

Compatibility of Partnered Students in Computer Science Education

Joshua Sennett
University of Virginia
Department of Computer Science
Charlottesville, VA 22904
+1-434-982-2688
jas7wh@virginia.edu

Mark Sherriff
University of Virginia
Department of Computer Science
Charlottesville, VA 22904
+1-434-982-2688
sherriff@virginia.edu

ABSTRACT

This paper details the results of an investigation into the compatibility of partnered computer science students. The study involved approximately 290 students at the University of Virginia (UVA). This study builds on the work of researchers at North Carolina State University (NCSU). NCSU researchers have conducted a number of studies on the compatibility of pair programmers. We examined many of the factors that the NCSU researchers explored in their studies (including personality type, learning style, skill level, programming self esteem, work ethic, and time management choices) in order to determine whether the conclusions of the research at NCSU also hold true at UVA. Consistent with the NCSU studies, we found that skill level continues to be the most important factor in student compatibility.

Categories and Subject Descriptors

K.3.1 [Computers and Education]: *Collaborative learning*,

K.3.2 [Computer and Information Science Education]:
Computer science education

General Terms

Measurement, Experimentation, Human Factors

Keywords

Pair Programming, Computer Science Education, Pair Evaluation

1. INTRODUCTION

Pairing students together for group work has long been used in schools and classrooms as a pedagogical tool to increase students' comprehension of subject material. This strategy has been applied to computer science education in the form of "pair programming," wherein two students work together at one computer on programming assignments [3]. Research has shown that involving students in pair programming activities can lead to increased comprehension of the material and higher

success rates in the associated class [7]. Unfortunately, incompatibility between students can limit the method's effectiveness [7].

This problem has prompted researchers at North Carolina State University (NCSU) to explore the possible factors contributing to partner compatibility in computer science education [3, 4, 5, 7, 10]. These factors include personality type, learning style, skill level, programming self esteem, work ethic, and time management choices [10]. The goal of this study is to contribute to the body of knowledge on student compatibility. By analyzing many of the same factors examined in the NCSU studies we hope to determine whether the conclusions of the NCSU researchers also hold true in University of Virginia (UVA) computer science classes.

In conjunction with this study, we set out to design and implement a web-based tool to facilitate the use of groups and the collection of compatibility data. The tool used in UVA classes to gather data for this study is the "PairEval" tool made available by the NC State researchers. The tool we designed was based on PairEval and expands the original system's functionality. This new system allows professors to more easily add questions to the surveys posed to the students. It also simplifies the group forming process. These new features should make the use of group work and pair programming easier for computer science professors, as well as facilitate the collection of student compatibility data for use in future work.

In Section 2 we provide a discussion of the relevant background material and related work. Section 3 describes the details of our study and its methodology. The results of the study are presented in Section 4. In Section 5 we describe the additions we have made to the PairEval web tool. Section 6 presents our conclusions and possibilities for future work.

2. RELATED WORK

This section discusses the related work and background information for the Myers-Briggs personality types, the Felder-Silverman learning styles, pair programming, and student compatibility research.

2.1 Myers Briggs

The Myers-Briggs Type Indicator (MBTI) was developed by Isabel Briggs Myers and Katharine Briggs in the 1940s and first published in the MBTI Manual in 1962[6]. It aims to place Carl

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. *SIGCSE'10*, March 10–13, 2010, Milwaukee, Wisconsin, USA. Copyright 2010 ACM 978-1-60558-885-8/10/03...\$10.00.

Jung's theory of psychological types into a usable and understandable format. The Myers-Briggs personality test classifies a respondent into categories representing their personality traits in four different dimensions, or dichotomies, consisting of opposite personality qualities. For each of the four dimensions an individual receives a classification and a number (from 0 to 100) that indicates how strongly they respond to that classification. The four dimensions of the Myers-Briggs Type Indicator are Introvert-Extrovert, Sensing-Intuition, Thinking-Feeling, and Judging-Perceiving.

2.2 Felder-Silverman

In 1988 Felder and Silverman proposed a model to describe student learning styles that would be particularly relevant to engineering students [2]. Recent publications of the original paper include a 2002 preface by Felder, explaining a few changes made. The Felder-Silverman learning style model borrows from a number of other systems, including the Myers-Briggs personality type indicator and a learning style model developed by Kolb. Like Myers-Briggs, the Felder-Silverman learning styles model consists of four dimensions (although, it was originally five). Individuals are placed into a given classification for each dimension based on their responses to a number of questions. The scale for each dimension ranges from -11 to 11, with values given at the odd integers (-11, -9, -7, etc.). The four dimensions of the Felder-Silverman learning styles model are Sensing-Intuitive, Visual-Verbal, Active-Reflective, and Sequential-Global.

2.3 Pair Programming

Pair programming is a practice wherein two programmers sit at the same computer to work on a program, algorithm, or piece of code [8]. One programmer, the "driver", controls the keyboard and types code while the other, the "navigator", observes, watching for errors. The two programmers switch roles frequently (every 20-30 minutes). A large amount of research into pair programming has shown it to be an effective tool in computer science education [1, 3, 7, 9].

Nagappan et al. [7] provide the results of a pair programming study conducted over three semesters at NCSU. In this study some lab sections were taught using pair programming, while other sections retained solo programming, functioning as a control. The study found that in all cases the pair programming lab sections had an equal or higher percentage of students finish the course with a grade of C or better. This indicates that pair programming can promote increased comprehension and retention of course material. Nagappan et al. also conclude that the pair programming students had a positive view of their group learning experience, that participation in pair programming does not harm students' future performance on individual assignments, and that pair programming reduces work load for instructors and teaching assistants.

Ferzli et al. [3] present the trends they observed during several focus group discussions regarding pair programming. The focus groups were conducted with students and teaching assistants after working with pair programming in their lab sections over the course of the semester. The overall impressions were positive, with students commenting on the method's effectiveness in helping them learn. Some, however,

commented on the negative experiences that resulted from incompatible partners. Teaching assistants also found the method effective for facilitating student learning. They also commented that the method resulted in less work for them, as students were able to figure out more on their own and did not have to continuously seek help from their teaching assistants.

2.4 Student Compatibility

There is already a significant body of research into student compatibility in pair programming. These studies examine the factors that may lead to student compatibility. The goal of this body of research is to maximize the group learning experience by minimizing incompatibilities between students. Here we summarize several of the studies into pair compatibility.

In 2004, Katira et al. [5] released the results of a study conducted at NCSU over the Fall 2002 and Spring 2003 semesters. The study involved 564 students, both graduate and undergraduate. The courses involved were a freshman introductory computer science course (CS1), an upperclassman software engineering course (SE), and a graduate level object-oriented programming course (OO). The study looked at the significance of several factors in the compatibility of students in the different classes: Myers-Briggs personality type, skill level, students' perceived skill level, and self esteem. The study concludes that the significance of personality type differed depending on the course, with personality type being found to be significant in the introductory course, but not in software engineering course (this hypothesis was not tested on the graduate course). The researchers found that skill level (based on midterm scores) was only significant in the graduate course. Students' perception of their partner's skill level was found to be significant in all courses. The significance of students' programming self esteem was only tested in the introductory course and was found not to be significant.

A study released in 2006 by Williams et al. [10] continues to build upon this body of work. Their study incorporated the data from the Fall 2002-Fall 2003 CS1, SE, and OO courses (Phase 1 of the study) as well as data from Fall 2004-Fall 2005 SE courses (Phase 2). In Phase 1, the study looked at the significance of actual and perceived skill level in all three classes and personality type and programming self esteem in the two undergraduate courses. In Phase 2 the study examined Felder-Silverman learning style, work ethic, and time management skills in the SE course. The researchers found personality type to be insignificant, except in Phase 2, where individual dimensions were analyzed. The researchers found the pairing of a Sensor and Intuit to be more likely to result in a Very Compatible rating. Similarly, in the Felder-Silverman learning styles, the sensing-intuition dimension was the only one to be found significant. Actual skill level was insignificant except for in Phase 1 SE courses. Perceived skill was significant in all cases. Programming and problem solving self esteem was found to be significant in Phase 1 and 2 SE. Work ethic was found to be significant in SE Phase 2. Time management skills were not found to be significant in the Phase 2 portion of the study.

3. STUDY DETAILS

In this section we detail the courses that contributed to the data pool used in this study, our hypotheses, and the methodology.

3.1 Course Description

The data used in this study is taken from two courses at the University of Virginia: CS150 and CS201.

CS150 is an introductory computer science course titled “Computing: From Ada to the Web”. It is intended for students with no previous computer science background and generally aimed at non-engineering students. The data from CS150 used in this study was gathered during the Spring 2008 semester, when the course had an enrollment of 68 students. Students were assigned group homework assignments. The students filled out partner evaluations after submitting each assignment.

CS201, “Software Development Methods”, is also a lower-level computer science course. It is generally the second programming computer science course students will take. CS201 is aimed at engineering students. The data incorporated in this study from CS201 was gathered from two sections of the course taught during the Fall 2008 semester and one section during the Spring 2008 semester. Each lecture session has several lab sections associated with it. The total enrollment across the three sections was 226 students. The partner evaluation data received from the CS201 came from several group homework assignments and a final group project.

3.2 Hypotheses

The goal of this study was to reproduce the results of the previous work in this field at NCSU (discussed in the Related Work section). Our data did not contain race or gender information. Therefore, these factors have been omitted from our hypotheses. Our data also did not include the test scores or GPA information used by the NCSU studies as metrics for “actual skill”. As such, this hypothesis has been excluded as well.

The following table lists our research hypotheses:

Table 1: Research Hypotheses

Hypothesis: Pairs are more compatible if students with ... are grouped together.	
H-1	... different personality types ...
H-2	... different learning styles ...
H-3	... similar perceived skill levels ...
H-4	... similar programming self-esteem ...
H-5	... similar work ethic ...
H-6	... similar time management skills ...

3.3 Methodology

We used a similar methodology to test the viability of each hypothesis. As part of their partner evaluation, students gave their teammate an overall compatibility and effort score. For each hypothesis we ran a Spearman rank-order correlation with the overall score and number of different scenarios.

For personality type we examined the partners’ values for each dimension of the Myers-Briggs type indicator. First, we ran a Spearman correlation with the overall score and a value representing whether the partners fell in the same category for that dimension (i.e. both Extrovert, both Sensing, etc.). Then, we ran a Spearman correlation with the absolute value of the difference between the two partners’ numeric Myers-Briggs score for the given dimension. Finally, we created a number of binary variables representing how close two users were in this dimension. If the difference between the two users numerical scores was less than or equal to a given number, the variable would be 1, otherwise 0. These variables were created in the range from 0 to 200 (the maximum possible difference in two Myers-Briggs numerical scores) in intervals of 10. We ran a Spearman correlation for each of these variables.

The method used for the learning styles is almost identical to that used for personality type; each dimension was examined individually. The only difference in methodology between personality type and learning style is that the range of binary variable spans from 0 to 22 in intervals of 2.

The methodology used to examine perceived skill levels is much simpler. On the partner evaluation form students were asked to rate their partner’s technical competency in relation to their own as better, about the same, or worse. In examining the relationship between perceived skill and compatibility we ran a Spearman correlation with the overall score and skill rating.

For hypotheses four through six, students provided answers at the beginning of the course to a self evaluation survey which included questions regarding their programming self esteem, work ethic, and time management skills (the questions relating to programming self esteem were split into separate questions). For these questions students placed themselves on a scale that ranged from 1 to 9. These hypotheses were analyzed in the same fashion as the personality type and learning style hypotheses, with the difference being that the range binary variables tested ranged from 0 to 8 (the maximum possible difference) .

4. RESULTS

In this section we summarize the results found in performing the Spearman rank-order correlations discussed in the previous section.

4.1 Personality Type

Overall, we found personality type to be insignificant in student compatibility. There were no significant correlations in any of the four dimensions based on the category match or the numerical difference between partners.

While there are some scattered statistically significant results ($p < .050$) among the binary variables, the correlation coefficients in these cases indicate a very weak correlation. For example, the variable indicating that a pair is within 130 on the Sensing-Intuition dimension is statistically significant, but has a correlation coefficient of -0.049 , not a very strong correlation. This is typical of the correlation numbers we observed, generally somewhere in the range of 0.050 to -0.050 . Also, if this were

truly an indicator of compatibility, we would expect to see other significant results near this one.

There is, however, a group of significant values present in the judging-perceiving dimension when using a one-tailed significance test. The variables representing partners within 60, 70, 80, and 90 of each other on the judging-perceiving scale are all significant. Still, the correlation is not extremely strong, with the highest coefficient being 0.074 at variable 80. The results of our study do not indicate any strong correlation between personality type and partner compatibility.

4.2 Learning Styles

The results of our learning styles correlations were even less significant than for personality type. There were no significant correlations in any of the four dimensions based the category match or the numerical difference. The active-reflective and sensing-intuitive dimensions had no significant results in their binary variables. The visual-verbal dimension had no significant results using a two-tailed significance test and only a few scattered significant results with weak correlations ($\rho < 0.050$) using a one-tailed significance test. The sequential-global dimension had one significant correlation when the difference between partners is within ten. This produced a correlation coefficient of 0.082. This still a very slight correlation and there are no other significant results surrounding this one. The results seem to indicate that there is no correlation between learning style and compatibility.

4.3 Skill Level

Perceived skill level was by far the most accurate predictor of partner compatibility. The Spearman correlation of the overall rating and skill rating resulted in a statistically significant correlation ($p = 0.000$) with a $\rho = -0.464$. This is a fairly strong correlation, indicating that students do not enjoy working with partners who they feel to be less technically competent than them.

4.4 Programming Self Esteem

The survey questions regarding programming self esteem are split into four questions which required students to place themselves somewhere on a scale from 1 to 9 (the names with which we refer to them by are in brackets):

- [Speed] When working on a programming assignment, do you think that: 1) It takes me longer to complete programming tasks than my colleagues. 9) I am much quicker at solving programming tasks than others.
- [Problem Solving] When working on a programming problem: 1) I have difficulty knowing where to begin on new programming problems. 9) I have no difficulty in starting the problem.
- [Helping] When helping others with their computer science work, do you: 1) Feel that you do not know enough to help others. 9) I know enough to help others.
- [Roadblock] When I run into a roadblock in my thinking: 1) My mind goes around and around in circles. 9) I know that I will find the answer somewhere.

For each of these four questions we ran Spearman correlations as described in the methodology section. The speed and problem solving questions did not return overall significant results. There were a few significant results, but as with the personality types and learning styles they were scattered and had weak correlation coefficients. The roadblock question had significant results for partners within 3, 4, and 5 of each other. The correlation was fairly weak (between -0.050 and -0.060), but this may improve in further investigations.

The helping question returned significant results ($p = 0.023$) for the Spearman correlation with overall score and the difference between the partners' helping responses using a two-tailed significance test. The correlation is slight ($\rho = 0.056$), however this result is still worth noting. Future investigations may find stronger correlations between students' responses to this question and partner compatibility. This question may prove an accurate indicator of how students perceive skill level, as the question requires them to relate their computer science experience to that of their peers.

4.5 Work Ethic

Work ethic does not seem to be a significant factor in compatibility. We received one significant result ($p = 0.048$), when the difference between the two numbers is 0 (a perfect match). This correlation resulted in a ρ of .049. This is not a very strong correlation, but this factor may prove more telling in future work.

As an interesting aside, a Spearman correlation with the overall score and only the work ethic response of the person being evaluated returns a significant result ($p = 0.000$) with a correlation coefficient of $\rho = 0.128$, indicating that students with stronger work ethics tend to receive more positive reviews from their partners.

4.6 Time Management Skills

Our results did not show time management to be a significant factor in compatibility. No results were statistically significant. Williams et al. [10] also found time management to be insignificant. They theorized that students' busy schedules dictated that they meet whenever both partners were available. Since these windows of time are most likely rare, students have to disregard their personal time management habits and meet whenever those times may be.

5. PAIREVAL

In conjunction with this study we began developing a web-based tool in PHP and MySQL that would facilitate the use of group work in computer science courses. This app is based off of the open source tool from NCSU called PairEval. NCSU's PairEval provided functionality that allowed students to respond to pre-course surveys (including taking a Myers-Briggs personality test and a Felder-Silverman learning styles test), enroll in courses and evaluate their partners for a given assignment. Our version of PairEval aims to extend the functionality of NCSU's app to make it more usable by professors. The two key functionalities added to the system are survey customization and group formation.

5.1. Survey Customization

In NCSU's PairEval tool the student pre-course survey questions are hard-coded in. If a professor wanted to add or remove questions from a survey he/she would have to modify the application code containing the questions and update the database accordingly. We have simplified the process of adding and removing questions and surveys by providing an interface to do so within the application. With our expanded functionality, professors can create and delete surveys and add and remove questions from their surveys without having to edit the application code or database. Our hope is that this will make the system easier for professors to use and foster the gathering of student compatibility data for future research.

5.2 Group Formation

The PairEval tool provided by NCSU does not provide an easy way for professors to form student groups for assignments. The professor must enter the number of groups he/she wishes to create and then manually place each student into a group. In our app, the formation of groups has been automated. In this system the professor selects the course and assignment, enters a desired number of students per group (obviously, it will not always be possible to have exactly the requested number per group), and clicks a button to have groups automatically generated for that assignment. This should make the use of groups in computer science courses easier for professors, hopefully encouraging the use of pair programming. As the research into the field of partner compatibility progresses we also hope to include an algorithm that will incorporate this new knowledge in order to maximize the compatibility of the created groups.

6. CONCLUSIONS AND FUTURE WORK

The main finding of this study is that students' perceived skill level continues to be the most accurate predictor of partner compatibility. The other categories were found to be insignificant on the whole. The results of our study are summarized in the following table:

Table 2: Summary of Results

Hypothesis: Pairs are more compatible if students with ... are grouped together.	Result
H-1 ... different personality types ...	No
H-2 ... different learning styles ...	No
H-3 ... similar perceived skill levels ...	Yes
H-4 ... similar programming self-esteem ...	No
H-5 ... similar work ethic ...	No
H-6 ... similar time management skills ...	No

We plan to continue to gather data in University of Virginia computer science courses. This data will be used in future studies to analyze any changes in trends. In addition to continuing to gather data in the courses discussed in this study we would like to expand our efforts into other courses. We would like to gather data from higher level programming courses to see if the conclusions found for the introductory level courses examined in this study also hold in upper level classes. We would also like to gather data from non-programming courses, such as UVA's Digital Logic and Design (DLD) course. DLD is an introduction to the design and implementation of logic circuits. The lab sections for this class more closely resemble a

traditional science lab. We would like to examine whether the factors that affect compatibility between partners in a more traditional lab setting are similar to those affecting pair programmers.

We would also like to continue to refine the questions presented to students in the beginning of course surveys. The new PairEval tool should make this process easier with its ability to easily modify the surveys. We would like to introduce new questions that could possibly serve as indicators of compatibility or of students' perceived skill level and examine their correlation with pair compatibility.

We also plan to continue to expand the functionality of the PairEval system. As more information becomes available on the factors that lead to compatibility between students we plan to update the system with an algorithm that will pair students together for maximum compatibility.

7. REFERENCES

- [1] Cockburn, Alistair and Williams, Laurie. *The Costs and Benefits of Pair Programming in Extreme Programming Examined*. Addison Wesley, 2001.
- [2] Felder, R. M. and Silverman, L. K., "Learning and Teaching Styles in Engineering Education," *Engineering Education*, vol. 78, 1988, pp. 674-681.
- [3] Ferzli, M., Wiebe, E., and Williams, L., "Paired Programming Project: Focus Groups with Teaching Assistants and Students", NCSU Technical Report, TR-2002-16, November 25, 2002.
- [4] Katira, N., Williams, L., and Osborne, J., "Towards Increasing the Compatibility of Student Pair Programmers", International Conference on Software Engineering (ICSE) 2005, pp. 625-626.
- [5] Katira, N., Williams, L., Wiebe, E., Miller, C., Balik, S., and Gehringer, E., "On Understanding Compatibility of Student Pair Programmers", ACM Technical Symposium on Computer Science Education, SIGCSE 2004, pp. 7-11.
- [6] Myers, I.B., McCaulley, M., Quenk, N., and Hammer, A. *MBTI Manual: A Guide to the Development and Use of the Myers Briggs Type Indicator*. 3rd ed. Consulting Psychologists Press. 1998.
- [7] Nagappan, N., Williams, L., Wiebe, E., Miller, C., Balik, S., Ferzli, M., and Petlick, M., "Pair Learning: With an Eye Toward Future Success", *Extreme Programming/Agile Universe* 2003, pp. 185-198.
- [8] Williams, L. and Kessler, R. *Pair Programming Illuminated*, Addison Wesley, 2003.
- [9] Williams, L., Kessler, R., Cunningham, W., and Jeffries, R. "Strengthening the Case for Pair-Programming", *IEEE Software*, 2000. 17: p. 19-25.
- [10] Williams, L., Layman, L., Osborne, J., and Katira, N., "Examining the Compatibility of Student Pair Programmers," Proceedings of Agile 2006, Minneapolis, MN, July 23-28, 2006, pp. 411-420.