competencies, genders, and personalities. The objective is to partition students into teams so that each team is even in size and balanced in competencies as well as personalities, and gender. We term those teams as *synergistic teams*.

In this context, the paper makes the following contributions:

- We identify and formally define a new type of real-life problem, the so-called synergistic team composition problem (STCP). The goal of the STCP is to partition a set of individuals into a set of *synergistic teams*: teams that are diverse in personality and gender and whose members cover all required competencies to complete a task. Furthermore, the STCP requires that all teams are balanced in that they are expected to exhibit similar performances when completing the task.

- We introduce two different algorithms to tackle the STCP: (i) STCPSolver, an algorithm that employs a reformulation of the problem which is then solved to optimality by an off-the-shelf integer linear programming (ILP) solver, and (ii) SynTeam, an anytime heuristic that can produce solutions of high quality within a limited computation time.

- We perform an exhaustive computational comparison of STCPSolver and SynTeam over realistic settings in education, considering actual-world data. Overall, our analysis indicates that STCPSolver is efficient for rather small problem instances, whereas SynTeam is able to cope with larger problem instances. First, we notice that the runtime of the optimal algorithm greatly increases with a growing team size and a growing number of students, which causes the algorithm not to be applicable to larger instances of the problem. This is not the case for SynTeam, which is capable of composing teams for larger problem instances while providing good quality approximate solutions (whose values range, in the worst case, between 75% and 95% of the value of an optimal solution). Second, we compare the anytime performance of both algorithms. We observe that SynTeam outperforms STCPSolver for large team sizes (beyond 3), whereas the opposite occurs for small team sizes. We also compared our optimal approach (i.e., STCPSolver) to ODP-IP [13], the state-of-the-art algorithm to solve the *coalition structure generation* problem. Results show that STCPSolver outperforms ODP-IP both in terms of runtime (since ODP-IP cannot exploit the presence of cardinality constraints to reduce the space of feasible solutions) and scalability (due to ODP-IP's exponential memory requirements).

Outline. The remainder of this paper is structured as follows. Section 2 provides an overview of the related work. Section 3 introduces the basic definitions used in this paper. Section 4 introduces the key notions used to measure the synergistic value of a team and formally defines the synergistic team composition problem. Sections 5 and 5.2 describe STCPSolver and SynTeam, the two algorithms that we introduce in this work. Then, Section 7 discusses our empirical comparison of the proposed algorithms over synthetically generated instances of the STCP. Section 8 briefly introduces a web application that is freely available and offers team composition as its main functionality. Finally, Section 9 discusses both the conclusions and directions for future research.

2. Related work

In this section we review relevant related work. Sections 2.1 and 2.2 go through related work in the education literature and the organisational psychology literature respectively. Sections 2.3 and 2.4 revise related work in the computer science literature; section 2.3 revises the multiagent systems literature, whereas section 2.4 discusses relevant work in the coalition formation literature.

2.1. Relation with the education literature

There are many works that advise on how to *handcraft* heterogeneous teams with the purpose of increasing team-based learning and improving team performance, for instance, [14] or [15].

[14] offers a *manual* method to divide a classroom based on students' personalities and genders. In this paper, we extend the method in [14] by adding competencies and propose an algorithm to compose teams in an *automatic* way.

[15] advises beginning a team composition process by simply asking questions to a group of students. These questions are used to gather information about those competencies that are important for the successful completion of a given task. Students respond to each question either orally or with a show of hands. Then, students are lined up based on the number of required competencies that they have, derived from the answers to the questions. Ties are broken randomly. Finally, teams are built by asking students to count off down the line. For instance, if teams of five students are required, the count is as follows: $1, 2, 3, 4, 5, 1, 2, \ldots$. Thereby, the number associated to a student indicates the team to which he/she is assigned.

Some authors have tried to automatise the team composition process. That is, they have aimed at composing a set of teams so that all teams are as similar as possible with regard to the mean values of multiple attributes [16, 17, 18, 19, 20]. As opposed to our approach, none of these works imposes heterogeneity in a direct way when composing teams. They are rather limited to studying a set of fixed constraints (such as avoiding clustering particular majors, ensuring that no international student and no female are isolated on a team, etc). Additionally, compared to our approach where we compose teams for particular tasks, they do not explicitly consider the notion of the task when composing balanced teams. This is also the case in the work by Agrawal et al. [21], though the authors diverge from the above-mentioned approaches to team composition. Their team composition approach focuses on grouping students so that, in the end, the value gained by less capable students through collaboration is maximised. Despite the novelty of their team composition (grouping) problem, Agrawal et al. only consider students capabilities (disregarding the findings of the organisational psychology literature about other individual attributes, i.e. personality and gender, that we include), consider that students count on abilities for a single competence (instead of multiple competencies as we do), and are not concerned about yielding balanced team compositions, which is our main goal.

To the best of our knowledge, the only available web tool supporting team composition is the Comprehensive Assessment of Team Member Effectiveness (CATME)[1] that composes teams based on individual students' responses to an online survey. Teachers define student surveys by selecting the desired students' characteristics from a given inventory [19]. The application calculates a “question score” for each characteristic that informs how well each team's distribution of that characteristic satisfies the teacher's aims. The application also measures a global “compliance score” for each composed team characterizing how well the team satisfies the teacher's wishes. The higher these values the better the team. Their team composition algorithm starts by randomly distributing students across teams of a pre-specified size. Next, the algorithm calculates both question and compliance scores. Then, it iteratively changes the teams with the purpose of maximising the minimum compliance score of all teams. This work is similar to our approach, however, there are also substantial differences. In addition to the

[1] http://www.catme.org

differences discussed above, the authors do not analyze their solutions' quality. They assume that the groupings produced by their algorithm are near-optimal. The analysis performed by [22] shows, however, that it is implausible the CATME method achieves near-optimal results.

2.2. Relation with the organisational psychology literature

As far as we are concerned, there are no methods in the organisational psychology literature that would provide a complete guideline on how to compose teams. Instead, the researchers in this field study how individual characteristics influence team performance.

The most studied individual characteristic that is associated with team performance is cognitive ability. [23] define it as the "capacity to understand complex ideas, learn from experience, reason, solve problems, and adapt" [23, p.507]. It is a very wide concept that—in addition to competencies, broadly used in multiagent systems research—covers many other characteristics such as experience, gender or even age. [24] and [23] discovered the positive correlation between team performance and the average team values of cognitive ability. [23] also showed that the variance of team members' cognitive ability was not a good predictor of team performance. Additionally, these authors observed that the average value is two times more informative for the prediction of team performance than the lowest and the highest member scores. [25] suggested the existence of collective intelligence in teams that can predict team performance. This collective intelligence is not strongly correlated with the maximum or average intelligence of team members. Instead, it is positively correlated with increasing equality in conversational turn-taking, the mean social sensitivity of group members, and gender balance [26].

The organisational psychology literature, in addition to cognitive ability, has examined the impact of personality traits on team performance [27, 6]. The most popular questionnaires to determine personality include: (1) the Five Factor Model (known as "Big Five" or FFM), which uses five traits to define individual personality [28]; (2) Belbin theory [29], which provides a theory of nine different personality role types; and (3) the Myers-Briggs Type Indicator (MBTI) questionnaire that uses four traits to specify psychological preferences concerning the way people perceive the world and make decisions [30].

Concerning FFM, [24] found that for each personality trait examined individually (i.e. extraversion, agreeableness, conscientiousness, emotional stability, openness to experience), team means were associated with team performance. The study in [31] confirmed these findings for all traits except for the Openness to Experience trait which was not considered. However, the sizes of studied samples were small and it is unclear whether these findings are statistically significant [32]. [33] reported contradictory findings after studying student teams. Each team was asked to improve processes based on problems encountered in organisations. The researchers measured team orientation, extraversion, agreeableness, conscientiousness, and emotional stability of each team using the team average and team variability. Interestingly, they did not find any meaningful connection between team performance and any of these personality traits (when examined individually).

According to Belbin, there are nine team roles that should be covered in every team [29]. These roles are: completer–finisher, coordinator, implementer, monitor evaluator, plant, resource investigator, shaper, specialist and teamworker. Although some studies with very limited sample sizes (such as 10 teams in [34]) reported support for the theory, studies based on a larger number of samples did not find the relation between the Belbin roles and team performance [35, 36, 37].

Finally, the MBTI has four binary dimensions, that is: intuition vs sensing (N–S), thinking vs feeling (T–F), extraversion vs introversion (E–I), and perceiving vs judging (P–J). Within this questionnaire, every individual can be categorised into one of the sixteen possible four-letter combinations, where each letter represents one personality dimension. This approach is easy to interpret by non-psychologists. Reliance on dichotomous preference scores rather than on continuous scores, however, excessively restricts the level of statistical analysis [38].

2.3. Relation with the multiagent systems literature

To our knowledge, the only computational model in the context of team composition that takes both personality and competencies into account was presented in [39]. In particular, the influence of personality on different strategies for allocating tasks is studied in this paper. However, there are substantial differences with our work. Firstly, instead of proposing an algorithm for the composition of teams both based on personality and competence, they only describe a model to evaluate teams. Secondly, they give no importance to gender balance. And finally, they do not evaluate their algorithm with real data (only via agent-based simulation).

We separate the remaining literature that is relevant to this article into the following two categories: works that deal with agent competencies (individual and social capabilities of agents), and works that consider agent personality (individual behaviour models).

Competencies. Various previous works have focused on the competency dimension. However, in contrast to our work, in which competencies are graded, the majority of works assume agents to have multiple binary skills (either an agent has a required skill or not). In [40] and [10], for instance, one k-robust team is composed for a single task, based on the agents' capabilities. Hereby, a team is called k-robust if by removing any k members from the team, the completion of the task is not compromised. In [41], each task requires a specific set of competencies. Moreover, tasks arrive sequentially over time. The team composition algorithm, whose focus is on balancing the workload of the agents across teams, builds teams based on competencies and communication cost.

Personality. There are, to our knowledge, two works in the literature that consider agents' personality to compose teams, namely [7] and [9]. In [7], Belbin's theory is used to obtain human predominant *roles* (see Section 2.2). As discussed in subsection 2.2, these roles do not tend to be related to team performance. Additionally, gender is not considered for the composition of heterogeneous teams.

In [9], Farhangian et al. make use of the classical MBTI personality test (see Section 2.2). Their aim is to build the best possible team around a selected leader. In other words, they compose the *best* possible team for a particular task. However, gender balance is again not considered. Finally, although real data was considered in [9], the resulting teams' performance was not validated. Instead, Bayesian theory was used to predict the success probability in a variety of team composition conditions.

2.4. Relation with the coalition formation literature

The STCP can be seen as a *coalition structure generation* (CSG) problem [13] over the entire set of students with a characteristic function that assigns a synergistic value to every *feasible* coalition (i.e., with the desired number of students), and $-\infty$ to every *unfeasible* coalition (since the characteristic function has to be defined for every possible subset of agents in the standard definition of CSG). Solving the STCP requires to compute the coalition structure (team partition) with the largest total value, i.e., the optimal solution to the CSG problem. In principle, state-of-the-art CSG approaches such as ODP-IP [13] could be used to solve the STCP problem. Unfortunately, these approaches are not able to exploit the presence of cardinality constraints to reduce the space of feasible solution, and hence, are limited to problem instances of up to 25 agents, due to their exponential memory requirements, as shown by our experiments in Section 7.2.

On the other hand, given a STCP we can also define a constrained coalition formation (CCF) [42] game $\mathcal{G} = \langle A, \mathcal{P}_m(A), s \rangle$, where $\mathcal{P}_m(A)$ is the set of feasible coalition structures. More precisely, the STCP poses a specific type of CCF game, namely, a *basic* CCF game [42]. Intuitively, basic CCF game express constraints in the form of: (1) allowed sizes of coalitions that can be formed; and (2) subsets of agents whose presence in any coalition is permitted or not. On the one hand, a STCP naturally defines constraints on the size of coalitions. On the other hand, expressing a STCP as a CCF problem would require to define one positive constraint per feasible team, while the set of negative constraints would be empty. As a consequence, the number of positive constraints quickly becomes very large (i.e., > 3000 in our case), hence making the use of the approach by Rahwan *et al.* [42] impossible.

3. Team Composition Model

There are three diversity dimensions of students used in our model: gender, personality, and competencies. We measure personality using the theory of personality called Post-Jungian [14] which is a reduced variant of the Myers-Briggs Type Indicator (MBTI) [20]. The numbers are obtained from the answers of a short questionnaire of 20 quick questions (much shorter than the common 93 questions of the Boolean MBTI). This is very efficient in terms of time and effort for both teachers and students, as completing the test takes only a few minutes (see [14, p.21] for details). Douglass J. Wilde claimed that this numerical method is a coherent extension of the psychological dimensions of MBTI [43]. The test is based on the personality model proposed by C. G. Jung [44] containing two sets of functions and attitudes:

1. Sensing – Intuition (SN),
2. Thinking – Feeling (TF)
3. Extroversion – Introversion (EI),
4. Perception – Judgment (PJ).

The numerical values along each dimension (SN, TF, EI, PJ) are the result of combining the answers to the questionnaire mentioned above where each question can be answered by selecting one out of five possible answers; Each possible answer has a value (in a scale from -1 to +1). Each personality trait is assessed by five questions. The values of the answers are added up and divided by 5 (the number of questions) to give the final value along each personality dimension. This method seems promising as—within one decade—Prof. Wilde multiplied the number of teams of Stanford that were awarded prizes by the Lincoln Foundation [45] by three. Accordingly, the definition of a personality profile in our context is as follows.

Definition 1. *A* personality profile *is a tuple* $\langle sn, tf, ei, pj\rangle \in [-1,1]^4$ *of personality traits.*

A competence is understood as the knowledge, skills and attitudes that enable a student to successfully solve tasks and face challenges [45]. Moreover, a student possesses every competence with a certain level. Let $C = \{c_1, \ldots, c_k\}$ be the set of competencies.

Definition 2. *A* student *is represented as a tuple* $\langle id, g, \mathbf{p}, l\rangle$ *such that:*

- id *is the student's identifier;*
- $g \in \{man, woman\}$ *stands for the student's gender;*
- $\mathbf{p}$ *is a personality profile tuple;*
- $l : C \to [0,1]$ *gives the students competence levels, that is,* $l(c)$ *is the student* competence level *for competence c. We assume that when a student does not have a competence (or we do not know about it),* $l(c) = 0$.

Henceforth, the set of considered students is denoted by $A = \{a_1, \ldots, a_n\}$.

The notion of a team is defined as follows, in a straightforward way, as a group of two or more students.

Definition 3 (Team). *A* team *is any subset of* A *with at least two students. We denote by* $\mathcal{K}_A = (2^A \setminus \{\emptyset\}) \setminus \{\{a_i\}|a_i \in A\}$ *the set of all possible teams from students in* A.

$w(K)$ and $m(K)$ are the number of women and men respectively in team K. Students are organized in teams to solve tasks. We understand a task as an instance of a *task type*. A task type not only determines the competences that are required to successfully solve any instance, but also specifies the competence levels and the relative importance of competences. Task types thus differ in requiring different competence levels. A specific task type may require, for instance, a high level of creativity (e.g. to design a city brochure), while another one may require analitycal competences (e.g. to solve mathematical equations). This is formalized as follows.

Definition 4. *A task type* τ *is a tuple* $\langle \lambda, \{(c_i, l_i, w_i)\}_{i \in I_\tau}\rangle$ *where:*

- I_τ *is the index set of the required competencies.*

- $\lambda \in [0, 1]$ *is the importance given to proficiency; the higher the value of* λ*, the higher the team proficiency importance.*
- $c_i \in C$ *is a competence required to perform the task;*
- $l_i \in [0, 1]$ *is the required competence level for* c_i*;*
- $w_i \in [0, 1]$ *is the importance of competence* c_i *for the success in solving an instance of task type* τ*; and* $\sum_{i \in I_\tau} w_i = 1$.

Tasks are instances of task types plus a required number of students.

Definition 5. *A* task t *is a tuple* $\langle \tau, m \rangle$ *such that* τ *is a task type and* m *is the required number of students, where* $m \geq 2$.

We note by T the set of tasks and by $\mathcal{T}$ the set of task types. We will note by $C_\tau = \{c_i | i \in I_\tau\}$ the set of competencies required by task type τ.

Given a team and a task, we must consider how to assign responsibilities for the competencies within the team. This *competence assignment* is defined as follows.

Definition 6. *Given a task type* τ *and a team* $K \in \mathcal{K}_\mathcal{A}$*, a competence assignment is a function* $\eta_\tau : K \to 2^{C_\tau}$ *satisfying that* $C_\tau = \bigcup_{a \in K} \eta_\tau(a)$*. We note by* Θ_τ^K *the set of competence assignments for task type* τ *and team* K.

The list of students assigned to each competence is defined as follows.

Definition 7. *Given a task type* τ*, a team* K*, and competence assignment* η_τ*, the set* $\delta(c_i, K, \eta_\tau) = \{a \in K | c_i \in \eta_\tau(a)\}$ *stands for those students in team* K *responsible for competence* c_i.

In team-based learning, it is a key requirement that students share responsibilities in order to achieve a successful performance. Hence, our objectives are: (a) to distribute responsibilities in a balanced way across a team; and (b) to have each team member responsible of at least one competence. This is especially important in an education context, where no one should be cornered within a team. We shall refer to such an assignment as a *balanced competence assignment*. Note that we will be concerned with this particular assignment in this paper. Hereafter, we note by $\bar{\Theta}_\tau^K$ the set of balanced competence assignments for task type τ and team K, where $\bar{\Theta}_\tau^K \subseteq \Theta_\tau^K$.

4. The Problem of Composing Synergistic Teams

Next, we present our computational model to compose and evaluate teams. First, we introduce a way of measuring *proficiency*, namely the degree of matching between a competence assignment and the competences of the members of a team. Thereafter, we provide a measure of *congeniality*, namely of the diversity of personalities of the members in a team. Then, the *synergistic* value of a team results from combining both proficiency and congeniality values.

4.1. How to assess the proficiency value of a team

Our goal is to calculate the *proficiency degree* of a team for a particular task from a competence assignment. With this aim, our measure of proficiency will adhere to the following principle: the closer the competence levels of the team members in a competence assignment to the competence levels required by the task, the larger the proficiency degree of the team. In this way, we pursue to avoid both *under-proficient* and *over-proficient* competence assignments, since they involve under-qualified and over-qualified teams. On the one hand, students in under-proficient teams may get frustrated because of their lack of knowledge to undertake their assignments. On the other hand, as argued in [46], students in over-qualified teams are bound to lose attention and motivation because of the lack of challenge in their assignments.

Our formal definitions of under-proficiency degree and over-proficiency degree are based on measuring the distance between what is required (in terms of competence levels) by a task and what a team offers to perform the task (according to a competence assignment).

Definition 8 (Under-proficiency degree). *The under-proficiency degree of a team K to perform a task of type τ according to a competence assignment η_τ is:*

$$u(\eta_\tau) = \sum_{i \in I_\tau} w_i \cdot \frac{\sum_{a \in \delta(c_i, K, \eta_\tau)} |\min(l^a(c_i) - l_i, 0)|}{|\delta(c_i, K, \eta_\tau)| + 1}$$

Definition 9 (Over-proficiency degree). *The over-proficiency degree of a team K to perform a task of type τ according to a competence assignment η_τ is:*

$$o(\eta_\tau) = \sum_{i \in I_\tau} w_i \cdot \frac{\sum_{a \in \delta(c_i, K, \eta_\tau)} \max(l^a(c_i) - l_i, 0)}{|\delta(c_i, K, \eta_\tau)| + 1}$$

We combine the under-proficiency and over-proficiency degrees of a team as a weighted average to finally obtain the proficiency degree of a team as follows:

Definition 10. *The proficiency degree of a team K to perform a task of type τ following a competence assignment η_τ, and considering an under-proficiency penalty $\upsilon \in [0, 1]$ is:*

$$u_{prof}(K, \tau) = \max_{\eta_\tau \in \bar{\Theta}_\tau^K} (1 - (\upsilon \cdot u(\eta_\tau) + (1 - \upsilon) \cdot o(\eta_\tau)). \quad (1)$$

Definition 10 is restricted to the set of balanced competence assignments $\bar{\Theta}_\tau^K$, which are the relevant competence assignments in education scenarios, as discussed above.[2] Furthermore, It is worth noticing that function $u_{prof}(K, \tau)$ in Definition 10 is well defined for any team, task type and competence assignment. Indeed, for any task type τ, team K, and $\eta \in \Theta_\tau^K$, we observe that $u(\eta_\tau) + o(\eta_\tau) \in [0, 1)$ and

[2] A more general definition of proficiency could be readily obtained by considering the set of all competence assignments Θ_τ^K instead. However, we propose this definition for the sake of simplicity.

$0 \leq u_{prof}(K, \tau) < 1$. This is true since no student can be over-proficient and under-proficient at the same time.

From equation 1 we observe that the larger the value of importance of the proficiency penalty (υ), the larger the importance of the over-proficiency degree. And the other way around, the lower the proficiency penalty, the less important the under-proficiency degree. Hence, setting large values to the proficiency penalty guarantees that competence assignments that make a team under-competent (unable to cope with competence requirements) are penalised. Analogously, small proficiency penalties are meant to penalise over-competent teams. The correct setting of the proficiency penalty parameter will depend on each task type. On the one hand, if our objective is to foster *effective* teams, then we must set the proficiency penalty to a large value to penalise more under-proficiency.

In order to computer $u_{prof}(K, \tau)$ we must solve a constrained optimisation problem: find the balanced competence assignment with minimum *cost* (in terms of under- and over-proficiency). This problem can be formulated and solved as a minimum cost flow problem. More precisely, given a team $K \subseteq \mathcal{K}_{\mathcal{A}}$ and a task type τ, we derive the balanced competence assignment η that minimises equation 1 by solving a minimum cost assignment problem, which in turn can be expressed as an integer linear program (ILP) as follows. The ILP employs a binary variable x_{ij} to encode the decision of whether student $a_i \in K$ is tasked with competence $c_j \in C_\tau$, where C_τ is the set of competencies required by task τ. Hereby, the cost of assigning a student a_i to a competence c_j, denoted by p_{ij}, is defined as follows:

$$p_{ij} := \begin{cases} (l^{a_i}(c_j) - l_j) \cdot (1-\upsilon) \cdot w_j & \text{if } l^{a_i}(c_j - l_j) \geq 0 \\ -(l^{a_i}(c_j) - l_j) \cdot \upsilon \cdot w_j & \text{if } l^{a_i}(c_j - l_j) < 0 \end{cases}$$

where $\upsilon \in [0, 1]$ is the penalty applied to the under-proficiency of team K (see Section 4.1 for a detailed introduction of this term) and $w_j \in [0, 1]$ weighs the importance of competence c_j to succeed in completing a task of type τ (see definition 4).

The above-mentioned minimum cost assignment problem can then be expressed in the following way as an ILP model.

$$\min \sum_{a_i \in K} \sum_{c_j \in C_\tau} x_{ij} \cdot p_{ij} \tag{2}$$

subject to:

$$\sum_{c_j \in C_\tau} x_{ij} \leq \left\lceil \frac{|C_\tau|}{|K|} \right\rceil \quad \forall\, a_i \in K \tag{3}$$

$$\sum_{c_j \in C_\tau} x_{ij} \geq 1 \quad \forall\, a_i \in K \tag{4}$$

$$\sum_{a_i \in K} x_{ij} = 1 \quad \forall\, c_j \in C_\tau \tag{5}$$

Constraint (3) makes sure that each student is assigned to at most $\left\lceil \frac{|C_\tau|}{|K|} \right\rceil$ competencies, while constraint (4) makes sure that each student is assigned to at least one competence.

Note that constraints (4) are only used if $|C_\tau| \geq |K|$. Finally, constraint (5) ensures that each competence has exactly one student assigned to it.

The solution to the ILP above allows us to build the balanced competence assignment required to compute u_{prof} in equation 1 as follows: for each student a_i in team K, $\eta_\tau(a_i) = \{c_j \in C_\tau | x_{ij} = 1\}$.

At this point we have learned how to compute the proficiency value for a team given a particular competence assignment. However, as argued in the introduction, the degree of proficiency alone is not enough for a team to succeed. Next we introduce a function to measure the *congeniality* within a team from the personalities and genders of its team members. Thus, our congeniality measure does not consider any competence assignment, hence differing from our above-defined proficiency measure.

4.2. How to assess the congeniality value of a team

Recent studies in organisational psychology have proven the existence of a trade-off between the creative productivity caused by "meta-cognitive conflict" and "harmony" —good feeling— in a team [47]. On the one hand, meta-cognitive conflict stems from the different views of the world that people exhibit based on opposing personality and gender. On the other hand, harmony originates in agreements between people with similar personalities [14].

Based on such observations, in [43], Wilde proposes several heuristics to target the composition of successful teams. Along these lines, here we propose to build cognitively diverse teams by employing psychological functions (the SN and TF pairs), psychological attitudes (PJ and EI), and gender. With the aim of mathematically formalising Wilde's heuristics, we introduce a novel function to measure *congeniality*, u_{con}, based on the following objectives:

1. the more diverse a team (in terms of the sensing-intuition (SN) and thinking-feeling (TF) personality dimensions of its team members), the larget its congeniality value u_{con};
2. u_{con} values more teams with at least one member that is extrovert, thinking and judging (with positive EI, TF and PJ personality dimensions), namely exhibiting an ETJ personality;
3. u_{con} prefers teams that with at least one introvert member (with negative EI personality dimension); and
4. the more the gender blanace in a team, the larger its congeniality value u_{con}.

Definition 11. *Given a team K and a task type τ, we define the congeniality degree of the team to perform the task as:*

$$u_{con}(K) = u_{SNTF}(K) + u_{ETJ}(K) + u_I(K) + u_{gender}(K), \tag{6}$$

with

1. *$u_{SNTF}(K) = \sigma(K, SN) \cdot \sigma(K, TF)$ measures team diversity, where $\sigma(K, SN)$ and $\sigma(K, TF)$ are the standard deviations over the distributions of the SN and TF personality traits for the members of team K. the SN and TF personality trait distributions of the members of team K. The larger the values of those*

deviations, the larger the personality diversity with respect to the SN and TF dimensions, and the larger their product.[3]

2. $u_{ETJ}(K) = \max\{0, \max\{(0, \alpha, \alpha, \alpha) \cdot \mathbf{p^a} | a \in K\}\}$ *represents the utility of ETJ personalities, where the importance of each dimension TF, EI and PJ (the second, third and fourth dimensions of a personality profile) is considered equal and bounded by* α.
3. $u_I(K) = \max\{0, \max\{(0, 0, -\beta, 0) \cdot \mathbf{p^a} | a \in K\}\}$ *measures the utility of an introvert student, being* β *a value to measure the relevance of introvert students.*
4. $u_{gender}(K) = \gamma \cdot \sin(\pi \cdot g(K))$ *measures the preference over gender balance. Function* $g(K) = \frac{w(K)}{w(K)+m(K)}$ *yields the ratio of women in a team considering the number of women* ($w(K)$) *and men* ($m(K)$)*. The* γ *parameter* ($\gamma \leq 1$) *weighs the importance of gender balance. A team* K *is perfectly gender-balanced iff* $w(K) = m(K)$*, and hence* $g(K) = 1/2$ *and* $\sin(\pi \cdot g(K)) = 1$*. Observe that when the number of women and men is equal, it follows that* $g(K) = 1/2$ *and* $\sin(\pi \cdot g(K)) = 1$*. In this case, we say that a team is perfectly balanced.*

Given the above definition, we now discuss how to choose the values of parameters α, β and γ, as they affect the congeniality degree of a team.

If all factors in $u_{con}(K)$ are equally important, then the values of α, β and γ are interdependent and will ultimately depend on the shape of the distribution of the personality traits. Next we analyse two extreme cases and give the actual values that have been used in the experiments.

- *Distribution with maximal variance*. The maximum value of $u_{ETJ}(K)$ will be 3α in case there is a student a such that $p^a = (k, 1, 1, 1)$, where $k \in [-1, 1]$. If $u_{SNTF}(K)$ and $u_{ETJ}(K)$ have to have the same importance then we need to equate the maximum value of $\sigma(K, SN) \cdot \sigma(K, TF)$ to 3α. The distribution over an interval $[a, b]$ that has maximal variance corresponds to the distribution with the elements evenly situated at the extremes of the interval with $\sigma^2 \leq ((b-a)/2)^2$. That distribution would correspond to teams whose students have values on dimensions SN and TJ at the extremes of the interval $[-1, 1]$. Regarding our particular case, which considers the $[-1, 1]$ for personality traits ($b = 1$, $a = -1$), the value of σ for that distribution would be 1, and hence that would imply that $3\alpha = \sigma(K, SN) \cdot \sigma(K, TF) \leq 1$ and thus $\alpha \leq 1/3 = 0.33$.

- *Uniform distribution*. If the distributions for SN and TF values follow a uniform distribution, then the variance of each distribution is $\sigma^2 \leq \frac{(b-a)^2}{12}$ and thus $\sigma(K, SN) \cdot \sigma(K, TF) = \frac{(b-a)^2}{12} = 3\alpha$ which implies that $\alpha \leq 0.11(1)$.

[3] Other diversity measures could be used. A possibility would be to understand students as charged particles that distribute in the space as to minimise the overall energy (maximum entropy point). This analogy is appropriate as what is needed in a truly diverse team is that everybody is far from one another as repelling particles are. The repelling force between two particles is proportional to $1/d^2$ where d is the distance between the particles. So given n particles/students, the values (in the dimensions SN, TJ, or EI) that give the minimum energy are: $\arg\min_{f \in F} \sum_{i,j \in A} \frac{1}{(f(i)-f(j))^2}$. Where $f \in F$ is a function that assigns values to students in a particular dimension (SN, TJ or EI).

These cases represent extreme situation, whereas real-world scenarios usually lie in the middle. Thus, we define $0.11(1) \leq \alpha \leq 0.33(3)$. Then, in order to make $u_{SNTF}(K)$ and $u_I(K)$ equally important, it follows that $\beta \approx 2\alpha$. Finally, by setting $\gamma \approx 3\alpha$ we make the gender factor equally important to the rest of factors in equation 6.

4.3. Evaluating synergistic teams

We now define our performance measure to evaluate each team. Specifically, a team K is effective when it is both *proficient* and *congenial*. This means that a team counts on the required competences required to perform a task, and also that it shows a balance of gender and personalities so that students will work well together. The *synergistic value* of a team results from aggregating its proficiency and congeniality values as follows:

Definition 12. *Given a team K, its synergistic value to perform an instance of τ is:*

$$s(K, \tau) = \lambda \cdot u_{prof}(K, \tau) + (1 - \lambda) \cdot u_{con}(K), \tag{7}$$

where $\lambda \in [0, 1]$ weighs the importance of proficiency.

As the value of λ determines the relative importance of the congeniality and proficiency factors, its definition depends on the task type. As an example, congeniality is more important to solve tasks that require a high level of creativity (e.g., tasks tackled for the first time), hence, in this case, $\lambda < 0.5$. On the other hand, proficiency is crucial to complete tasks appearing in, for instance, sport competitions or disaster management, which require the fast operation of teams. In those cases, we must set λ to a value greater than 0.5

4.4. Problem definition

Given a set of students A, we aim at partitioning A into teams so that each team is balanced (in terms of competencies, gender and personality) and team sizes are even. Henceforth, we shall refer to balanced (i.e., both congenial and proficient) teams as *synergistic teams*. Any partition of A into teams is denoted as a team partition. Furthermore, we are interested in forming team partitions whose teams are constrained by size m as follows, as this constraint usually applies in educational contexts.

Definition 13. *Given a set of students A, a team partition P_m of A is constrained by size m, $2 \leq m \leq |A|$, iff for every team $K \in P_m$, $m \leq |K| \leq m + 1$[4] holds.*

Henceforth, we will focus on the set $\mathcal{P}_m(A)$ of team partitions of set A constrained by some size m.

As mentioned above, our objective is to compute a partition whose teams are as good as possible. Thus, we want to disregard unbalanced partitions composed of some

[4]Since $|K|/m$ is not necessarily a natural number, we allow $m \leq |K| \leq m + 1$. In practice, we want partitions whose teams differ in size by at most one student.

teams that perform well and some other teams that perform badly. Therefore, we target at partitions whose teams display homogeneous behaviours (similar performances). This leads to the definition of a measure for the overall performance of a partition that prefers homogeneous teams. Thus, our definition below defines the synergistic value of a team partition as the Bernoulli-Nash product of the teams' synergistic values. In this way, this function ensures that we give larger values to *fair* partitions [48] (containing homogeneous teams), unlike other functions like, e.g., the addition.

Definition 14. *Given a team partition P_m and task type τ, the synergistic value of P_m is*

$$S(P_m, \tau) = \prod_{K \in P_m} s(K, \tau) \tag{8}$$

Now we are ready to formally define the Synergistic Team Composition Problem (STCP) as the problem of finding the partition with the largest synergistic value.

Definition 15. *Given a set of students A and task type τ, the* synergistic team composition problem (STCP) *is the problem of finding a team partition constrained by size m, $P_m^* \in \mathcal{P}_m(A)$, that maximises $S(P_m, \tau)$, namely*

$$P_m^* = \arg\max_{P_m \in \mathcal{P}_m(A)} S(P_m, \tau).$$

5. A complete algorithm for the STCP

We now propose a complete algorithm to solve the STCP. As a first step, in Section 5.1 we show how we linearise the problem, allowing us to model the STCP as an ILP. Then, in Section 5.2 we detail a complete algorithm for the STCP that solves such an ILP.

5.1. Linearising the STCP

Given a set A of n students, a task t of type $\langle \tau, m \rangle$, we define the total number of teams $b = \lfloor n/m \rfloor$. Depending on the cardinality of A and the desired team size m, the number of students in each team may vary. Let $Q(n, m)$ denote a set of couples such that each $\langle x, y \rangle \in Q(n, m)$ indicates that we consider x teams of size y.[5] We refer to $Q(n, m)$ as the *quantity distribution of team sizes*.

Let $K_1, \ldots, K_q$ denote the complete set of feasible teams that can be generated on the basis of the students from A, and $s(K_1, \tau), \ldots, s(K_q, \tau)$ their synergistic values given a task $t = \langle \tau, m \rangle$. Finally, let C be a matrix of size $n \times q$ such that $c_{ij} = 1$ if student a_i is part of team K_j, and $c_{ij} = 0$ otherwise.

For each team K_j ($j = 1, \ldots, q$) we consider a binary decision variable x_j. The value of x_j indicates whether team K_j is selected or not as part of the optimal solution

---

[5] For simplicity, in our experiments in Section 7 we consider that the number of students is a multiple of the desired team size, i.e., $n \bmod m = 0$. In this case, $Q(n, m) = \{(b, m)\}$.

of the STCP. Solving the STCP, then, amounts to solving the following non-linear integer program:

$$\max \prod_{j=1}^{q} s(K_j, \tau)^{x_j} \tag{9}$$

subject to:

$$\sum_{j=1}^{q} x_j = b \tag{10}$$

$$\sum_{j=1}^{b} c_{ij} \cdot x_j = 1 \quad \forall 1 \leq i \leq n \tag{11}$$

$$x_j \in \{0, 1\} \quad 1 \leq j \leq q \tag{12}$$

Constraint 10 ensures that any valid solution consists of exactly b teams, whereas constraint 11 enforces that each student belongs to exactly one of the selected teams. Notice that the objective function (see Equation 9) is non-linear. Nevertheless, it is rather easy to linearise this objective function by maximising the logarithm of $\prod_{j=1}^{q} s(K_j, \tau)^{x_j}$ instead. Thus, solving the non-linear integer program above is equivalent to solving the following binary linear program:

$$\max \sum_{j=1}^{q} x_j \cdot log(s(K_j, \tau)) \tag{13}$$

subject to: equations 10, 11, and 12.

5.2. Solving the ILP model

Algorithm 1 presents the pseudo-code of our complete approach to solve the STCP by means of the above detailed ILP. First, we generate the input for this ILP (see lines 2 to 4). Specifically, line 2 generates all possible teams of size m as determined by the quantity distribution $Q(|A|, m)$. The best synergistic values of these teams are computed in lines 3 and 4. This involves solving an optimisation problem, as discussed at the end of Section 4.1. We then generate the ILP according to equation 13 and solve it with the aid of an off-the-shelf ILP solver such as, for example, CPLEX, Gurobi, or GLPK. If given sufficient time, the algorithm returns an optimal solution (that is, an optimal team partition) together with the competence assignments (line 7).
We remark that generating the input for STCPSolver takes linear time with respect to the number of feasible teams q, which grows rapidly with increasing m and n.

6. A heuristic algorithm for the STCP

In this section we present *SynTeam*, an algorithm based on local search. The pseudo-code of SynTeam is provided in Algorithm 2. SynTeam starts by generating an initial solution/partition (line 1). This is done by randomly ordering the set of students

Algorithm 1 STCPSolver

Require: A ▷ The set of students
Require: $t = \langle \tau, m \rangle$ ▷ Task
Ensure: $(P, \boldsymbol{\eta}^*)$ ▷ Best partition found and best assignments
$P \leftarrow \emptyset$
$[K_1, \ldots, K_q] \leftarrow GenerateTeams(A, Q(|A|, m))$
for $i \in [1..q]$ **do**
    $(s(K_i, \tau), \eta^i_\tau(K_i, \tau)) \leftarrow getBestSynergisticValue(K_i, t)$
$ILP \leftarrow generateILP([K_1, \ldots, K_q], [s(K_1, \tau), \ldots, s(K_q, \tau)], b)$
$P \leftarrow solve(ILP)$
return $(P, \{\eta^i_\tau(K_i, \tau)\}_{K_i \in P})$

and assigning them, one after the other, in this order, to a number of teams whose sizes are determined by $Q(|A|, m)$; see Section 5.1 for the definition of $Q(.,.)$. This initial solution is denoted by $(P, S(P, \tau), \boldsymbol{\eta})$, where $\boldsymbol{\eta}$ is the vector of balanced competence assignments used to compute the proficiency degrees of the teams in P. The assignment of students to competencies is done as described in Section 4.1. The main part of the algorithm consists in a local search procedure which makes use of two different neighbourhoods. This first one, which is applied by default, consists in randomly selecting two teams from the current solution. Then, the set of students contained in these two teams is redistributed in the optimal way into two (possibly new) teams and the resulting solution, together with the corresponding competence assignments, is stored in $(P', S(P', \tau), \boldsymbol{\eta}')$; see line 4. In addition, whenever the algorithm detects that n_l not necessarily consecutive, non-improving iteration were performed, the second—more fine-grained—neighbourhood is applied to the current solution $(P, \boldsymbol{\eta})$ in the following way in line 6 of Algorithm 2.[6] The second neighbourhood tries to identify—in ascending order determined by team and student indexes—two students from different teams whose swap results in an improved solution. The first improved solution that is found in this way (if any) is stored as $(P', \boldsymbol{\eta}')$. Moreover, counter c_l regarding the non-consecutive non-improving iterations is re-initialized. The algorithm terminates after a number of n_r consecutive non-improving iterations.

7. Computational Results

In this section we conduct a comprehensive experimental evaluation in order to compare the two STCP solvers proposed in this work: (1) the optimal solver (STCPSolver), and (2) the SynTeam solver which is based on local search. In particular, we compare the two approaches regarding their run-times, as team sizes and the number of students increase. Moreover, we study the quality of the solutions provided by SynTeam in comparison to the optimal solutions. Finally, we also examine the anytime performance of SynTeam with respect to STCPSolver.

[6] Note that an iteration is called *improving* in case the current solution is improved in line 4 of Algorithm 2.

Algorithm 2 SynTeam

Require: A ▷ List of students
Require: $t = \langle \tau, m \rangle$ ▷ Task
Require: n_r ▷ Max. number of consecutive non-improving iterations
Require: n_l ▷ Number of non-improving iterations before student-swap
Ensure: $(P, \boldsymbol{\eta})$ ▷ Best solution found
$(P, S(P,\tau), \boldsymbol{\eta}) \leftarrow \mathit{GenerateRandomSolution}(A, Q(|A|, m))$
$c_r \leftarrow 1, c_l \leftarrow 1$
while $c_r \leq n_r$ **do**
  $(P', S(P',\tau), \boldsymbol{\eta}') \leftarrow \mathit{GenerateNeighbour}(P, \boldsymbol{\eta})$
  if $S(P',\tau) \leq S(P,\tau)$ **and** $c_l = n_l$ **then**
    $(P', S(P',\tau), \boldsymbol{\eta}') \leftarrow \mathit{ApplyImprovingSwap}(P, \boldsymbol{\eta})$
    $c_l \leftarrow 1$
  if $S(P',\tau) > S(P,\tau)$ **then**
    $(P, S(P,\tau), \boldsymbol{\eta}) \leftarrow (P', S(P',\tau), \boldsymbol{\eta}')$
    $c_r \leftarrow 1, c_l \leftarrow 1$
  else
    $c_r \leftarrow c_r + 1, c_l \leftarrow c_l + 1$
return $(P, \boldsymbol{\eta})$

7.1. Computational Scenario

The empirical evaluation is done within the following scenario:

- **ILP Solver.** CPLEX Optimization Studio v12.7.1 [49] was used for solving the ILPs generated by STCPSolver.
- **Students.** Actual-world data, in terms of 210 students, each one identified by an ID, gender information, the personality profile, and the competence levels regarding seven competencies, was used.
- **Classroom size.** The total number of students (n) in a classroom ranges from 10 to 100.
- **Task type.** In contrast to our preliminary paper [50], we consulted with professionals from the educational sector in order to define four diverse task types. These task types, together with their required competencies and importance levels, are provided in Table 1. As usual in educational contexts, these task types require a subset of seven competencies that directly stem from Gardner's multiple intelligences [51], which is widely-used in education scenarios: LINGUISTIC, LOGIC_MATHEMATICS, VISUAL_SPATIAL, BODILY_KINESTHETIC, MUSICAL, INTRAPERSONAL, and INTERPERSONAL. Notice that requirement levels and importance levels were set by educators to *qualitative* values to ease their specification. We employed five qualitative values for requirement levels (*fundamental awareness*, *novice*, *intermediate*, *advanced*, *expert*) and five qualitative

values for importance degrees (*unimportant*, *slightly important*, *important*, *fairly important*, *very important*). Thereafter, we evenly mapped the qualitative labels in Table 1 to quantitative values within the $[0, 1]$ interval.

- **Task.** The team size (m) ranged from 2 to 6. On the one hand, this limitation comes from the fact that larger team sizes are rather rare in an educational context (see e.g. [52, 53, 54])[7] On the other hand, the computational burden for STCPSolver when handling team sizes larger than 6 was too high, making comparisons with optimal solutions too costly.

- **Team proficiency.** An intermediate value of $\nu = 0.5$ was used for all computational tests, because the specific team proficiency value is rather irrelevant for the study of the algorithm properties.

- **Team Congeniality.** With the aim of making the personality and gender requirements equally relevant, the importance values were set as follows: (1) $\alpha = 0.11$, (2) $\beta = 3 \cdot \alpha$, (3) $\gamma = 0.33$. Note that a detailed description of the meaning of these values can be found in Section 4.2.

- **Balance between Proficiency and Congeniality.** The value of parameter $\lambda \in [0, 1]$ determines the balance between team proficiency and team congeniality. The experiments are performed with $\lambda \in \{0.2, 0.5, 0.8\}$. However, for space reasons we do not report on the results for $\lambda = 0.5$. They can be obtained from the authors upon request.

- **Number of iterations without improvement (n_r).** This value was fixed to $1.5 \cdot b$, with the following rationale behind. The computation time requirements of SynTeam can be expected to relate strongly to the number of teams in a solution (b): the larger b, the higher the computation time requirements. Moreover, after studying the evolution of SynTeam over time, we decided to scale this value with 1.5.

- **Frequency of local search (n_l).** It was observed that, after performing $\approx \frac{n_r}{6}$ applications of the first neighbourhood without improvement, the probability of finding an improvement in this way was very low. Hence, the value of n_l was set to $\frac{n_r}{6}$.

7.2. Computational Results

The experiments were performed on a cluster of machines with Intel® Xeon® CPU 5670 CPUs with 12 cores of 2933 MHz and a minimum of 40 Gigabytes of RAM. Moreover, IBM ILOG CPLEX v12.7.1 was used within STCPSolver and SynTeam.

[7] In fact, this actual-world constraint is in line with studies in the organisation psychology literature showing an inverse relationship between the size of a team and its performance [55, 56, 57, 58]. [56] observes that in the case of a team containing more than six people there is a tendency to split the team into two, which brings about negative effects. The cause is twofold: high coordination costs and loss of motivation by team members.

Table 1: Specification of the four task types, designed by education professionals.

(a) Task type `body rythm`

Competence	Req. level	Importance
BODILY_KINESTHETIC	advanced	very important
MUSICAL	intermediate	fairly important
LINGUISTIC	intermediate	slightly important
INTERPERSONAL	advanced	very important
VISUAL_SPATIAL	novice	slightly important

(b) Task type `entrepreneur`

Competence	Req. level	Importance
LINGUISTIC	advanced	fairly important
LOGIC_MATHEMATICS	intermediate	very important
VISUAL_SPATIAL	novice	slightly important
MUSICAL	novice	slightly important
INTERPERSONAL	advanced	very important
INTRAPERSONAL	intermediate	important

(c) Task type `arts design`

Competence	Req. level	Importance
LINGUISTIC	novice	slightly important
VISUAL_SPATIAL	advanced	very important
INTRAPERSONAL	intermediate	fairly important

(d) Task type `English`

Competence	Req. level	Importance
LINGUISTIC	intermediate	very important
INTRAPERSONAL	novice	important
INTERPERSONAL	advanced	very important

Remember, in this context, that CPLEX is used internally by SynTeam for the calculation of the optimal assignment of students to tasks for any given team and task type. CPLEX was run, in all cases, in one-threaded mode for ensuring a fair comparison.

In order to generate problem instances (classrooms) we considered the following parameters: task type, balance between proficiency and congeniality (λ), total number of students (n), and team size (m). For each combination of these parameters, we generate 20 problem instances, each one by randomly selecting n students out of the set of 210 available students. For instance, when considering team size 2, we have 46 different values for n, 3 values for λ, and 4 task types. Thus, we generate $11040 = 46 \cdot 3 \cdot 4 \cdot 20$ different problem instances. Overall, considering all team sizes and parameters' values, we generate 31[illegible]40 problem instances. Then, we solved all these problem instances with both STCPSolver and SynTeam. The results will be shown as averages over the 20 random sets of students.

Runtime Analysis. The graphics in Figure 1 show the performance of the algorithms in terms of the total running time (in seconds). Each graphic is dedicated to the results concerning one of the task types and one of the two considered settings for λ. Each data point represents the average over 20 random sets of students of size n. The total running time of STCPSolver consists of two components: the data generation time and the computation time. Hereby, the data generation time measures the time for generating all possible teams and calculating their synergistic value (lines 1-5 in Algorithm 1), while the computation time measures the time needed by CPLEX to load the corresponding ILP model and solve it to optimality. The graphics in Figure 1 show that, as the team size (m) increases, the total time needed by STCPSolver becomes prohibitively costly. In fact, the whole range of results could only be generated up to a team size of $m = 4$. For $m = 5$ the calculations were stopped after a total number of 60 students, and for $m = 6$ after 42 students. This was done because for larger values of n and m, the size of the corresponding ILP models was simply too large for CPLEX.[8] In general, it can be observed that the runtime of STCPSolver dramatically increases with the number of students (n) and team size (m). Note that for a team size

[8]For instance, the ILP model for $n = 48$ and $m = 6$ consists of 12.271.512 binary variables.

of $m = 6$ and for $n = 42$ students, SynTeam is at least two orders of magnitude faster than STCPSolver. On the other side, for $m = 2$ the total time requirements of the two techniques is nearly equal.

To better understand this result, we compared the computation times of STCPSolver (disregarding its data generation time) with the computation times of SynTeam. Figure 2 provides this comparison in a graphical way. It can be observed that —even in this case— SynTeam is more efficient for larger instances, that is, for team sizes of $m > 3$ and a growing number of students.

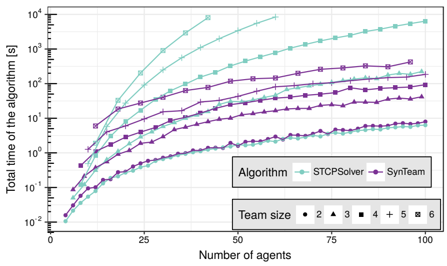
Quality Analysis. The relative quality of the best solutions provided by SynTeam are shown in Figure 3. Note that the relative quality of SynTeam is obtained by dividing the value of the best solution provided by SynTeam by the value of an optimal solution calculated by STCPSolver. The following observations can be made. First, in all cases the relative quality of SynTeam seems to decrease with decreasing team size. However, even in the case of the smallest considered team size ($m = 2$), the quality ratio of SynTeam is always above 95% in the context of the experiments with $\lambda = 0.8$, and above 75% in the case of $\lambda = 0.2$. The apparent difference in the quality ratios between $\lambda = 0.2$ and $\lambda = 0.8$ can be explained as follows. After an in-depth study, we noticed that the proficiency values of the different teams are nearly all from the range $[0.9, 1.0]$, while the congeniality values are spread over the whole range $[0, 1]$. Therefore, when preferring proficiency over congeniality ($\lambda = 0.8$), there is a large concentration of teams with synergistic values of a rather high relative quality. This is not the case when $\lambda = 0.2$.

Anytime performance. We also decided to show examples of the anytime performance of the two different algorithms. For that purpose, we chose for each task type a different exemplary combination of n and m. The corresponding graphics can be seen in Figure 4. Note that the considered combinations of n and m are detailed in the figure caption. Moreover, they do not include the data generation time of STCPSolver.

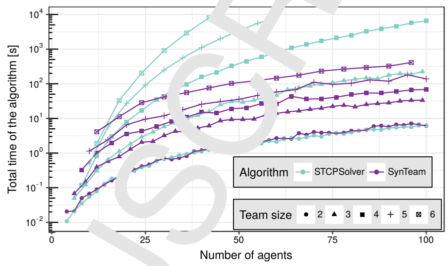
Figure 4 clearly shows that the anytime performance of SynTeam is superior for team sizes with $m > 3$. In Figure 4a, for example, observe that SynTeam provides very good solutions after approximately 11 seconds, while STCPSolver needs approximately 21 seconds (in addition to more than 1000 seconds of data generation time) to come up with a first, low-quality solution. However, when team sizes are small (as in the case of Figure 4e and 4f), the anytime performance of STCPSolver is better.

Comparison to ODP-IP. Furthermore, we compared SynTeam to ODP-IP [13], the state-of-the-art coalition structure generation algorithm that can be employed to solve the STCP, as discussed in Section 2.4. We employed it to solve all cases in which this was possible, that is, to all cases with $n \leq 25$.[9] Figure 5 shows a comparison of the average computation times of STCPSolver and ODP-IP, exemplary for task `arts`

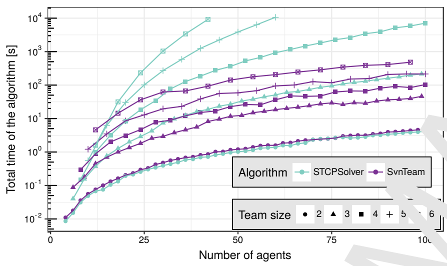
[9]Note that, due to exponential memory requirements, the ODP-IP algorithm can not be applied to cases with more than 25 students.



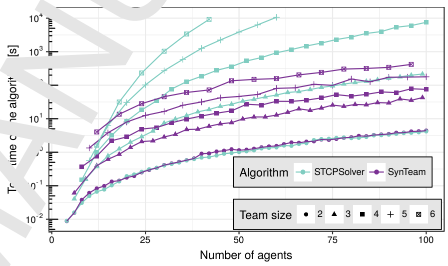
(a) body rythm, $\lambda = 0.2$.



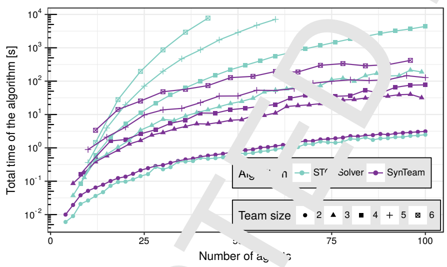
(b) body rythm, $\lambda = 0.8$.



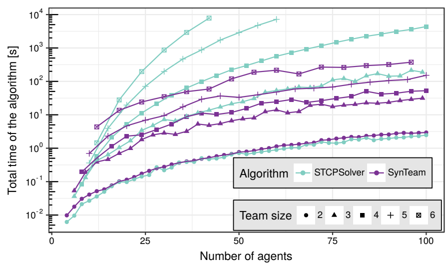
(c) entrepreneur, $\lambda = 0.2$.



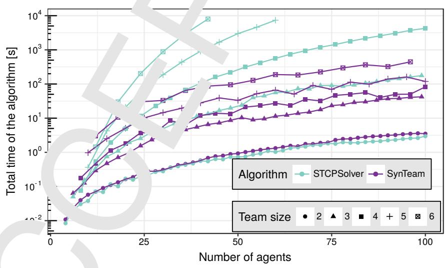
(d) entrepreneur, $\lambda = 0.8$.



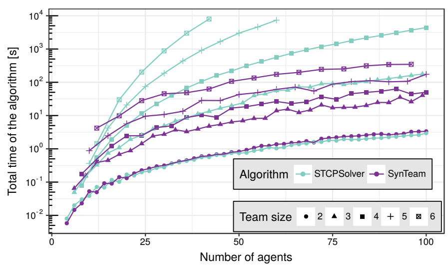
(e) arts design, $\lambda = 0.2$.



(f) arts design, $\lambda = 0.8$.

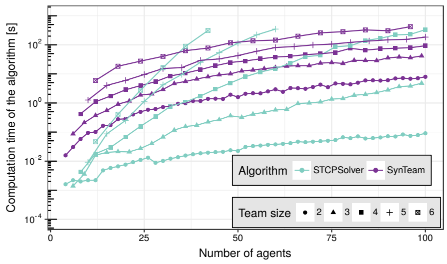


(g) English, $\lambda = 0.2$.

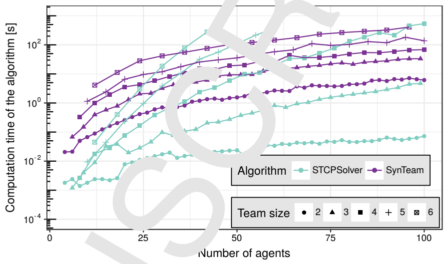


(h) English, $\lambda = 0.8$.

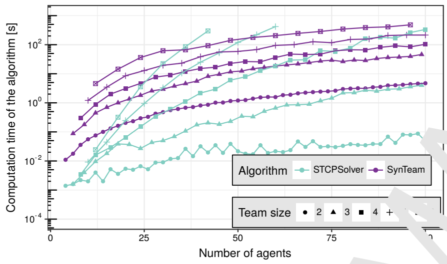
Figure 1: Total time needed by the two techniques.



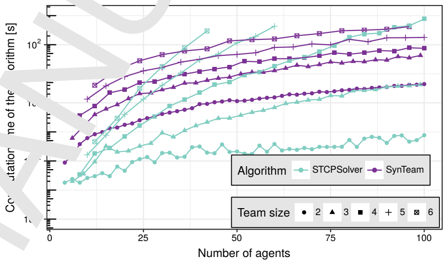
(a) `body rythm`, $\lambda = 0.2$.



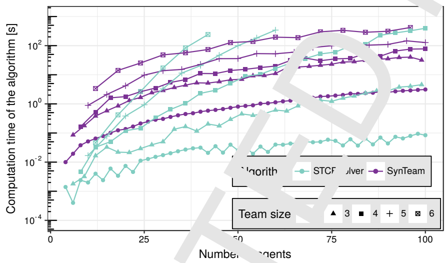
(b) `body rythm`, $\lambda = 0.8$.



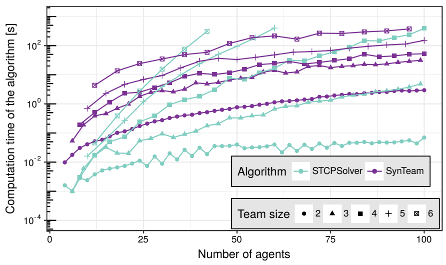
(c) `entrepreneur`, $\lambda = 0.2$.



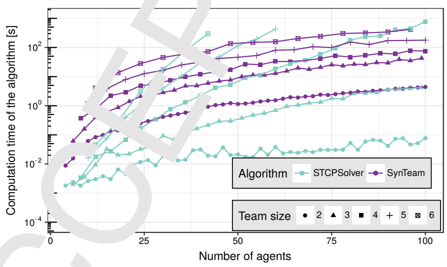
(d) `entrepreneur`, $\lambda = 0.8$.



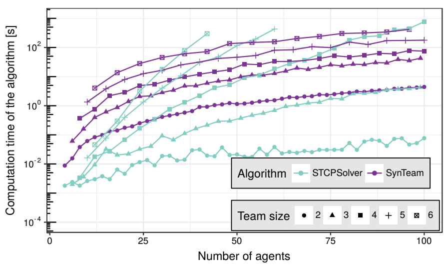
(e) `arts design`, $\lambda = 0.2$.



(f) `arts design`, $\lambda = 0.8$.

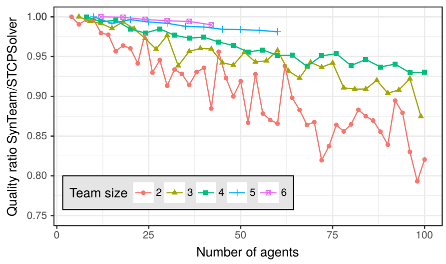


(g) `English`, $\lambda = 0.2$.

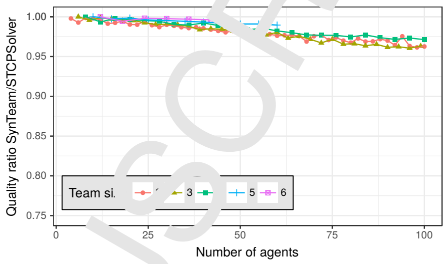


(h) `English`, $\lambda = 0.8$.

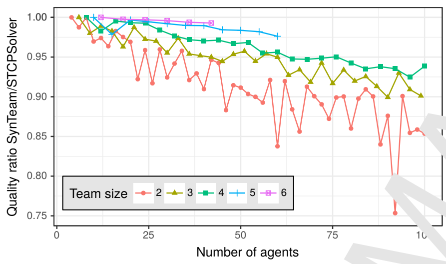
Figure 2: Computation time needed by the two techniques (disregarding the data generation time of STCPSolver).



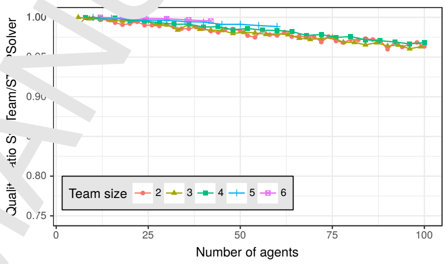
(a) `body rythm`, $\lambda = 0.2$.



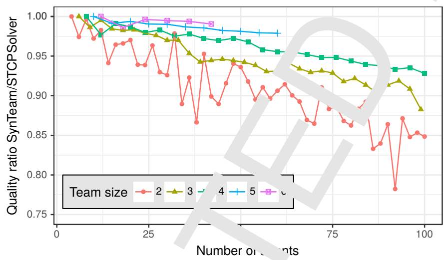
(b) `body rythm`, $\lambda = 0.8$.



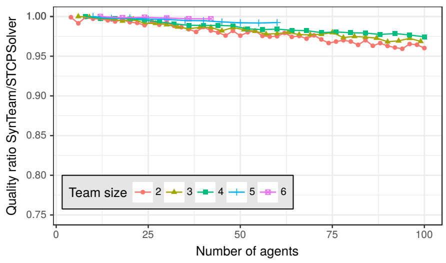
(c) `entrepreneur`, $\lambda = 0.2$.



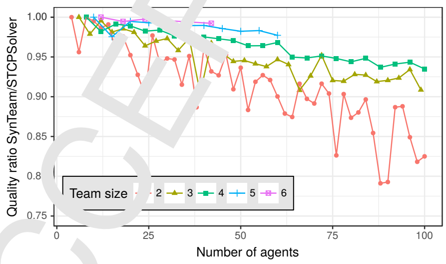
(d) `entrepreneur`, $\lambda = 0.8$.



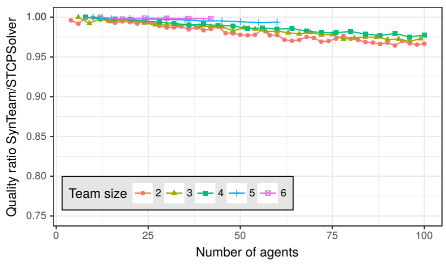
(e) `arts design`, $\lambda = 0.2$.



(f) `arts design`, $\lambda = 0.8$.

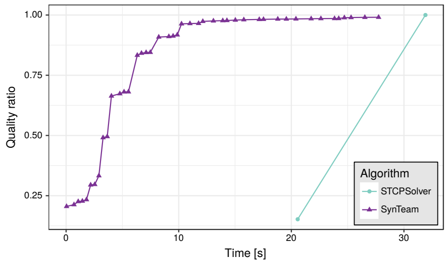


(g) `English`, $\lambda = 0.2$.

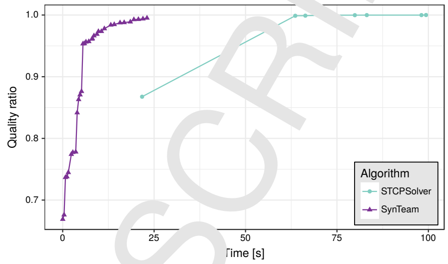


(h) `English`, $\lambda = 0.8$.

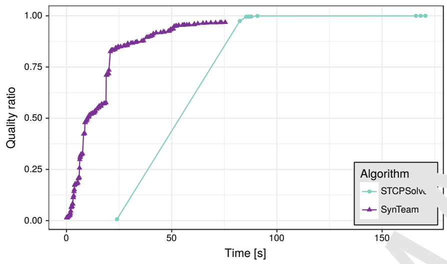
Figure 3: SynTeam relative quality plots.



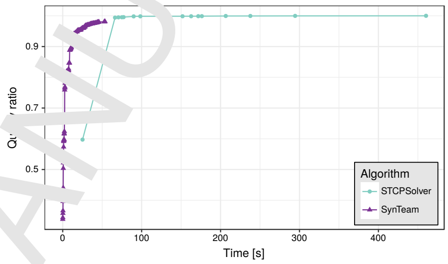
(a) body rythm, $\lambda = 0.2$.



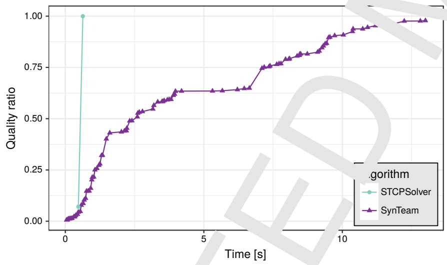
(b) body rythm, $\lambda = 0.8$.



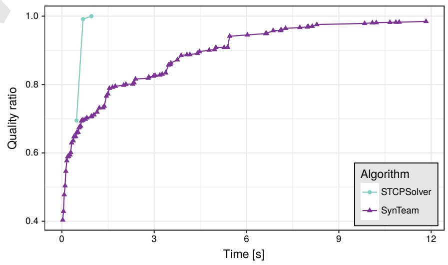
(c) entrepreneur, $\lambda = 0.2$.



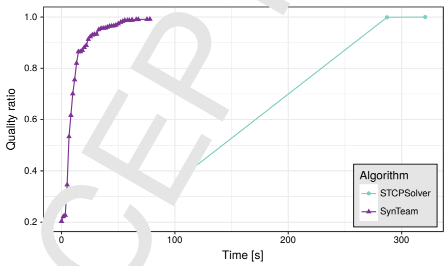
(d) entrepreneur, $\lambda = 0.8$.



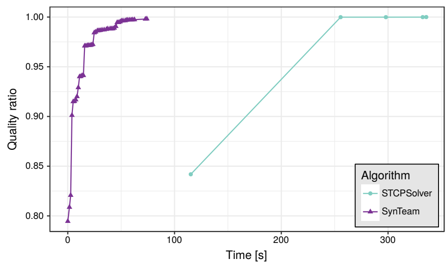
(e) arts design, $\lambda = 0.2$.



(f) arts design, $\lambda = 0.8$.

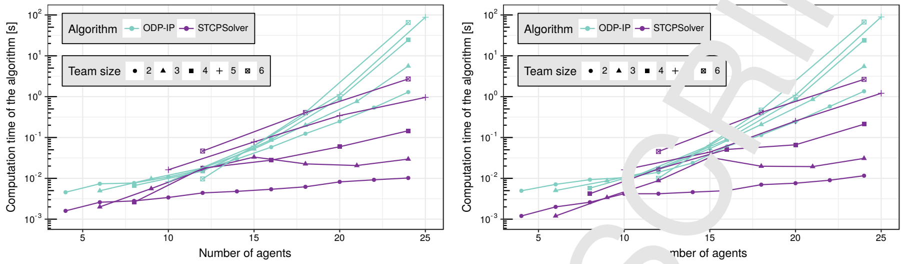


(g) English, $\lambda = 0.2$



(h) English, $\lambda = 0.8$

Figure 4: Anytime performance of SynTeam vs. STCPSolver. The curves show the quality ratio (with respect to the optimal solutions). Graphics (a) and (b) are for 45 students and a team size of [illegible]. Graphics (c) and (d) are for 80 students and a team size of 4. Graphics (e) and (f) are for [illegible]0 students and a team size of 3. Finally, graphics (g) and (h) are for 42 students and a team size of 6.



(a) `arts design`, $\lambda = 0.2$.

(b) `arts design`, $\lambda = 0.8$.

Figure 5: Comparison of the computation times (disregarding the data generation time) between STCPSolver and ODP-IP in the context of task `arts design`. Note that ODP-IP is limited to a total number of 25 students.

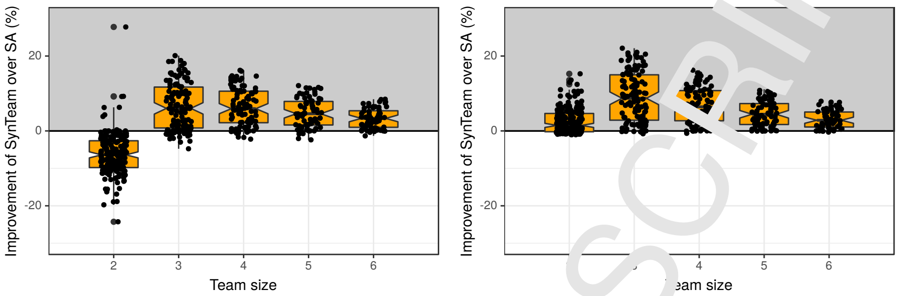
`design`. The main observation is that, even though ODP-IP seems to be slightly faster than STCPSolver for team sizes greater than 2 when n is small, with a growing value of n the computation time requirements of ODP-IP increase much more raplidly than those of STCPSolver. In fact, for the cases with $n \geq 24$, the computation time of ODP-IP is about two orders of magnitude higher than the one of STCPSolver. Moreover, the data generation time for ODP-IP, which is not taken into account in these graphics, is longer than for ODP-IP, due to the fact that *every* subset of students (even unfeasible teams of size different from m) must be visited in the generation phase (see Section 2.4).

Comparison to a baseline metaheuristic: Simulated Annealing. Finally, we compared SynTeam to a standard implementation of simulated annealing (SA) [59], which is one of the baseline metaheuristics. The algorithm starts from a random initial solution. At each step, a random neighbour of the current solution is chosen by (1) randomly selecting two agents from two different teams, and by (2) swapping the two agents. This neighbouring solution is accepted as the new current solution (1) if it is at least as good as the current solution, or (2) with probability

$$P_{accept} = \exp\left(-\frac{\Delta}{T}\right) = \exp\left(-\frac{(S(P',\tau) - S(P,\tau))/S(P',\tau)}{T}\right), \tag{14}$$

where (P,τ) denotes the current solution, (P',τ) denotes the neighbouring solution, Δ is the percent change in synergistic value, and T is an important control parameter called *temperature*. At the beginning of the search process, the value of T is relatively large so that even worsening moves are frequently accepted. During the optimization process, the temperature is gradually decreased so that fewer and fewer worsening moves are accepted.

We adopt the scheme for setting the value of T at each iteration proposed by Sampels et al. [60]. The initial temperature is set such that the probability to accept a move with $\Delta = \delta = 0.01$ is $P_{start} = 0.9$. Moreover, at the end of the optimization process, we aim for a probability to accept a move with $\Delta = \delta = 0.01$ of $P_{end} = 0.1$. With these requirements, we define the temperature value at time x to be $T := r^x \tau_{\max}$,



(a) $\lambda = 0.2$. (b) $\lambda = 0.8$.

Figure 6: The boxplots show the improvement (in percent) of SynTeam over SA. Each box contains all comparisons concerning all different considered numbers of students (n) and all four tasks. Note that points in the area below zero (y-axis) indiacate that SA performed better than SynTeam.

where $\tau_{\max} := -\delta/\ln P_{start}$, $r := \sqrt[t_{\max}]{\delta(\ln(1/P_{end})\cdot\tau_{\max})}$, and where $t_{\max}$ is the computation time limit of the run.

SA was applied to all problems to which SynTeam was applied. Moreover, the computation time limit given to SA was the total time required by SynTeam in each case. The results of the comparison between SynTeam and SA are presented in Figure 6a (for $\lambda = 0.2$) and Figure 6b (for $\lambda = 0.8$) in terms of boxplots. The data points show, for each team size, the improvement of SynTeam over SA (note that these graphics summarize over all numbers of students (n) and all four tasks). Hereby, values below zero (area with white background) indicate that SA performed better than SynTeam. The following conclusions can be drawn: first, there is just one single case ($\lambda = 0.2$, team size 2) in which SA is generally better than SynTeam. In all other cases, SynTeam outperforms SA with statistical significance (as indicated by the notches of the boxplots). Interestingly, the relative performance of SynTeam with respect to SA is better with a higher value of λ.

Summarizing, we would like to remark that we were not able to observe significant differences in (relative) algorithm performance when comparing between the different task types. This is a strong indication for the robustness of the developed techniques. Moreover, we obtained a confirmation of the well known principle that *no algorithm is the best-performing one in all possible cases*. More specifically, when team sizes of $m \leq 3$ are concerned, we recommend the use of STCPSolver, while when $m > 3$ the use of SynTeam is indicated.

8. EduTeams: a publicly-available application to compose teams

We developed a web application, EduTeams, which is publicly available.[10]

[10] `http://eduteams.iiia.csic.es`

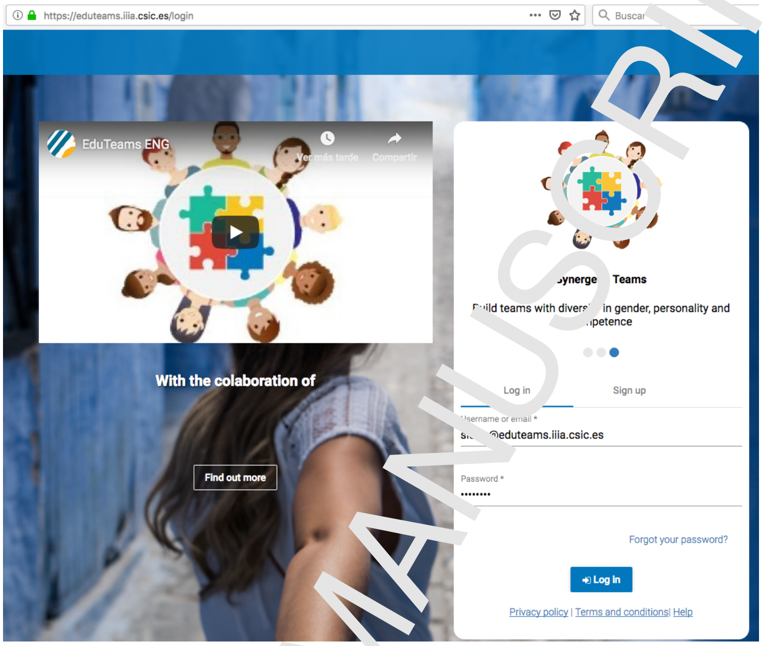


Figure 7: EduTeams user registration page.

EduTeams allows teachers to employ SynTeam to partition their classrooms into synergistic teams to perform collaborative tasks requiring multiple competences. In short, after signing up in EduTeams, a teacher can create her own classrooms. For a classroom, the teacher can specify tasks using the competences, competence requirements and importance levels specified in section 7.1. Several examples of task definitions are shown in Table 1. Every classroom has a unique code that students need to set up their own accounts and thus join classrooms. After signing up, each student is asked to fill in one competence and one personality test. Once all students in a classroom complete both tests, the teacher owning the classroom can proceed with team composition. Figures 7 and 8 display a couple of snapshots of EduTeams using artificial data so that no privacy is breached.

9. Conclusions and future work

In this paper, we defined the Synergistic Team Composition Problem (STCP) in the domain of student team composition. We introduced two different solutions to solve our problem, that is, to partition students' classrooms into teams that are balanced in size, competences, personality and gender. First, we proposed an algorithm called STCPSolver to optimally solve our problem. Second, we proposed an algorithm called SynTeam, a heuristic that yields close to optimum, but not necessarily optimal solutions. Our computational results show that the benefits of SynTeam with respect to STCPSolver grow with the increasing number of students and team sizes. Moreover,

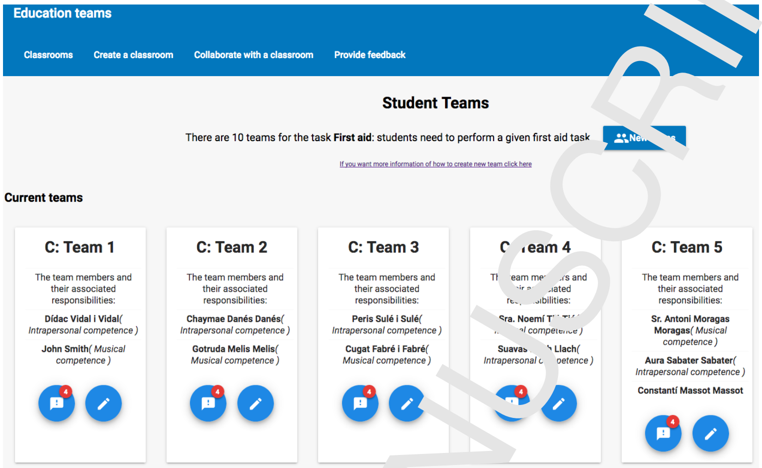


Figure 8: Teams as shown in EduTeams.

SynTeam gives good quality approximate solutions depending on the trade-off between proficiency and congeniality. Thus, even in the case of the smallest considered team size ($m = 2$), the quality ratio of SynTeam is always above 95% when preferring proficiency over congeniality ($\lambda = 0.8$), and above 75% when preferring congeniality over proficiency ($\lambda = 0.2$).

This paper identified and formalised an interesting real-world case as a new type of constrained coalition formation problem. This case requires a *balanced* coalition structure in terms of both coalitional values and coalition sizes. The computational comparison of both algorithms offers the guidance for their use by any institution that is in need for automatic team composition (e.g. classrooms, research units, private companies). Note that the algorithms compose teams in a completely automated way without experts knowledge, which is a tremendous advantage for settings where there are no guidelines or expertise available.

The STCP problem opens new research paths. First, there is the need for considering more general and richer models to better express the different determinants of the team performance. For instance, we plan to extend our approach so as to consider preferences and constraints coming from human experts (e.g., conflicts of interests). Furthermore, we plan to extend our STCP model to deal with multiple task types, along the lines of the work by Präntare and Heintz [61]. Finally, we aim at investigating how to exploit paralellism so that our search process can benefit from the capabilities offered by modern computer architectures.

Acknowledgements

This work was supported by projects CIMBVAL (MINECO, TIN2017-89758-R), AppPhil (RecerCaixa 2017), 2017 SGR 172, Collectiveware (MINECO/FEDER, TIN2015-66863-C2-1-R) and AI4EU (H2020-825619). Bistaffa was supported by the H2020-MSCA-IF-2016 HPA4CF project. The research was partially supported by the ST

Engineering - NTU corporate Lab through the NRF corporate lab@university scheme. The first author thanks Enzyme Advising Group for partial sponsorship of the Industrial PhD programme to develop ideas introduced in this paper.

References

[1] S. Vosniadou, How children learn?, in: D. Bhaskara Rao (Ed.), Successful Schooling, Discovery Publishing House, New Delhi, India, 2003, Ch. 2, pp. 16–33.

[2] E. F. Barkley, K. P. Cross, C. H. Major, Collaborative learning techniques: A handbook for college faculty, John Wiley & Sons, 2014.

[3] J. Arnold, R. Randall, Work psychology, Pearson Education Limited, Harlow, England, 2010.

[4] M. Mount, M. Barrick, G. Stewart, Five-factor model of personality and performance in jobs involving interpersonal interactions, Human Performance 11 (2-3) (1998) 145–165.

[5] M. A. West, Effective Teamwork: Practical Lessons Learned from Organizational Research, Wiley-Blackwell, West Sussex, 2012.

[6] K. White, Mis project teams: An investigation of cognitive style implications, MIS Quarterly 8 (2) (1984) 95–101.

[7] J. M. Alberola, E. Del Val, V. Sanchez-Anguix, A. Palomares, M. D. Teruel, An artificial intelligence tool for heterogeneous team formation in the classroom, Knowledge-Based Systems 101 (2016) 1–14.

[8] B. Chen, X. Chen, A. Timsina, L. Soh, Considering agent and task openness in ad hoc team formation, in: Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2015, Istanbul, Turkey, May 4-8, 2015, 2015, pp. 1861–1862.

[9] M. Farhangian, M. K. Purvis, M. Purvis, B. T. R. Savarimuthu, Modeling the effect of personality on team formation in self-assembly teams, in: PRIMA 2015: Principles and Practice of Multi-Agent Systems - 18th International Conference, Bertinoro, Italy, October 26-30, 2015, Proceedings, 2015, pp. 538–546.

[10] T. Okimoto, N. Schwind, M. Clement, T. Ribeiro, K. Inoue, P. Marquis, How to form a task-oriented robust team, in: Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems, AAMAS '15, International Foundation for Autonomous Agents and Multiagent Systems, 2015, pp. 395–403.

[11] S. S. Rangapuram, T. Bühler, M. Hein, Towards realistic team formation in social networks based on densest subgraphs, CoRR abs/1505.06661.

[12] S. T. Acuña, M. Gómez, N. Juristo, How do personality, team processes and task characteristics relate to job satisfaction and software quality?, Information and Software Technology 51 (3) (2009) 627–639.

[13] T. Michalak, T. Rahwan, E. Elkind, M. Wooldridge, N. R. Jennings, A hybrid exact algorithm for complete set partitioning, Artificial Intelligence 230 (2016) 14–50.

[14] D. Wilde, Post-Jungian Personality Theory for Individuals and Teams, SYDROSE LP, 2013.

[15] L. K. Michaelsen, D. X. Parmelee, K. K. McMahon, R. E. Levine, D. M. Billings, Team-based learning for health professions education: A guide to using small groups for improving learning, The Journal of Chiropractic Education 23 (1) (2009) 47–48.

[16] R. Cutshall, S. Gavirneni, K. Schultz, Indiana university's kelley school of business uses integer programming to form equitable, cohesive student teams, Interfaces 37 (3) (2007) 265–276.

[17] R. G. Dear, Z. Drezner, Applying combinatorial optimization metaheuristics to the golf scramble problem, International Transactions in Operational Research 7 (4-5) (2000) 331–347.

[18] J. Desrosiers, N. Mladenović, D. Villeneuve, Design of balanced mba student teams, Journal of the Operational Research Society 56 (1) (2005) 60–66.

[19] R. A. Layton, M. L. Loughry, M. W. Ohland, G. D. Ricco, Design and validation of a web-based system for assigning members to teams using instructor-specified criteria., Advances in Engineering Education 2 (1) (2010) n1.

[20] P. A. Rubin, L. Bai, Forming competitively balanced teams, IIE Transactions 47 (6) (2015) 620–633.

[21] R. Agrawal, B. Golshan, E. Terzi, Grouping students in educational settings, in: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), acm sigkdd international conference on knowledge discovery and data mining (kdd) Edition, ACM, 2014.
URL `https://www.microsoft.com/en-us/research/publication/grouping-students-in-educational-settings/`

[22] T. R. Henry, Forming productive student groups using a massively parallel brute-force algorithm, in: Proceedings of the World Congress on Engineering and Computer Science, Vol. 1, 2013, pp. 23–25.

[23] D. J. Devine, J. L. Philips, Do smarter teams do better a meta-analysis of cognitive ability and team performance, Small Group Research 32 (5) (2001) 507–532.

[24] S. Bell, Deep-level composition variables as predictors of team performance: A meta-analysis, Journal of Applied Psychology 92 (3) (2007) 595–615.

[25] A. W. Woolley, I. Aggarwal, T. W. Malone, Collective intelligence and group performance, Current Directions in Psychological Science 24 (6) (2015) 420–424.

[26] A. W. Woolley, C. F. Chabris, A. Pentland, N. Hashmi, T. W. Malone, Evidence for a collective intelligence factor in the performance of human groups, science 330 (6004) (2010) 686–688.

[27] M. West, Effective teamwork: Practical lessons from organizational research, John Wiley and Sons, 2012.

[28] J. Costa, P.T., R. McCrae, Revised NEO Personality Inventory (NEO-PI-R) and NEO Five-Factor Inventory (NEO-FFI) manual, Odessa, FL: Psychological Assessment Resources, 1992.

[29] A. Aritzeta, S. Swailes, B. Senior, Belbin's team role model: Development, validity and applications for team building, Journal of Management Studies 44 (1) (2007) 96–118.

[30] I. Briggs, P. Myers, Gifts Differing: Understanding Personality Type, Mountain View, CA: Davies-Black Publishing, 1995.

[31] M. S. Prewett, A. A. Walvoord, F. R. Stilson, M. E. Rossi, M. T. Brannick, The team personality–team performance relationship revisited: The impact of criterion choice, pattern of workflow, and method of aggregation, Human Performance 22 (4) (2009) 273–296.

[32] M. A. Peeters, H. F. Van Tuijl, C. G. Rutte, I. M. Reymen, Personality and team performance: a meta-analysis, European Journal of Personality 20 (5) (2006) 377–396.

[33] S. Mohammed, L. C. Angell, Personality heterogeneity in teams: Which differences make a difference for team performance?, Small group research 34 (6) (2003) 651–677.

[34] B. Senior, Team roles and team performance: is there 'really'a link?, Journal of occupational and organizational psychology 70 (3) (1997) 241–258.

[35] D. Partington, H. Harris, Team role balance and team performance: an empirical study, Journal of Management Development 18 (8) (1999) 694–705.

[36] R. Batenburg, W. van Walbeek, W. in der Maur, Belbin role diversity and team performance: is there a relationship?, Journal of Management Development 32 (8) (2013) 901–913.

[37] H. van de Water, K. Ahaus, R. Rozier, Team roles, team balance and performance, Journal of Management Development 27 (5) (2008) 499–512.

[38] A. J. Devito, Review of myers-briggs type indicator, The ninth mental measurements yearbook 2 (1985) 1030–1032.

[39] M. Farhangian, M. Purvis, M. Purvis, T. B. R. Savarimuthu, Agent-based modeling of resource allocation in software projects based on personality and skill, in: Advances in Social Computing and Multiagent Systems, Springer, 2015, pp. 130–146.

[40] C. Crawford, Z. Rahaman, S. S., Evaluating the efficiency of robust team formation algorithms, International Workshop on Optimization in Multi-Agent Systems.

[41] A. Anagnostopoulos, L. Becchetti, C. Castillo, A. Gionis, S. Leonardi, Online team formation in social networks, in: Proceedings of the 21st international conference on World Wide Web, 2012, pp. 839–848.

[42] T. Rahwan, T. Michalak, E. Elkind, P. Faliszewski, J. Sroka, M. Wooldridge, N. Jennings, Constrained coalition formation, in: W. Burgard, D. Roth (Eds.), AAAI, Vol. 11, AAAI Press, 2011, pp. 719–725.

[43] D. J. Wilde, Teamology: The Construction and Organization of Effective Teams, Springer-Verlag, London, 2009.

[44] C. Jung, Psychological types, Princeton University Press, Princeton, 1921.

[45] R. Roe, Competences-a key towards the integration of theory and practice in work psychology, Gedrag en Organisatie 15 (4) (2002) 203–224.

[46] R. Bashur, A. Hernandez, J. Peiro, The impact of underemployment on individual and organizational performance, Underemployment: Psychological, Economic, and Social Challenges (2011) 187–213.

[47] J. H. Bradley, F. J. Hebert, The effect of personality type on team performance, Journal of Management Development 16 (1997) 337–353.

[48] J. Nash, The Bargaining Problem, Econometrica 18 (2) (1950) 155–162.

[49] IBM, Ibm ilog cplex optimization studio (2017).
URL https://www.ibm.com/us-en/marketplace/ibm-ilog-cplex

[50] E. Andrejczuk, F. Bistaffa, C. Blum, J. A. Rodríguez-Aguilar, C. Sierra, Heterogeneous teams for homogeneous performance, in: T. Miller, N. Oren, Y. Sakurai, I. Noda, B. T. R. Savarimuthu, T. C. Son (Eds.), Proceedings of PRIMA 2018 – 21st International Conference on Principles and Practice of Multi-Agent Systems, Vol. 11224 of Lecture Notes in Computer Science, Springer, 2018, pp. 89–105.

[51] H. Gardner, The theory of multiple intelligences, Annals of Dyslexia 37 (1) (1987) 19–35.

[52] X. S. Apedoe, M. R. Ellefson, C. D. Schunn, Learning together while designing: Does group size make a difference?, Journal of Science Education and Technology 21 (1) (2012) 83–94. doi:10.1007/s10956-011-9284-5.
URL https://doi.org/10.1007/s10956-011-9284-5

[53] P. Pujolàs, Atención a la diversidad y aprendizaje cooperativo en la educación obligatoria, Ediciones ALJIBE, 2001.

[54] P. Pujolàs, Aprendre junts alumnes diferents. Els equips d'aprenentatge cooperatiu a l'aula, Vic: Eumo Editorial, 2003.

[55] J. M. Levine, R. L. Moreland, Progress in small group research, Annual review of psychology 41 (1) (1990) 585–634.

[56] C. Oyster, Groups: A User's Guide, McGraw-Hill, Boston, 1999.

[57] K. Bartol, Building synergistic EDP teams, in: Proceedings of the fifteenth annual SIGCPR conference, ACM, 1977, pp. 18–30.

[58] I. C. Moon, K. M. Carley, M. Schneider, O. Shigiltchoff, Detailed analysis of team movement and communication affecting team performance in the america's army game, Working paperCASOS, Carnegie Mellon, Pittsburgh PA.

[59] S. Kirkpatrick, C. D. Gelatt, M. P. Vecchi, Optimization by simulated annealing, Science 220 (4598) (1983) 671–680.

[60] M. Sampels, C. Blum, M. Mastrolilli, O. Rossi-Doria, Metaheuristics for group shop scheduling, in: J. M. G. et al. (Ed.), Proceedings of PPSN-VII – Seventh International Conference on Parallel Problem Solving from Nature, no. 2439 in Lecture Notes in Computer Science, Springer, 2002, pp. 631–640.

[61] F. Präntare, F. Heintz, An anytime algorithm for simultaneous coalition structure generation and assignment, in: T. Miller, N. Oren, Y. Sakurai, I. Noda, B. T. R. Savarimuthu, T. Cao Son (Eds.), PRIMA 2018: Principles and Practice of Multi-Agent Systems, Springer International Publishing, Cham, 2018, pp. 158–174.

Declaration of Interest

We wish to confirm that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome.

We confirm that the manuscript has been read and approved by all named authors and that there are no other persons who satisfied the criteria for authorship but are not listed. We further confirm that the order of authors listed in the manuscript has been approved by all of us.

We confirm that we have given due consideration to the protection of intellectual property associated with this work and that there are no impediments to publication, including the timing of publication, with respect to intellectual property. In so doing we confirm that we have followed the regulations of our institutions concerning intellectual property.

We understand that the Corresponding Author is the sole contact for the Editorial process (including Editorial Manager and direct communications with the office). He/she is responsible for communicating with the other authors about progress, submissions of revisions and final approval of proofs. We confirm that we have provided a current, correct email address which is accessible by the Corresponding Author and which has been configured to accept email from any email address.

Sincerely,

Ewa Andrejczuk
Filippo Bistaffa
Christian Blum (corresponding author)
Juan Antonio Rodríguez Aguilar
Carles Sierra

Signature of the corresponding author, on behalf of all authors: