Meet your Maker: A Social Identity Analysis of Robotics Software Engineering

Carlos Gavidia-Calderon
The Alan Turing Institute
United Kingdom
The Open University
United Kingdom
cgavidia-calderon@turing.ac.uk

Amel Bennaceur
The Open University
United Kingdom
amel.bennaceur@open.ac.uk

Tamara Lopez
The Open University
United Kingdom
tamara.lopez@open.ac.uk

Anastasia Kordoni
Lancaster University
United Kingdom
a.kordoni@lancaster.ac.uk

Mark Levine
Lancaster University
United Kingdom
mark.levine@lancaster.ac.uk

Bashar Nuseibeh
The Open University
United Kingdom
Lero - University of Limerick
Ireland
bashar.nuseibeh@open.ac.uk

ABSTRACT

Software systems often reflect the values of the people that engineered them: it is vital to understand and engineer those values systematically. This is crucial for autonomous systems, where human interventions are not always possible. The software engineering community shows some positive values —like altruism— and lack others —like diversity. In this project, we propose to elicit the values of the engineers of autonomous systems by analysing the artefacts they produce. We propose to build on the social identity theory to identify encouraged and discouraged behaviours within this collective. Our goal is to understand, diagnose, and improve the engineering culture behind autonomous system development.

CCS CONCEPTS

• **Software and its engineering** → **Programming teams**; • **Social and professional topics** → *Computing profession.*

KEYWORDS

robotics software engineering, social identity, autonomous systems, game theory, empirical software engineering

ACM Reference Format:
Carlos Gavidia-Calderon, Amel Bennaceur, Tamara Lopez, Anastasia Kordoni, Mark Levine, and Bashar Nuseibeh. 2023. Meet your Maker: A Social Identity Analysis of Robotics Software Engineering. In *First International Symposium on Trustworthy Autonomous Systems (TAS '23), July 11–12, 2023, Edinburgh, United Kingdom.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3597512.3600206

TAS '23, July 11–12, 2023, Edinburgh, United Kingdom

ACM ISBN 979-8-4007-0734-6/23/07.
https://doi.org/10.1145/3597512.3600206

1 INTRODUCTION

> We have not completely overcome the longing for the tribe, when a man was still an inseparable part of the collective.
>
> Mario Vargas Llosa, The Call of the Tribe, 2018

In the early days of computing, it was common for a single developer, or coder, to build production-ready software. Nowadays, software development is an increasingly social endeavour, in which many coders across the globe work together. Software developers have a large impact on modern life through the tools they build. Given their influence, researchers have used methods from the social sciences to address the question of "who software engineers are". Ethnographic studies have found evidence of a culture among software engineers, with shared beliefs and values [30]. Research examining software development practice has identified aspects of work that motivate [2] or satisfy developers within the workplace [9], and ways in which developers perceive that they are successful [36] or belong to communities [37].

Software development is unique, exhibiting significant differences to other professions. To mention a few, from a sociological perspective developers are unusually altruistic [15]. The open-source movement develops and maintains software artefacts for free, relying mostly on volunteers. For example, the Log4J utility —powering iCloud and Twitter— is free to use. Also, and unlike other engineering disciplines, the profession is mostly unregulated. There is actually a resistance to professional licensing among software professionals [21] and often times, developers do not even need a university degree to be involved in large software projects. On the negative side, software development has a severe gender-diversity problem, even when compared with other STEM disciplines [20]. Social psychology studies show the presence of implicit gender biases that influence hiring decisions in software teams [40].

Robotics software engineers, or just *roboticists*, have some characteristics that distinguish them from the wider software community. For one, they have diverse educational backgrounds: in addition to training in computer science, many roboticists are formally trained in electrical and mechanical engineering [5, 10, 43]. Roboticists have also been found to anthropomorphise the systems they build [6]. In common with other STEM disciplines, robotics engineering has a gender diversity problem [22, 38], and sometimes a disregard for ethical concerns [43].

Robotics engineering, in common with other software engineering professions, is a team effort, requiring the cooperation of software engineers, testers, hardware engineers, and product designers to develop autonomous systems [6]. Team members share values and develop new ones during system development. Often unintentionally, they embed these values in the autonomous systems they build. It is crucial to understand the values and relationships within the robotics software engineering community, given that the systems they build are becoming part of our daily lives. Many robotic systems —like drones, self-driving cars, or industrial robots— operate with some degree of autonomy, sharing space and interacting regularly with human beings. A system's relationship with humans is defined by the values of the engineering team that built it. For society to trust these autonomous systems, they need to trust that the values of its engineering team are aligned with societal values. The values of the makers are reflected in the products they fabricate [5], raising the need to promote the development of an inclusive, altruistic culture within robotics engineering teams.

Decades of research have identified multiple behaviours emerging within the software engineering collective [4]. However, the group dedicated to building software for autonomous systems —like drones, self-driving cars, or industrial robots— is still understudied [43]. In this extended abstract, we propose research to identify the behaviours that originate within the robotics engineering community.

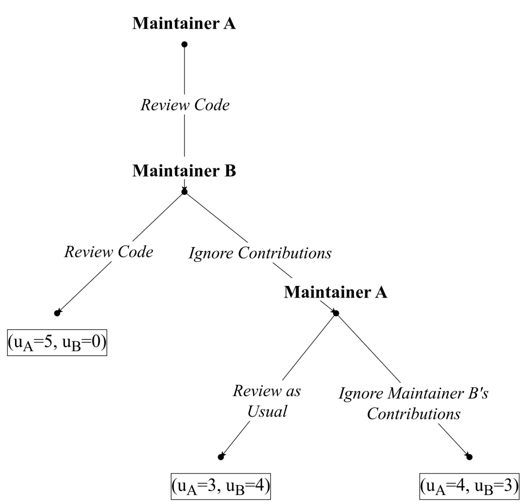

Figure 1: A game-theoretic model of an identity-driven behaviour within open-source. It models two maintainers (A and B), that can either review code contributions or ignore them. Maintainer B would prefer to ignore them, but this might trigger a response from maintainer A, given that reviewing code is an identity expectation.

Our approach, shown in Figure 2, builds game-theoretic models of identity-driven behaviours, informed by previous work examining cultural aspects of software engineering communities (eg. [2, 9, 36]), Based on these models, we can propose interventions that discourage negative behaviours and promote positive ones. To explain these behaviours, we propose using the social identity approach: a social psychology perspective that explains group behaviour [34]. It establishes that human behaviour can be explained by group membership. These groups can be a function of, for example, gender, nationality, or profession. Social identity has explanatory power. For instance, gender discrimination in the workplace, and division of labour in a household can be explained with social identity models, but resist classic economic analysis [1]. Our focus is on the identities associated to the robotics engineering practice. We will study identity at different levels of granularity: from wide categories like "roboticist", to more specific ones like "testing engineer".

In section 2, we present our approach for modelling identity-driven behaviours to investigate interactions within the roboticist community. In section 3, we elaborate on how those models can support the development of an altruistic and inclusive robotics engineering culture. In section 4, we briefly discuss related work. Finally, section 5 concludes the paper.

2 MODELLING THE ROBOTICIST IDENTITY

In software engineering, game-theoretic models have been used to diagnose and remove problems in software processes [13, 14]. In this extended abstract, we propose using game-theoretic models to understand identity dynamics within robotics teams. Game theory is an adequate framework for reasoning about relationships within a social identity [1]. Myerson defines game theory as "the study of mathematical models of conflict and cooperation between intelligent rational decision-makers" [23]. These decision-makers, or *players*, interact in a game using *strategies*, that determine the *utility* they perceive. Using game theory, we can predict player behaviour by obtaining the Nash equilibrium of the game-theoretic model. At equilibrium, each player adopts a strategy that is the best response to the strategies of the rest of the players.

In identity-based game-theoretic models, identity expectations favour some strategies and discourage others [1]. For example, open-source projects receive code contributions from developers outside the core team. These candidate contributions constitute pull requests. In order to be incorporated to the project's codebase, or *merged*, they need to be approved by senior developers, called *maintainers* [16]. Solidarity and cooperation are core values of open-source software [15]. Hence, we expect the "open-source maintainer" identity to favour the strategy of volunteering to review code contributions from the community.

Group members that do not align to identity expectations experience anxiety and discomfort, that we model as utility loss [1]. This negative effect is not limited to the offending member, but extended to the rest of the social group. Following the open-source maintainer example, let us imagine that personal reasons made a maintainer neglect or delay code reviews. This would produce a utility loss on all the project's maintainers. To restore identity, group members can penalise the offending person to make them comply [39]. In the open-source project example, this penalty can take the form of ignoring code contributions from maintainers that do not actively engage in reviewing for the community. This behaviour could indicate a low level of kinship or altruism within the group, two attributes that have been positively associated with a sense of virtual community in open source development [37].

Our goal is to build game-theoretic models, like the one in Figure 1, of the identities *within* robotics engineering teams. These models should reflect the expectations and behaviours that arise by being part of the robotics engineering collective. In them, players are team members sharing an identity, like maintainers A and B in Figure 1, and their strategies are behaviours of interest, like "Review Code" or "Ignore Maintainer's contributions". Player utility —u_A and u_B in Figure 1— depends on a metric team members try to maximise, like number of accepted code contributions for open-source developers.

We can obtain players, strategies, and utility values by mining process data. Model validation will also be driven by process data. For example, the "open-source maintainer" identity model in Figure 1 establishes a dependency between a maintainer's pull request merges and their own accepted code contributions. Using time series data from code repositories, we can analyse the relationship between these two quantities. For example, a *Granger causality test* can determine if past values of merged pull-requests contribute to the prediction of future values of accepted code contributions [12]. We can use statistical techniques to find if an identity model's assumptions and predictions hold. Software engineering is in a privileged position for statistical data analysis. Every activity, from requirements gathering to defect management, is supported by software that generates and exposes data.

From the many dimensions of social identities, our identity models —like the one in Figure 1— only represent a behavioural expectation and the penalty for members that do not comply with it. This modelling scope is defined by the available process data, the behaviours reflected in them, and the statistical methods to determine causal relations between these behaviours. Formulating plausible hypothesis about the behaviours that define an engineering team's identity is challenging. We plan to gather our candidate identity-driven behaviours from the team under analysis, using observation, semi-structured interviews and surveys. We believe robotics engineers are in a better position to identify the group-level behaviours of their workplace.

3 A BETTER ENGINEERING CULTURE

We can use these game-theoretic models to determine behaviours that can be explained by identity dynamics. These behaviours correspond to the strategies resulting from the equilibrium analysis of an identity's game-theoretic model. These identity models are not only a diagnosis tool for robotics engineering teams. We can also use them to test interventions for the removal of undesired behaviour, using the process shown in Figure 2.

Let us use the "open-source maintainer" identity example to showcase our approach. In the *social identity model* shown in Figure 1, the *equilibrium analysis* predicts some maintainers ignoring code reviews, and other maintainers ignoring their code contributions as a response. This *identity-driven behaviour* has a negative impact on team morale and productivity, so we will attempt its removal. During *process improvement*, we posit that by publishing review reminders in the project's messaging platform would make ignoring reviews unappealing. Under the assumption that this reminders would reduce the utility of the offending maintainer from $u_A = 4$ to $u_A = 2$ (bottom-right leaf in Figure 1), the updated model has an equilibrium where maintainers *do not* reciprocate by ignoring contributions. These results can serve as an input to later deploy this practice within the team.

Other identity-driven behaviours can be addressed with our approach. For instance, let us imagine we use *process data* from project management systems to build the *social identity model* of the "startup roboticist" identity. Then, *equilibrium analysis* predicts engineers working long hours, discouraging leaving early by scheduling late meetings. This *identity-driven behaviour* can lead to burnout and a corresponding low level of job satisfaction [9]. Pressure for working extra hours is specially challenging for women, that sadly have the lion's share of family chores in many households [35]. This outcome signal areas for improvement in the work culture: during *process improvement*, we can test if some form of "taxation" can remove the long-hours behaviour at equilibrium. In practice, taxation can be implemented by requiring a justification and senior management approval to schedule a late-hours meeting.

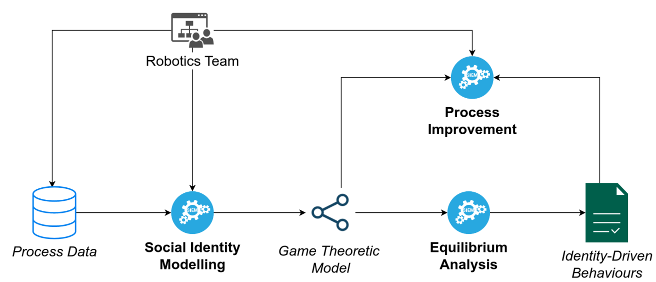

Figure 2: Improving robotics teams by social identity modelling. Using process data and observing the teams, we build game theoretic models of their social identity (like the one in Figure 1). Equilibrium analysis of these models produce a list of identity-driven behaviours. If any of these behaviours are harmful, we can use the model to test interventions for its removal.

4 RELATED WORK

Existing work has promoted the need to consider ethics [27] and values[41] during the development of software systems. Some approaches have been proposed to assess and study values in software engineering [42], to incorporate social values in software design patterns [19], and to measure the impact of values in requirements engineering activities [26]. Values are well studied in human-computer interaction and information systems [7], and especially in healthcare [18]. Existing approaches focus on early stages of the development process [29], with little attention given to the satisfaction of values in deployed software systems or on the design of sociotechnical interventions to operationalise these values. We will build on existing work in Values@Runtime [3] to consider values during operation in order to monitor and reflect on these values. We will also explore the use of emotional goals of roboticists to design inclusive software and processes [17].

Social identities may also help guide the planning of adaptive interventions [24, 28] that require engagement and cooperation of humans. Existing work on enabling software systems to reason about identities to enable cooperation [11] provides a direction for planning identity-aware interventions.

5 CONCLUSION AND FUTURE WORK

In this extended abstract, we used two identity-driven behaviours to showcase our approach: code reviewing in open-source projects, and work-life balance in startup companies. For demonstration purposes, this paper relied heavily on assumptions. In future work, we will use the procedure described in Figure 2 to explain each of these behaviours using process data. While data for the work-life balance issue would require an industrial collaborator, the code reviewing study can take advantage of the features and data publicly available at GitHub.

We will also embed Responsible Research and Innovation (RRI) principles in our work, by adopting the AREA (anticipate, reflect, engage, and act) framework [25]. Our proposed approach models identity-driven behaviours within a robotics engineering team, using software process data from individual engineers' contributions. We anticipate these models —along with the data used to build them— can be used to identify specific engineers and how they conform to a team's identity. Our approach is designed to diagnose groups, and its application to assess individuals is a clear misuse. Using the model's outputs to single out and penalise team members is a risk we need to mitigate. Our plan is to apply differential privacy techniques [8] to the process data we extract, to guarantee the privacy of the members of a robotics team under study.

Acting in line with a group identity enables a sense of responsibility. Research has shown that a strong common identity involves a sense of duty, for acting in certain ways consistent with in-group values. By understanding a robotics team value system, responsibility within the team is turned from an individual's expectation into a team's expectation, meaning that taking the praise or blaming becomes a group attribute. Thinking about responsibility in this way may reduce psychological distress, especially when creating autonomous systems for time-constrained situations, such as emergencies.

In taking a mixed methods approach to examine cultural aspects of software engineering for autonomous systems [33], our vision is to translate an improved understanding of robotics engineering into societal gains. We will accomplish this by focusing on understanding the identity of developers and designing appropriate sociotechnical interventions for improving the working culture of robotics engineering teams. We can orient the altruism of the open-source movement towards urgent and neglected engineering challenges, like misinformation detection in non-English languages [31]. We may also alleviate the robotics engineers shortage, increasing the hiring pool by addressing its gender-diversity problem [32].

ACKNOWLEDGMENTS

This work was supported by the Engineering and Physical Sciences Research Council [grant numbers EP/V026747/1, EP/R013144/1]; and Science Foundation Ireland [grant number 13/RC/2094_P2].

REFERENCES

[1] George A. Akerlof and Rachel E. Kranton. 2000. Economics and Identity*. *The Quarterly Journal of Economics* 115, 3 (08 2000), 715–753. https://doi.org/10.1162/003355300554881 arXiv:https://academic.oup.com/qje/article-pdf/115/3/715/5174909/115-3-715.pdf

[2] Sarah Beecham, Nathan Baddoo, Tracy Hall, Hugh Robinson, and Helen Sharp. 2008. Motivation in Software Engineering: A systematic literature review. *Information and software technology* 50, 9-10 (2008), 860–878.

[3] Amel Bennaceur, Diane Hassett, Bashar Nuseibeh, and Andrea Zisman. 2023. Values@ Runtime: An Adaptive Framework for Operationalising Values. (2023).

[4] Alan F. Blackwell, Marian Petre, and Luke Church. 2019. Fifty years of the psychology of programming. *Int. J. Hum. Comput. Stud.* 131 (2019), 52–63. https://doi.org/10.1016/j.ijhcs.2019.06.009

[5] EunJeong Cheon and Norman Makoto Su. 2016. Integrating Roboticist Values into a Value Sensitive Design Framework for Humanoid Robots. In *The Eleventh ACM/IEEE International Conference on Human Robot Interation, HRI 2016, Christchurch, New Zealand, March 7-10, 2016*, Christoph Bartneck, Yukie Nagai, Ana Paiva, and Selma Sabanovic (Eds.). IEEE/ACM, 375–382. https://doi.org/10.1109/HRI.2016.7451775

[6] Bohkyung Chun and Heather Knight. 2020. The Robot Makers: An Ethnography of Anthropomorphism at a Robotics Company. *J. Hum.-Robot Interact.* 9, 3, Article 16 (jun 2020), 36 pages. https://doi.org/10.1145/3377343

[7] G Cockton. 2004. Value-centric HCI. In *NordHCI*.

[8] Cynthia Dwork. 2008. Differential Privacy: A Survey of Results. In *Theory and Applications of Models of Computation—TAMC* (theory and applications of models of computation—tamc ed.) *(Lecture Notes in Computer Science, Vol. 4978)*. Springer Verlag, 1–19. https://www.microsoft.com/en-us/research/publication/differential-privacy-a-survey-of-results/

[9] César França, Fabio QB Da Silva, and Helen Sharp. 2018. Motivation and satisfaction of software engineers. *IEEE Transactions on Software Engineering* 46, 2 (2018), 118–140.

[10] Sergio García, Daniel Strüber, Davide Brugali, Thorsten Berger, and Patrizio Pelliccione. 2020. Robotics software engineering: a perspective from the service robotics domain. In *ESEC/FSE '20: 28th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering, Virtual Event, USA, November 8-13, 2020*, Prem Devanbu, Myra B. Cohen, and Thomas Zimmermann (Eds.). ACM, 593–604. https://doi.org/10.1145/3368089.3409743

[11] Carlos Gavidia-Calderon, Amel Bennaceur, Anastasia Kordoni, Mark Levine, and Bashar Nuseibeh. 2022. What Do You Want From Me? Adapting Systems to the Uncertainty of Human Preferences. In *44th IEEE/ACM International Conference on Software Engineering: New Ideas and Emerging Results ICSE (NIER) 2022, Pittsburgh, PA, USA, May 22-24, 2022*, Liliana Pasquale and Christoph Treude (Eds.). IEEE/ACM, 126–130. https://doi.org/10.1109/ICSE-NIER55298.2022.9793539

[12] Carlos Gavidia-Calderon, DongGyun Han, and Amel Bennaceur. 2022. Quid Pro Quo: An Exploration of Reciprocity in Code Review. In *19th IEEE/ACM International Conference on Mining Software Repositories, MSR 2022, Pittsburgh, PA, USA, May 23-24, 2022.* ACM, 738–740. https://doi.org/10.1145/3524842.3528522

[13] Carlos Gavidia-Calderon, Federica Sarro, Mark Harman, and Earl T. Barr. 2020. Game-theoretic analysis of development practices: Challenges and opportunities. *J. Syst. Softw.* 159 (2020). https://doi.org/10.1016/j.jss.2019.110424

[14] Carlos Gavidia-Calderon, Federica Sarro, Mark Harman, and Earl T. Barr. 2021. The Assessor's Dilemma: Improving Bug Repair via Empirical Game Theory. *IEEE Trans. Software Eng.* 47, 10 (2021), 2143–2161. https://doi.org/10.1109/TSE.2019.2944608

[15] Robert L. Glass. 2000. Loyal Opposition - The Sociology of Open Source: Of Cults and Cultures. *IEEE Softw.* 17, 3 (2000), 104–105. https://doi.org/10.1109/MS.2000.10027

[16] Georgios Gousios, Andy Zaidman, Margaret-Anne D. Storey, and Arie van Deursen. 2015. Work Practices and Challenges in Pull-Based Development: The Integrator's Perspective. In *37th IEEE/ACM International Conference on Software Engineering, ICSE 2015, Florence, Italy, May 16-24, 2015, Volume 1*, Antonia Bertolino, Gerardo Canfora, and Sebastian G. Elbaum (Eds.). IEEE Computer Society, 358–368. https://doi.org/10.1109/ICSE.2015.55

[17] Diane Hassett, Amel Bennaceur, and Bashar Nuseibeh. 2023. Feel It, Code It: Emotional Goal Modelling for Gender-Inclusive Design. (2023).

[18] C. Heath, C. Crivellaro, and L. Coles-Kemp. 2019. Relations are more than Bytes: Re-thinking the Benefits of Smart Services with People and Things. In *CHI Conf. on Human Factors in Computing Systems*. ACM.

[19] W Hussain, D Mougouei, and J Whittle. 2018. Integrating social values into software design patterns. In *International Workshop on Software Fairness*. ACM/IEEE.

[20] Brian Kennedy, Richard Fry, and Cary Funk. 2021. 6 facts about America's STEM workforce and those training for it. (2021).

[21] John C. Knight and Nancy G. Leveson. 2002. Should software engineers be licensed? *Commun. ACM* 45, 11 (2002), 87–90. https://doi.org/10.1145/581571.581601

[22] Ruxandra Lupu, Massimo Caccia, Enrica Zereik, and Rosangela Barcaro. 2022. Women in Blue: Toward a Better Understanding of the Gender Gap in Marine Robotics [Women in Engineering]. *IEEE Robotics Autom. Mag.* 29, 4 (2022), 138–140. https://doi.org/10.1109/MRA.2022.3213467

[23] Roger B. Myerson. 1997. *Game theory - Analysis of Conflict.* Harvard University Press. http://www.hup.harvard.edu/catalog/MYEGAM.html

[24] Inbal Nahum-Shani, Shawna N Smith, Bonnie J Spring, Linda M Collins, Katie Witkiewitz, Ambuj Tewari, and Susan A Murphy. 2018. Just-in-time adaptive interventions (JITAIs) in mobile health: key components and design principles for ongoing health behavior support. *Annals of Behavioral Medicine* 52, 6 (2018), 446–462.

[25] Richard Owen, Phil Macnaghten, and Jack Stilgoe. 2012. Responsible research and innovation: From science in society to science for society, with society. *Science and public policy* 39, 6 (2012), 751–760.

[26] H Perera, R Hoda, R Shams, A Nurwidyantoro, M Shahin, W Hussain, and J Whittle. 2021. The impact of considering human values during requirements engineering activities. *IEEE Transactions on Software Engineering* (2021).

[27] A Rashid, J Weckert, and R Lucas. 2019. Software engineering ethics in a digital world. *IEEE Computer* (2019).

[28] Irum Rauf, Marian Petre, Thein Tun, Tamara Lopez, Paul Lunn, Dirk van der Linden, John N. Towse, Helen Sharp, Mark Levine, Awais Rashid, and Bashar Nuseibeh. 2022. The Case for Adaptive Security Interventions. *ACM Trans. Softw. Eng. Methodol.* 31, 1 (2022), 9:1–9:52. https://doi.org/10.1145/3471930

[29] Mojtaba Shahin, Waqar Hussain, Arif Nurwidyantoro, Harsha Perera, Rifat Ara Shams, John C. Grundy, and Jon Whittle. 2021. Operationalizing Human Values in Software Engineering: A Survey. *ArXiv* abs/2108.05624 (2021).

[30] Helen Sharp, Mark Woodman, Fiona Hovenden, and Hugh Robinson. 1999. The Role of 'Culture' in Successful Software Process Improvement. In *25th EUROMICRO '99 Conference, Informatics: Theory and Practice for the New Millenium, 8-10 September 1999, Milan, Italy*. IEEE Computer Society, 2170. https://doi.org/10.1109/EURMIC.1999.794778

[31] Renato Moraes Silva, Roney L. S. Santos, Tiago A. Almeida, and Thiago A. S. Pardo. 2020. Towards automatically filtering fake news in Portuguese. *Expert Syst. Appl.* 146 (2020), 113199. https://doi.org/10.1016/j.eswa.2020.113199

[32] Laurel Smith-Doerr, Sharla N Alegria, and Timothy Sacco. 2017. How diversity matters in the US science and engineering workforce: A critical review considering integration in teams, fields, and organizational contexts. *Engaging Science, Technology, and Society* 3 (2017), 139–153.

[33] Margaret-Anne D. Storey, Neil A. Ernst, Courtney Williams, and Eirini Kalliamvakou. 2020. The who, what, how of software engineering research: a socio-technical framework. *Empir. Softw. Eng.* 25, 5 (2020), 4097–4129. https://doi.org/10.1007/s10664-020-09858-z

[34] Henri Tajfel, John C Turner, William G Austin, and Stephen Worchel. 1979. An integrative theory of intergroup conflict. *Organizational identity: A reader* 56, 65 (1979), 9780203505984–16.

[35] Bianca Trinkenreich, Ricardo Britto, Marco Aurélio Gerosa, and Igor Steinmacher. 2022. An Empirical Investigation on the Challenges Faced by Women in the Software Industry: A Case Study. In *44th IEEE/ACM International Conference on Software Engineering: Software Engineering in Society ICSE (SEIS) 2022, Pittsburgh, PA, USA, May 22-24, 2022*. IEEE/ACM, 24–35. https://doi.org/10.1109/ICSE-SEIS55304.2022.9793931

[36] Bianca Trinkenreich, Mariam Guizani, Igor Wiese, Tayana Conte, Marco Gerosa, Anita Sarma, and Igor Steinmacher. 2021. Pots of Gold at the End of the Rainbow: What is Success for Open Source Contributors? *IEEE Transactions on Software Engineering* 48, 10 (2021), 3940–3953.

[37] Bianca Trinkenreich, Klaas-Jan Stol, Anita Sarma, Daniel M German, Marco A Gerosa, and Igor Steinmacher. 2023. Do I Belong? Modeling Sense of Virtual Community Among Linux Kernel Contributors. *International Conference of Software Engineering* (2023).

[38] Gentiane Venture. 2014. Pioneer Women in Robotics [Women in Engineering]. *IEEE Robotics Autom. Mag.* 21, 2 (2014), 98–99. https://doi.org/10.1109/MRA.2014.2314034

[39] Eugene Vinitsky, Raphael Köster, John P. Agapiou, Edgar A. Duéñez-Guzmán, Alexander Sasha Vezhnevets, and Joel Z. Leibo. 2021. A learning agent that acquires social norms from public sanctions in decentralized multi-agent settings. *CoRR* abs/2106.09012 (2021). arXiv:2106.09012 https://arxiv.org/abs/2106.09012

[40] Yi Wang and David F. Redmiles. 2019. Implicit gender biases in professional software development: an empirical study. In *Proceedings of the 41st International Conference on Software Engineering: Software Engineering in Society, ICSE 2019, Montreal, QC, Canada, May 25-31, 2019*, Rick Kazman and Liliana Pasquale (Eds.). ACM, 1–10. https://doi.org/10.1109/ICSE-SEIS.2019.00009

[41] Jon Whittle. 2019. Is Your Software Valueless? *IEEE Software* (2019).

[42] Emily Winter, Stephen Forshaw, Lucy Hunt, and Maria Angela Ferrario. 2019. Advancing the study of human values in software engineering. In *2019 IEEE/ACM 12th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE)*. IEEE, 19–26.

[43] Karolina Zawieska. 2020. Disengagement with ethics in robotics as a tacit form of dehumanisation. *AI Soc.* 35, 4 (2020), 869–883. https://doi.org/10.1007/s00146-020-01000-3

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009