The Journal of Systems & Software 194 (2022) 111489

Contents lists available at ScienceDirect

The Journal of Systems & Software

journal homepage: www.elsevier.com/locate/jss

# The effect of advice network connectedness on problem-solving competence among software developers ☆

Matthew O. Ajimati [a,b,*], Eoin Whelan [a], Rick Aalbers [c], Lorraine Morgan [d], Hans van Kranenburg [b]

[a] Business Information Systems, J.E. Cairnes School of Business & Economics, National University of Ireland, Galway, Ireland
[b] Institute for Management Research, Radboud University, Heyendaalseweg 141 6525AJ, Nijmegen, The Netherlands
[c] Institute for Management Research, Centre for Organisation Restructuring, Radboud University, Heyendaalseweg 141 6525AJ, Nijmegen, The Netherlands
[d] Business Information Systems, Lero – Irish Software Engineering Centre, J.E. Cairnes School of Business & Economics, National University of Ireland, Galway, Ireland

ARTICLE INFO

*Article history:*
Received 18 November 2021
Received in revised form 11 August 2022
Accepted 12 August 2022
Available online 23 August 2022

*Keywords:*
Advice network
Software development
Problem-solving competence
Software developers
Contact quality
Brokerage

ABSTRACT

Software development requires software developers to share knowledge and solve problems together. Although researchers have considered the business and technical knowledge germane to performing software development tasks, empirical studies investigating business and technical advice networks on problem-solving competence is scarce. Using social network theory, we argue that software developers must be embedded for knowledge brokering within and across business and technical advice connectedness for improving problem-solving competence. Moreover, we argue that contact quality matters in increasing or decreasing individual problem-solving competence. We present data collected via an online survey from 153 respondents in a professional software organisation. Our findings suggest that software developers who engage in knowledge brokering in business and technical advice connectedness will increase problem-solving competence in the software development effort. Our findings also reveal no significant effect of contact quality between these advice networks and problem-solving competence. We discuss our findings' implications for theory and practice.



## 1. Introduction

There is much empirical evidence relating to the effect of social networks on different performance outcomes in software development (Bernardi et al., 2018; Peng et al., 2013; Sykes et al., 2014). For example, prior work has emphasised the role of social networks, specifically advice networks, in helping individuals search for knowledge required to fill a work-related knowledge gap (Brown and Duguid, 1998; McGrath et al., 2003). The quality of contacts in social networks, however, has also been considered beneficial in addressing work demands and improving socio-emotional benefits among coworkers (Burmeister et al., 2021; Fasbender et al., 2020). Social network antecedents have been linked to the ability to create, generate, and share knowledge with others to solve problems, promote learning, and achieve task performance (Gibbons, 2004; Siciliano, 2015; Sykes et al., 2014). As part of the rich body of literature, studies have shown that an individual actor's embeddedness in technical advice networks as a broker systematically facilitates knowledge sharing, improving the software development effort and performance of software development teams as a result (Cataldo and Herbsleb, 2008; Ehrlich and Cataldo, 2014). Other studies have linked software developers' embeddedness in both the business domain (knowledge of general business processes and customer needs) and technical advice (technology knowledge needed to solve and deliver customer solutions) networks, for example on team performance (Mehta and Bharadwaj, 2015; Sykes et al., 2014; Tiwana, 2004). In an empirical examination of enterprise systems implementation and conceptually drawing from social network theory, Sykes et al. (2014) for instance, suggest that access to workflow advice helps to fix non-software related organisational tasks while software advice helps to fix system errors and constraints related to technology tasks in performing software applications.

Software development is a problem-solving activity that often demands software developers within information systems and software engineering to drive solutions to technical problems (e.g., fixing software bugs) and business domain problems

☆ Editor: Kelly Blincoe.

* Corresponding author at: Business Information Systems, J.E. Cairnes School of Business & Economics, National University of Ireland, Galway, Ireland.
*E-mail addresses:* m.ajimati1@nuigalway.ie (M.O. Ajimati), eoin.whelan@nuigalway.ie (E. Whelan), r.aalbers@fm.ru.nl (R. Aalbers), lorraine.morgan@nuigalway.ie (L. Morgan), h.vankranenburg@fm.ru.nl (H. van Kranenburg).

https://doi.org/10.1016/j.jss.2022.111489
0164-1212/

(e.g., interpreting requirements) during the development process (Conaldi and Lomi, 2013; Rus et al., 2002; Tiwana, 2004). Social network matters in such problem-solving. The literature on organisation networks has pointed out that solving problems stands to benefit from access to and integration of both business domains and technical advice to improve problem-solving skills (Sykes et al., 2014; Tiwana, 2004). Lin et al. (2015) for example find that knowledge complement (i.e., "*heterogeneous capability, experience, knowledge, and skill posses by various experts*") increases the team's problem-solving competence and project performance (Lin et al., 2015, p. 1695). Likewise, other literature has advocated for reactive and anticipatory (proactive) mechanisms (e.g., available and findable expertise, experience and knowledge resources) in the organisation to be linked to increasing general team problem-solving competence as a requirement to address requirements uncertainty at different development stages in software development to maintain software quality (Li et al., 2011). Further studies have examined the alignment between the social and technical capability of software development teams on software development performance (Maheshwari et al., 2012). Overall, research has shown that individuals' interactions with a pool of information and knowledge resources and their collective integration drive timely, less costly and quality solutions to problems, and have resulted in efficient and cost-effective product performance (Sheremata, 2000).

Despite these contributions, limited empirical evidence exists regarding the role of advice networks and their impact on problem-solving competence (i.e., developing the capacity to solve problems) of software development professionals (Lin et al., 2015; Siciliano, 2015). This is important as numerous studies demonstrate that advice networks are important in explaining performance outcomes. To date, however, we do not have a complete picture of the causality undergirding this interaction, as prior studies often rely on a uniplex instead of a multiplex view of networks. Prior work, for example, resides on a conceptual design where business domain and technical advice interactions are studied separately in their respective software development settings (Cataldo and Herbsleb, 2008; Ehrlich and Cataldo, 2014). These prior studies offer limited insights into the dynamics of these networks, which the multiplex network approach can address (Ajimati et al., 2021; Schreiber and Zylka, 2020; Tiwana, 2012; Vermerris et al., 2014). This study extends our understanding of the effect of advice networks, in particular, the patterns of behaviour on problem-solving competence by incorporating a multiplex network approach to these advice networks. In doing so we shed light on the behavioural patterns of software developers in diverse advice networks in a software development context.

A multiplex view, as opposed to a uniplex view, considers more than one advice network relation or observations between actors, hence, allowing for conceptual overlap in a single study (Ajimati et al., 2021; Aalbers and Dolfsma, 2015; Belso-Martínez et al., 2017; Snijders et al., 2013). For example, a multiplex study by Belso-Martínez et al. (2017) found that structural brokerage (the role of actors in connecting disconnected others with knowledge resources) in business and technical advice networks results in improved innovation performance between organisations. Others have found that the focus of previous research on uniplex ties can be enriched by including rich ties in a multiplex network, for instance also as predictive of ideation activity within a firm (Aalbers and Dolfsma, 2015). Understanding the antecedents of problem-solving competence has typically been at the team level of analysis in prior software development research (e.g., Atuahene-Gima and Wei, 2011; Li et al., 2011; Lin et al., 2015), which encourage us to apply a multiplex view to such team interactions. Insofar, our research question is: *What is the relationship between diverse advice networks and individual problem-solving competence in software development?* To theoretically and empirically capture the dynamic of different knowledge interactions in informal advice networks in software development, we present our theory and hypotheses development in the next section.

## 2. Theory and hypotheses development

### 2.1. Problem-solving competence and software development

Problems are inherent to human activity, be it task-related or not, and are varied in different intensities. The extent to comprehending such problems can determine whether it can be self-managed or requires seeking interpersonal advice (MacGeorge et al., 2016), in particular because individuals differ in their cognitive load to process complex work-related problems (Burmeister et al., 2021; Funke, 2010). Problems are described as knowledge gaps between the present state and an end state (Lin et al., 2015; Thomas et al., 2002). They can be simple or complex, but complex problems are dynamic and interconnected with different issues, and thus require greater information processing and solution processes (Funke, 2010) in terms of problem-solving requirements. Competence, on the other hand, refers to a specialised system of knowledge, abilities, cognitive and domain skills, proficiencies, and behavioural attitude of individuals and teams to be proactive and reactive to performing familiar and unfamiliar tasks (Amabile, 1983; Li et al., 2011; Lin et al., 2015; Šmite et al., 2017). Perry-Smith (2006) links cognitive and domain skills to creativity by being divergent, flexible, and evaluative in synthesising unrelated concepts and making sense of them in guiding solutions to problems (Aldave et al., 2019). This is why problem-solving through creative solutions goes beyond an established convention in synthesising processes, information, and knowledge, to charting a new solution path forward toward achieving performance (Amabile, 1983; Hoegl and Parboteeah, 2007).

Problem-solving competence, therefore, is concerned with the process of identifying, clarifying, evaluating, and reviewing alternatives, in implementing creative and cost-saving solutions to a problem (Açıkgöz and Ö. İlhan, 2015; Atuahene-Gima and Wei, 2011; Lin et al., 2015; Sheremata, 2000). Primarily software development revolves around requirement analysis, design, development (coding/configuring), testing, and deployment activity (Iden and Bygstad, 2018; Sawyer et al., 2010). Hence, software developers are often charged to solve quality-related problems such as eliminating bugs and addressing design, comprehension, and redundancy problems (Balaban et al., 2015; Coughlan et al., 2003; Pitts and Browne, 2007). However, research suggests that brokers are needed to facilitate socialising and communicating with others in order to share knowledge and improve shared understanding, which is often required to comprehend and solve work problems during software development (Boden and Avram, 2009; Whelan et al., 2015, 2011). Researchers have also argued that recipients of knowledge can experience overloading and depletion of knowledge if they consult with low-quality contacts. However, such cognitive costs can be mitigated when the contact quality is high, as concrete and less ambiguous knowledge are shared by the providers to drive action and influence positive outcomes (Burmeister et al., 2021). Therefore, the need to develop or increase problem-solving competence is critical in driving software development efforts since individual competence requires cultivating the right knowledge, skills, attitude, and motivation to put them into action towards providing solutions to problems (Amabile, 1983; Gurteen and Associates, 1998); Yuzhu (Li et al., 2011; Lin et al., 2015). It is also worth noting that problem-solving requires people to collaborate and share different knowledge to get work done (Aladwani, 2002; Burmeister

et al., 2021), particularly when creative solutions are required in generative tasks (Gloor et al., 2007). We focus on business domain and technical knowledge to examine the role of informal advice networks on the problem-solving competence of software developers. Considering the pertinent role of knowledge in performing development tasks and its access in advice networks, we will now expatiate their relationship to develop our hypotheses for this study.

### 2.2. Advice network connectedness and problem-solving competence

The basic notion of social network theory has focused on the link between social connections, brokers and knowledge resource benefits between people and organisations and what it means for network outcomes (Allen, 1984; Freeman, 1978; Wellman, 1983). In general, the theoretical lens of social networks explains the structural patterns of advice networks as mechanisms to understand how information and knowledge diffuse among individuals or groups and build social capital to their advantage (Reagans and McEvily, 2003). McGrath et al. (2003, p. 3) assert that "*when individuals seek advice, they use interpersonal connections to fill gaps in their knowledge or experience to resolve problems or take advantage of opportunities more quickly*" (Burmeister et al., 2021). Thus, individuals seek advice to derive solutions to problems, gain meta-knowledge, reformulate problems, and validate perspectives (Cross et al., 2001). Knowledge itself can be described as a mix of information, experience, skills, belief, values, rules, norms, and standards that enforces the ability to act effectively. Knowledge forms can be explicit (in semantic language — coded, e.g., books, documents) and tacit (human memory or uncoded, e.g., experiences, observations, imagination, education, creativity). While the two are important, tacit knowledge cannot be easily articulated compared to explicit knowledge (Anwar et al., 2017; Aurum et al., 2008; Stenmark, 2002). However, social interactions are needed, more so for tacit knowledge, to share, internalise and or externalise any of this knowledge for performing tasks and deriving competitive advantage (Tassabehji et al., 2019; Wang and Wang, 2012). Addressing software development requirements requires social interactions for interpersonal connections with these knowledge types and are directly linked to business domain and technical knowledge resources among software developers (Ghobadi, 2015; Iden and Bygstad, 2018). Studies found advice sharing to be beneficial to advice-givers and advice seekers during problem-solving based on shared learning and self-evaluation to improve performance (Muthusamy and White, 2005; Shah et al., 2018; Williamson and Clark, 1989). Nonetheless, to make sense of the benefits individuals' software developers accrue when they share knowledge requires exploring the impact of business domain and technical advice networks on individual problem-solving competence.

#### *Business domain advice and problem-solving competence*

Business domain advice can be described as knowledge about business processes, rules, system support, functions, and design objectives of customers that are shared (give and/or receive) to perform software tasks among software developers (Iden and Bygstad, 2018; Rus et al., 2002; Šmite et al., 2017; Tiwana, 2004). For example, Kobayashi (2015) argues that for software developers to develop a medical system, they must first understand the practical use of the system such as listing prescription drugs before the actual software design. In other words, software developers must understand the business processes and customer needs before integrating technologies to deliver solutions. For example, when dealing with changes in project requirements or when new technology system support is introduced, it is argued that the ability to manage uncertainty demand relying on others for advice concerning the new workflow and software applications before jump-starting development (Nidumolu, 1995; Sykes et al., 2014). In doing so, task-related uncertainty can be reduced, and new insights can be generated for making good decisions. To drive quality work, individuals need to be competent to deal with task-related uncertainty, which arises because of the inability of individuals to generate sufficient information or knowledge required to predict a positive outcome. This is important in eliminating frustration and vulnerability caused by the low knowledge application in driving solutions to problems (Bunger et al., 2018; Nidumolu, 1995; Whelan and Conboy, 2014; Zand, 1972)

Lin et al. (2015) also suggest that access to available domain knowledge tends to improve problem-solving competence. However, Pitts and Browne (2007) argue that not all software developers are well trained in searching for required information as well as interpreting requirements due to limited cognitive processes to drive elicitation success. Nevertheless, a low problem-solving effort will be required when they face familiar problems to manage the situation better (Koo and Ko, 2017). In this vein, the function of brokers as experienced knowledge agents play a key role in ensuring that different perspectives are explored intuitively and analytically to provide answers to unresolved task problems within advice networks (Lopes and Botelho, 2013; Parise et al., 2006). Overall, the application of business domain knowledge has been considered suitable in solving problems, applying knowledge to tasks, and understanding the users (Nidumolu, 1995; Rus et al., 2002). Rus et al. (2002) suggest that a lack of business domain knowledge can lead to delays in software development efforts and project success. In other words, business domain knowledge is needed to drive technical competence, making connectedness by means of business advice ties an important antecedent to achieving technology solutions to problems a business might be experiencing (Perry-Smith, 2006; Yilmaz, 2008).

We thus posit that individual connectedness in the advice network of individuals with diverse business domain knowledge influences their ability to develop the required effort needed to solve business domain problems more effectively. In doing so, we expect such connectedness to have a positive bearing on the technical knowledge of software developers in solving business problems and performing software development tasks. Thus, we present our first hypothesis (denote H1a/H1b):

***H1a**: A software developer's connectedness in the business (domain) advice network will have a positive effect on their problem-solving competence.*

*H1b: A software developer's connectedness in the business (domain) advice network will have a positive effect on technical advice connectedness.*

#### *Technical advice and problem-solving competence*

Technical advice comprises technical knowledge being shared with others with the intent to further various technical activities; in the context of software development, the design, tool development, testing, debugging, programming, or general understanding of a coding environment, shared (provided and/or received) with the intent to perform software tasks among software developers (Iden and Bygstad, 2018; Rus et al., 2002; Šmite et al., 2017; Tiwana, 2004). In such a context, Ozer and Vogel (2015) suggest that software developers learn during problem-solving how to use certain software design tools to code and tackle software problems. Thus, individuals can freely choose who they approach for advice to solve task-related problems (Tassabehji et al., 2019). For example, studies have emphasised that software developers would need to have high problem-solving skills to define and provide solutions to technical-related quality problems (Aladwani, 2002; Balaban et al., 2015; Saltz

et al., 2016; Tekinerdogan and Aksit, 2006). In doing so, software developers would need to know who has the right knowledge when exploring diverse knowledge options required to solve specific technical problems such as fixing bugs and correcting feature errors (Conaldi and Lomi, 2013; Coughlan et al., 2003; Lin et al., 2015; Šmite et al., 2017). This is because an individual problem-solving competence to timely detect and analyse bugs will facilitate meeting commercial objectives of an organisation (Colazo, 2014). This can also inform software developers to be more knowledgeable and creative in clarifying and understanding problems where related solutions have not been earlier documented (Açıkgöz and Ö. İlhan, 2015). Researchers have attributed competent, benevolent, integrity and systems trust or psychological climate, including mutual influence, as factors for driving collaborative behaviours to learn and share advice in joint problem-solving activities (Ebner et al., 2009; Levin et al., 2002; Muthusamy et al., 2005; Purvis and Zagenczyk, 2018). Similar studies found that experience of positive knowledge and reciprocal interaction, particularly face-to-face meetings with community awareness, contributes to the overall health of open-source software development practices (Naparat et al., 2015). Additionally, individuals' shared traits and work history, supportive organisational culture and structure, serve as enabling factors for advice sharing required to develop problem-solving competence (Açıkgöz and Ö. İlhan, 2015; McGrath et al., 2003; Meister et al., 2018; Siciliano, 2015).

Studies further reveal that connecting to personal networks of technical managers can serve as a great resource for knowledge assistance and opportunities to gain more confidence and competence to quickly solve difficult technical problems to maintain performance (Ehrlich and Cataldo, 2014; McGrath et al., 2003; Sykes et al., 2014). This includes managing IT deployment and addressing other software problems (Xu et al., 2010). To generate non-redundant knowledge and display a creative approach to solving complex problems, individuals are expected to reach out to dissimilar others (i.e., weak ties) and not similar others (i.e., strong ties) (Granovetter, 1973; Levin et al., 2002; Perry-Smith, 2006). However, both social connections are necessary to generate new knowledge and disseminate it within the advice networks. According to structural hole theory, individuals who can bridge the gap between others as brokers tend to be more creative to benefit from and recombine diverse knowledge in increasing problem-solving efforts (Burt, 2004; Kislov et al., 2016; Whelan and Conboy, 2014). This is because such individuals would be more knowledgeable to alternatively and properly deal with various software problems to improve innovation and performance (Açıkgöz and Ö. İlhan, 2015; Thomas et al., 2002). In fact, the quality of workplace contact relationships between individuals in the community of practice matters for their motivation, productivity, satisfaction and network continuity (Burmeister et al., 2021; Naparat et al., 2015; van Bel et al., 2009). Finally, Gotel and Finkelstein (1994) argue that diverse techniques are needed to solve traceability problems when user requirements change during the development process. Software developers would need to know what to do (purpose-driven), how it should be done (solution-driven), and relevant information needed (information-driven) to tackle specific problems (direction-driven). Hence, we posit that individual connectedness in the advice network of individuals with diverse technical knowledge should influence their ability to solve technical problems more effectively during software development. We present our second hypothesis (denote H2):

**H2**: *A software developer's connectedness in the technical advice network will have a positive effect on their problem-solving competence*

### 2.3. Brokerage in advice networks and problem-solving competence

In social networks, and particularly in an organisational context, the functional role of brokers in the brokerage process of acquiring and connecting others with knowledge and informational resources, such as business domain and technical knowledge, have been linked to different network outcomes such as individual and innovation performances (Allen, 1984; Belso-Martínez et al., 2017; Obstfeld, 2005; Sawyer et al., 2010). Obstfeld et al. (2014, p. 141) define brokerage as "*behaviour by which an actor influences, manages, or facilitates interactions between other actors*". Their studies revealed that strategic brokerage action can involve sharing information or knowledge between two parties (*conduit*), where sharing of such resources may necessitate bringing such parties together to foster connections and coordination of tasks (*tertius iungens*, the third who joins). Another brokerage action involves a situation where brokers delight in keeping individuals apart to sustain their relevance by controlling knowledge resources access between the parties (*tertius gaudens or separans*, the third who enjoys) (Kwon et al., 2020; Obstfeld, 2005; Obstfeld et al., 2014). Studies suggest that individuals who perform extra-role behaviours in providing work-related advice to the need of others are more central, influential, and approachable in the advice network (Erdogan et al., 2020).

Moreover, the understanding of brokerage revolves around gaps between two disconnected actors called structural holes. Such a brokerage position has been considered beneficial for generating new ideas for business solutions such as innovation development (Burt, 2004). However, the revisit of the brokerage process indicates that a brokerage can occur when knowledge is served between connected members rather than their disconnection — an absolute brokerage (Li et al., 2018; Obstfeld et al., 2014). Given that software knowledge requirements are complementary, and a new member of staff such as a software developer might not hold them simultaneously, how they source for example business knowledge to support the technical knowledge or vice versa would be important for brokering activity, particularly how it is acquired, negotiated, and integrated into software development. One of the benefits of brokers is helping to orchestrate learning processes when they connect others with tacit or implicit knowledge, which in turn, improves performance (Conklin et al., 2013; Shah et al., 2018).

Taken together, scholars have suggested that advice networks with brokers will be more effective than without them because brokers can bridge communication, physical, cultural, cognitive and trust gaps in a network of relationships by facilitating knowledge interactions. This includes pointing to knowledgeable others when required in improving the problem-solving process and eventually driving solutions to work problems (Haas, 2015; Lam, 2018; Long et al., 2013; Verbeke et al., 2010). In the context of software development, studies have linked the role of brokers (e.g., gatekeeper, liaison) in facilitating business domain and technical knowledge resources to tackle software-related problems such as fixing bugs, handling sudden changes in requirements on the individual performance (Cataldo and Herbsleb, 2008; Sawyer et al., 2010), knowledge distribution (Boden and Avram, 2009; Whelan et al., 2013) and knowledge integration (Mehta and Bharadwaj, 2015; Šmite et al., 2017). Beyond software environments, other brokerage benefits include driving innovative knowledge transfer (Aalbers and Dolfsma, 2015) and knowledge expansion (Reus et al., 2020) in offline and online network studies. Work requirements are interpreted to relate to both technical and business dependencies, making boundary-spanning activity one of the prominent relational mechanisms utilised to gain external knowledge, solve problems, and achieve performance. This is especially important considering the timeliness

(length of time to access information and knowledge resources) and efficiency (amount and accuracy of information and knowledge accessed to the relevance of tasks) that come with coordination activities in software development (Blincoe et al., 2015; Mishra et al., 2012; Strode et al., 2012). Therefore, brokerage in business advice connectedness is needed to acquire business knowledge to solve business problems. Similarly, brokerage in the technical advice connectedness is needed to acquire technical knowledge to drive solutions to various technical problems. These together should increase the software development effort of software developers. Hence, our we present third hypotheses (denote H3a/H3b):

**H3a**: *A software developer's connectedness through a brokerage in the business advice network will have a positive effect on their problem-solving competence*

**H3b**: *A software developer's connectedness through a brokerage in the technical advice network will have a positive effect on their problem-solving competence*

### *2.4. The moderating role of contact quality in advice networks and problem-solving competence*

It is often said that people's social networks are their net worth. The quality of social contacts individuals maintain can decide their fate to succeed in organisational life. Where certain network configurations can land jobs, secure promotions, and may render the next trailblazing idea, not all social contacts result in meaningful positive outcomes (Fasbender et al., 2020; McManus et al., 2010). In this context, we consider contact quality as a moderator between business advice networks, technical advice networks, and problem-solving competence. A moderator is a variable that either increases or decreases the effect or association between two or more variables (Dawson, 2014; Farooq and Vij, 2017). Fasbender et al. (2020, p. 409) define contact quality as "*employees' experience of positive, natural, and cooperative interactions with peers of similar status that help employees to gain socioemotional meaning at work*". These authors suggest that high-quality contacts that coworkers maintain can increase their psychological well-being and job satisfaction. For example, Loury's (2006) study of informal advice networks found that job seekers connected with highly knowledgeable high-wage workers were able to receive more quality information to help match employers, which in turn resulted in good employment offers with high compensation and better satisfaction than low-quality-referred workers (cf. Snijders et al., 2013). Similar studies have shown why personal networks with quality contacts for advice seeking have resulted in generating quality information and hence positive task impact and good performance feedback in organisations (see for example, Parker et al., 2015; Sasidharan et al., 2012).

Su and Contractor (2011) emphasise the need for individuals (focal actors) to assess, recognise and seek knowledge expertise in others to improve their cognitive functioning in addressing their work-related problems (Burmeister et al., 2021; Parise et al., 2006). However, it is surmised that the focal actors as advice seekers must be willing to take and utilise such perspectives since it can be time and effort-consuming for the more knowledgeable coworkers in providing knowledge assistance (Cross et al., 2001; Fasbender et al., 2020). While seeking knowledge in others may be warranted to attain work goals, Burmeister et al. (2021) bemoan that knowledge overload can reduce the effectiveness of the knowledge seeker. These authors suggest that lower contact quality of a co-worker will result in "*lower germane cognitive load*" and "*higher extraneous cognitive load*", respectively. Nonetheless, as the quality of a contact increases, there will be a corresponding "*higher germane cognitive load*" and "*lower extraneous cognitive load*" (Burmeister et al., 2021). In other words, access to the network of quality contacts will create a positive reinforcement rather than a negative one to have a more motivated, cooperative, intelligent, and productive problem-solving interaction for higher problem-solving competence. In short, studies suggest that network affiliation (ties) to connected, competent and committed members such as brokers will be more trusted and highly instrumental in affecting individual and team performance in software development (Espinosa et al., 2018; Purvis and Zagenczyk, 2018; Sasidharan et al., 2012; Sykes et al., 2014). Therefore, quality over quantity of contacts matters for deriving intrinsic and extrinsic satisfaction in interpersonal advice networks (Fasbender et al., 2020; McManus et al., 2010; van Bel et al., 2009). Hence, our fourth hypotheses (denote H4a/H4b) is presented below.

**H4a**: *The positive effect of the relationship between a software developer's connectedness in the business advice network and problem-solving competence will be stronger when contact quality is high.*

**H4b**: *The positive effect of the relationship between a software developer's connectedness in the technical advice network and problem-solving competence will be stronger when contact quality is high.*

In sum, and based on our developed hypotheses, we posit that business advice connectedness (H1a/H1b), technical advice connectedness (H2), including brokerage in business advice connectedness (H3a) and brokerage in technical advice connectedness (H4a), directedly affect an individual problem-solving competence. However, we argue that contact quality moderates the effect of both business advice connectedness (H4a) and technical advice connectedness (H4b) on problem-solving competence. Fig. 1 presents our conceptual model for this study, which shows dynamic informal advice interactions as potential network antecedents shaping the problem-solving competence of individual software developers.

## 3. Research methodology

### *Data*

This study applied a social network survey design to test our conceptual model. A social network survey is a survey design to assess how individuals reflect on their knowledge interactions or social ties with others within their work roles or community of practice (Lorant et al., 2015). The accessible information can be described as social network data (Wuchty and Uzzi, 2011). The use of a social network survey guides in operationalising and measuring our research concepts, which is needed for building theoretical reasoning for our formulated research question. We gathered our social network data through self-reports following one of the established practices in this field. For instance, on the one hand, such social network data can be collected through staff lists (i.e., roster name generator questions) where participants are asked to reflect by indicating with whom (colleagues) they interact to give and/or get knowledge within their work roles. On the other hand, where access to such personal information was not available for privacy and confidentiality concerns, participants would be asked to reflect on their advice or knowledge interactions with the people in their social networks (e.g., advice/knowledge) (Sykes et al., 2014; van Bel et al., 2009). Hence, the latter survey design approach was adopted for this study. A further example of a typical roster name generator methodology is capturing information and knowledge interactions among students who just completed simulation tasks (Wong, 2008). Another is using an "event-driven methodology" to capture informational exchanges between information seekers who broadcast questions and the information providers who respond to them

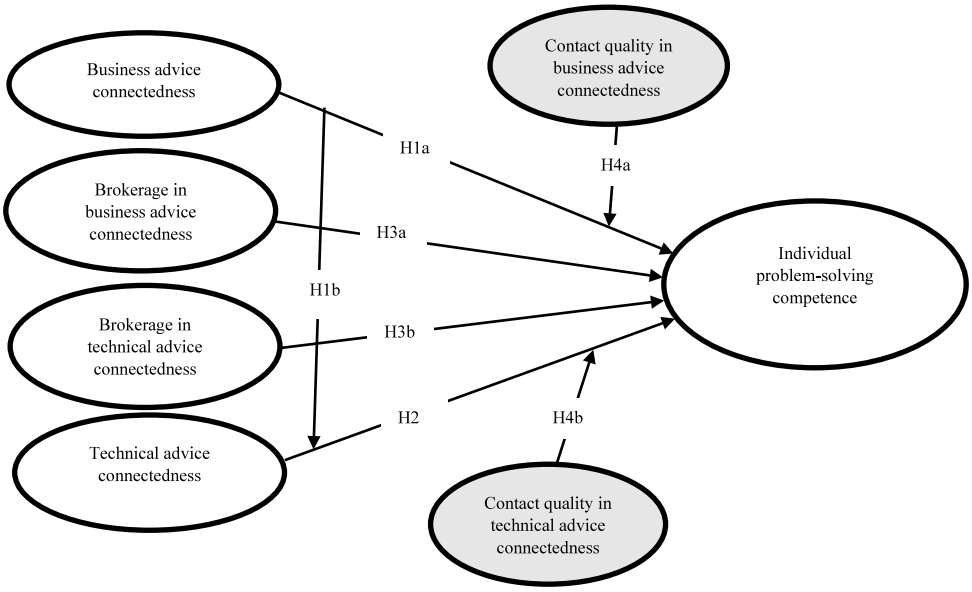


**Fig. 1.** A conceptual model for advice network antecedents of problem-solving competence.

(Constant et al., 1996). These are different avenues for collecting data for understanding patterns of behaviours in network contexts.

We subsequently collected social network data from software development professionals in a technology firm in Europe that specialises in cloud computing, web servers, and developing and implementing software solutions for businesses and end-users. There were on average 6 to 8 members in a team with roles including database administrator, analyst, business domain expert, programmer, technical lead architect, tester, scrum master, and project manager. In general, the various software development tasks performed included system integration, shell scripting, version control, debugging, functional testing, UI design, and virtualisation. A sampling frame of 370 respondents was contacted via an email online questionnaire distribution. A total of 161 responses were received, which accounted for a response rate of 43.51 percent. We excluded a total of 8 questionnaires with incomplete responses (missing data), leaving 153 complete questionnaires being used for our Partial Least Square Structural Equation Modelling (PLS-SEM) analysis. Our data were analysed using PLS-SEM analytic technique (Smart PLS-3). The demographic characteristics in this sample indicate that 79 (49%) respondents were females while 73 (47%) respondents were males. The remaining 4% was neither belonging to males nor females (non-binary, third/prefer not to say category). The average age of the respondents was 26 years old. Most of the respondents hold bachelor's degrees and professional diplomas. The discipline or educational background of the respondents represented diverse industries namely, engineering, humanities, health sciences, natural sciences, and social sciences, with the majority operating in the field of engineering. Similarly, the respondents represented diverse ethnic backgrounds: Whites, Blacks, Asians, and Hispanics, with the most representative sample being Whites. Lastly, the majority of the respondents were possessed between 1- and 4-years professional experience. The proportion of junior software developers also far exceeded those of senior software developers.

By adhering to the PLS-SEM guidelines on the appropriateness of sample size, it is recommended that the representative sample for analysis should be ten times the number of arrows directed at the target construct (Hair et al., 2014, 2017). In our case, there are five arrows of antecedents pointing toward individual problem-solving competence. Ideally, the minimum representative sample size would be 50 respondents. However, considering the recent criticism of this approach, we have calculated an estimated minimum sample size using the G*Power 3.1 tool (Faul et al., 2009; Hair et al., 2019; Tehseen et al., 2017; Wirth et al., 2021). This was calculated based on *Linear multiple regression: Fixed model, single regression coefficient*. By setting tails of 2, an effect size ($f^2$) of 0.15, an $\alpha$ error probability of 0.05, corresponding power of 0.95 and 4 predictors. It generated 89 respondents. Thus, the current sample size of 153 is sufficient to validate statistically, our conceptual model.

### *Measures*

The survey items for our constructs were developed based on a 7-point Likert scale (1= completely disagree, 7 = completely agree) to maintain consistency. Our survey items measured business advice and technical advice connectedness, a brokerage in business advice and technical advice connectedness, and problem-solving competence.

*Independent variables*

Business and technical advice connectedness were measured by three items adapted from the social connectedness scale of measuring advice experience in social networks by van Bel et al. (2009). Firstly, respondents were asked to reflect on the statements regarding their business advice connectedness: "*I often know what people in my business advice network think; I often know what people in my business advice network feel; I am often aware of my relationships with people in my business advice network*". Secondly, respondents were asked to reflect on the statements regarding their technical advice network connectedness: "*I often know what people in my technical advice network think; I often know what people in my technical advice network feel; I am often aware of my relationships with people in my technical advice network*". However, business and technical advice network questions were tweaked to follow similar network research in the extant studies (Belso-Martínez et al., 2017; Parker et al., 2015; Sykes et al., 2014). Respondents were presented with the following statements for self-reflection. "*This questionnaire is about how you experience your social relationship and social contacts with people in your business and technical advice networks. A business advice network relates to the network of people with information and knowledge needed to address business processes and customer needs such as analysing requirements. A technical advice network*

*relates to the network of people with information and knowledge needed to solve specific technical problems to develop solutions such as writing and changing source codes to fix bugs. With contacts, we mean problem-solving conversations/communications in each other's company or media (telephone, chat, email, text messaging, etc.) towards developing software products*". Firstly, respondents were further asked to "*please reflect on the statements below about your experience within your business advice network in providing you with information and knowledge related to problem-solving to get your work done"*. Secondly, respondents were asked to "*please reflect on the statements below about your experience with people within your technical advice network in providing you with information and knowledge related to problem-solving to get your work done*". Brokerage in business and technical advice connectedness was measured by six items adapted from tertius iungens orientation scale proposed by Obstfeld (2005). Firstly, respondents were asked to "*please rate the extent to which the statements below reflect your social orientation in connecting other work colleagues within your business advice network"*. Secondly, respondents were asked to *"please rate the extent to which the statements below reflect your social orientation in connecting other work colleagues within your technical advice network"*. Specifically, either for business advice network or technical advice network respectively, respondents were asked to reflect on these statements: "I introduce people to each other who might have a common strategic work interest; I will try to describe an issue in a way that will appeal to a diverse set of interests; I see opportunities for collaboration between people; I point out the common ground shared by people who have different perspectives on an issue; I introduce two people when I think they might benefit from becoming acquainted; I forge connections between different people dealing with a particular issue".

*Moderator variable*

Contact quality was measured by three items adapted from the contact quality scale proposed by van Bel et al. (2009). Firstly, respondents were asked to *"please rate the extent to which the statements below reflect the quality of your social interactions within your business advice network"*. Secondly, respondents were asked to *"please rate the extent to which the statements below reflect the quality of your social interactions within your technical advice network"*. Once again, the term *"moderator is any variable that affects the association between two or more other variables; moderation is the effect the moderator has on this association"* (Dawson, 2014, p. 1). It is the third variable that tends to change the relationship between an independent variable and a dependent variable. Studies suggest that a moderating effect helps test the causal tendency of a hypothetical model, which is different to an interaction effect that seems to test the model hypothesis without assuming causality. Arguably *"moderation effect is an interaction effect but interaction effect is not necessarily a moderating effect"* (Farooq and Vij, 2017, p. 35). In other words, we established the conditions under which contact quality-related effects on business and technical advice connectedness materialise (i.e., moderation) on the problem-solving competence of software developers. Considering a seemingly increasing demand for promoting and sustaining trustworthiness in organised work routines among employees in organisations (Kähkönen et al., 2021), there is a need to examine the value of the moderating role of contact quality between business and technical advice connectedness and problem-solving competence.

*Dependent variable*

Problem-solving competence was measured by eight items (four items each measured problem-solving speed and problem-solving creativity, respectively) adapted from Atuahene-Gima and Wei (2011). Problem-solving speed is "*the ability to find and speedily implement a large number of solutions*" while problem-solving creativity is the "*ability to discover and implement novel and cost-effective solutions*" (Atuahene-Gima and Wei, 2011, p. 83). Hence, respondents were asked to "*please rate the extent to which the statements below reflect how your social relationship and social contacts in the advice networks help you…"*

*Control variables*

To understand possible confounding factors that can influence problem-solving competence, we included age, gender, qualification, discipline, rank, ethnicity, computer self-efficacy and conscientiousness personality. Diversity of qualifications, ethnicities and disciplines has been shown to improve a firm's level of innovativeness (Mohammadi et al., 2017) and advice-giving interactions (Vargas and Schafer, 2013). Computer self-efficacy was measured by four items, and conscientiousness personality was measured by 5 items, both variables being adapted from Zhang and Venkatesh (2013). Age, gender, rank, computer self-efficacy and conscientiousness personality have been linked to affecting individual performance (Sykes et al., 2014; Zhang and Venkatesh, 2013). The details of the measurement scales and items are further provided in Table 10 in Appendix.

## 4. Data analysis and results

*Assessment of Measurement Model*

A measurement model reveals the interacting relationship between constructs and their indicator items. In assessing a measurement model, studies suggest considering the nature of the construct, the causal direction between the latent construct and its indicator and how the indicator characteristics measure the construct (Coltman et al., 2008; Hair et al., 2019). Given our conceptual model, we decided to use only reflective indicators. First, we examined *convergent validity*, and whether our model values met the acceptable threshold set by the PLS-SEM guidelines for the *indicator loadings* ($\geq$0.70). Second, the *internal consistency* reliability using *composite reliability* (CR $\geq$0.70) and *Cronbach's alpha*. Third, the *convergent validity* of each construct using *average variance extracted* (AVE $\geq$ 0.50). Fourth, *the discriminant validity* using *Fornell–Larcker criterion* and *heterotrait–monotrait* (HTMT< 0.90) ratio (Hair et al., 2017, 2014, 2019; Henseler et al., 2015).

As part of the rules of thumb in PLS-SEM guidelines, we removed nine items (BBAC6, BTAC6, CP3, CP4, CP5, CSE1, PSC4, PSC5, PSC6) from the analysis due to low indicator (factor) loadings (<0.70). However, we ensured due diligence by cross-checking to see that these items removed were captured in the remaining items of the construct. We tested for the reliability of the constructs using Cronbach's alpha and composite reliability (CR). The CR values met the required acceptable threshold (<0.70). The Average Variance Extracted (AVE) values were also valid (>0.50). Hence, the Convergent Validity was acceptable (see Table 1). We assessed discriminant validity using the Fornell–Larcker criterion, which shows the square root of AVE for the construct greater than the inter-construct correlations (see Table 2). We also assessed the discriminant validity using heterotrait–monotrait (HTMT) ratio of correlations, with values within the required threshold (<0.90). Hence, discriminant validity was established (see Table 3).

*Assessment of Structural Model*

The structural model explains the relationship in the direction of paths between the constructs proposed in the conceptual model. It is assessed based on the required values for variance inflated factor (VIF), $R^2$, $Q^2$, $f^2$, root mean square residual (SRMR) and significance of paths. Our VIF values are less than 3 (see Table 1), the $R^2$ value is greater than 0.5, the $Q^2$ value is greater than 0.25, and the SRMR value is less than .10 (Hair et al., 2019;

**Table 1**
Loadings, reliability, and validity.

| | Loadings | Cronbach Alpha | Composite Reliability | Average Variance Extracted (AVE) | VIF |
|---|---|---|---|---|---|
| BAC1 | 0.895 | 0.805 | 0.886 | 0.721 | 2.049 |
| BAC2 | 0.853 | | | | |
| BAC3 | 0.796 | | | | |
| BBAC1 | 0.783 | 0.865 | 0.902 | 0.649 | 2.597 |
| BBAC2 | 0.819 | | | | |
| BBAC3 | 0.821 | | | | |
| BBAC4 | 0.779 | | | | |
| BBAC5 | 0.825 | | | | |
| CQBAC1 | 0.886 | 0.830 | 0.896 | 0.742 | 2.027 |
| CQBAC2 | 0.865 | | | | |
| CQBAC3 | 0.832 | | | | |
| TAC1 | 0.819 | 0.746 | 0.855 | 0.663 | 2.120 |
| TAC2 | 0.843 | | | | |
| TAC3 | 0.779 | | | | |
| BTAC1 | 0.860 | 0.911 | 0.934 | 0.738 | 2.510 |
| BTAC2 | 0.885 | | | | |
| BTAC3 | 0.869 | | | | |
| BTAC4 | 0.856 | | | | |
| BTAC5 | 0.825 | | | | |
| CQTAC1 | 0.878 | 0.853 | 0.906 | 0.763 | 1.950 |
| CQTAC2 | 0.855 | | | | |
| CQTAC3 | 0.888 | | | | |
| PSC1 | 0.894 | 0.903 | 0.928 | 0.720 | |
| PSC2 | 0.833 | | | | |
| PSC3 | 0.857 | | | | |
| PSC7 | 0.845 | | | | |

**Table 2**
Discriminant validity using Fornell–Larcker criterion.

| | BAC | BBAC | BTAC | CQBAC | CQTAC | PSC | TAC |
|---|---|---|---|---|---|---|---|
| BAC | *0.849* | | | | | | |
| BBAC | 0.504 | *0.806* | | | | | |
| BTAC | 0.345 | 0.732 | *0.859* | | | | |
| CQBAC | -0.222 | -0.118 | -0.074 | *0.861* | | | |
| CQTAC | -0.117 | -0.096 | -0.039 | 0.683 | *0.874* | | |
| PSC | 0.514 | 0.754 | 0.633 | -0.204 | -0.105 | *0.849* | |
| TAC | 0.620 | 0.450 | 0.504 | -0.068 | -0.113 | 0.453 | 0.814 |

Note: Values in italic represent the square root of AVE.

**Table 3**
Discriminant validity using Heterotrait–Monotrait (HTMT) ratio.

| | BAC | BBAC | BTAC | CQBAC | CQTAC | PSC | TAC |
|---|---|---|---|---|---|---|---|
| BAC | | | | | | | |
| BBAC | 0.599 | | | | | | |
| BTAC | 0.404 | 0.827 | | | | | |
| CQBAC | 0.262 | 0.133 | 0.088 | | | | |
| CQTAC | 0.143 | 0.119 | 0.065 | 0.809 | | | |
| PSC | 0.601 | 0.844 | 0.696 | 0.228 | 0.113 | | |
| TAC | 0.796 | 0.557 | 0.607 | 0.092 | 0.136 | 0.548 | |

**Table 4**
Variance in Goodness of Fit of the model.

| | Included | Excluded | f-squared (variance) | Effect-size |
|---|---|---|---|---|
| R-squared | 0.636 | 0.620 | 0.016 | |
| | | | | |
| Effect size $f^2$ | 0.02 | 0.15 | 0.35 | Weak |
| | Weak | Moderate | Strong | |

Tehseen et al., 2017). Hence, our VIF values indicate there are no multicollinearity issues. The goodness of fit of the model is based on the value of $R^2$ for problem-solving competence, which reflects the strength of each structural path of the model. In moderating analysis, change in the value of $R^2$ in explaining variance in PSC must be considered. For the base (main effect excluding the moderator) model, our explained variance of $R^2$ is 0.620 and 0.636 for the interaction (moderating) effect. The $R^2$ change of 0.016 indicates 1.6% of the additional variance. Hence, the goodness of fit increases from 62.1% to 63.6%. We assessed the effect size of $f^2$ values as it shows the weakness and strength of the latent constructs on target constructs. Given 0.016 in our additional variance of PSC, we can state that the introduction of two interaction terms (CQBAC/CQTAC) only has a weak effect size $f^2$ on PSC (see Table 4). In addition, the $Q^2$ value ($Q^2$ = .441>0) reveals and confirms the *predictive relevance* of PSC. Furthermore, our assessment of SRMR value (SRMR = .082), indicated an acceptable model fit.

In assessing the goodness of fit, we tested our hypotheses to ascertain the significance of the relationships between the constructs. H1a evaluates whether BAC has a significant impact on PSC. The results revealed that BAC has an insignificant impact on PSC ($\beta$ = .138, t = 1.896, p <.058). Hence, H1a was not supported. H1b evaluates whether BAC has a significant impact on TAC. The results revealed that BAC has a significant impact on TAC ($\beta$ = .620, t = 7.942, p <.000). Hence, H1b was supported. H2 evaluates whether TAC has a significant impact on PSC. The results revealed that TAC has an insignificant impact on PSC ($\beta$ = .043, t = .542, p = .588). Hence, H2 was not supported. H3a evaluates whether BBAC has a significant impact on PSC. The results revealed that BBAC has a significant impact on PSC ($\beta$ = .535, t = 6.231, p <.000). Hence, H3a was supported. H3b evaluates whether BTAC has a significant impact on PSC.

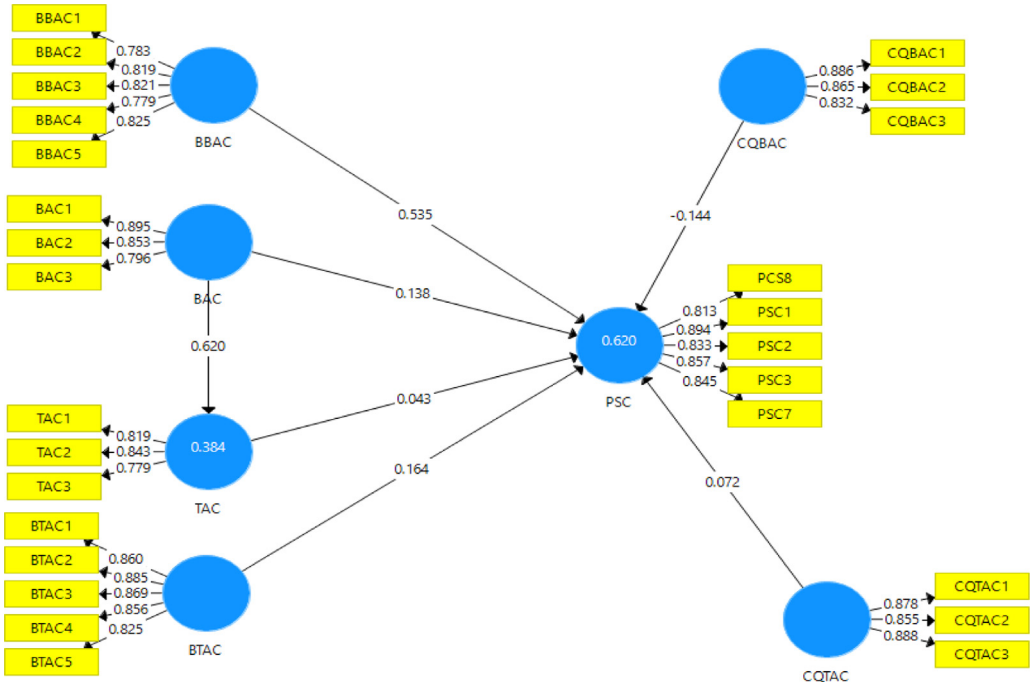


**Fig. 2.** Assessing the level of indicator weights and interaction variance before moderating effect.

**Table 5**
Hypotheses testing before adding moderating effect.

| | $\beta$ | STDEV | T Statistics | P Values | 0.025 | 0.975 |
|---|---|---|---|---|---|---|
| BAC → PSC | 0.138 | 0.073 | 1.896 | 0.058[a] | −0.009 | 0.275 |
| BAC → TAC | 0.620 | 0.078 | 7.942 | **0.000** | 0.433 | 0.748 |
| BBAC → PSC | 0.535 | 0.086 | 6.231 | **0.000** | 0.365 | 0.688 |
| CQBAC → PSC | −0.144 | 0.079 | 1.835 | 0.067 | −0.316 | −0.004 |
| TAC → PSC | 0.043 | 0.079 | 0.542 | 0.588 | −0.107 | 0.203 |
| BTAC → PSC | 0.164 | 0.078 | 2.101 | **0.036** | 0.004 | 0.310 |
| CQTAC → PSC | 0.074 | 0.073 | 1.027 | 0.305 | −0.052 | 0.224 |

[a]BAC → PSC (A positive relationship requires T Statistics ≥ 1.96 and P Values ≤ 0.05).

**Table 6**
Hypotheses testing after adding moderating effect.

| | $\beta$ | STDEV | T Statistics | P Values |
|---|---|---|---|---|
| BAC → PSC | 0.148 | 0.072 | 2.058 | **0.040** |
| BAC → TAC | 0.620 | 0.076 | 8.110 | **0.000** |
| BBAC → PSC | 0.498 | 0.082 | 6.048 | **0.000** |
| TAC → PSC | 0.004 | 0.075 | 0.058 | 0.954 |
| BTAC → PSC | 0.174 | 0.073 | 2.382 | **0.017** |
| CQ mod BAC-PSC → PSC | −0.016 | 0.086 | 0.181 | 0.856 |
| CQ mod TAC-PSC → PSC | −0.108 | 0.082 | 1.315 | 0.188 |

**Table 7**
Summary of hypotheses results.

| Direct Relationship | Result | Moderation effect | Result |
|---|---|---|---|
| H1a | Not supported | H1a | Supported |
| H1b | Supported | H1b | Supported |
| H2 | Not supported | H2 | Not supported |
| H3a | Supported | H3a | Supported |
| H3b | Supported | H3b | Supported |
| H4a | Not supported | H4a | Not supported |
| H4b | Not supported | H4b | Not supported |

The results indicate that BTAC has a significant impact on PSC ($\beta$ = .164, t = 2.101, p<.036). Hence, H3b was supported (see Fig. 2 and Table 5).

Furthermore, the direct relationship of our moderation analysis before the moderating (interaction) effect is as follows. H4a evaluates whether CQBAC has a significant relationship between BAC and PSC. The results revealed that CQBAC has an insignificant moderating effect between BAC and PSC ($\beta$ = −.144, t = 1.835, p<.067). Hence. H4a was not supported. H4b evaluates whether CQTAC has a significant relationship between TAC and PSC. The results revealed that CQTAC has an insignificant moderating effect between TAC and PSC ($\beta$ = .074, t = 1.027, p <.305). Hence, H4b was not supported.

In this study, our bootstrapping results of 5000 resamples also generated 95% confidence intervals because confidence intervals with values different from zero indicate significant relationships. Fig. 3 shows the interaction variance of the moderation, while Table 5 provides the summary of our tested hypotheses.

After controlling for the moderating effect, surprisingly, we found BAC having a significant impact on PSC ($\beta$ = .148, t = 2.058, p = .040) and TAC ($\beta$ = .620, t = 8.110, p = .000). While the significant impact of BBAC between BAC and PSC ($\beta$ = .498, t = 6.048, p = .000), including the significant impact of BTAC between TAC and PSC ($\beta$ = .174, t = 2.273, p = .023), and the insignificant impact of TAC on PSC ($\beta$ = .004, t = 0.058, p = .934), remain unchanged. Thus, H1a and H1b, H3a and H3b were supported, but H2 remain unsupported. Similarly, we found the moderating effect of CQ between BAC and PSC ($\beta$ = . − .016, t = .181, p = .856), and TAC and PSC ($\beta$ = −.108, t = 1.315, p = .188) to be insignificant. Thus, H4a and H4b were not supported, showing a lack of moderation. This revealed that the effect of BAC and TAC on PSC needs no moderation (see Tables 6 and 7).

*Assessment of Importance — Performance Map Analysis (IPMA)*

We calculated importance-performance map analysis (IPMA) to identify and assess the relative importance and performance of our latent constructs and how they perform with the target construct (endogenous variables) in prioritising areas of improvement (Ringle and Sarstedt, 2016). The IPMA values show that

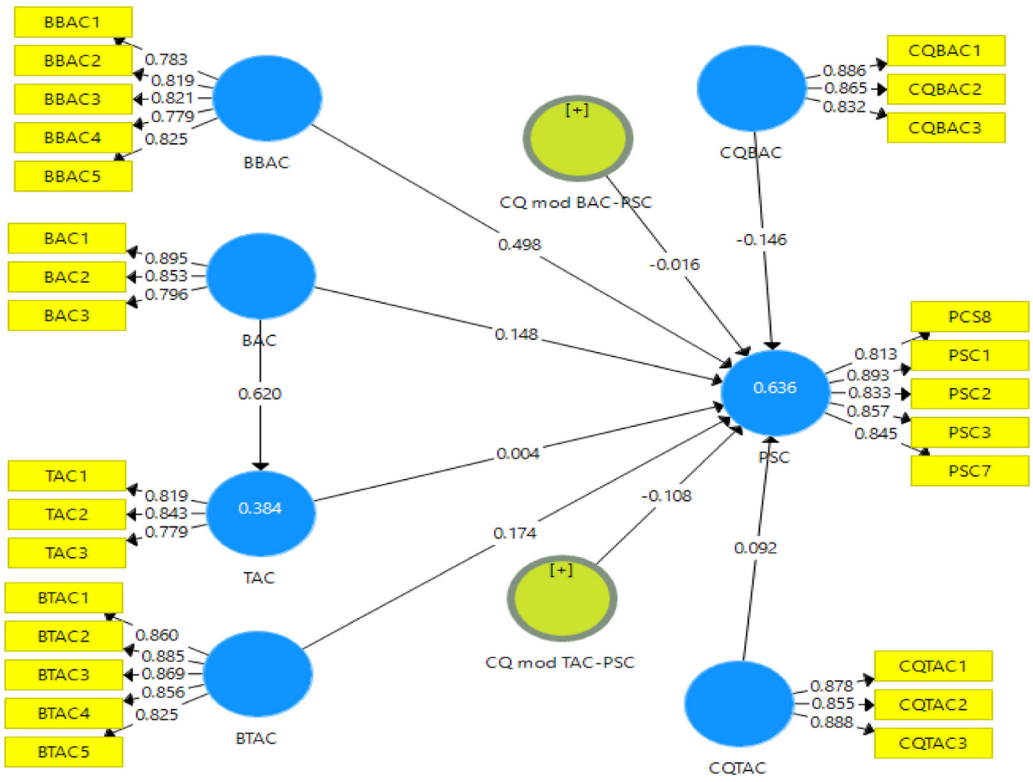


**Fig. 3.** Assessing the level of indicator weights and interaction variance after moderating effect.

**Table 8**
Importance-Performance map results.

| Construct | Importance (Total effect) | Performance (Index values) |
|---|---|---|
| BBAC | 0.519 | 72.007 |
| BTAC | 0.154 | 70.458 |
| BAC | 0.143 | 63.944 |
| CQBAC | −0.109 | 47.989 |
| CQTAC | 0.066 | 48.592 |
| TAC | 0.004 | 66.877 |

BBAC and BTAC followed by BAC have high importance and corresponding performance scores (see Table 8). Thus, we used this to inform managerial decisions in interpreting the managerial implications for our findings as conducted in other research (e.g., Wirth et al., 2021).

The PLS-SEM regression in Table 9 indicates that only two of our control variables: computer self-efficacy ($\beta = .187$, t = 2.776, p = .006) and conscientiousness personality ($\beta = .218$, t = 2.972, p = .003), were significant on PSC. Whereas only BBAC ($\beta = .338$, t = 3.762, p = .000), as part of our independent variables, maintains significance and support for hypothesis H3a.

In addressing similarity issues, studies suggest that two constructs can be distinct or similar, conceptually (Hair et al., 2017), such as business advice connectedness and technical advice connectedness. In this context, these observed network constructs may tend to be similar but share different underlying measurements. Our convergent validity (or confirmatory factor analyses) revealed that their performance indicators loaded well ($\geq 0.70$) and also met the acceptable level of discriminant validity (quality robustness check). Hair et al. (2017, p. 455) suggest that "*Discriminant validity – HTMT Values lower than 0.85 for conceptually distinct constructs and below 0.90 for conceptually similar constructs; confidence intervals should not include a value of 1*". Hence, our study has put this HTMT ratio criterion into consideration based on PLS-SEM guidelines in the literature.

## 5. Discussion and conclusion

The purpose of this study was to empirically examine the effect of advice networks on the problem-solving competence of software developers. This is important to establish how an individual's advice network shapes their software development effort. We theorised that business advice and technical advice connectedness and a brokerage in business advice and technical advice connectedness for knowledge resources would enhance the problem-solving competence of software developers. However, that the effect of business advice connectedness and technical advice connectedness on problem-solving competence depends on the quality of contacts within these networks. Our contributions reveal that software developers who engage in knowledge brokering in both business and technical advice (network) connectedness will particularly increase problem-solving competence to drive their software development effort. Surprisingly, we found that contact quality played no significant moderating role in strengthening and weakening the impact of advice networks on improving the problem-solving competence of software developers.

### 5.1. Theoretical implications

This study provides interesting findings and implications. First, in understanding the relationship between advice networks and problem-solving competence, our findings revealed a statistically significant positive association between advice networks and individual problem-solving competence. The approach to work interactions often requires the flow of different knowledge resources, such as business advice and technical advice, as pertinent for driving the software development effort of software developers. Therefore, studies focusing on either of these advice networks offer a limited understanding of their performance benefits. In this study, our multiplex network approach

**Table 9**
Hypotheses testing after adding control variables.

| | $\beta$ | STDEV | T Statistics | P Values |
|---|---|---|---|---|
| *Control variables* | | | | |
| Age → PSC | −0.010 | 0.047 | 0.221 | 0.825 |
| Computer self-efficacy → PSC | 0.187 | 0.067 | 2.776 | **0.006** |
| Conscientiousness personality → PSC | 0.218 | 0.073 | 2.972 | **0.003** |
| Discipline → PSC | −0.037 | 0.051 | 0.729 | 0.466 |
| Ethnicity → PSC | −0.029 | 0.064 | 0.456 | 0.648 |
| Gender → PSC | 0.053 | 0.064 | 0.828 | 0.408 |
| Qualification → PSC | −0.017 | 0.052 | 0.316 | 0.752 |
| Rank → PSC | 0.034 | 0.054 | 0.623 | 0.533 |
| Tenure → PSC | 0.012 | 0.053 | 0.234 | 0.815 |
| *Independent variables* | | | | |
| BAC → PSC | 0.067 | 0.069 | 0.968 | 0.333 |
| BBAC → PSC | 0.338 | 0.090 | 3.762 | **0.000** |
| TAC → PSC | 0.025 | 0.077 | 0.329 | 0.742 |
| BTAC → PSC | 0.140 | 0.078 | 1.782 | 0.075 |
| *Moderating variables* | | | | |
| CQBAC → PSC | −0.143 | 0.075 | 1.900 | 0.058 |
| CQTAC → PSC | 0.046 | 0.072 | 0.641 | 0.522 |
| CQ mod BAC-PSC → PSC | 0.014 | 0.080 | 0.174 | 0.862 |
| CQ mod TAC-PSC → PSC | −0.073 | 0.079 | 0.927 | 0.354 |

provides joint advantages on the problem-solving competence of software developers. In doing so, we add to a stream of prior work on multiplex social interaction that outline the combined effect of various advice network relations to be more beneficial than uniplex ones to various individual and team level tasks, including ideation and problem-solving capability (Aalbers et al., 2014; Belso-Martínez et al., 2017; Shah et al., 2017; Snijders et al., 2013). In addition to these prior insights and based on the strong association between brokerage in business advice connectedness and problem-solving competence, our findings suggest that individuals need to engage in brokerage activities for knowledge resources to develop the required effort for amplifying their problem-solving competence. This finding makes our study one of the first to directly link business and technical advice connectedness via brokerage (as antecedents) to problem-solving competence (as outcome). This is of both scholarly and managerial relevance, as the linkage is often needed for the software development effort to tackle software-related business and technical problems. This contribution additionally extends existing literature by examining business communication networks while providing support for individual connectedness such as the role of the gatekeeper in technical communication networks and the resulting positive impact on improved software development efforts and performance of distributed software development teams (Cataldo and Herbsleb, 2008; Ehrlich and Cataldo, 2014).

Interestingly, these findings reveal an inverse (direct) relationship when considering the impact of business advice connectedness and technical advice connectedness (without brokerage) on problem-solving competence. Moreover, after considering the moderating effect of contact quality, there is a significant positive relationship between business advice connectedness (but not technical advice connectedness) and problem-solving competence. In spite of this effect, the moderating role of contact quality is still not significant. Instead, we found that business advice connectedness has a direct relationship with technical advice connectedness, suggesting a positive interaction between both networks. A relationship, however, does not translate to a significant impact on problem-solving competence to make sense of further performance implications since our findings suggest a weak and insignificant interaction with problem-solving competence. For example, our findings indicate that it is not incontestable that business advice connectedness would be required for implementing technical advice, since a software developer requires applying technological competence and tools to drive solutions to business problems. Hence, it is the understanding of business problems that makes such technical knowledge applications possible. To this end, we can only make an assumption that a potentially high relationship between business and technical advice connectedness might, in turn, translate into a rather positive relationship between technical advice connectedness and problem-solving competence. This invites further examination of our conceptual model.

Second, though our findings revealed there was no significant moderating role of contact quality, still this moderation turned an insignificant relationship between business advice connectedness (direct relationship) into a significant relationship with problem-solving competence (moderating effect) and arguably advancing research contributions. In evidence of this insight, Fig. 4 above shows two graph slopes for advice network sensemaking. In the business advice connectedness graph, the blue line indicates a corresponding increase in both business advice connectedness and problem-solving competence when quality contact in business advice connectedness decreases. However, counterintuitively, a decrease in quality contacts in business advice connectedness (i.e., green line) also increases business advice connectedness and problem-solving competence. In the technical advice connectedness, the blue line is pointing upward, which indicates that both technical advice connectedness and problem-solving increases when contact quality decreases. However, the green line represents the effect of high contact quality in technical advice connectedness and problem-solving competence. The green line is not flat but slightly points downwards as it goes forward, which shows that the technical advice connectedness increases and problem-solving competence decreases, albeit slightly, when contact quality in the technical advice connectedness is high. This is counterintuitive to our expectations. This finding is not consistent with a recent study by Burmeister et al. (2021), which found that a significant moderating role of high contact quality reduced knowledge overload during knowledge-seeking, and in turn, increased the productivity of the knowledge seeker.

Third, while some existing studies have contributed to research and practice by linking antecedents of team problem-solving competence to the combined effects of anticipation and reaction mechanisms (Li et al., 2011), knowledge complement, knowledge location and knowledge deployment (Lin et al., 2015), project-size and staff expertise (Aladwani, 2002), market knowledge competence (Atuahene-Gima and Wei, 2011), and centrifugal and centripetal forces (Sheremata, 2000); however, they have not conducted research on business and technical advice networks in software development settings. Our findings extend

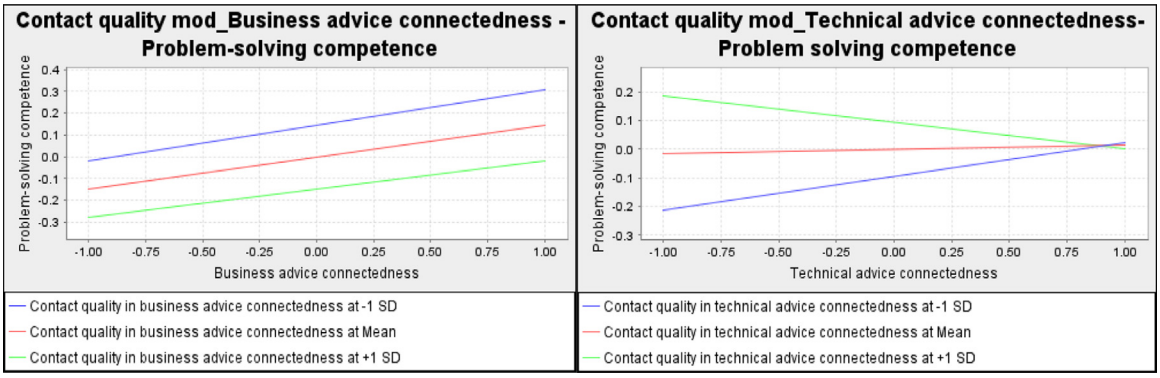


**Fig. 4.** The moderating role of contact quality in business and technical advice connectedness on problem-solving competence.

these contributions by looking at the behavioural patterns of both networks on problem-solving competence of software developers. Our study also provided support for previous studies that have drawn a positive relationship between problem-solving ties and network performance (see for example, Shah et al., 2018), including brokerage for business and technical knowledge sharing on other performance-related outcomes (e.g., knowledge integration) (for example, Mehta and Bharadwaj, 2015; Sawyer et al., 2010; Šmite et al., 2017). Our findings further extend research on advice networks by stating that an individual access to workflow advice and software advice resources may be contingent on their brokerage action in workflow advice network and software advice network to achieve better job performance in software development settings (Sykes et al., 2014). In the same vein, individuals may need network connectedness of a brokering kind to gain access to business and technical knowledge resources before knowledge integration can occur to drive software development performance (Tiwana, 2004).

Fourth, and more importantly, studies suggest that social connectedness partially determines the propensity for brokerage (e.g., Aalbers et al., 2013; Parise et al., 2006; Sheremata, 2000; van Bel et al., 2009) because connectedness can enable individuals to gain diverse information and knowledge resources, help solve problems faster, and make trade-off decisions (Sheremata, 2000). However, a tie only exists in direct social relationships, hence, connectedness can signal the presence or absence of ties or a form of network membership. Hence, we posit that an advice network can involve an exchange of resources between two individuals without brokerage playing a role. For example, on one hand, the moderation analysis here revealed a positive relationship between business advice connectedness and problem-solving competence. On the other hand, the positive interaction and moderating effect suggest a positive impact of brokerage in business advice connectedness on problem-solving competence. Thus, the interesting question here is *what triggers the need for knowledge or advice sharing and problem-solving assistance?* While this may invite obvious answers based on examined literature, it can be argued that people who know a lot may be happy to share knowledge with others they know and have been approached for knowledge support. However, the tendency to share advice requires a need to be met by someone, another person who may not necessarily be known by the knowledge source, except by indirect effect via a connector, the middleman or the broker. Research on this domain suggests that a person who has a lot of contacts and knows those in need of knowledge support may not be willing to link two people together (knowledge source/knowledge recipient), by sourcing knowledge from one and sharing it with the other person (e.g., *tertius gaudens*) (Obstfeld, 2005). This is with the exemption of *tertius iungens* who may be willing to bring the knowledge source/knowledge recipient together with no prior connections, without taking the glory or assuming to be the direct knowledge source/originator (Allen, 1984; Long et al., 2013; Obstfeld, 2005; Whelan et al., 2011). Thus, these findings thereby capture this specificity of the direct effect of business advice connectedness and the indirect effect of brokerage in business advice connectedness and technical advice connectedness, respectively, on problem-solving competence. However, and more importantly, while we do not expect the two networks to be different, there is certainly some overlap between them. What is different is the level of connectedness different software (or information systems) development professionals will have in a business advice network versus a technical advice network. For example, Cross et al. (2001) argue that individuals have different levels of connectedness in different types of networks. Therefore, it is possible that a technical engineer is a go to person in the technical advice network, but on the periphery of business advice network. In doing so, our findings contribute to contemporary thinking in viewing brokerage as a process to explain patterns of connectedness, thus advancing contributions to this research enquiry (Obstfeld et al., 2014) even though brokerage in business advice connectedness provided the most robust association with problem-solving competence after controlling for other factors (e.g., conscientiousness personality, computer self-efficacy).

Furthermore, we considered these network mechanisms for theorising as having different implications for problem-solving competence. In this instance, the importance of conscientiousness personality as a link between networking behaviour and knowledge sharing in expressive and/or instrumental advice networks and performance cannot be over-emphasised (e.g., Fang et al., 2015; Licorish and Macdonell, 2015; Sykes et al., 2014). In line with Fang et al. (2015, p. 1245) who propose that "*conscientiousness concerns the extent to which a person is industrious, organized, dutiful, prepared, persistent, and detail-oriented*", our findings also share the positive relationship between conscientiousness personality and the problem-solving competence of software developers. Also, we found that the significant impact of computer self-efficacy, "*an individual's belief of capabilities to perform a specific task using a computer*" (Homburg et al., 2010, p. 162) on problem-solving competence can be related to brokerage action. Because when individuals get stuck with technical problems related to software development or implementation (Sykes et al., 2014), they can invite knowledge seeking of a brokering kind with and through other people. By controlling these effects on problem-solving competence, it appears that this might have created more variance affecting the extent of the significance of our results, for example, a brokerage in technical advice connectedness that once maintained significance in both interactions (i.e., direct relationship) and moderating effects. As aforementioned, a moderating effect implies some measure of

causality between two or more variables. By considering the interaction (direct), moderating, and controlling effects altogether, our findings still suggest a highly significant positive relationship between brokerage in business advice connectedness and problem-solving competence. In this instance, we can make a number of assumptions that people go to brokers who have knowledge of business requirements by being just closer to the customers have sufficient experience with solving business problems, can be trusted, have well connected knowledgeable others for knowledge support (social capital), more socially approachable and communicative (in terms of personality), and are more committed to helping others when they are confronted with problem-solving challenges during software development process. Such brokers may therefore be subject-matter experts in cross-functional domains (business/technical related) as software developers operating within an organisation, or better still, get hired on a consulting basis.

However, in limiting these factors, the use of control variables should be taken into consideration. For example, despite the fact that empirical evidence revealed that gender diversity (e.g., men/women) and their perceptions in network positions vary in networking behaviours (Brands and Kilduff, 2014; Merluzzi, 2017) and that location (i.e., proximity) thus affect advice and friendship networking behaviours (Borgatti and Cross, 2003; Ellwardt et al., 2012), these have no empirical support for our results, with the exemption of conscientiousness personality and computer self-efficacy that seem rather significant in this case.

### 5.2. Managerial implications

These findings are relevant for software practitioners. Problem-solving activity is core to software development projects as software organisations are tasked to tailor business solutions to the needs of their clients, users, and business partners. Successful project development requires the right application of knowledge to solve different problems, which often depend on collaboration between software developers to work towards achieving project goals. Our findings reveal that organisational leaders need to be cognisant of the advice networks for knowledge resources within their organisations and how best to promote effective networking within and between them for individual and collective knowledge resourcefulness. By promoting networking for example in both business and technical advice networks, individuals would be able to identify and utilise their diverse knowledge expertise, experience, and status of the more knowledgeable others, such as technical lead architects, business domain experts, or knowledge brokers. Moreover, by being connected to a network of knowledgeable and experienced problem solvers in these networks, individuals can increase their problem-solving skills to drive better software development efforts for project success. Realising the need to increase problem-solving competence demand greater brokerage and connectedness, especially for less connected and isolated individuals in the advice networks for business and technical knowledge resources.

Our findings suggest that brokerage in business advice connectedness is considered of high importance (see Fig. 5) and high performance for increasing problem-solving competence for successful software development. The reason for this that understanding, analysing, and interpreting business problems is a necessity for matching requirements with appropriate technologies and applying technical competence to drive solutions to customer needs.

Our findings further reveal that technical advice connectedness, business advice connectedness and brokerage in technical advice connectedness are of high importance yet with low performance on problem-solving competence in the software development effort. Hence, project managers need to pay more attention to these organisational knowledge mechanisms in ensuring that they are improved and optimised to deliver high performance as germane to increasing problem-solving competence require for individuals, teams, and organisational performance. In addition, our findings revealed that contact quality is of low importance and low performance in both business and technical advice connectedness to improve problem-solving competence. While this is the least to focus on, the lack of contact quality sources for business and technical advice interactions can affect the ability to receive information quality or knowledge quality which will be needed to become creative, make timely good decisions, solve problems faster, and increase productivity to meet budget and satisfy customer needs. In addition, we can relate contact quality to issues of trust between two people and their organisation based on the level of expertise findable and available to individuals, the reputation of such individuals, and the usefulness of knowledge to be shared especially tacit knowledge (than explicit knowledge) — which often requires social interactions for shared understanding and learning benefits needed to solve problems and achieve performance. The onus is on organisations to provide expert support network systems as a form of transactive knowledge systems for such attainment. Hence, organisational managers must consider and configure the structure and protocol of communication, task, team, and project development in such a way that permits and promotes contact quality engagement across all levels of their community of practice. Software development activity tends to be driven interdependently within software development teams and other stakeholders; therefore, project managers need to give attention to this in shaping problem-solving competence. It can encourage intentional expert interventions and the flow of quality advice required to optimise the software development effort of their software developers. We are not unaware that software engineering is a collaborative process where communication, cooperation and coordination serve as the social infrastructure in software development projects. Hence, such network mechanisms are extremely important for driving development activities and software development success. Overall, software engineering is a team pursuit. Software developers have to interact and network together. We build on the literature that shows such advice network patterns are important in software engineering settings. For example, we show that the interaction between different types of networks has important implications for software engineering. Organisations can avail of these insights to better look into their support network processes to facilitate access to and integration of these knowledge resources for greater software development performance. Managers thus need to promote and facilitate both business and technical networks.

### 5.3. Limitations and future research

These findings have a number of limitations for further research. First, our analysis was focused on the self-report measure of individual perceived experiences of their advice interactions with people in the whole or network level of analyses. Hence, this might have biased our results somewhat. Notwithstanding studies have found that people are generally good at assessing themselves and their networks (Fox and Dinur, 1988), as well as their recurring behavioural patterns of social interactions (Roth et al., 2021), regardless of the potential for bias. Similar studies have also found reliable individuals demonstrating better predictive information accuracy based on their observable patterns of recall and interaction data (Kimball Romney and Weller, 1984). In other words, we feel that our data collected through this self-retrospection has a sense of reliability and validity since all measurement scales and items used for the survey design have

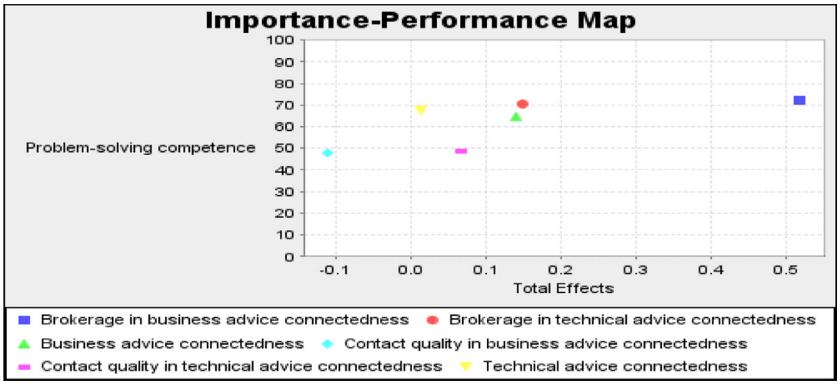


**Fig. 5.** Importance-performance results of our network constructs.

been validated in prior studies. While our measurements and structural model analyses revealed a model fit for the hypothesised constructs, convergent validity demonstrated the statistical significance of the factor loadings, and composite reliabilities indicated an internal consistency and predictive relevance as detailed in the methodology section.

Furthermore, because of confidentiality issues and COVID-19 disruptions to work routines, we were unable to access relational data which would have warranted using UCINET 6.0 software analytics tool in performing social network analysis for this study. Nonetheless, our choice of data collection was within survey designs in social network research. Relational data would have generated insights into the direction of ties (e.g., reciprocated ties between two people), the level of centralisation and decentralisation, including reciprocity and brokerage roles within these advice networks, on problem-solving competence. While the lack of relational data made it difficult to establish a more interdependent relationship between business advice connectedness and technical advice connectedness and what it means for achieving higher problem-solving competence for software developers, this at least leaves some room for further research to build on this study. This is particularly important for multiplex network relations where a relationship can, but not necessarily have to, involve one network relation or let alone share either business advice or technical advice (c.f. Aalbers et al., 2014). In other words, two software developers may engage in role multiplexity by being connected in business and technical advice networks to share business and technical advice at the same time in a multiplex (overlapping) network (cf. Maruping and Matook, 2020). Our study only followed a multiplex view by combining both business and technical advice networks in a single study compared with some existing studies. However, we did not measure the multiplex effect among software developers because we need access to relational data to capture such interpersonal advice interactions for social network analysis. Again, this will require using UCINET 6.0 to compute such multiplex matrix or configurations technically, which we hope to do in future research. We posit that multiplex ties can increase creativity and the speed to develop greater problem-solving competence in solving complex problems. More importantly, it can specifically address multiplex advantages over uniplex relations among software developers towards problem-solving competence in software development efforts. In doing so, our study would have been able to generate more contributions to the work of Sheremata (2000), where some of these network properties, including reach and free flow of information on problem-solving outcomes, were examined outside the software development domain. For example, network relationships rich in diverse yet balanced and mutual reciprocity in knowledge exchange and problem-solving interactions may facilitate greater shared understanding, and in turn, improve problem-solving competence between team members than their lack thereof. Recent studies have suggested the need for developers to develop trust relationships that promote reciprocity in the exchange of knowledge to sustain healthy participation in the open-source software development effort (Naparat et al., 2015). Additional access to data on individual performance metrics set out in the organisation would provide further insights on variations in software developer networks and problem-solving performance outcomes. However, recent studies have begun to debate that it is not so much about the list of social contacts an individual has but identifying the social network of contacts where relevant resources are accessed that matters in the digital communication era (Wuchty and Uzzi, 2011). By comparing brokerage behaviour and connectedness with contact quality in business and technical advice networks, there is some sense of the patterns of effects in the network behaviours affecting problem-solving competence. Based on these patterns, it would be good to establish whether people with multiplex network (and advice) orientations are central in both networks, discuss through offline and online communications, are friends or colleagues, etc. There is a need to understand the patterns of network interconnectedness and their performance consequences in multiplex network research (Ajimati et al., 2021). These can have profound implications for affecting problem-solving competence and software development performance. This can also invite a mix-method research approach to generate deeper insights into theory and practice in software development networks.

Second, the lack of empirical support for part of our hypotheses such as the direct impact of business and technical advice connectedness (without brokerage) and partial support for business advice connectedness (after moderating the effect of contact quality), needs further investigation in similar settings. This includes understanding what contact quality means for problem-solving and performance in the software development effort of software developers. We assume that contact quality will be based on the usefulness of knowledge, trust in the knowledge source, status within the organisation, network position (e.g., closing structural holes), network cohorts (being connected to well-connected others), and their availability to commit to providing knowledge assistance when called upon. This is because researchers have drawn relevance to the importance of competent, benevolent and integrity-driven trust in learning and knowledge-seeking by individuals or partners (Levin et al., 2002; Muthusamy and White, 2005), as well as system trust (perceptions of the level of provisional support receivable from their organisations) (Ebner et al., 2009) or psychological climate (Purvis and Zagenczyk, 2018) and generally trust repair issues where individuals operate on (Kähkönen et al., 2021). We posit that this can influence quality contact connectedness in advice networks

to drive knowledge quality. However, researchers have considered information quality measurements within advice networks (e.g., Sasidharan et al., 2012), how this compares with contact quality measurements in a similar study on problem-solving competence or other network outcomes would be worth research enquiry. We have adapted the scale for measuring contact quality proposed by van Bel et al. (2009). However, due to an oversight, a recent scale by Fasbender et al. (2020) or their combinations might have predicted a different result for our moderation analysis and more empirical support. Specifically, van Bel et al.'s (2009) scale reads: "*The social contacts with people in my social network feel superficial*; *My relationships with people in my social network feel superficial*; *I derive little satisfaction from my social contacts.*" However, Fasbender et al.'s (2020) scale read: "*Contact with my coworkers is generally positive, negative and cooperative.*" To note, the word "*superficial*" is not explicit and so might mean different things but does not really imply whether the perceived relationship between their social contacts is negative or not cooperative. The word "*satisfaction*" can also mean a natural and cooperative relationship. We submit that this might alter the lack of support, though surprisingly, in these software development networks, which is necessary for further investigation.

Third, studies suggest that individuals' connectedness in offline and online networks matters for knowledge resourcefulness (e.g., van Bel et al., 2009; Zhang and Venkatesh, 2013). However, the position of researchers on whether different communication networks affect performance differently is mixed in extant studies (Ding et al., 2019; Rico and Cohen, 2005; Zhang and Venkatesh, 2013). For example, Rico and Cohen (2005) suggest that individuals interacting synchronously (face-to-face/virtual meetings) and asynchronously (virtual meetings only) can experience different performance benefits. This includes controlling for computer self-efficacy as well as adding computer experience to the existing list of control variables and replicating them using our model, in drawing insights into the plausible generalisation of this research. Extant studies have shown these to be linked to job performance in online communication networks (e.g., Zhang and Venkatesh, 2013). Research evidence suggests that support technologies help improve planning, processes, and problem-solving competence, and in turn, drive project performance (Aladwani, 2002). Regarding brokerage, social psychology theorists have found personality and social cognitive factors — e.g., "*self-monitoring*" (adaptive behaviour), "*openness to experience*" (creative potentials) and "*extraversion*" (social demeanour) including "*conscientiousness*" (professionalism) to be influencing networking behaviours and performance in social networks (Fang et al., 2015; Meng et al., 2016; Sasovova et al., 2010; Stajkovic et al., 2018). However, our study only considered and tested conscientiousness personality as a control variable, in order to align with extant studies (e.g., Sykes et al., 2014). These psychosocial dynamics can affect the intensity and quality of network brokerage and motivation toward problem-solving activities and the potential to vary the problem-solving competence of software developers. Overall, our study does not compare these variations of personality and the effect of business and technical advice connectedness in both offline and online networks and their resulting impact on varying individual problem-solving competence.

Lastly, for the benefit of the above considerations, our findings cannot be generalised but subject to the organisational setting where this data was collected. Notwithstanding, our conceptual model needs more empirical testing before it can be better validated and generalisable. Although our causal model was largely validated, the model is still a cross-sectional representation, inviting future work to focus on longitudinal assessments to validate the causal effects our study implies.

### 5.4. Conclusion

This study responds to calls to investigate the advice network of software developers regarding how their access to and sharing of knowledge improve their problem-solving competence. Our study contributes to the social network literature by exploring the patterns of brokerage processes in business and technical networks and the ways these network patterns affect problem-solving competence in software development settings. In doing so, this study revealed that business and technical advice network connectedness via brokerage for business and technical advice has a positive relationship with improving the problem-solving competence of software developers. However, contact quality has no moderating relationship concerning the effect of both business domain and technical advice connectedness and problem-solving competence. This includes controlling for and establishing a positive relationship that conscientiousness personality and self-efficacy have on problem-solving competence and possibly brokerage activities in both advice networks. While it is expected that a new and long-term member of staff, working in the same or different function, floor, building and office (collocated/distributed) might exhibit different networking behaviours, these different control variables have no significant support for our results. Thus, there is a need for researchers to further investigate these relationships in case there might be a similar or different outcome from our research model. In sum, we conclude that focusing on either technical or business domain advice connectedness based on uniplex network approaches does not explain the overarching network effect on problem-solving competence but by studying both networks simultaneously from a multiplex network perspective. In doing so, it will allow for the variance in individual problem-solving competence in boosting software development effort, solving software problems, and producing quality software products within budgets that run more efficiently meeting requirements, and satisfying customers' commercial objectives. Overall, we suggest that organisational managers must promote organisation-wide advice support networks that cut across business and technical domains, in achieving software development success at individual, team and project levels.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Data availability

The authors do not have permission to share data.

**Table 10**
Measurement scales and items.

| ***Business Advice Connectedness (BAC)*** |
|---|
| Please reflect on the statements below about your experience with people within your ***business advice network*** in providing you with information and knowledge related to problem-solving to get your work done. A *business advice network* relates to the network of people with information and knowledge needed to address business and customer needs such as analysing requirements [1 = completely disagree, 7 = completely agree]. |

(*continued on next page*)

**Table 10** (*continued*).

| |
|---|
| BAC1. I often know what people in my business advice network think<br>BAC2. I often know what people in my business advice network feel<br>BAC3. I am often aware of my relationships with people in my business advice network |
| ***Brokerage in Business Advice Connectedness (BBAC)***<br>Please rate the extent to which the statements below reflect your ***social orientation*** in connecting other work colleagues within your ***business advice network***...? [1 = completely disagree, 7 = completely agree] |
| BBAC1. I introduce people to each other who might have a common strategic work interest<br>BBAC2. I will try to describe an issue in a way that will appeal to a diverse set of interests<br>BBAC3. I see opportunities for collaboration between people<br>BBAC4. I point out the common ground shared by people who have different perspectives on an issue<br>BBAC5. I introduce two people when I think they might benefit from becoming acquainted<br>BBAC6. I forge connections between different people dealing with a particular issue[a] |
| ***Contact Quality in Business Advice Connectedness (CQBAC)***<br>Please rate the extent to which the statements below reflect ***the quality of your social interactions*** within your ***business*** *advice network*...? [1 = completely disagree, 7 = completely agree] |
| CQBAC1. The social contacts with people in my business advice network feel superficial<br>CQBAC2. My relationships with people in my business advice network feel superficial<br>CQBAC3. I derive little satisfaction from my social contacts<br>***Technical Advice Network (TAC)***<br>Please reflect on the statements below about your experience with people within your ***technical advice network*** in providing you with information and knowledge related to problem-solving to get your work done. A *technical advice network* relates to the network of people with information and knowledge needed to solve specific technical problems to develop solutions such as writing and changing source codes to fix bugs [1 = completely disagree, 7 = completely agree].<br>TAC1.I often know what people in my technical advice network think<br>TAC2. I often know what people in my technical advice network feel<br>TAC3. I am often aware of my relationships with people in my technical advice network |
| ***Brokerage in Technical Advice Connectedness (BTAC)***<br>Please rate the extent to which the statements below reflect your *social orientation* in connecting other work colleagues within your ***technical*** *advice network*...? [1 = completely disagree, 7 = completely agree] |
| BTAC1. I introduce people to each other who might have a common strategic work interest<br>BTAC2. I will try to describe an issue in a way that will appeal to a diverse set of interests<br>BTAC3. I see opportunities for collaboration between people<br>BTAC4. I point out the common ground shared by people who have different perspectives on an issue<br>BTAC5. I introduce two people when I think they might benefit from becoming acquainted<br>BTAC6. I forge connections between different people dealing with a particular issue[a] |
| ***Contact Quality in Technical Advice Connectedness (CQTAC)***<br>Please rate the extent to which the statements below reflect ***the quality of your social interactions*** within your ***technical*** *advice network*...? [1 = completely disagree, 7 = completely agree] |
| CQTAC1. The social contacts with people in my technical advice network feel superficial<br>CQTAC2. My relationships with people in my technical advice network feel superficial<br>CQTAC3. I derive little satisfaction from my social contacts |
| ***Problem-Solving Competence (PSC)***<br>Please rate the extent to which the statements below reflect how your ***social relationship*** and ***social contacts*** in the advice networks help you...? [1 = completely disagree, 7 = completely agree] |
| PSC1. Quickly define, find, and implement solutions to the problems<br>PSC2. Evaluate a large number of alternative solutions for each problem quickly<br>PSC3. Found and implemented the right number of alternative solutions to problems<br>PSC4. For each problem encountered with a spoiled choice of solutions[a]<br>PSC5. Implemented solutions found at less cost than in previous projects[a]<br>PSC6. Came up with lower-cost solutions for problems than expected[a]<br>PSC7. Found and implemented creative solutions<br>PSC8. Generate fresh thinking from the solutions found |
| ***Conscientiousness Personality (CP)***<br>Please rate the extent to which the statements below reflect your ***approach to work*** with colleagues...? [1 = completely disagree, 7 = completely agree] |
| CP1. I am always prepared<br>CP2. I pay attention to details<br>CP3. I make plans and stick to them[a]<br>CP4. I waste my time[a]<br>CP5. I find it difficult to get down to work[a] |
| ***Computer Self-Efficacy (CSE)***<br>Please rate the extent to which the statements below reflect your behaviour: ***I could complete a job or task using a technology system...*** [1 = completely disagree, 7 = completely agree] |
| CSE1. If there was no one around to tell me what to do as I go[a]<br>CSE2. If I could call someone for help if I got stuck<br>CSE3. If I had a lot of time to complete the job for which the software was provided. |

[a]Items removed from PLS-SEM regression model due to low factor loadings (<0.70).

## Acknowledgements

This work was supported, in part, by Science Foundation Ireland grant 13/RC/2094_P2 and co-funded under the European Regional Development Fund through the Southern and Eastern Regional Operational Programme to Lero — the Science Foundation Ireland Research Centre for Software (http://www.lero.ie),

National University of Ireland Galway, and co-funded by Radboud University, the Netherlands.

We would like to thank the area editor, Dr. Kelly Blincoe, for the opportunity and time given to revise this manuscript. We sincerely appreciate our anonymous reviewers for their good comments on this revision.

## Appendix

See Table 10.

## References

Aalbers, H.L., Dolfsma, W., 2015. Bridging firm-internal boundaries for innovation: Directed communication orientation and brokering roles. J. Eng. Technol. Manag. - JET-M 36, 97–115. http://dx.doi.org/10.1016/j.jengtecman.2015.05.005.

Aalbers, R., Dolfsma, W., Koppius, O., 2013. Individual connectedness in innovation networks: On the role of individual motivation. Res. Policy 42 (3), 624–634. http://dx.doi.org/10.1016/j.respol.2012.10.007.

Aalbers, R., Dolfsma, W., Koppius, O., 2014. Rich ties and innovative knowledge transfer within a firm. Br. J. Manag. 25 (4), 833–848. http://dx.doi.org/10.1111/1467-8551.12040.

Açıkgöz, A., Ö. İlhan, Ö., 2015. Climate and problem solving in software development teams. Procedia - Soc. Behav. Sci. 207, 502–511. http://dx.doi.org/10.1016/j.sbspro.2015.10.120.

Ajimati, M.O., Aalbers, R., Whelan, E., Kranenburg, H. Van, Morgan, L., 2021. Applying a multiplex network perspective to understand performance in software development. Int. J. Bus. Inf. Syst. 1 (1), 1. http://dx.doi.org/10.1504/IJBIS.2021.10041085.

Aladwani, A.M., 2002. An integrated performance model of information systems projects. J. Manage. Inf. Syst. 19 (1), 185–210. http://dx.doi.org/10.1080/07421222.2002.11045709.

Aldave, A., Vara, J.M., Granada, D., Marcos, E., 2019. Leveraging creativity in requirements elicitation within agile software development: A systematic literature review. J. Syst. Softw. 157, http://dx.doi.org/10.1016/j.jss.2019.110396.

Allen, T.J., 1984. Managing the Flow of Technological Information Within the R & D Organization. MIT Press, Cambridge, Massachusetts.

Amabile, T.M., 1983. The social psychology of creativity: A componential conceptualization. J. Personal. Soc. Psychol. 45 (2), 357–376. http://dx.doi.org/10.1037/0022-3514.45.2.357.

Anwar, R., Rehman, M., Wang, K.S., Amin, A., Akbar, R., 2017. Conceptual framework for implementation of knowledge sharing in global software development organizations. In: IEEE Symposium on Computer Applications and Industrial Electronics. pp. 174–178. http://dx.doi.org/10.1109/ISCAIE.2017.8074972.

Atuahene-Gima, K., Wei, Y., 2011. The vital role of problem-solving competence in new product success. J. Prod. Innov. Manage. 28 (1), 81–98. http://dx.doi.org/10.1111/j.1540-5885.2010.00782.x.

Aurum, A., Daneshgar, F., Ward, J., 2008. Investigating knowledge management practices in software development organisations - An Australian experience. Inf. Softw. Technol. 50 (6), 511–533. http://dx.doi.org/10.1016/j.infsof.2007.05.005.

Balaban, M., Maraee, A., Sturm, A., Jelnov, P., 2015. A pattern-based approach for improving model quality. Software Syst. Model. 14 (4), 1527–1555. http://dx.doi.org/10.1007/s10270-013-0390-0.

Belso-Martínez, J.A., Mas-Tur, A., Roig-Tierno, N., 2017. Synergistic effects and the co-existence of networks in clusters. Entrepreneurship Reg. Dev. 29 (1–2), 137–154. http://dx.doi.org/10.1080/08985626.2016.1255429.

Bernardi, M.L., Canfora, G., Di Lucca, G.A., Di Penta, M., Distante, D., 2018. The relation between developers' communication and fix-inducing changes: An empirical study. J. Syst. Softw. 140, 111–125. http://dx.doi.org/10.1016/j.jss.2018.02.065.

Blincoe, K., Valetto, G., Damian, D., 2015. Facilitating coordination between software developers: A study and techniques for timely and efficient recommendations. IEEE Trans. Softw. Eng. 41 (10), 969–985. http://dx.doi.org/10.1109/TSE.2015.2431680.

Boden, A., Avram, G., 2009. Bridging knowledge distribution - The role of knowledge brokers in distributed software development teams. In: Proceedings of the 2009 ICSE Workshop on Cooperative and Human Aspects on Software Engineering. CHASE 2009, pp. 8–11. http://dx.doi.org/10.1109/CHASE.2009.5071402.

Borgatti, S.P., Cross, R., 2003. A relational view of information seeking and learning in social networks. Manage. Sci. 49 (4), 432–445. http://dx.doi.org/10.1287/mnsc.49.4.432.14428.

Brands, R.A., Kilduff, M., 2014. Just like a woman? Effects of gender-biased perceptions of friendship network brokerage on attributions and performance. Organ. Sci. 25 (5), 1530–1548. http://dx.doi.org/10.1287/orsc.2013.0880.

Brown, J.S., Duguid, P., 1998. Organizing knowledge. Calif. Manage. Rev. 40 (3), 29–46. http://dx.doi.org/10.4324/9780203791189-4.

Bunger, A.C., Doogan, N., Hanson, R.F., Birken, S.A., 2018. Advice-seeking during implementation: A network study of clinicians participating in a learning collaborative. Implement. Sci. 13 (1), 1–12. http://dx.doi.org/10.1186/s13012-018-0797-7.

Burmeister, A., Alterman, V., Fasbender, U., Wang, M., 2021. Too much to know? The cognitive demands of daily knowledge seeking and the buffering role of coworker contact quality. J. Appl. Psychol. http://dx.doi.org/10.1037/apl0000925.

Burt, R.S., 2004. Structural holes and good ideas. Am. J. Sociol. 110 (2), 349–399. http://dx.doi.org/10.1086/421787.

Cataldo, M., Herbsleb, J.D., 2008. Communication networks in geographically distributed software development. In: Proceedings of the ACM Conference on Computer Supported Cooperative Work. CSCW, pp. 579–588. http://dx.doi.org/10.1145/1460563.1460654.

Colazo, J., 2014. Performance implications of stage-wise lead user participation in software development problem solving. Decis. Support Syst. 67, 100–108. http://dx.doi.org/10.1016/j.dss.2014.08.007.

Coltman, T., Devinney, T.M., Midgley, D.F., Venaik, S., 2008. Formative versus reflective measurement models: Two applications of formative measurement. J. Bus. Res. 61 (12), 1250–1262. http://dx.doi.org/10.1016/j.jbusres.2008.01.013.

Conaldi, G., Lomi, A., 2013. The dual network structure of organizational problem solving: A case study on open source software development. Social Networks 35 (2), 237–250. http://dx.doi.org/10.1016/j.socnet.2012.12.003.

Conklin, J., Stolee, P., Harris, M., Lusk, E., 2013. Knowledge brokers in a knowledge network: The case of SHRTN knowledge brokers. Implement. Sci. 8 (7), 1–10.

Constant, D., Sproull, L., Kiesler, S., 1996. The kindness of strangers: The usefulness of electronic weak ties for technical advice. Organ. Sci. 7 (2), 119–135. http://dx.doi.org/10.1287/orsc.7.2.119.

Coughlan, J., Lycett, M., Macredie, R.D., 2003. Communication issues in requirements elicitation: A content analysis of stakeholder experiences. Inf. Softw. Technol. 45 (8), 525–537. http://dx.doi.org/10.1016/S0950-5849(03)00032-6.

Cross, R., Borgatti, S.P., Parker, A., 2001. Beyond answers: Dimensions of the advice network. Soc. Networks 23, 215–235.

Dawson, J.F., 2014. Moderation in management research: What, why, when, and how. J. Bus. Psychol. 29 (1), 1–19. http://dx.doi.org/10.1007/s.

Ding, D., Bhattacharya, P., Phan, T.Q., Kong, H., 2019. A tale of two networks: Characterizing the interplay between online and offline communication networks. In: Fortieth International Conference on Information Systems, Munich 2019. pp. 1–9.

Ebner, W., Leimeister, J.M., Krcmar, H., 2009. Community engineering for innovations: The ideas competition as a method to nurture a virtual community for innovations. R D Manag. 39 (4), 342–356. http://dx.doi.org/10.1111/j.1467-9310.2009.00564.x.

Ehrlich, K., Cataldo, M., 2014. The communication patterns of technical leaders: Impact on product development team performance. In: Proceedings of the ACM Conference on Computer Supported Cooperative Work. CSCW, pp. 733–744. http://dx.doi.org/10.1145/2531602.2531671.

Ellwardt, L., Steglich, C., Wittek, R., 2012. The co-evolution of gossip and friendship in workplace social networks. Social Networks 34 (4), 623–633. http://dx.doi.org/10.1016/j.socnet.2012.07.002.

Erdogan, B., Karaeminogullari, A., Bauer, T.N., Ellis, A.M., 2020. Perceived overqualification at work: Implications for extra-role behaviors and advice network centrality. J. Manage. 46 (4), 583–606. http://dx.doi.org/10.1177/0149206318804331.

Espinosa, J.A., Clark, M.A., Carter, D.R., 2018. Understanding shared familiarity and team performance through network analytics. In: Proceedings of the 51st Hawaii International Conference on System Sciences. pp. 1–10.

Fang, R., Landis, B., Zhang, Z., Anderson, M.H., Shaw, J.D., Kilduff, M., 2015. Integrating personality and social networks: A meta-analysis of personality, network position, and work outcomes in organizations. Organ. Sci. 26 (4), 1243–1260.

Farooq, R., Vij, S., 2017. Moderating variables in business research. IUP J. Bus. Strategy 14 (4), 34–54.

Fasbender, U., Burmeister, A., Wang, M., 2020. Motivated to be socially mindful: Explaining age differences in the effect of employees' contact quality with coworkers on their coworker support. Pers. Psychol. 73 (3), 407–430. http://dx.doi.org/10.1111/peps.12359.

Faul, F., Erdfelder, E., Buchner, A., Lang, A.G., 2009. Statistical power analyses using G*Power 3.1: Tests for correlation and regression analyses. Behav. Res. Methods 41 (4), 1149–1160. http://dx.doi.org/10.3758/BRM.41.4.1149.

Fox, S., Dinur, Y., 1988. Validity of self-assessment a field evaluation.pdf. Pers. Psychol..

Freeman, L.C., 1978. Centrality in social networks conceptual clarification. Social Networks 1 (3), 215–239. http://dx.doi.org/10.1016/0378-8733(78)90021-7.

Funke, J., 2010. Complex problem solving: A case for complex cognition? Cogn. Process. 11 (2), 133–142. http://dx.doi.org/10.1007/s10339-009-0345-0.

Ghobadi, S., 2015. What drives knowledge sharing in software development teams: A literature review and classification framework. Inf. Manag. 52 (1), 82–97. http://dx.doi.org/10.1016/j.im.2014.10.008.

Gibbons, D.E., 2004. Friendship and advice networks in the context of changing professional values. Adm. Sci. Q. 49 (2), 238–262.
Gloor, P.A., Oster, D., Putzke, J., Fischbach, K., Schoder, D., Ara, K., et al., 2007. Studying microscopic peer-to-peer communication patterns. In: Association for Information Systems - 13th Americas Conference on Information Systems, AMCIS 2007: Reaching New Heights, Vol. 7. pp. 4391–4402. http://dx.doi.org/10.2139/ssrn.1097111.
Gotel, O.C.Z., Finkelstein, A.C.W., 1994. An analysis of the requirements traceability problem. In: Proceedings of the First International Conference on Requirements Engineering. pp. 94–101.
Granovetter, M.S., 1973. Strength of weak ties. Am. J. Sociol. 78 (6), 1360–1380. http://dx.doi.org/10.1007/sl.
Gurteen, D., Associates, K., 1998. Knowledge, creativity and innovation. J. Knowl. Manag. 2 (1), 5–13.
Haas, A., 2015. Crowding at the frontier: Boundary spanners, gatekeepers and knowledge brokers. J. Knowl. Manag. 19 (5), 1029–1047. http://dx.doi.org/10.1108/JKM-01-2015-0036.
Hair, J., Hollingsworth, C.L., Randolph, A.B., Chong, A.Y.L., 2017. An updated and expanded assessment of PLS-SEM in information systems research. Ind. Manag. Data Syst. 117 (3), 442–458. http://dx.doi.org/10.1108/IMDS-04-2016-0130.
Hair, Joseph F., Risher, J.J., Sarstedt, M., Ringle, C.M., 2019. When to use and how to report the results of PLS-SEM. Eur. Bus. Rev. 31 (1), 2–24. http://dx.doi.org/10.1108/EBR-11-2018-0203.
Hair, Joe F., Sarstedt, M., Hopkins, L., Kuppelwieser, V.G., 2014. Partial least squares structural equation modeling (PLS-SEM): An emerging tool in business research. Eur. Bus. Rev. 26 (2), 106–121. http://dx.doi.org/10.1108/EBR-10-2013-0128.
Henseler, J., Ringle, C.M., Sarstedt, M., 2015. A new criterion for assessing discriminant validity in variance-based structural equation modeling. J. Acad. Mark. Sci. 43 (1), 115–135. http://dx.doi.org/10.1007/s11747-014-0403-8.
Hoegl, M., Parboteeah, K.P., 2007. Creativity in innovative projects: How teamwork matters. J. Eng. Technol. Manag. - JET-M 24 (1–2), 148–166. http://dx.doi.org/10.1016/j.jengtecman.2007.01.008.
Homburg, C., Wieseke, J., Kuehnl, C., 2010. Social influence on salespeople's adoption of sales technology: A multilevel analysis. J. Acad. Mark. Sci. 38 (2), 159–168. http://dx.doi.org/10.1007/s11747-009-0157-x.
Iden, J., Bygstad, B., 2018. The social interaction of developers and IT operations staff in software development projects. Int. J. Project Manag. 36 (3), 485–497. http://dx.doi.org/10.1016/j.ijproman.2017.12.001.
Kähkönen, T., Blomqvist, K., Gillespie, N., Vanhala, M., 2021. Employee trust repair: A systematic review of 20, years of empirical research and future research directions. J. Bus. Res. 130 (October 2020), 98–109. http://dx.doi.org/10.1016/j.jbusres.2021.03.019.
Kimball Romney, A., Weller, S.C., 1984. Predicting informant accuracy from patterns of recall among individuals. Social Networks 6 (1), 59–77. http://dx.doi.org/10.1016/0378-8733(84)90004-2.
Kislov, R., Hodgson, D., Boaden, R., 2016. Professionals as knowledge brokers: The limits of authority in healthcare collaboration. Public Adm. 94 (2), 472–489. http://dx.doi.org/10.1111/padm.12227.
Kobayashi, M., 2015. Requirement for engineers in embedded software development outsourcing. Ann. Bus. Adm. Sci. 14 (2), 97–107. http://dx.doi.org/10.7880/abas.14.97.
Koo, H.M., Ko, I.Y., 2017. Construction and utilization of problem-solving knowledge in open source software environments. J. Syst. Softw. 131, 402–418. http://dx.doi.org/10.1016/j.jss.2016.06.062.
Kwon, S.W., Rondi, E., Levin, D.Z., De Massis, A., Brass, D.J., 2020. Network brokerage: An integrative review and future research agenda. J. Manag. 46 (6), 1092–1120. http://dx.doi.org/10.1177/0149206320914694.
Lam, A., 2018. Boundary-crossing careers and the 'third space of hybridity': Career actors as knowledge brokers between creative arts and academia. Environ. Plan. A: Econ. Space 50 (8), 1716–1741. http://dx.doi.org/10.1177/0308518X17746406.
Levin, D.Z., Cross, R., Abrams, L.C., 2002. The strength of weak ties you can trust: The mediating role of trust in effective knowledge transfer. Acad. Manag. J. 1–39.
Li, Yuan, Li, N., Guo, J., Li, J., Harris, T.B., 2018. A network view of advice-giving and individual creativity in teams: A brokerage-driven, socially perpetuated phenomenon. Acad. Manag. J. 61 (6), 2210–2229. http://dx.doi.org/10.5465/amj.2016.0212.
Li, Yuzhu, Yang, M.H., Klein, G., Chen, H.G., 2011. The role of team problem solving competency in information system development projects. Int. J. Project Manag. 29 (7), 911–922. http://dx.doi.org/10.1016/j.ijproman.2010.09.004.
Licorish, S.A., Macdonell, S.G., 2015. Communication and personality profiles of global software developers. Inf. Softw. Technol. 64, 113–131. http://dx.doi.org/10.1016/j.infsof.2015.02.004.
Lin, T.C., Chen, C.M., Hsu, J.S.C., Fu, T.W., 2015. The impact of team knowledge on problem solving competence in information systems development team. Int. J. Project Manag. 33 (8), 1692–1703. http://dx.doi.org/10.1016/j.ijproman.2015.07.007.
Long, J.C., Cunningham, F.C., Braithwaite, J., 2013. Bridges, brokers and boundary spanners in collaborative networks: A systematic review. BMC Health Serv. Res. 13, 158.
Lopes, A.L., Botelho, L.M., 2013. Distributed coordination of heterogeneous agents using a semantic overlay network and a goal-directed graphplan planner. PLoS One 8 (5), http://dx.doi.org/10.1371/journal.pone.0062931.
Lorant, V., Soto, V.E., Alves, J., Federico, B., Kinnunen, J., Kuipers, M., et al., 2015. Smoking in school-aged adolescents: Design of a social network survey in six European countries. BMC Res. Not. 8 (91), 1–12. http://dx.doi.org/10.1186/s13104-015-1041-z.
Loury, L.D., 2006. Some contacts are more equal than others: Informal networks, job tenure, and wages. J. Labor Econ. 24 (2), 299–318. http://dx.doi.org/10.1086/499974.
MacGeorge, E.L., Feng, B., Guntzviller, L.M., Feng, B., 2016. Advice: Expanding the communication paradigm. Ann. Int. Commun. Assoc. 40 (1), 213–243. http://dx.doi.org/10.1080/23808985.2015.11735261.
Maheshwari, M., Kumar, U., Kumar, V., 2012. Alignment between social and technical capability in software development teams: An empirical study. Team Perform. Manag.: Int. J. 18 (1–2), 7–26. http://dx.doi.org/10.1108/13527591211207680.
Maruping, L.M., Matook, S., 2020. The multiplex nature of the customer representative role in agile information systems development. MIS Q. 44 (3), 1411–1437. http://dx.doi.org/10.25300/misq/2020/12284.
McGrath, C.A., Vance, C.M., Gray, E.R., 2003. With a little help from their friends: Exploring the advice networks of software entrepreneurs. Creativity Innov. Manag. 12 (1), 2–10. http://dx.doi.org/10.1111/1467-8691.00262.
McManus, J.L., Feyes, K.J., Saucier, D.A., 2010. Contact and knowledge as predictors of attitudes toward individuals with intellectual disabilities. J. Soc. Pers. Relationsh. 28 (5), 579–590. http://dx.doi.org/10.1177/0265407510385494.
Mehta, N., Bharadwaj, A., 2015. Knowledge integration in outsourced software development: The role of sentry and guard processes. J. Manage. Inf. Syst. 32 (1), 82–115. http://dx.doi.org/10.1080/07421222.2015.1029381.
Meister, M., Böing, T., Batz, S., Metternich, J., 2018. Problem-solving process design in production: Current progress and action required. Procedia CIRP 78, 376–381. http://dx.doi.org/10.1016/j.procir.2018.08.316.
Meng, L., Hulovatyy, Y., Striegel, A., Milenković, T., 2016. On the interplay between individuals' evolving interaction patterns and traits in dynamic multiplex social networks. IEEE Trans. Netw. Sci. Eng. 3 (1), 32–43. http://dx.doi.org/10.1109/TNSE.2016.2523798.
Merluzzi, J., 2017. Gender and negative network ties: Exploring difficult work relationships within and across gender. Organ. Sci. 28 (4), 636–652. http://dx.doi.org/10.1287/orsc.2017.1137.
Mishra, D., Mishra, A., Ostrovska, S., 2012. Impact of physical ambiance on communication, collaboration and coordination in agile software development: An empirical evaluation. Inf. Software Technol. 54 (10), 1067–1078. http://dx.doi.org/10.1016/j.infsof.2012.04.002.
Mohammadi, A., Broström, A., Franzoni, C., 2017. Workforce composition and innovation: How diversity in employees' ethnic and educational backgrounds facilitates firm-level innovativeness. J. Prod. Innov. Manage. 34 (4), 406–426. http://dx.doi.org/10.1111/jpim.12388.
Muthusamy, S.K., White, M.A., 2005. Learning and knowledge transfer in strategic alliances: A social exchange view. Organ. Stud. 26 (3), http://dx.doi.org/10.1177/0170840605050874.
Muthusamy, S.K., White, M.A., White, M.A., 2005. Learning and knowledge transfer in strategic alliances : A learning and knowledge transfer in strategic alliances : A social exchange view, (March). http://dx.doi.org/10.1177/0170840605050874.
Naparat, D., Cahalane, M., Finnegan, P., 2015. Healthy community and healthy commons: Opensourcing as a sustainable model of software production. Australas. J. Inf. Syst. 19, 1–13. http://dx.doi.org/10.3127/ajis.v19i0.1221.
Nidumolu, S., 1995. The effect of coordination and uncertainty on software project performance: Residual performance risk as an intervening variable. Inf. Syst. Res. 6 (3), 191–219. http://dx.doi.org/10.1287/isre.6.3.191.
Obstfeld, D., 2005. Social networks, the Tertius Iungens orientation, and involvement in innovation. Adm. Sci. Q. 50 (1), 100–130.
Obstfeld, D., Borgatti, S.P., Davis, J., 2014. Brokerage as a process: Decoupling third party action from social network structure. Res. Soc. Organ. 40, 135–159. http://dx.doi.org/10.1108/S0733-558X(2014)0000040007.
Ozer, M., Vogel, D., 2015. Contextualized relationship between knowledge sharing and performance in software development. J. Manage. Inf. Syst. 32 (2), 134–161. http://dx.doi.org/10.1080/07421222.2015.1063287.
Parise, S., Cross, R., Davenport, T.H., 2006. Strategies for preventing a knowledge-loss crisis. MIT Sloan Manag. Rev. 47 (4), 31–38.
Parker, A., Halgin, D.S., Borgatti, S.P., 2015. Dynamics of social capital: Effects of performance feedback on network change. Organ. Stud. 37 (3), 375–397. http://dx.doi.org/10.1177/0170840615613371.
Peng, G., Wan, Y., Woodlock, P., 2013. Network ties and the success of open source software development. J. Strateg. Inf. Syst. 22 (4), 269–281. http://dx.doi.org/10.1016/j.jsis.2013.05.001.
Perry-Smith, J.E., 2006. Social yet creative: The role of social relationships in facilitating individual creativity. Acad. Manag. J. 49 (1), 85–101. http://dx.doi.org/10.5465/AMJ.2006.20785503.
Pitts, M.G., Browne, G.J., 2007. Improving requirements elicitation: An empirical investigation of procedural prompts. Inf. Syst. J. 17, 89–110.
Purvis, R., Zagenczyk, T., 2018. The development of psychological climate within information systems projects and its impact on project performance. In: European Conference on Information Systems. pp. 1–15.

Reagans, R., McEvily, 2003. Network structure and knowledge transfer: The effects of cohesion and range. Adm. Sci. Q. 48, 240–267.
Reus, B., Moser, C., Groenewegen, P., 2020. Expanding organisational knowledge online: The role of bridging members in knowledge expansion in online groups. Innov.: Organ. Manag. 1–24. http://dx.doi.org/10.1080/14479338.2020.1840381.
Rico, R., Cohen, S.G., 2005. Effects of task interdependence and type of communication on performance. J. Manag. Psychol. 20 (3–4), 261–274. http://dx.doi.org/10.1108/02683940510589046.
Ringle, C.M., Sarstedt, M., 2016. Gain more insight from your PLS-SEM results the importance-performance map analysis. Ind. Manag. Data Syst. 116 (9), 1865–1886. http://dx.doi.org/10.1108/IMDS-10-2015-0449.
Roth, A.R., Peng, S., Coleman, M.E., Finley, E., Perry, B.L., 2021. Network recall among older adults with cognitive impairments. Social Networks 64 (September 2020), 99–108. http://dx.doi.org/10.1016/j.socnet.2020.08.005.
Rus, I., Lindvall, M., Sinha, S.S., 2002. Knowledge management in software engineering. IEEE Software 19 (3), 26–38. http://dx.doi.org/10.1109/MS.2002.1003450.
Saltz, J.S., Yilmazel, S., Yilmazel, O., 2016. Not all software engineers can become good data engineers. In: Proceedings - 2016 IEEE International Conference on Big Data, Big Data 2016. pp. 2896–2901. http://dx.doi.org/10.1109/BigData.2016.7840939.
Sasidharan, S., Santhanam, R., Brass, D.J., Sambamurthy, V., 2012. The effects of social network structure on enterprise systems success: A longitudinal multilevel analysis. Inf. Syst. Res. 23 (3), 658–678.
Sasovova, Z., Mehra, A., Borgatti, S.P., Schippers, M.C., 2010. Network churn: The effects of self-monitoring personality on brokerage dynamics. Adm. Sci. Q. 55 (4), 639–670. http://dx.doi.org/10.2189/asqu.2010.55.4.639.
Sawyer, S., Guinan, P.J., Cooprider, J., 2010. Social interactions of information systems development teams: A performance perspective. Inf. Syst. J. 20, 81–107. http://dx.doi.org/10.1111/j.1365-2575.2008.00311.x.
Schreiber, R.R., Zylka, M.P., 2020. Social network analysis in software development projects: A systematic literature review. Int. J. Software Eng. Knowl. Eng. 30, http://dx.doi.org/10.1142/S021819402050014X.
Shah, N.P., Cross, R., Levin, D.Z., 2018. Performance benefits from providing assistance in networks: Relationships that generate learning. J. Manag. 44, http://dx.doi.org/10.1177/0149206315584822.
Shah, N.P., Parker, A., Waldstrøm, C., 2017. Examining the overlap: Individual performance benefits of multiplex relationships. Manag. Commun. Q. 31 (1), 5–38. http://dx.doi.org/10.1177/0893318916647528.
Sheremata, W.A., 2000. Centrifugal and centripetal forces in radical new product development under time. Acad. Manag. 25 (2), 389–408.
Siciliano, M.D., 2015. Advice networks in public organizations: The role of structure, internal competition, and individual attributes. Public Adm. Rev. 75 (4), 548–559. http://dx.doi.org/10.1111/puar.12362.
Šmite, D., Moe, N.B., Šāblis, A., Wohlin, C., 2017. Software teams and their knowledge networks in large-scale software development. Inf. Softw. Technol. 86, 71–86. http://dx.doi.org/10.1016/j.infsof.2017.01.003.
Snijders, T.A.B., Lomi, A., Jasmine, V., 2013. A model for the multiplex dynamics of two-mode and one-mode networks, with an application to employment preference, friendship, and advice. Soc. Networks 35, 265–276.
Stajkovic, A.D., Bandura, A., Locke, E.A., Lee, D., Sergent, K., 2018. Test of three conceptual models of influence of the big five personality traits and self-efficacy on academic performance: A meta-analytic path-analysis. Pers. Individ. Differ. 120, 238–245. http://dx.doi.org/10.1016/j.paid.2017.08.014.
Stenmark, D., 2002. Information vs. knowledge: The role of intranets in knowledge management. In: Proceedings of the Annual Hawaii International Conference on System Sciences. pp. 928–937. http://dx.doi.org/10.1109/HICSS.2002.994043.
Strode, D.E., Huff, S.L., Hope, B., Link, S., 2012. Coordination in co-located agile software development projects. J. Syst. Softw. 85 (6), 1222–1238. http://dx.doi.org/10.1016/j.jss.2012.02.017.
Su, C., Contractor, N., 2011. A multidimensional network approach to studying team members's information seeking from human and digital knowledge sources in consulting firms. J. Am. Soc. Inf. Sci. Technol. 62 (7), 1257–1275.
Sykes, T.A., Venkatesh, V., Johnson, J.L., 2014. Enterprise system implementation and employee job performance: Understanding the role of advice networks. MIS Q. 38 (1).
Tassabehji, R., Mishra, J.L., Dominguez-Péry, C., 2019. Knowledge sharing for innovation performance improvement in micro/SMEs: An insight from the creative sector. Prod. Plan. Control 30 (10–12), 935–950. http://dx.doi.org/10.1080/09537287.2019.1582101.
Tehseen, S., Sajilan, S., Gadar, K., Ramayah, T., 2017. Assessing cultural orientation as a reflective-formative second order construct-a recent PLS-SEM approach. Rev. Integr. Bus. Econ. Res. 6 (2), 38, Retrieved from http://buscompress.com/journal-home.html.
Tekinerdogan, B., Aksit, M., 2006. Integrating the concept of synthesis in the architecture design process. Trans. SDPS 10 (1), 45–56.
Thomas, J.C., Lee, A., Danis, C., 2002. Enhancing creative design via software tools. Commun. ACM 45 (10), 112–115. http://dx.doi.org/10.1145/570907.570944.
Tiwana, A., 2004. An empirical study of the effect of knowledge integration on software development performance. Inf. Softw. Technol. 46 (13), 899–906. http://dx.doi.org/10.1016/j.infsof.2004.03.006.
Tiwana, A., 2012. Novelty-knowledge alignment: A theory of design convergence in systems development. J. Manage. Inf. Syst. 29 (1), 15–52. http://dx.doi.org/10.2753/MIS0742-1222290101.
van Bel, D., Smolders, K.C.H., Ijsselsteijn, W.A., de Kort, Y., 2009. Social connectedness: Concept and Measurement. In: Intelligent Environments, IOS Press, http://dx.doi.org/10.3233/978-1-60750-034-6-67.
Vargas, N., Schafer, M.H., 2013. Diversity in action: Interpersonal networks and the distribution of advice. Soc. Sci. Res. 42 (1), 46–58. http://dx.doi.org/10.1016/j.ssresearch.2012.08.013.
Verbeke, W., Dietz, B., Verwaal, E., 2010. Drivers of sales performance: A contemporary meta-analysis, have salespeople become knowledge brokers? J. Acad. Mark. Sci. 38 (3), 407–428. http://dx.doi.org/10.1007/s11747-010-0211-8.
Vermerris, A., Mocker, M., Van Heck, E., 2014. No time to waste: The role of timing and complementarity of alignment practices in creating business value in IT projects. Eur. J. Inf. Syst. 23 (6), 629–654. http://dx.doi.org/10.1057/ejis.2013.11.
Wang, Z., Wang, N., 2012. Knowledge sharing, innovation and firm performance. Expert Syst. Appl. 39 (10), 8899–8908. http://dx.doi.org/10.1016/j.eswa.2012.02.017.
Wellman, B., 1983. Network analysis: Some basic principles. Sociol. Theory 1, 155–200. http://dx.doi.org/10.2307/j.ctv5rf5rj.4.
Whelan, E., Anderson, J., van den Hooff, B., Donnellan, B., 2015. How IT and the rest of the business can innovate together. Commun. Assoc. Inf. Syst. 36, 261–271. http://dx.doi.org/10.17705/1cais.03614.
Whelan, E., Conboy, K., 2014. Understanding the role of advice networks in software development. In: ECIS 2014 Proceedings - 22nd European Conference on Information Systems. pp. 1–9.
Whelan, E., Golden, W., Donnellan, B., 2013. Digitising the R & D social network: Revisiting the technological gatekeeper. Inf. Syst. J. 23 (3), 197–218. http://dx.doi.org/10.1111/j.1365-2575.2011.00384.x.
Whelan, E., Parise, S., De Valk, J., Aalbers, R., 2011. Creating employee networks that deliver open innovation. MIT Sloan Manag. Rev. 53 (1), 37–44.
Williamson, G.M., Clark, M.S., 1989. Providing help and desired relationship type as determinants of changes in moods and self-evaluations. J. Personal. Soc. Psychol. 56 (5), 722–734. http://dx.doi.org/10.1037/0022-3514.56.5.722.
Wirth, J., Maier, C., Laumer, S., Weitzel, T., 2021. Laziness as an explanation for the privacy paradox: A longitudinal empirical investigation. Internet Res. (2015), http://dx.doi.org/10.1108/INTR-10-2019-0439.
Wong, S., 2008. Task knowledge overlap and knowledge variety: The role of advice network structures and impact on group effectiveness. J. Organ. Behav. 29, 591–614. http://dx.doi.org/10.1002/job.
Wuchty, S., Uzzi, B., 2011. Human communication dynamics in digital footsteps: A study of the agreement between self-reported ties and email networks. PLoS One 6 (11), http://dx.doi.org/10.1371/journal.pone.0026972.
Xu, Y., Kim, H.W., Kankanhalli, A., 2010. Task and social information seeking: Whom do we prefer and whom do we approach? J. Manage. Inf. Syst. 27 (3), 211–240. http://dx.doi.org/10.2753/MIS0742-1222270308.
Yilmaz, L., 2008. Innovation systems are self-organizing complex adaptive systems. In: AAAI Spring Symposium - Technical Report, SS-08-03. pp. 142–148.
Zand, D.E., 1972. Trust and managerial problem solving. Adm. Sci. Q. 3 (3), 229–239. http://dx.doi.org/10.1177/105960117800300306.
Zhang, X., Venkatesh, V., 2013. Explaining employee job performance: The role of online and offline workplace communication networks. MIS Q. 37 (3), 695–722.

**Matthew O. Ajimati** holds a Bachelor's degree with honours and a Master of Science in Strategy and Innovation from Maynooth University, Ireland. He is a Doctoral Researcher working on a joint Ph.D. project with the National University of Ireland Galway and Radboud University, the Netherlands. His research interests include multiplexity, social network, knowledge brokerage, problem-solving and innovation within and beyond software development settings. He is particularly interested in using a multilevel network approach to understand and explain how software development teams can optimally influence their ability to solve a variety of software problems, enhance creative interactions and performance towards software/project development success.

**Eoin Whelan** is a Senior Lecturer in Business Information Systems at the National University of Ireland, Galway, Ireland. He is also a Visiting Professor at the Institute d'Economie Scientifique et de Gestion (IESEG), France, and visiting researcher at Stellenbosch University, South Africa. His research explores the psychology underlying engagement with interactive digital media such as smartphones, social networking sites, fitness tracking apps and online gambling and gaming sites. His publications have appeared in *Information Systems Journal*, *Journal of Information Technology*, *European Journal of Information Systems*, *MIT Sloan Management Review* and R&D Management.

**Rick (H.L.) Aalbers** is an Associate Professor of Strategy and Innovation at Radboud University, the Netherlands. His main research area is on collaborative innovation strategies and organisational change, with a special interest in

conditions of sudden intervention and shock. His work has been published in *Research Policy*, *Long Range Planning*, *Journal of Product Innovation Management*, *Journal of Engineering and Technology Management*, *MIT Sloan Management Review*, *Management Decision* and *British Journal of Management*, among others.

**Lorraine Morgan** is a Lecturer in Business Information Systems and Senior Researcher with Lero (the Irish Software Engineering Research Centre) at the National University of Ireland Galway. Her principal research interests include open innovation, open-source software, inner source, and crowdfunding. She has also been involved in a number of collaborative research projects involving international- and national-based industry partners, attracting over €20 million in funding. Her research has also been published in leading journals including the *Journal of Strategic Information Systems*, *European Journal of Information Systems*, *Database for Advances in Information Systems* and *Information and Software Technology*.

**Hans van Kranenburg** is a Full Professor of Corporate Strategy at Radboud University, the Netherlands. He is the Head of the Strategic Management group. He was a visiting scholar at the Department of Politics and International Relations, Reuters Institute and Green Templeton College at the University of Oxford (UK), Media Management and Transformation Center at Jönköping International Business School (Sweden), University of Navarra (Spain) and the University of Chicago (USA). He published on the strategic behaviour of organisations, industry dynamics, alliances and networks, foreign direct investments, non-market strategies and stakeholder management. He also is an expert in media management and economics.