Exploring the Impact of Gender Bias on Pair Programming

Aslıhan Akalın
UC Berkeley
Berkeley, California, United States

Nathaniel Weinman
UC Berkeley
Berkeley, California, United States

Katherine Stasaski
UC Berkeley
Berkeley, California, United States

Armando Fox
UC Berkeley
Berkeley, California, United States

CCS CONCEPTS

• **Software and its engineering → Pair programming**; • **Social and professional topics → Gender**; • **Applied computing → Collaborative learning**.

KEYWORDS

Pair programming, Gender bias, CS Education

ACM Reference Format:
Aslıhan Akalın, Nathaniel Weinman, Katherine Stasaski, and Armando Fox. 2021. Exploring the Impact of Gender Bias on Pair Programming. In *Proceedings of the 17th ACM Conference on International Computing Education Research (ICER 2021), August 16–19, 2021, Virtual Event, USA.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3446871.3469790

Pair programming, two partners working on a programming task together, is an effective tool for teaching computer science (CS), as measured by performance, confidence, and improved retention in CS programs [4]. These positive effects are especially impactful for women [9, 11]. In pair programming, mutual student engagement is key. But what makes a good pair?

Gender affects the experience of (any pairing of) students due to phenomena such as implicit gender bias (e.g., assuming a woman will be less technically competent than a man). Previous work has found conflicting results about whether same-gender or mixed-gender pairings are more effective [2, 3, 6, 8]. One explanation is that gender correlates with other dimensions that may affect collaboration, such as relative skill level, personality traits, or existing friendships [1–3, 5, 7, 10, 13]. However, it is not feasible to control for these other factors in a between-subject study design.

We propose an IRB-approved within-subject methodology to gain insight into the effect of the *perceived* gender of a partner and the *actual* gender of a partner (Figure 1). This allows us to separate effects of factors such as implicit gender bias, which rely on perceived gender, and larger systemic factors, which affect people based on their actual gender. For *perceived* gender, we acknowledge the current study focuses on binary gender roles.

We will have undergraduate CS students solve programming questions in a *distributed* pair programming setting, allowing *remote* pairings to coordinate their actions as they would in-person. Participants will only communicate through text-based chat and a shared code editor. Inspired by Whiting et al. [12], participants will believe that they are paired with two different partners, but they will actually be paired with the same partner twice. Using gendered pseudonyms, pronouns, and avatars, we will deceive one participant to perceive their partner as two different genders across sessions. Between these sessions, participants will work alone in a solo session. Participants will answer survey questions after each session and after completing the study.

Our participants will be categorized into two groups: the "gender-deceived" group that believe they worked with two partners of different genders (Participant B in Figure 1), and the "bystander" group that believe they worked with two partners of the same gender (Participant A in Figure 1).

We will use code editor logs, chat logs, and self-rated survey results to probe different aspects of the pair programming experience. For instance, we can measure if "gender-deceived" participants rated perceived-female partners as less technically competent, causing them to write more code and make fewer uncertainty statements in the chat. We can also see if "bystanders" rate the session as less effective when this occurs.

We will run this study in two phases. The first phase will help us understand the impact and extent of gender bias in this setting; the second will use interventions designed to reduce biased behavior (e.g., explicit "skill level" visualizations), allowing us to measure their effectiveness. Understanding the efficacy of these interventions could create more equitable pair programming environments, helping to close the gender gap in CS.

ACKNOWLEDGMENTS

This work was supported by two National Science Foundation (NSF) Graduate Research Fellowships (DGE 1752814) and an AWS Machine Learning Research Award. We thank the anonymous reviewer as well as Pablo Fernandez Montes, Amador Dúran Toro, Beatriz Bernárdez, An Ju, Hezheng Yin, Qi Zhong, and the Hearst and Fox Lab Research Groups for their helpful comments.

REFERENCES

[1] Edgar Acosta Chaparro, Aybala Yuksel, Pablo Romero, and Sallyann Bryant. 2005. Factors Affecting the Perceived Effectiveness of Pair Programming in Higher Education. In *Proceedings of the 17th Annual Workshop of the Psychology of Programming Interest Group, PPIG 2005, Brighton, UK, June 29 - July 1, 2005.* Psychology of Programming Interest Group, 2. http://ppig.org/library/paper/factors-affecting-perceived-effectiveness-pair-programming-higher-education

[2] Kyungsub S. Choi. 2013. Evaluating Gender Significance within a Pair Programming Context. In *Proceedings of the 2013 46th Hawaii International Conference on System Sciences (HICSS '13).* IEEE Computer Society, USA, 4817–4825. https://doi.org/10.1109/HICSS.2013.209

ICER 2021, August 16–19, 2021, Virtual Event, USA

ACM ISBN 978-1-4503-8326-4/21/08.
https://doi.org/10.1145/3446871.3469790

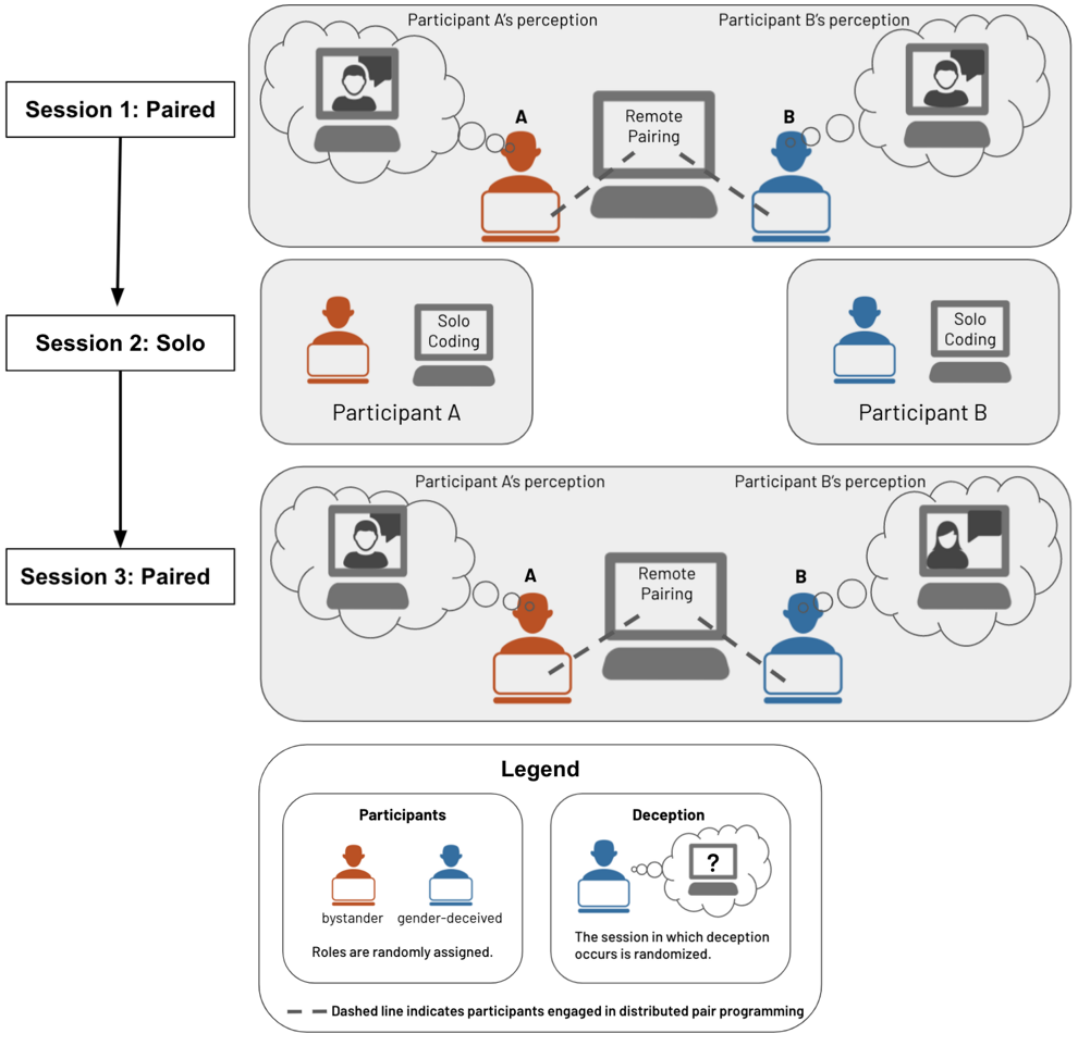

Figure 1: Study design diagram.

[3] Kyungsub Stephen Choi. 2015. A comparative analysis of different gender pair combinations in pair programming. *Behaviour & Information Technology* 34, 8 (2015), 825–837. https://doi.org/10.1080/0144929X.2014.937460 arXiv:https://doi.org/10.1080/0144929X.2014.937460

[4] Bernardo José da Silva Estácio and Rafael Prikladnicki. 2015. Distributed Pair Programming: A Systematic Literature Review. *Information and Software Technology* 63 (2015), 1–10. https://doi.org/10.1016/j.infsof.2015.02.011

[5] Jo E. Hannay, Erik Arisholm, Harald Engvik, and Dag I.K. Sjoberg. 2010. Effects of Personality on Pair Programming. *IEEE Transactions on Software Engineering* 36, 1 (2010), 61–80. https://doi.org/10.1109/TSE.2009.41

[6] Sarah I. Hofer. 2015. Studying Gender Bias in Physics Grading: The role of teaching experience and country. *International Journal of Science Education* 37, 17 (2015), 2879–2905. https://doi.org/10.1080/09500693.2015.1114190 arXiv:https://doi.org/10.1080/09500693.2015.1114190

[7] Lindsay Jarratt, Nicholas A. Bowman, K.C. Culver, and Alberto Maria Segre. 2019. A Large-Scale Experimental Study of Gender and Pair Composition in Pair Programming. In *Proceedings of the 2019 ACM Conference on Innovation and Technology in Computer Science Education* (Aberdeen, Scotland Uk) *(ITiCSE '19)*. Association for Computing Machinery, New York, NY, USA, 176–181. https://doi.org/10.1145/3304221.3319782

[8] Sandeep Kaur Kuttal, Kevin Gerstner, and Alexandra Bejarano. 2019. Remote Pair Programming in Online CS Education: Investigating through a Gender Lens. In *2019 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*. 75–85. https://doi.org/10.1109/VLHCC.2019.8818790

[9] Fernando J. Rodríguez, Kimberly Michelle Price, and Kristy Elizabeth Boyer. 2017. Exploring the Pair Programming Process: Characteristics of Effective Collaboration. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education* (Seattle, Washington, USA) *(SIGCSE '17)*. Association for Computing Machinery, New York, NY, USA, 507–512. https://doi.org/10.1145/3017680.3017748

[10] Lynda Thomas, Mark Ratcliffe, and Ann Robertson. 2003. Code Warriors and Code-a-Phobes: A Study in Attitude and Pair Programming. In *Proceedings of the 34th SIGCSE Technical Symposium on Computer Science Education* (Reno, Navada, USA) *(SIGCSE '03)*. Association for Computing Machinery, New York, NY, USA, 363–367. https://doi.org/10.1145/611892.612007

[11] Linda L. Werner, Brian Hanks, and Charlie McDowell. 2004. Pair-Programming Helps Female Computer Science Students. *J. Educ. Resour. Comput.* 4, 1 (March 2004), 4–es. https://doi.org/10.1145/1060071.1060075

[12] Mark E. Whiting, Allie Blaising, Chloe Barreau, Laura Fiuza, Nik Marda, Melissa Valentine, and Michael S. Bernstein. 2019. Did It Have To End This Way? Understanding The Consistency of Team Fracture. *Proc. ACM Hum.-Comput. Interact.* 3, CSCW, Article 209 (Nov. 2019), 23 pages. https://doi.org/10.1145/3359311

[13] Stelios Xinogalos, Maya Satratzemi, Alexander Chatzigeorgiou, and Despina Tsompanoudi. 2017. Student perceptions on the benefits and shortcomings of distributed pair programming assignments. In *2017 IEEE Global Engineering Education Conference (EDUCON)*. 1513–1521. https://doi.org/10.1109/EDUCON.2017.7943050