Published in final edited form as:

Learn Individ Differ. 2015 December 1; 44: 33–39. doi:10.1016/j.lindif.2015.11.002.

Dyadic Instruction for Middle School Students: Liking Promotes Learning

Amy C. Hartl^{a,*}, Dawn DeLay^b, Brett Laursen^a, Jill Denner^c, Linda Werner^d, Shannon Campe^c, and Eloy Ortiz^c

^aDepartment of Psychology, Florida Atlantic University

^bT. Denny Sanford School of Social and Family Dynamics, Arizona State University

^cResearch Department, ETR Associates

^dDepartment of Computer Science, University of California, Santa Cruz

Abstract

This study examines whether friendship facilitates or hinders learning in a dyadic instructional setting. Working in 80 same-sex pairs, 160 (60 girls, 100 boys) middle school students ($M = 12.13$ years old) were taught a new computer programming language and programmed a game. Students spent 14 to 30 ($M = 22.7$) hours in a programming class. At the beginning and the end of the project, each participant separately completed (a) computer programming knowledge assessments and (b) questionnaires rating their affinity for their partner. Results support the proposition that liking promotes learning: Greater partner affinity predicted greater subsequent increases in computer programming knowledge for both partners. One partner's initial programming knowledge also positively predicted the other partner's subsequent partner affinity.

Keywords

friendship; peer influence; dyadic learning; computer programming; Actor-partner Interdependence Model

Many teachers require students to work in dyads. Some teachers allow friends to work together; others do not. There are good reasons for each classroom strategy. Friends often collaborate enthusiastically, but there is concern they may be disruptive or digress from the task at hand (Jackson, Kutnick, & Kington, 2001; Zajac & Hartup, 1997). It is not clear from previous research if liking promotes learning in collaborative education. This is not just because research on the topic is limited but also because of the statistical obstacles posed by the nonindependent nature of dyadic data. The present study was designed to examine the effect of liking (or affinity for the partner) on learning in a collaborative computer

*Corresponding author: amy.hartl@live.com. Tel. + 1 484 744 6679.

Publisher's Disclaimer: This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final citable form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

programming environment. Analyses designed for nonindependent data assessed within-dyad, over time associations between partner affinity and the acquisition of computer skills.

Peers can be instrumental to learning. Peers are an important source of assistance, encouragement, and motivation, in that they model learning strategies and proffer tuition (Blatchford, Kutnick, Baines, & Galton, 2003; Kindermann, 2007; Tudge & Winterhoff, 1999). In collaborative settings, one peer usually has greater knowledge or experience than the other. Expert-novice interchanges are mutually beneficial because the novice learns content from the expert, and the expert's knowledge is strengthened by teaching (Azmitia, 1988; Howe, 2013; Rosenshine, Meister, & Chapman, 1996; Tudge & Rogoff, 1999; Vygotsky, 1978). According to Vygotsky (1978), when less skilled peers interact with more advanced peers, new skills are mastered when children work within their zone of proximal development, which represents the distance between actual and potential abilities. More advanced peers also cognitively benefit from collaboration. More specifically, in cooperative learning, each student within a group has specific responsibilities, actively participates in discussion, helps group members learn the material, assesses the progress of the group, and is held accountable for personally learning the material and helping the group learn the material (Johnson, Johnson, & Smith, 2007). The active, engaged nature of cooperative learning allows all group members to cognitively benefit, either through the instruction they offer others, which helps to consolidate and apply their knowledge or through the discussion with more skilled peers (Brown & Ciuffetelli, 2009). Finally, peers foster learning through discussion and negotiation (Azmitia & Perlmutter, 1989; Webb & Palincsar, 1996). It is clear that children profit from collaborative instructional opportunities; what is not clear is whether they profit more when they like their partner.

Conceptual views regarding the optimal partner for collaborative learning fall into two camps. Some argue that friends are advantageous; others argue that friends are a liability. The benefits side of the ledger includes views that closeness enhances the influence that peers wield over a variety of domains, including instruction and learning. Group work is most effective when there is trust, respect, mutuality, and equality among members (Kutnick & Colwell, 2010), characteristics usually present among friends. Friends work to maintain their relationship through open communication and successful conflict resolution (Laursen & Pursell, 2009), which provide a foundation for working together to solve problems and achieve a common goal. Considerable research supports claims that the closer peers are to one another, the more they influence each other's academic performance (Molloy, Gest, & Rulison, 2010). Many children and young adolescents working on collaborative tasks perform better when paired with friends than with nonfriends. This advantage extends to musical composition and communication (among girls) (MacDonald, Miell, & Mitchell, 2002; Miell & MacDonald, 2000), creativity box exploration and recall¹ (Newcomb & Brady, 1982), problem solving (Azmitia & Montgomery, 1993), and science reasoning (Kutnick & Kington, 2005). Collaboration with friends also promotes cognitive advances and task motivation (Fonzi, Schneider, Tani, & Tomada, 1997). Put simply, children prefer to work with friends, and there are many examples in which they benefit from doing so.

¹Children explored a box that included interesting internal and external features. The features could be successfully manipulated by (1) only one child at a time, (2) one or two children at a time, or (3) a coordinated effort of two children.

But not everyone agrees that children learn best in the company of friends. Some have argued the presence of friends can distract from learning and interfere with productivity (Zajac & Hartup, 1997). Working with a friend may relax partner expectations such that pressure to perform is reduced (Maldonado, Klemmer, & Pea, 2009). When friends are present, off-task behaviors can increase, which tends to limit on-task discussion (So & Brush, 2008). Together, these conspire to reduce effort invested in task completion. In one study of college undergraduates, ratings of friendship were negatively correlated with performance on a design project (Maldonado et al., 2009). Compared to nonfriends, friends proposed plans that were less well-developed and submitted final projects that were of poorer quality. The poor performance of friends was attributed to an aversion to criticize, pressure to agree, and time spent off-task.

Conclusions from prior research must be tempered by concerns about statistical analyses conducted on nonindependent dyadic data. Traditional parametric statistics are inappropriate for dyadic data, because correlated partner reports violate assumptions of independence, biasing error estimates (Kenny, 1996). Significance tests are compromised, which typically inflates Type II error. The problem is compounded in path analyses, because model misspecification biases parameter estimates.

Recent statistical advances overcome these difficulties. The Actor-Partner Interdependence Model (APIM: Kenny, Kashy, & Cook, 2006) partitions variance shared across partners on the same variables from variance that uniquely describes within- and between-partner associations. Modifications for longitudinal data specifically address over time influence between members of a dyad (Laursen, Popp, Burk, Kerr, & Stattin, 2008). A longitudinal APIM is akin to a residual change model, in that autoregressive effects describe the stability of a variable (Popp, Laursen, Kerr, Stattin, & Burk, 2008). By controlling for stability, the residual change in scores can be predicted. Applied to the present study, the APIM provides an unbiased estimate of the influence of liking or affinity on the acquisition of computer skills over the course of a dyadic computer programming class. One partner's initial perception of liking or affinity was used to predict improvement in computer programming knowledge of the same partner and of the other partner. Conversely, one partner's initial computer programming knowledge was used to predict changes in perceptions of liking or affinity for the partner.

Two research questions were addressed. Does liking promote learning? We tested two competing hypotheses. In keeping with the notion that friendship promotes motivation, we tested the hypothesis that greater partner affinity at the outset would anticipate greater increases in computer programming knowledge, for the self and for the partner. The alternative hypothesis, consistent with the notion that friends are often a distraction, holds that initial partner affinity would be inversely related to the rate of increase in computer programming knowledge. Does task knowledge promote liking? Little is known about whether children grow closer to those who excel in educational tasks, although one study suggests this might be the case (Tesser, Campbell, & Smith, 1984). If so, we would expect that initial computer programming knowledge would be positively associated with increases in partner affinity.

Method

Participants

The final sample of 160 students (60 girls and 100 boys in a total of 80 same-sex dyads) included 61 6th graders, 49 7th graders, and 50 8th graders. Students ranged from 10 to 14 ($M = 12.13$, $SD = 1.00$) years old. Of this total, 64.4% ($n = 103$) lived with both biological parents, 18.1% ($n = 29$) lived with one biological parent, 3.1% ($n = 5$) lived in other households, and 14.4% ($n = 23$) did not report living arrangements. In the final sample, 45.6% ($n = 73$) described themselves as “White Caucasian”, 27.5% ($n = 44$) as “Mixed Race”, 13.8% ($n = 22$) as “Hispanic Latino”, 5.0% ($n = 8$) as “Asian Pacific Islander”, 3.1% ($n = 5$) as “Native American”, 1.9% ($n = 3$) as “African American”, and 3.1% ($n = 5$) did not report their race or ethnicity.

Measures

Participants separately completed the same assessments at the beginning of the project (pretest) and at the conclusion of the project (posttest).

Partner affinity was assessed with 6 items from the Friendship Quality Scale (Bukowski, Hoza, & Boivin, 1994), which measured perceptions of satisfaction with and affection for the partner (e.g., *I feel happy when I am with my partner*; *My partner is my friend*; *If my partner had to move away, I would miss him or her*; *When I do a good job at something my partner is happy for me*; *Sometimes my partner does things for me or makes me feel special*; *I can count on my partner for help*). Items were rated on a 1 (*not true*) to 5 (*really true*) scale. Scores were averaged across items. Internal reliability was good (Cronbach’s $\alpha = .90$).

Computer programming knowledge was assessed with an 8-item measure of sample programming code created with the Alice interface similar to problems seen in introductory computer programming textbooks (Dann, Cooper, & Pausch, 2006). Items were created to measure skill in the programming environment and an understanding of programming code. Each item was designed to measure one or more of aspects of computational thinking, including algorithmic thinking, sequence of operations, and abstraction. Questions were similar to those found in computer science textbooks for children and novices. Questions tested students’ declarative (e.g., facts about Alice) and procedural (e.g., ability to apply rules to solve problems) knowledge. Students were presented with screenshots of the programming interface (e.g., *What would happen if you were to play the above 3 lines of code?*). Students selected 1 of 5 possible response options. Each student received a score that indicated the number of correct responses ($Range = 0 - 8$). Internal reliability was acceptable (Cronbach’s $\alpha = .76$).

Procedure

Participants were recruited from 8 classrooms and 4 extended learning programs in 7 public middle schools in 4 lower and middle class communities in Northern California. Schools were selected for participation if the technology teacher was interested in the project. In three schools, students ($n = 62$ boys, 40 girls) participated as part of their regular academic instruction in a technology elective course. In four schools, students ($n = 38$ boys, 20 girls)

participated as part of an after school enrichment program. All students in the technology elective classrooms were invited to participate in the study through in-class announcements and a letter sent home to parents. Grades for the elective course were based on task participation and project completion. Recruitment for the after school program involved teacher announcements. One student from the classroom setting and two students from the after school setting participated in the instruction but declined to complete the assessments and surveys; these students were not included in the study. Written parent consent and student assent were required for participation. Most students participated for 10 weeks, for an average of 22.66 hours (*Range* = 14 – 30, *SD* = 3.19).

During the first two sessions students worked with several different partners of their choosing, then privately submitted the names of three students from the class they wanted to work with, in rank order. Teachers assigned students to dyads according to preferences, attempting to place students with their first choice partner. Unfortunately, teachers did not keep records about partner rankings, so we could not determine the proportion of first choice partner dyads.

Students spent the remainder of the course working in assigned dyads. Each dyad shared a single computer and keyboard during the pair programming (Williams & Kessler, 2002). At any given time, each partner was assigned either the role of “driver” or “navigator.” In line with ideas of positive interdependence and accountability (Johnson et al., 2007), success depended on students’ abilities to fulfill their individual role responsibilities and work together to learn the Alice software. The responsibilities of each pair programmer role were (1) discussed with the students, (2) viewed in a video, (3) demonstrated in a role-play scenario, and (4) reiterated throughout the length of the program by the teacher. The responsibilities of the driver included: operating the keyboard and mouse and discussing with the navigator about what they are doing on the computer. The responsibilities of the navigator included: following what the driver is doing on screen, suggesting driver actions, preventing driver mistakes, taking notes, and organizing, holding, and reading flowcharts, binders, game design materials, and books. Students were given instructions on how to most effectively work with their partner. Students were instructed to respect the responsibilities of each role, switch roles without arguing, pay attention to each other, talk to each other about what they are doing, angle the screen so it is visible to both partners, handle disagreement respectfully, share ownership of the project, and help each other.

During the first half of the course, students learned programming skills and concepts by playing games, completing tutorials, and following step-by-step written instructional materials. During the second half of the course, students designed and programmed their own interactive computer game. Computer knowledge assessments were administered as individual online surveys at the beginning of the first and at the end of the last class meetings. Partner affinity assessments were administered in writing at the beginning of the third class meeting, after partners were assigned, and online at the end of the last class meeting. To ensure privacy, students did not sit next to partners to complete surveys and were not permitted to collaborate or share answers.

Surveys were administered by a member of the research staff. Teachers and research staff used varied management techniques to keep students on-task. Standard teaching behavior management techniques, such as proximity (walking near students who were not paying attention during teacher instruction), walking around the room, checking in with students and pairs about what they were doing, stepping in when a partner seemed disengaged, and reminding students to switch drive and navigator positions. There was always at least 1 adult (teacher) per class, with class sizes of about 20 – 30 students. About half of the time, an additional adult (research staff) was present.

Of the 189 students who attended introductory sessions, all but 18 indicated a preference for and were paired with a same-sex partner. Most early adolescent friendships are same-sex, so we omitted the 18 students assigned to other-sex dyads. One dyad was excluded because the partners were biological twins. An additional 9 students dropped out of the study. Individuals with partners who dropped out were immediately paired with students who did not have partners. New partners were assigned after 2 to 12 hours of class time ($M = 7.38$, $SD = 4.04$). Reassigned partners completed the measure of partner affinity after their first session of joint instruction. Participants in dyads with reassigned partners did not differ from the remainder of the sample on any study or demographic variable. Neither were there differences between students who participated in the after school program and those who participated during school hours.

Plan of Analysis

Path analyses were conducted in a structural equation modeling framework using Mplus v7.0 (Muthén & Muthén, 2014). Figure 1 depicts the fully saturated longitudinal APIM measurement model. Tests of indistinguishability were conducted, including intraclass correlations, paired t-tests, and Fisher's r to z correlational contrasts. The results indicated that there were no systematic within-dyad differences between partners on computer programming knowledge or partner affinity. Equality constraints reflect the interchangeable nature of partners (Olsen & Kenny, 2006). Within-dyad equality constraints were imposed on within-individual stability paths (actor paths a and b), within-individual influence paths (actor paths c and d), between-individual influence paths (partner paths e , f , g , and h), within-individual correlations (w and x), and between-individual correlations (y and z), as well as on the means (m and n) and variances (v and u) of pretest scores, and the intercepts (i and j) and residuals (r and s) of posttest scores.

We compared three different models to maximize power and to obtain the most parsimonious, best-fitting model: (1) a model in which nonsignificant within-individual influence paths were trimmed; (2) a model in which nonsignificant between-individual influence paths were trimmed; and (3) a model in which nonsignificant between-individual influence paths and nonsignificant within-individual influence paths were trimmed. The first and third models significantly worsened model fit; the second model did not significantly worsen model fit. Thus, the most parsimonious, best-fitting model excluded nonsignificant between-individual influence paths but included nonsignificant within-individual influence paths.

Fit indices for the indistinguishable dyad APIM were adjusted as recommended (Olsen & Kenny, 2006). The adjustment required consideration of three models: (a) a null model, with all effects fixed to zero; (b) a saturated model, which included all possible paths; and (c) the model of interest, which represented the most parsimonious, best-fitting model. The adjusted χ^2 , degrees of freedom (*df*), Tucker–Lewis index (*TLI*), and root mean square error of approximation (*RMSEA*) for the model of interest were adjusted in response to results from the null and saturated models. A nonsignificant χ^2 indicates the model fits the data. *TLI* compares the model of interest with the null model. Results typically range from 0 to 1, with values of .90 and higher indicating acceptable fit. The *RMSEA* compares the observed and estimated covariance matrices in the model of interest. Results range from 0 to 1, with values of .08 and lower indicating acceptable fit.

An average of 2.3% (*Range* = 1.2 – 3.7%) of the data were missing on pretest and posttest affinity and computer programming knowledge variables. Little’s test indicated data were missing completely at random, $\chi^2(44, N = 80 \text{ same-sex dyads}) = 53.63, p > .05$. Full-information maximum likelihood estimation procedures were used to handle missing values.

Combination ratios provide a measure of the effect size of the influence paths (Wickham & Knee, 2012). Combination ratios indicate the relative strength of the beta weights of two paths, typically a within-individual influence path and a between-individual influence path (e.g., β_c/β_g). Positive ratios describe concordant influence, such that individuals influence their own outcomes and their partner’s outcomes in the same manner; negative ratios represent discordant influence, such that individuals influence their own outcomes and their partner’s outcomes in the opposite manner. Combination ratios may also be used to describe the magnitude of influence paths relative to stability paths (e.g., β_c/β_a).

Additional multiple group analyses were conducted with instructional setting (during and after school) and sex as moderators. There were no statistically significant χ^2 differences, indicating that patterns of association did not differ between boys and girls or between during school and after school classes. Due to the small sample size, control variables (age, grade, hours of instruction, ethnicity, and household structure) were separately entered into the model. In each case, model fit significantly worsened and the pattern of statistically significant results did not change. Including the nine opposite-sex dyads in the APIM yielded a similar pattern of results, but significantly worsened model fit, likely because of the different functions and processes involved in opposite-sex relationships compared to same-sex relationships (Weiss, 1974), and the lesser significance adolescents assign opposite-sex friends compared to same-sex friends (Lempers & Clark-Lempers, 1993).

Results

Preliminary Analyses

Pretest scores were positively correlated with posttest scores, for both computer programming knowledge ($r = .47, p < .001$) and partner affinity ($r = .62, p < .001$). There were no statistically significant concurrent or over time correlations between computer programming knowledge and partner affinity.

Separate 2 (Time: pretest and posttest) by 2 (Sex) repeated measures ANOVAs were conducted with computer programming knowledge and partner affinity as dependent variables. Data from both members of a dyad were not independent, so one member was randomly selected for inclusion in ANOVAs. For computer programming knowledge, there was a main effect of time, $F(1,75) = 133.87, p < .001$. There was a large ($d = 2.67$) increase in programming knowledge from pretest ($M = 2.03, SD = 1.55$) to posttest ($M = 4.23, SD = 1.72$), which was qualified by a time by sex interaction, $F(1,75) = 7.44, p = .008$. Follow-up T-tests revealed that girls' computer programming knowledge ($d = 3.94$; pretest $M = 1.76, SD = 1.57$, posttest $M = 4.66, SD = 1.37$) increased more than boys' ($d = 1.96$; pretest $M = 2.19, SD = 1.53$, posttest $M = 3.98, SD = 1.87$). There were no significant main effects or interactions involving partner affinity, $F(1,74) = 2.52, p = .12$ (pretest $M = 3.24, SD = 1.05$; posttest $M = 3.04, SD = 1.08$).

Intraclass correlations (interpreted as r_2) established nonindependence between partners on predictor variables, a necessary precondition for APIM analyses (Kenny et al., 2006). Statistically significant within-dyad intraclass correlations emerged for pretest computer programming knowledge (intraclass $r = .25, p = .026$) and pretest partner affinity (intraclass $r = .42, p < .001$).

Overtime associations between partner affinity and computer programming knowledge

Figure 2 depicts the final model. The model fit the data, $\chi^2(2) = 2.61, p > .05, TLI = .96, RMSEA = .06$.

Pretest partner affinity was significantly associated with changes in the student's own computer programming knowledge from pretest to posttest ($\beta = .20$), as well as with changes in the partner's computer programming knowledge from pretest to posttest ($\beta = .21$). The closer an individual felt toward the partner at the outset, the more his or her own computer programming knowledge improved and the more the partner's computer programming knowledge improved from pretest to posttest. The ratio of the within-individual partner affinity influence path to the between-individual partner affinity influence path was 0.95, indicating that initial partner affinity is somewhat more associated with changes in the partner's computer programming knowledge than with changes in one's own computer programming knowledge. Influence paths for partner affinity were 0.36 to 0.38 the size of the partner affinity stability path, and 0.43 to 0.46 the size of the computer programming knowledge stability path, indicating substantially greater effects for stability than for influence.

Pretest computer programming knowledge was significantly associated with changes in partner reports of partner affinity from pretest to posttest ($\beta = .12$), but not with changes in self-reports of partner affinity ($\beta = .08$). The ratio of the within-individual computer programming knowledge influence path to the between-individual computer programming knowledge influence path was 0.67, indicating that initial computer programming skills are more associated with changes in the partner's affinity than with changes in one's own affinity. The influence path for computer programming knowledge was 0.22 the size of the partner affinity stability path and 0.26 the size of the computer programming knowledge stability path.

Discussion

Our findings come down firmly on the side of partner affinity as a facilitator of pair programming learning. In so doing, we replicate and extend previous work on collaborative learning. Replication comes in the form of findings indicating that adolescents learn more the more they like their partners. Extension comes in the form of findings that partner affinity shapes subsequent learning by both the individual and by the partner. New findings suggest that initial abilities anticipate changes in partner affinity.

Unique to this study is the finding that partner affinity promotes computer programming learning for both partners. Indeed, changes in computer programming knowledge were more strongly predicted by the partner's initial reports of affinity than by self-reports. Our findings are consistent with studies indicating that friends learn more in a collaborative environment than nonfriends (MacDonald et al., 2002; Miell & MacDonald, 2000). Our findings are also in line with recent evidence that influence over school behaviors depends on friends' relative attitude toward the relationship, such that the friend who has a more positive view of the relationship influences changes in his or her friends' behavior in school (Hiatt, Laursen, Kerr, & Stattin, in press). As such, they offer further support for the view that working with friends is an advantage, not a liability.

The findings suggest that even if children are not able to work with their best friends, there is merit in encouraging them to work with someone whom they like. As partner affinity increased, participants learned more, suggesting that children should be allowed to select their collaborative learning partners in order to take full advantage of the benefits of prior affiliation. Similar social skills between partners may both underlie partner affinity and allow for more effective collaboration, although this was not evaluated in this study. Equally important is the conclusion that every effort should be made to avoid pairing children who dislike one another, because another way to interpret the results is that as closeness between partners decreased, so did the amount learned. We suspect that poor communication interfered with learning, but it may also be the case that partners who are not close to one another may differ in motivation.

Of interest to those who study peers is the finding that one partner's initial programming knowledge predicted changes in the other partner's affinity. Here too, between-individual influence effects were stronger than within-individual influence effects. Posttest partner affinity was better predicted by pretest partner computer programming knowledge than by one's own computer programming knowledge. We can only speculate as to the reason. Children may base assessments of attraction on the perceived costs and benefits of the relationship (Laursen & Hartup, 2002). Relationship satisfaction may be affected by the resources a partner has to offer. A friend with academic skills can be valuable in terms of the potential for tutelage and support, whereas a friend without academic skills can be burdensome and slow the learning process. These findings are consistent with previous work indicating that middle school students prefer friends who perform well on academic and extracurricular activities (Tesser et al., 1984).

This study is not without limitations. The small sample size precluded the use of age and ethnicity as moderators and limited power in models that used sex and instructional setting as moderators. This may explain why we were able to replicate the finding that girls benefit more from being paired with friends than do boys (Kutnick & Kington, 2005), but we did not find sex differences in the degree to which one friend influences another. The small sample size also precluded the investigation of subgroups of students who may benefit less from pair programming. Although the findings indicate that partner liking aids the learning process, certain students, such as those who are very easily distracted, may benefit more from individual work (Lucas & Thomas, 2012). It should not be assumed that friends are always beneficial learning partners. Students who are burned out from school tend to influence decreases in their friends' school engagement (Marion, Laursen, Kiuru, Nurmi, & Salmelo-Aro, 2014). This study did not investigate the underlying mechanisms of influence, thus it is difficult to determine what features (e.g., social skills, motivation, communication) of close partners drive influence. Influence is unlikely to be equally shared between friends within a dyad. For instance, we know that confidence and initial knowledge impact who is more greatly influenced within a dyad (DeLay et al., 2014), as do perceptions of the quality of friendship (Hiatt, Laursen, Kerr, & Stattin, in press). Children did not select friends, but instead expressed a preference for partners to teachers who assigned them to pairs, then rated the degree to which they liked their assigned partner. It is unknown to what extent children based their partner preferences on affinity, academic skills, or something else. It is important to note, however, that partner preference criteria is largely irrelevant to the interpretation of the findings because the results do not contrast friends and nonfriends, but rather measure the degree to which perceptions of partner liking influence performance. Our study concerned learning activities that were of interest to students, as indicated by the fact that the course or after school activity was optional. Friends may be inclined to be disruptive and unproductive if they are not engaged in the subject matter (So & Brush, 2008). Data were not collected on off-task behavior, so the amount of unproductive behavior that occurred is unknown. Only computational thinking was tested, so it is unclear whether the same findings would result from a less algorithmic, procedural task. An 8-item multiple choice measure may not have been the best measure of learning over the course of the study. However, the items are similar to those used in beginning computer programming courses to measure conceptual understanding providing some justification for why we chose the measure. Lastly, because all children worked in dyads, it is unclear whether the same findings would result with larger groups (e.g., triads) in which there is greater opportunity for variation in skill, interest, and partner liking.

Every child cannot be paired with a knowledgeable partner. Nor can teachers dictate the characteristics of partners. Teachers can, however, structure paired learning opportunities in such a way as to maximize the potential of each dyad. One way to do this is to ensure that students enjoy the company of their assigned partners.

Acknowledgement

The authors would like to thank Christopher A. Hafen for his early efforts on this project.

This project was supported by a grant to Jill Denner and Linda Werner from the U.S. National Science Foundation (0909733). Brett Laursen also received support from the U.S. National Science Foundation (0923745) and the U.S. National Institute of Child Health and Human Development (HD068421).

References

- Azmitia M. Peer interaction and problem solving: When are two heads better than one? *Child Development*. 1988; 59(1):87–96.
- Azmitia M, Montgomery R. Friendship, transactive dialogues, and the development of scientific reasoning. *Social Development*. 1993; 2(3):202–221.
- Azmitia, M.; Perlmutter, M. Social influences on children's cognition: State of the art and future directions. In: Reese, HW., editor. *Advances in child development and behavior*. Vol. 22. NY: Academic Press; 1989. p. 89-144.
- Blatchford P, Kutnick P, Baines E, Galton M. Toward a social pedagogy of classroom group work. *International Journal of Educational Research*. 2003; 39(1):153–172.
- Brown, H.; Ciuffetelli, DC., editors. *Foundational methods: Understanding teaching and learning*. Toronto: Pearson Education; 2009.
- Bukowski WM, Hoza B, Boivin M. Measuring friendship quality during pre-and early adolescence: The development and psychometric properties of the Friendship Qualities Scale. *Journal of Social and Personal Relationships*. 1994; 11(3):471–484.
- Dann, W.; Cooper, S.; Pausch, R. *Learning to program with Alice*. Upper Saddle River, NJ: Prentice-Hall; 2006.
- DeLay D, Hartl AC, Laursen B, Denner J, Werner L, Campe S, Ortiz E. Learning from friends: Measuring influence in a dyadic computer instructional setting. *International Journal of Research Methods in Education*. 2013; 37(2):190–205.
- Fonzi A, Schneider BH, Tani F, Tomada G. Predicting children's friendship status from their dyadic interaction in structured situations of potential conflict. *Child Development*. 1997; 68(3):496–506. [PubMed: 9249962]
- Hiatt C, Laursen B, Stattin H, Kerr M. Best friend influence over adolescent problem behaviors: Socialized by the satisfied. *Journal of Clinical Child and Adolescent Psychology*. (in press).
- Howe C. Scaffolding in context: Peer interaction and abstract learning. *Learning, Culture and Social Interaction*. 2013; 2(1):3–10.
- Jackson A, Kutnick P, Kington A. Principles and practical grouping for the use of drill and practice programs. *Journal of Computer Assisted Learning*. 2001; 17(2):130–141.
- Johnson DW, Johnson RT, Smith K. The state of cooperative learning in postsecondary and professional settings. *Educational Psychology Review*. 2007; 19:15–29.
- Kenny DA. Models of non-independence in dyadic research. *Journal of Social and Personal Relationships*. 1996; 13(2):279–294.
- Kenny, DA.; Kashy, DA.; Cook, WL. *Dyadic data analysis*. NY: Guilford Press; 2006.
- Kindermann TA. Effects of naturally existing peer groups on changes in academic engagement in a cohort of sixth graders. *Child Development*. 2007; 78(4):1186–1203. [PubMed: 17650133]
- Kutnick, P.; Colwell, J. Dialogue enhancement in classrooms. In: Littleton, K.; Howe, C., editors. *Educational dialogues: Understanding and promoting productive interaction*. NY: Routledge; 2010. p. 192-215.
- Kutnick P, Kington A. Children's friendships and learning in school: Cognitive enhancement through social interaction? *British Journal of Educational Psychology*. 2005; 75(4):521–538. [PubMed: 16318677]
- Laursen, B.; Hartup, WW. The origins of reciprocity and social exchange in friendships. In: Laursen, B.; Graziano, WG., editors. *Social exchange in development: New Directions for Child and Adolescent Development*. San Francisco, CA: Jossey-Bass; 2002. p. 27-40.
- Laursen, B.; Popp, D.; Burk, WJ.; Kerr, M.; Stattin, H. Incorporating interdependence into developmental research: Examples from the study of homophily and homogeneity. In: Card, NA.; Selig, JP.; Little, TD., editors. *Dyadic and interdependent data in developmental research*. Mahwah, NJ: Erlbaum; 2008. p. 11-37.

- Laursen, B.; Pursell, GR. Conflict in peer relationships. In: Rubin, KH.; Bukowski, WM.; Laursen, B., editors. *Handbook of peer interactions, relationships, and groups*. NY: Guilford; 2009. p. 267-286.
- Lempers JD, Clark-Lempers DS. A functional comparison of same-sex and opposite-sex friendships during adolescence. *Journal of Adolescent Research*. 1993; 8(1):89–108.
- Lucas, D.; Thomas, G. Organising classrooms to promote learning for all children: two pieces of action research. In: Clipson-Boyles, S., editor. *Putting research into practice in primary teaching and learning*. NY: Routledge; 2012. p. 25-36.
- MacDonald AR, Miell D, Mitchell L. An investigation of children's musical collaborations: The effect of friendship and age. *Psychology of Music*. 2002; 30(2):148–163.
- Maldonado, H.; Klemmer, SR.; Pea, RD. When is collaborating with friends a good idea?. In: O'Malley, C.; Suthers, D.; Reimann, P.; Dimitracopoulou, A., editors. *Proceedings of the 9th International Computer Supported Collaborative Learning Conference; International Society of the Learning Sciences*; 2009. p. 227-231.
- Marion D, Laursen B, Kiuru N, Nurmi JE, Salmela-Aro K. Maternal affection moderates friend influence on schoolwork engagement. *Developmental Psychology*. 2014; 50(3):766–771. [PubMed: 24015690]
- Miell D, MacDonald R. Children's creative collaborations: The importance of friendship when working together on a music composition. *Social Development*. 2000; 9(3):348–369.
- Molloy LE, Gest SD, Rulison KL. Peer influences on academic motivation: Exploring multiple methods of assessing youths' most "influential" peer relationships. *Journal of Early Adolescence*. 2010; 31(1):13–40.
- Muthén, LK.; Muthén, BO. *Mplus User's Guide*. Los Angeles: Muthén & Muthén; 1998–2014.
- Newcomb AF, Brady JE. Mutuality in boys' friendship relations. *Child Development*. 1982; 53(2): 392–395.
- Olsen JA, Kenny DA. Structural equation modeling with interchangeable dyads. *Psychological Methods*. 2006; 11(2):127. [PubMed: 16784334]
- Popp D, Laursen B, Kerr M, Stattin H, Burk WK. Modeling homophily over time with an actor-partner interdependence model. *Developmental Psychology*. 2008; 44(4):1028–1039. [PubMed: 18605832]
- Rosenshine B, Meister C, Chapman S. Teaching students to generate questions: A review of the intervention studies. *Review of Educational Research*. 1996; 66(2):181–221.
- So HJ, Brush TA. Student perceptions of collaborative learning, social presence and satisfaction in a blended learning environment: Relationships and critical factors. *Computers and Education*. 2008; 51(1):318–336.
- Tesser A, Campbell J, Smith M. Friendship choice and performance: Selfevaluation maintenance in children. *Journal of Personality and Social Psychology*. 1984; 46(3):561–574.
- Tsay M, Brady M. A case study of cooperative learning and communication pedagogy: Does working in teams make a difference? *Journal of the Scholarship of Teaching and Learning*. 2012; 10(2):78–89.
- Tudge, J.; Rogoff, B. Peer influences on cognitive development: Piagetian and Vygotskian perspectives. In: Lloyd, P.; Fernyhough, C., editors. *Lev Vygotsky: Critical assessments*. Vol. 3. London: Routledge; 1999. p. 32-56.
- Tudge, JR.; Winterhoff, PA. Vygotsky, Piaget, and Bandura: Perspectives on the relations between the social world and cognitive development. In: Lloyd, P.; Fernyhough, C., editors. *Lev Vygotsky: Critical assessments*. Vol. 1. London: Routledge; 1999. p. 311-338.
- Webb, NM.; Palincsar, AS. Group processes in the classroom. In: Berliner, DC.; Calfee, RC., editors. *Handbook of educational psychology*. NY: Macmillan; 1996. p. 841-873.
- Weiss, RS. The provisions of social relationships. In: Rubin, Z., editor. *Doing unto others*. Englewood Cliffs, NJ: Prentice-Hall; 1974.
- Wickham RE, Knee CR. Interdependence theory and the Actor-Partner Interdependence Model: Where theory and method converge. *Personality and Social Psychology Review*. 2012; 16(4):375–393. [PubMed: 22619276]
- Williams, L.; Kessler, R. *Pair programming illuminated*. Boston, MA: Addison- Wesley Longman Publishing Co., Inc.; 2002.

- Vygotsky, L. *Mind in society: The development of higher psychological processes*. Cambridge, MA: Harvard University Press; 1978.
- Zajac R, Hartup WW. Friends as coworkers: Research review and classroom implications. *Elementary School Journal*. 1997; 98(1):3–13.

Author Manuscript

Author Manuscript

Author Manuscript

Author Manuscript

Highlights

- Middle school students were taught computer programming in same-sex dyads.
- Each student completed computer skills and partner affinity assessments.
- Initial partner affinity positively predicted one's own later computer skills.
- Initial partner affinity positively predicted the partner's later computer skills.
- Initial computer skills positively predicted the partner's later partner affinity.

Figure 1.

Measurement model for indistinguishable dyad APIM assessing longitudinal associations between partner affinity and computer programming knowledge.

Note. The model includes stability paths (*a* and *b*), within-individual influence paths (*c* and *d*), between-individual influence paths (*e*, *f*, *g*, and *h*), within-individual correlations (*w* and *x*), and between-individual correlations (*y* and *z*).

Figure 2.

Initial partner affinity predicts changes in self and partner computer programming knowledge and initial computer programming knowledge predicts changes in partner reports of partner affinity.

Note. $N = 160$ (80 same-sex dyads). * $p < .05$, ** $p < .001$, two-tailed.