Trends and Updated Research Agenda for Autonomous Agile Teams: A Summary of the Second International Workshop at XP2019

Nils Brede Moe[1(✉)], Viktoria Stray[1,2], and Rashina Hoda[3]

[1] SINTEF, Trondheim, Norway
nilsm@sintef.no
[2] University of Oslo, Oslo, Norway
stray@ifi.uio.no
[3] University of Auckland, Auckland, New Zealand
r.hoda@auckland.ac.nz

Abstract. To succeed in complex environments and handle the innovation, development and support, organizations have to find ways to support and regulate the autonomy of teams according to the environmental demands and limitations. Furthermore, there is no one-size-fits-all autonomy approach. The process of forming and implementing autonomous teams, as well as the effective functioning of such teams, is not yet adequately addressed and understood in the context of complex knowledge-intensive organizations. The second international workshop on autonomous teams investigated barriers for team autonomy: "What are the real-world problems to be solved for autonomous teams?" and "What concepts from the literature can be used to solve the problems?"

Keywords: Autonomous teams · Agile · Team design · Coordination · Self-managing teams · Self-organizing teams

1 Introduction

Digitalization is changing the competitive landscape in many industries. A company conducting a digital transformation needs to (1) cultivate the leadership for such transformation, (2) develop an agile and scalable platform on which digital product and services can be delivered, (3) design new digitally enabled customer experiences, and (4) incubate and accelerate emerging digital innovations [1]. Handling these four capabilities at the same time is a complex task on many levels in an organization. Teams designing new digitally-enabled customer experiences and incubating and accelerating emerging digital innovations are challenged by increasingly complex problems that also involve external actors. For example, a design or development team needs to interact with many experts outside of their team and department [2], needs to rely on a number of technologies and work processes, and frequently makes decisions with economic consequences. High productivity, innovation, accuracy of problem

solving, and fast decision-making are handled best by autonomous teams [3–6] (also known as self-organizing or self-managing teams).

Autonomous teams are described as teams given freedom by management [7] that take on the responsibilities of their supervisors [8], and are composed of people with a variety of skills to effectively tackle the variety in their external environments [9]. There is a high adoption rate of autonomous teams in many sectors, such as ICT, telecom, finance and banking, energy, transport and manufacturing. In the ICT industry, autonomous teams are exemplified by extensive uptake of agile methods [10–12].

The process of forming and implementing teams with high autonomy, as well as the effective functioning of such teams, is not yet adequately addressed and understood in the context of complex team-based, knowledge-intensive organizations [5]. Moreover, research on teams has predominantly been based on cross-sectional survey data, often involving student teams, and has not sufficiently taken into account the complexity in which teams operate [5]. We argue that more research is needed on roles, management, leadership, authority, decision-making, learning, technology, and the role of such teams in networks of autonomous teams. We know so far that the emergence of informal self-organizing roles facilitates the transitions in team practices and management approaches in the process of becoming agile [13].

1.1   Roles in Autonomous Teams

Traditional software development teams are comprised of formal functional roles such as developers, testers, and project managers. Agile methods (e.g., Scrum) replaced these with new formal roles (e.g., the Scrum master and product owner) that represent a cross-functional collection of old traditional (e.g. developers and testers), while also increasingly including other formal roles formerly beyond the core technical team boundary (e.g. the business analysts, user interface designers, and, more recently, artificial intelligence and machine learning experts).

While the composition of the software development team became more diverse and inclusive, these new and expanded formal roles alone do not form the basis of autonomous teams. What makes a software development team autonomous is the presence of temporary and spontaneously emerging informal roles, such as mentor, coordinator, translator, champion, promoter, and terminator [10]. These roles focus on: (1) mentoring the new agile team into agile ways of working and autonomy, (2) coordinating with the customer on a regular and detailed basis, (3) translating between the business language used by the customer and the technical language employed by the team, (4) championing the cause of agile and autonomous teams with senior management in case of bottom-up adoptions and championing agile transformation and autonomy with the teams in case of top-down adoptions, (5) promoting agility with the customer and educating the customer on his or her role and responsibility in supporting autonomous teams, and (6) terminating personnel not contributing positively to the agile ways of working and threatening the autonomous functioning of the team. Through the emergence of these roles, the team becomes self-organizing, with both the team and managers moving toward team-driven practices and empowering management approach respectively [13].

2   The Workshop and Papers Presented

The workshop included one keynote speech, "What makes your team self-organizing?" by Rashina Hoda from the University of Auckland. Further, the workshop had six presentations by researchers who had had their papers peer- reviewed. In a multiple-case study of three large organizations that implemented the Scaled Agile Framework (SAFe), Gustavsson [14] found that when several teams gather in joint events the teams get a better overview, higher feeling of autonomy, and the ability to stop planned work when it becomes too much. However, SAFe required the use of detailed plans and routines that somewhat reduced the team autonomy because the team members felt they had less mandate in choosing what to work on. In their study on large-scale agility in Bosch, Speigler et al. [15] found that low external autonomy limited team autonomy because of factors related to hierarchies, learning, structural dependencies, and rigid processes. They suggest companies need to foster learning organizations by providing time for communities of practice and open space meetings and tools to support transparency. Mikalsen et al. [16] relied on the Modern Sociotechnical Theory (MST) to understand team autonomy in a large, agile program with cross-functional BizDev teams. Their findings suggest organizations need to balance between agility and more standardized ways of working, and the de-bureaucratization ideal from MST is challenging to accomplish in complex organizations. Barriers for autonomy and efficiency included team members being dispersed, lack of team continuity, and use of shared resources. Petit [17] collected data, such as the quality of prior releases, to assess team autonomy from 70 teams in a bank. The teams were assessed using five levels of autonomy, and the effects of the assessment included teams governing each other as opposed to managers doing it, improved accountability of teams, reduction in time required for release approval, and reduced attempts to find workarounds and loopholes. Salameh and Bass [18] investigated how a large organization tailored agile practices to balance team autonomy and alignment. They reported on factors that promote the effectiveness of autonomous teams, such as informal and on-demand knowledge-sharing, collective code ownership, and the use of Lean Startup. Finally, Hukkelberg and Berntzen [19] reported on the challenges associated with integrating the data science role in agile autonomous teams, such as a lack of knowledge of the data science role (leading to sub-optimal use of the data scientist), the use of agile practices, and the lack of infrastructure. They suggested teams that want to expand with data scientist roles should arrange team kick-off, adjust their agile practices, use communities of practice, and provide training about the data science role and machine learning.

In addition to the paper presentation there were two interactive sessions. In the first session, we collected feedback on team size for autonomous teams and the barriers for such teams using Mentimeter (a tool for receiving feedback from the participants). The second session was a group discussion on the real-world problems to be solved for autonomous teams and concepts from the literature that can be used to solve the problems. These real-world problems motivated a discussion that led to a research agenda.

3   Barriers for Autonomous Teams

Mutual adjustment tends to be the primary coordination mechanism in autonomous teams. However, mutual adjustment in its pure form requires everyone to communicate with everyone. Therefore, the teams need to be dense, and since our communication abilities are limited that means they also have to be small. At the same time more and more teams are becoming BizDevOps teams (business, development, and operations in a team) to increase their authority, which often leads to large teams. We asked the workshop participants about the best team size for autonomous agile teams; 23% answered four to five members, 23% answered eight to nine members, and 54% answered six to seven members.

The actual performance of an autonomous agile team depends not only on the competence of the team itself in managing and executing its work but also on the organizational context. In the 2018 workshop, barriers were identified [12]. During the 2019 workshop, eight barriers for external autonomy were rated on a scale from 1 (not a barrier) to 10 (extreme barrier) (see Fig. 1).
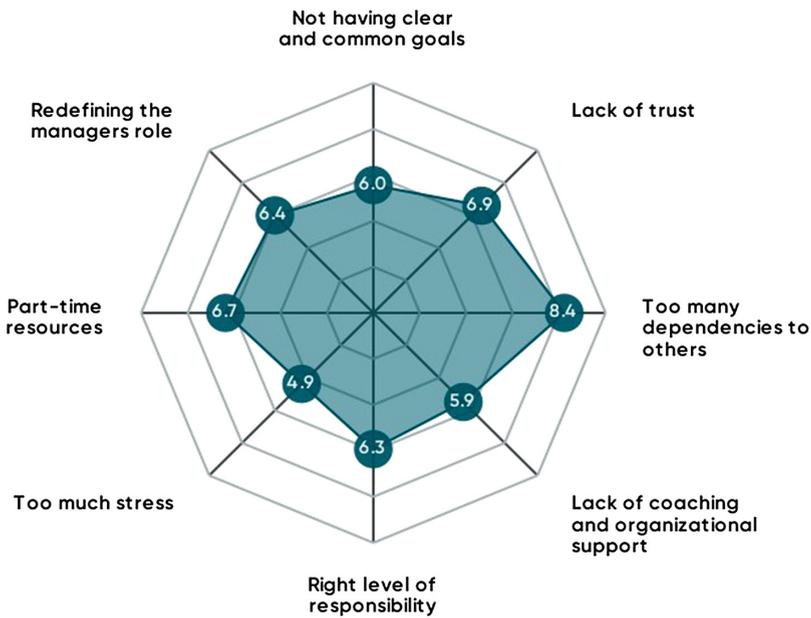

Fig. 1. Barriers for autonomous teams.

4   Research Agenda

Five areas of concern emerged at the workshop in 2018 [12] to help understand how companies can effectively enable autonomous agile teams: leadership, coordination, an organizational context supporting autonomy, design of autonomous agile teams, and

internal team processes. Each area suggested research questions that can be used to identify factors that increase, moderate, or limit the level of team autonomy, and the effects of autonomy on team performance were suggested. In this year's workshop, the above-mentioned topics were prioritized. The rating was (1) coordination, (2) organizational context supporting autonomy, (3) leadership, (4) design of autonomous agile teams, and (5) internal team processes.

In the last part of the workshop, the real-world problems that need to be solved were discussed (Table 1). Concepts from the literature that can be used to study these problems were suggested (e.g., Modern Sociotechnical Theory, Coordination Theory, Actor Network Theory, and the General Theory of Software Engineering). When investigating problems related to the personality of team members, theories like Big five and Myers Briggs were suggested. Other suggested areas of concern were onboarding, shared mental models, sourcing, team, and multiteam systems.

Table 1. Real world problems to be solved.

Area of focus	Problem to be solved
Complexity and roles	Complex products and domains increase the need for large cross functional teams (e.g. BizDevOps). But smaller teams are more efficient than larger teams
Distributed teams	Distribution requires formation of virtual and remote teams. Creating virtual autonomous teams is a challenge. How to collaborate with outsourced teams
Inter-team coordination	Large projects and programs slow down the team. How can coordination be more efficient? How to balance the alignment of many teams vs. team autonomy?
Organizational structure	Autonomous teams need large networks but also work without too many interruptions. There is a problem building a network fast and maintaining the network and, at the same time, have time to do solve the team's tasks
Personality	Personality can impact communication, coordination, and decision-making. However, you often have limited influence on who is on the team. What personality types or attributes foster autonomy? How to handle individual vs. team autonomy?
Middle management and governance	Middle management need to support the teams and give the teams authority. What is the role of middle management, who decides what, and how do you balance management control vs. team autonomy? What legacy roles are needed?
Team membership	Stable teams is one key factor. However, teams grow and need new competence. Further, people want to change teams, projects and even companies. So how do you handle rotation and onboarding of new members?
Define and measure	The organization needs some control and feedback. However, the team should not collect data that is not used for the team to improve

5   Conclusion

This paper gives an overview of what practitioners and researchers in the field of agile software development believe are emergent research themes for autonomous teams. Top-rated barriers for autonomous teams were (1) too many dependencies on others, (2) lack of trust, and (3) part-time resources. Further top-rated research topics for future research are coordination, organizational context supporting autonomy, and leadership.

References

1. Sia, S.K., Soh, C., Weill, P.: How DBS bank pursued a digital business strategy. MIS Q. Executive **15**, 105–121 (2016)
2. Šmite, D., Moe, N.B., Šāblis, A., Wohlin, C.: Software teams and their knowledge networks in large-scale software development. Inf. Softw. Technol. **86**, 71–86 (2017)
3. Cohen, S.G., Bailey, D.E.: What makes teams work: group effectiveness research from the shop floor to the executive suite. J. Manag. **23**, 239–290 (1997)
4. Hoegl, M., Parboteeah, K.P.: Autonomy and teamwork in innovative projects. Hum. Resour. Manag. **45**, 67–79 (2006)
5. Mathieu, J., Maynard, M.T., Rapp, T., Gilson, L.: Team effectiveness 1997–2007: a review of recent advancements and a glimpse into the future. J. Manag. **34**, 410–476 (2008)
6. Hollenbeck, J.R., Beersma, B., Schouten, M.E.: Beyond team types and taxonomies: a dimensional scaling conceptualization for team description. Acad. Manag. Rev. **37**, 82–106 (2012)
7. Takeuchi, H., Nonaka, I.: The new product development game. Harv. Bus. Rev. **64**, 137–146 (1986)
8. Trist, E.: The evolution of socio-technical systems: a conceptual framework and an action research program. Ontario Quality of Working Life Centre (1981)
9. Morgan, G.: Images of Organizations. SAGE Publications, Thousand Oaks (2006)
10. Hoda, R., Noble, J., Marshall, S.: Self-organizing roles on agile software development teams. IEEE Trans. Softw. Eng. **39**, 422–444 (2013)
11. Dingsøyr, T., Nerur, S., Balijepally, V., Moe, N.B.: A decade of agile methodologies: towards explaining agile software development. J. Syst. SW **85**(6), 1213–1221 (2012)
12. Stray, V., Moe, N.B., Hoda, R.: Autonomous agile teams: challenges and future directions for research. In: Proceedings of the 19th International Conference on Agile Software Development: Companion, pp. 1–5. ACM, Porto (2018)
13. Hoda, R., Noble, J.: Becoming agile: a grounded theory of agile transitions in practice. In: Proceedings of the 39th International Conference on Software Engineering (2017)
14. Gustavsson, T.: Voices from the teams - impacts on autonomy in large-scale agile software development settings. In: Hoda, R. (ed.) XP 2019 Workshops. LNBIP, vol. 364, pp. 29–36. Springer, Cham (2019)

15. Spiegler, S.V., Heinecke, C., Wagner, S.: The influence of culture and structure on autonomous teams in established companies. In: Hoda, R. (ed.) XP 2019 Workshops. LNBIP, vol. 364, pp. 46–54. Springer, Cham (2019)
16. Mikalsen, M., Næsje, M., Reime, E.A., Solem, A.: Agile autonomous teams in complex organizations. In: Hoda, R. (ed.) XP 2019 Workshops. LNBIP, vol. 364, pp. 55–63. Springer, Cham (2019)
17. Petit, Y., Marnewick, C.: Earn your wings: a novel approach to deployment governance. In: Hoda, R. (ed.) XP 2019 Workshops. LNBIP, vol. 364, pp. 64–71. Springer, Cham (2019)
18. Salameh, A., Bass, J.M.: Spotify tailoring for promoting effectiveness in cross-functional autonomous squads. In: Hoda, R. (ed.) XP 2019 Workshops. LNBIP, vol. 364, pp. 20–28. Springer, Cham (2019)
19. Hukkelberg, I., Berntzen, M.: Exploring the challenges of integrating data science roles in agile autonomous teams. In: Hoda, R. (ed.) XP 2019 Workshops. LNBIP, vol. 364, pp. 37–45. Springer, Cham (2019)