Kashf Journal of Multidisciplinary Research
Vol: 01 - Issue 12 (2024)
P-ISSN: 3007-1992 E-ISSN: 3007-200X
https://kjmr.com.pk

A SYSTEMATIC REVIEW ON SOCIO-TECHNICAL CONGRUENCE

Binish Raza
Department of Computer Science & Technologies, Emerson University Multan, Multan, Pakistan.
Sehrish Raza
Institute of Computer Science and Information Technology, The Women University, Multan, 60000, Pakistan.
****Muhammad Ahsan Raza***
Department of Information Science, University Of Education, Lahore, Multan Campus 60000, Pakistan.

****(Corresponding Author)*** (ahsan.raza@ue.edu.pk)
DOI: (*https://doi.org/10.71146/kjmr155*)

Article Info

Abstract

Software development is a critical task that requires team effort and effective coordination among team members. The optimized coordination proportion brings team members together to efficiently perform organizational activities. To control the level of coordination, there are numerous techniques applied in the literature. Socio-Technical Congruence (STC) is one of these existing techniques that help to measure the balance between technical and social aspects of an organization at various stages of software development. The key aim of the proposed work is to perform a systematic review of existing STC-related studies to identify their prominent features and significance in software development. The relevant publications were collected from different databases, including journal articles, conferences, and proceedings, mainly published in the period of 2008 to 2024. A detailed analysis was used to identify the related studies in accordance with the proposed research questions. The results of the review contain a sufficient number of studies that highlight different aspects, characteristics, significance, and new trends in the domain of STC. The proposed study outcomes provide various dimensions of STC with some future suggestions for researchers. Although the literature provides many studies related to STC, till now, only one study exists that mainly discusses the risk of overwhelming STC, so this area of research can be further investigated. The existing STC studies identify the congruence gap, but none have provided an appropriate solution to overcome the identified gap using or building a tool, so this area needs to be addressed. Additionally, the existing STC measurement techniques were generally adaptations of Cataldo et al.'s or Kwan et al.'s work, setting STC measurement for the development phase of the project life cycle. Only a few of the techniques have been discussed in other related phases, such as the requirements or testing phase, or throughout the project life cycle. As STC is a multi-dimensional technique that can cover all levels of the project life cycle, this area of study needs more research.

Keywords: *Socio-Technical Congruence (STC), Software Development Coordination, Systematic Literature Review, Congruence Measurement Techniques.*

1. Introduction

Today, the complex and time-consuming nature of tasks in software development needs years for completion, which increases the budget and forces developers to improve their skills (Omoronyia et al., 2010; Tahir, 2016). Nowadays, it is also considered a key activity of other industries that support their products with innovative software. Due to the multifaceted and time-intensive nature of tasks, today it is common for software development teams to be geographically separated and distributed. The geographical distribution of the development team and the modular nature of task assignment reduce labour costs, but on the other hand, it also increases the risks of team collaboration and coordination (Omoronyia et al., 2010; Baird and Riggins, 2012). In spite of more than 30 years of research, the existence of ample innovative tools and techniques in the domain of distributed development, issues, and failures is still a common and focused area for researchers, particularly in global software development (Abbasi et al., 2019; Maheshwari et al., 2012).

In distributed software development, the key complexity is to understand the way through which information circulates among teams and team members (Datta, 2017) and the identification of key individuals in the communication network (Sangwan et al.,2008). The critical nature of the development process needs team effort whose effectiveness depends on the interaction between the team members and numerous activities that bring team members together. The interdependencies among tasks can be tackled by introducing social capabilities, such as communication among team members, which help developers to coordinate and achieve the same goal (Kuhrmann et al., 2016). The structuring of coordination and collaboration is a complex activity that not only affects team performance but also impacts the software product quality. In literature, various techniques are utilized to control the level of coordination among team members and organizational activities.

Socio-Technical Congruence (STC) is one of these prominent existing approaches used to measure the fit between technical and social aspects of software development at different stages of development (Cataldo et al., 2006). The fit represents a sufficient level of coordination among team members with respect to social and technical abilities, which results in improved team performance and product quality (Cataldo & Herbsleb, 2013). STC helps to measure the level of coordination in early stages of development, which assists an organization in identifying the gap in coordination that can cause delays in work, resulting in project failure. In the literature, an ample amount of studies exist related to STC and its impact on the success of distributed software development. A systematic review can give a synthesized view of these existing studies that can help the researchers to find a new interpretation of STC other than that provided by a single study (Cruzes & Dybå, 2010).

Our main contribution is assorted; it provides a detailed analysis of relevant studies along with a new systematic view of STC. The proposed study work with a research objective, to investigate prominent aspects and the new trends in the field of STC. To achieve the defined objective, the studies synthesized from the literature that contain different descriptions of socio-technical congruence emphasize the benefits, issues, and challenges in STC measurement.

The review findings will help developers or organizations to minimize the risk associated with software development due to a lack of

coordination. In addition, the proposed work helps developers to detect the coordination issues at the early stage of development, which improves the team performance and product quality (J. Portillo-Rodríguez, 2013). Lastly, we consider the overall research efficiency and prospective research directions in this area of research.

This paper is divided into several sections. In the introduction section, the STC background, research gap, and motivation of the proposed study are explained. Section 2 discusses the methodology to perform a systematic review that helps to achieve the proposed objective. Next, presents the results and detailed discussion. The last section concluded the study with some future research directions.

2. Research Methodology

A systematic review assists in determining, evaluating, and interpreting the literature related to research questions that depict the latest trends and significance of the target topic in different fields or areas. The key idea of this study is to collect the evidence from the related studies on which the result is based. The steps of the review procedure adopted to perform this study are presented in Figure 1.

Figure 1: Review Steps

2.1 Define Research Questions (RQs)

The key purpose of this review is to discover the latest research, trends, and impact of STC in distributed software development. The following research questions (RQ's) are derived from the study objective to facilitate achieving the main aim of the study.

RQ1. How is STC defined in literature?
RQ2. What are the main features of STC?
RQ3. What are the benefits and issues related to STC?

2.2 Search the Literature

The initial search is performed on different databases to discover the key search terms to extract relevant literature. The search results obtained from these identified terms have many irrelevant studies that do not focus on the STC with respect to software development.

To refine irrelevant search, the study has added some contextual terms in the search query to get the maximum possible relevant results and to also validate the search accuracy (Kitchenham & Charters, 2007). Initially, we tried several combinations to search relevant studies and compared the findings with the initial search collection. Finally, a search query is formulated using Boolean "AND" or "OR" operators with selected key search terms and searched in prominent databases: WoS, ACM, ScienceDirect, and IEEE.

After utilizing the selected keywords, a total of 211 papers were gathered between 2008 and 2024 from the selected databases after refining the search query. The search

parameters used for each database are presented in Table 1.

Table 1. Databases with Search Parameters

Digital Source	Criteria
WoS	2008-2024, Journals, Conferences, Proceedings
ACM	2008-2024, Proceedings Paper, Journal Article
ScienceDirect	2008-2024, Conference Papers, Articles
IEEE	2008-2024, Conference Papers, Workshops, Journal Articles

2.3 Select the relevant Studies

After a systematic screening process (presented in Figure 2) that consists of: removal of duplicate studies, selection of relevant studies (firstly, reading of the title, abstract, and keywords, and then from the selected studies, full text screening was performed); 38 studies were selected. Additionally, to cover the grey literature, 8 more papers are selected from the reference lists of selected papers, resulting in a total pool of 46 papers.

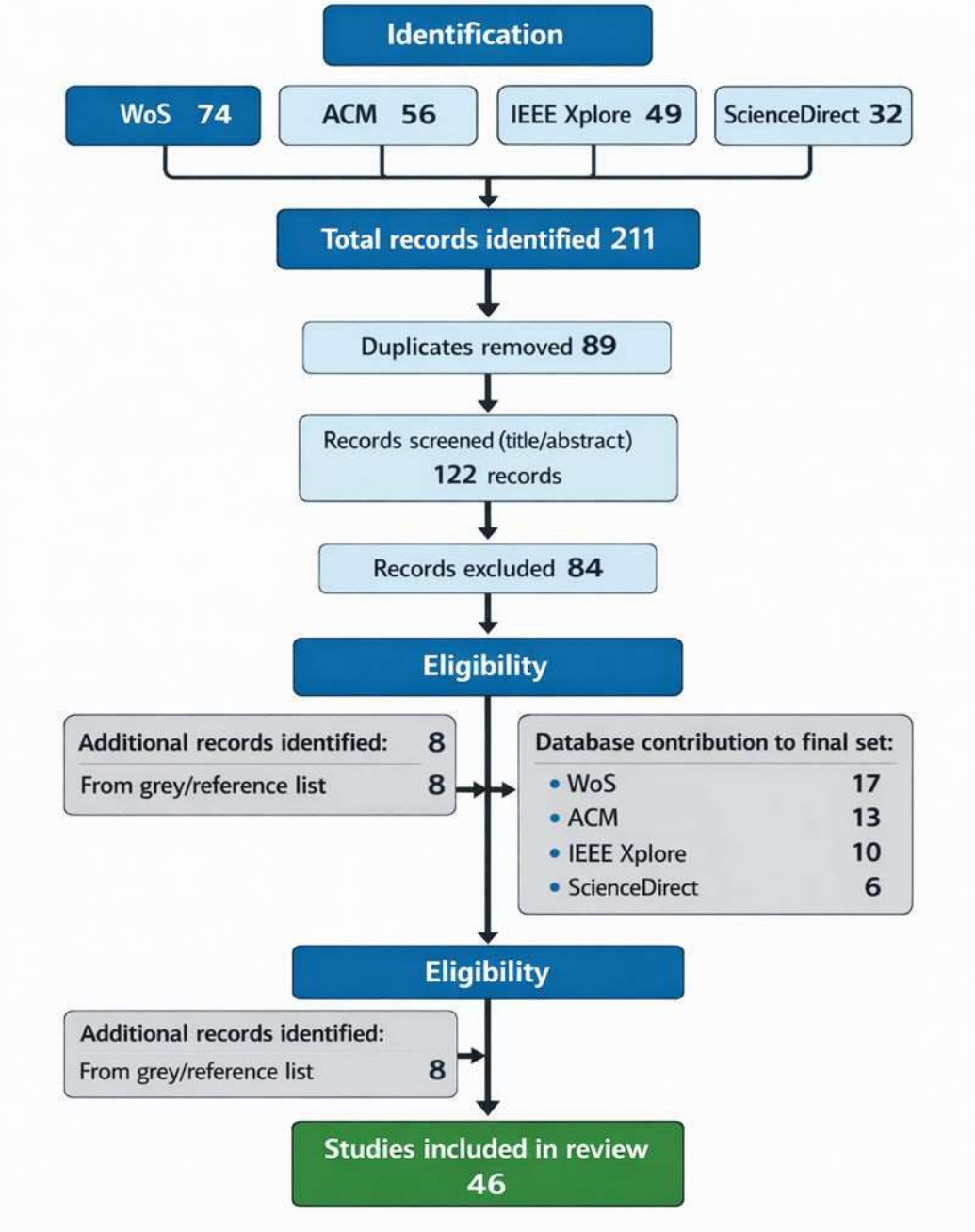

Figure 2: Relevant Studies Selection Procedure

2.4 Extract the required Data

The relevant data extraction mechanism was performed through collaboration among the main and co-authors. After a detailed discussion, the data selection procedure was organized, which was performed by screening the full content, introduction, results, and conclusion of each selected study. The data are extracted from selected studies according to the proposed study objective and research questions. The data refinement process continues till we reach a saturation point. The whole mechanism helps to address our research questions through the

assurance of full details of the articles under analysis.

2.5 Analyze Findings and Result Reporting

2.5.1. Result Analysis

Before finalizing the reporting of selected information, a demographic overview of selected studies is presented in Figure 3, which represents the year-wise distribution of publications over selected databases. Table 2 presents a key observation deduced from the findings. The trend depicted from the Table shows that the research on STC started around the year 2008 and rapidly gained attention in the period of 2009-2013. The highest numbers were analyzed in the years from 2010 to 2012. However, after 2014, there was a gradual decline in publications that indicate the evolution of STC into a well-developed research phase.

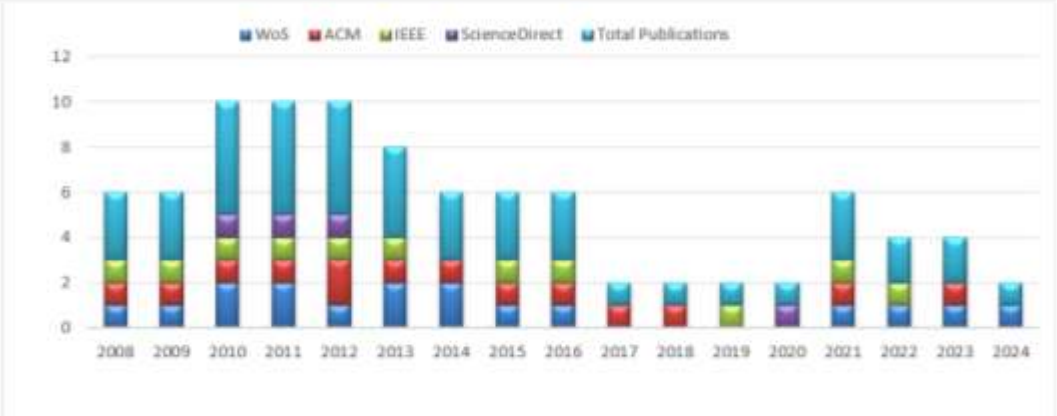

Figure 3: Year-wise Publication distribution over selected Databases

Table 2. Key Observation from Collected Data

Sr#	Key Features	Trends and Database Insights
1	**Emerging phase**	2007–2009
2	**Peak activity Period**	2010–2013
3	**Well-developed phase**	2014 onwards
4	Dominate Databases	**WoS & ACM**
5	Contribution in terms of conference papers	**IEEE**
6	High-impact journal articles	**ScienceDirect**

3. Result Reporting

After a thorough analysis of data collected from a systematic review, the results are reported in this section in terms of answering the proposed research questions. Table 3 presents a mapping of selected studies against each research question that helps to identify the relevance of each selected study to the concerned RQ.

Table 3. Selected Studies and RQs Mapping

Paper Identifier	Paper	RQ1	RQ2	RQ3
S1	Marcelo Cataldo et al.[2009]	X		
S2	A. J. Suali et al.[2017]	X	x	
S3	W. A. W. M. Sobri et al.[2017]			x
S4	Stefanie Betz et al.[2013]	x		
S5	Bonnie K. MacKellar[2013]	x		x
S6	Marcelo Cataldo et al.[2013]	x		x
S7	Weiqiang Zhang et al.[2018]	x		
S8	Irwin Kwan et al.[2011]	x		
S9	Christian Bird et al.[2011]	x		
S10	M.M. Mahbubul Syeed et al.[2013]		x	x
S11	Irwin Kwan et al.[2011]	x		
S12	Subhajit Datta[2017]	x		x
S13	Adrian Kuhn[2010]	x	x	
S14	Sabrina Marczak et al.[2009]	x		
S15	Nicolas Bettenburg [2011]	x		
S16	Darja Šmite et al.[2012]		x	
S17	Xiao Wang et al.[2018]	x		x
S18	Adrian Schröter[2010]			x
S19	Anderson M. de Santana et al.[2013]	x		x
S20	Anh Nguyen Duc et al.[2012]			
S21	McLeod, L.,et al.[2011]			x
S22	Marcelo Cataldo et al.[2008]	x		
S23	M. M. Mahbubul Syeed et al.[2014]			
S24	Terhi Kilamo et al.[2015]	x		
S25	Javier Portillo-Rodríguez et al.[2014]	x		
S26	Marc Palyar et al.[2017]		x	
S27	Sunny Wong[2010]	x		x
S28	Raghvinder S. Sangwan et al.[2008]	x		
S29	Saskia Bick et al.[2017]			x
S30	Mohd Shameem et al.[2015]		x	x
S31	Alexander Scheerer et al.[2014]			
S32	Anh Nguyen-Duc et al.[2013]	x	x	x
S33	Florian Skopik et al.[2012]			
S34	Anita Sarma et al.[2007]	x	x	
S35	Stefano et al. [2022]	x	x	x
S36	Colfer L et al.[2010]	x		
S37	Lopez et al.[2023]	x	x	x
S38	Choudhary et al. [2020]	x	x	x
S39	Irwin Kwan et al.[2009]	x	x	x

S40	D. A. A. Tamburri et al. [2019]	x	x	x
S41	W Zhang et al. [2019]	x		x
S42	M Golzadeh [2019]	x		x
S43	Mauerer et al. [2022]	x	x	x
S44	Di Stefano et al.[2020]	x	x	x
S45	Rajapakse & Szabo[2024]	x		x

3.1. How is STC defined in the Literature?

To cover all studies related to STC, we have also included interconnected studies from the references of selected publications. Numerous pieces of evidence are found in the literature about STC. In selected studies, several have mentioned the definition of STC from the same authors. Later on, each study defines a particular conceptualization of STC through some improvement or modification. Firstly, the term STC is coined from Conway's law (Conway, 1968), which states "any organization which designs a system will inevitably produce a design whose structure is a copy of the organization's communication structure" or which says "the structure of a software product reflects the physical layout of the development organization". The concept of STC in software development is credited to Conway (1968).

On the basis of Conway's law, Cataldo et al. (2006) define STC as" the matching of the communication and task dependency within a software organization." In another study, Cataldo et al. (2008) define STC as "when coordination needs are matched by appropriate coordinating actions, a state defined as socio-technical congruence". With more refinement, Cataldo et al. (2006) reported STC as," socio-technical congruence is defined as the degree of matching between two elements, coordination requirements and actual coordination activities". Whereas Anita et al. (2008) refined the STC concept as "a phenomenon in which an organization has aligned its coordination capabilities to meet the coordination demands of the technical products under development." Colfer & Baldwin (2010) define STC as, "The alignment between the communication structure of a team and the structure of dependencies in its work products has been studied for a long time". Datta (2017) considers Socio-Technical Congruence as a control variable to measure the team performance.

Wang et al. (2018) define Socio-Technical Congruence (STC) as a state that indicates the actual communication links among developers should be congruent with the expected communication needs implied by the technical dependencies among their tasks. Kwan et al. (2011) reported socio-technical congruence as the match between the coordination needs established by the technical domain (i.e., the architectural dependencies in the software) and the actual coordination activities carried out by project members (i.e., within the members of the developer community). Portillo-Rodríguez et al., (2014) have taken the definition given in (Bird et al., 2011; Kwan et al., 2009), where it is defined as ''an intuitive way to compare required coordination effort within a software development project with the actual ongoing coordination'' or ''the fit between an organization's coordination requirements and an organization 's social interactions''.

3.2. What are the main features of STC?

After the literature survey, we have found the following properties of STC:

(1) STC is a scale for organization stability. It describes the performance of an organization at any moment. The scale can measure how much tasks and teams are congruent with each other (Lopez et al., 2023). (2) STC is multi-dimensional: Various ways exist to measure coordinated work. Congruence is not just the coordination among the developers. It is just a one dimension of congruency, but some additional concepts also exist, such as comments in code, work communication patterns, expertise, awareness, and resource, which play important roles in congruence measurement (Sierra et al., 2018). (3) STC is a key index: It is used to indicate the relative amount of coordination among a team (Kwan & Damian, 2011). Congruence is dynamic, as communication and social structures and task dependencies, and because within an organization, may change over time (Valetto et al., 2008).

Lastly, (4) STC is a quantitative technique to analyze socio-technical concerns in SW development (Portillo-Rodríguez et al., 2014). It is a starting point for large-scale change that meets the test of any successful model (Mauerer et al., 2022). Congruence can be beneficial and harmful at the same time. There are many dimensions or levels that exist in an organization. It is possible that at one level, congruence is beneficial for the team, but at the same time, it creates incongruence at other levels of the organization. That shows the need for congruence coverage on all dimensions or levels of an organization (Sierra et al., 2018; Portillo-Rodríguez et al., 2014). Table 4 depicts the summary of identified STC features.

Table 4: STC prominent Features

Id	Features	Application
Mauerer at al., 2022	Scale for organization stability	Can measure organization performance
Sierra at el., 2018	Multi-dimensional	Can measure congruence among developers coordination as well as in code comments, task communication pattern, expertise, awareness and resource
Kwan & Damian, 2011	Key index	Can indicate the relative amount of coordination among a team
Valetto at el., 2008	Dynamic	Can measure communication among social structures and task dependencies that may change over time.
Portillo-Rodríguez et al., 2014	Quantitative technique	Can analyze socio technical concerns in Software development.

3.3. What are the benefits and issues of STC

Cataldo et al. (2006) reported that congruence decreases the task completion time, which validates the higher productivity. The research in (Ehrlich et al., 2008) presents an effective way to measure STC that can identify gaps and measures to overcome these gaps that cause a negative effect on work productivity.

With the help of STC measurement, a manager can manage the social coordination among his team members with their technical dependencies (Kwan & Damian, 2009). It can also act as a scale to indicate the level of missing coordination and can prioritize gaps according to their severity.

STC can provide the metrics for collaboration tool designing. A collaboration tool can help the software development team to coordinate, monitor, control, and improve the flow of information. They also indicate the need for communication among the team members. To design an effective collaboration tool, STC can help by identifying the need for communication and its measure of fulfilment (Cataldo et al., 2006).

After the literature survey, we identified several studies that show the effectiveness of STC measurement. Cataldo et al. (2006; Cataldo et al., 2008; Cataldo & Herbsleb, 2013; Bolici et al.,2009) suggested that the higher level of congruence can introduce decreased resolution time for change requests within the code. The reduced resolution time indicates the high performance of the development team. A high level of STC ensures the achievement of coordination needs. However, a lack of sufficient STC level causes a delay in task completion, leading to project failure. The evidence of this hypothesis is supported by a case study reported in (Smite & Galvina, 2012), where a lack of sufficient STC measurement was detected through some appropriate tools. The results depicted the insufficient coordination among team members, due to a hidden part of the organization's structure, which caused the project to fall short.

A multidimensional congruence approach is presented in Li et al. (2012). The technique introduces the awareness concept in STC by introducing ways that help team members identify task dependencies, knowledge, and expertise of other members.

Moreover, several studies (Rajapakse & Szabo et al., 2024; Kwan & Damian, 2009; Cataldo et al., 2006; Kwan et al., 2011; Ehrlich et al., 2008; Cataldo et al., 2008; Gokpinar et al., 2010) reported that a high level of congruence introduces an improved level of coordination in an organization, which results in an improvement in outcomes such as better team performance, high software quality, and reduction in development cost.

On the other hand, STC measurement can introduce some risks and an augmentation in cost. To close the coordination gap, we might introduce ways of communication and collaboration that overload the information among the team members. The excess information causes a burden on personnel at the individual level and results in a reduction in productivity (Choudhary et al., 2020; Valetto et al., 2008).

Xiao Wang et al. (2018) analyze a risk associated with overwhelming STC, which they named Transgressive Incongruence (TraIn), which causes a negative impact on the software quality. To investigate the term TraIn, a term communication coupling is used, which represents the phenomenon of excessive communication traffic among

developers. To identify TraIn, the proposed research investigates six Apache projects to present overwhelming communication between files that are not matched by any dependencies. After evaluation on six projects, the results show that the files that are part of TraIn are prone to bugs. A summarized form of STC benefits and issues is represented in Table 5.

Table 5: Summary of STC Benefits and Issues

Id	Benefits and Issues
	Benefits
Cataldo et al., 2006; Ehrlich et al., 2008	• Decreases the task completion time • Validate higher productivity
Li et al., 2012 Cataldo et al., 2006 Cataldo et al., 2006, Cataldo et al., 2008; Cataldo & Herbsleb, 2013; Bolici et al.,2009 Li et al., 2012	• Used as a management tool • An effective collaboration tool • Decreases resolution time for change request Validate high development performance • Help team member to identify task dependencies • Introduce awareness among team members.
Rajapakse & Szabo et al., 2024, Cataldo, et al., 2006; Kwan et al., 2011; Ehrlich et al., 2008 ; Cataldo et al., 2008; Gokpinar et al., 2010	• Better team performance, • High software quality • Reduction in development cost
	Issues
Choudhary et al., 2020; Valetto et al., 2008 Wang et al., 2018	• Burden on personal at individual level • Excessive amount can reduce productivity • Transgressive Incongruence (TraIn) • Excessive communication traffic among developers causes negative impact on the software quality

4. Discussion

4.1. Principal Findings

The detailed analysis of our research questions has identified several contexts, such as numerous studies that define STC with respect to their conceptualization on the basis of Conway's Law. However, the first operational STC model was presented by Cataldo et al. When we investigate studies to answer our second question, we have found STC has several properties, one of which is commonly defined by all studies as a quantitative technique to analyze socio and technical concerns of a project. From the practitioner's point of view, we identify that STC measurement can help managers to view and control the team members' activities that

can enhance team performance and software quality.

4.2. Future implications

After analyzing the selected studies, the following research directions have been identified: several studies explain the benefits and importance of STC, but only a few identify the risks associated with STC. As we have observed, only one study mainly discusses the risk of overwhelming STC, so this area of research needs further attention.

By the analysis of existing STC measuring techniques, we have found that there are numerous studies identified that set STC measurement for the development phase of the project life cycle. Only a few of the techniques have been discussed in other related phases, such as the requirement or testing phase, or for the whole project life cycle. As STC is a multi-dimensional technique that can cover all levels of the project life cycle, this area of study needs more research.

5. Conclusion

In this study, a systematic review of the literature is conducted on the aspects and evolution of new trends in STC. More specifically, the main aim of the presented review was to understand the concept and efficiency of the STC mechanism. Additionally, we aim to understand the impact of STC on the distributed team performance and the development of software quality.

In this paper, STC has been defined as a technique to measure the congruence among social and technical aspects of a software project. We have also analyzed that STC can identify the coordination needs among the team members and also analyze the communication that occurs among team members.

We have identified 46 studies relevant to three research questions from four large data sources. After a detailed analysis of selected studies, we have identified a relationship between the social and technical aspects of software development in terms of STC. Moreover, with detailed analysis, we have summarized the findings of each study in the form of our research question mapping. Findings reveal that STC is the fit among the social and technical aspects of an organization that can be measured through different techniques and tools, and each has some benefits or disadvantages. The analysis outcome follows areas for future suggestions: i) risks associated with high and low levels of STC, ii) tools that can identify the congruence gap, as well as provide an appropriate solution to overcome the identified gap, and iii) STC techniques that can cover the whole project life cycle.

References:-

1. Abbasi, F. A., Burdi, A., Khan, R. S., & Naqvi, S. H. F. (2019). Requirement Engineering Challenges in Distributed Software Development. Sindh University Research Journal -Science Series. 51. 465-. 10.26692/surj/2019.09.74.
2. Baird, A., & Riggins, F. J. (2012). Planning and Sprinting: Use of a Hybrid Project Management Methodology within a CIS Capstone Course. *Journal of Information Systems Education*, *23*(3), 243-258.
3. Bernardi, M. L., Canfora, G., Di Lucca, G. A., Di Penta, M., & Distante, D. (2012). *Do developers introduce bugs when they do not communicate? The case of Eclipse and Mozilla* 16th European Conference on Software Maintenance and Reengineering (CSMR), Szeged, Hungary.
4. Bettenburg, N. (2011). *Mining Development Repositories to Study the Impact of Collaboration on Software Systems* Proceedings of the 9th ACM SIGSOFT International Symposium on the Foundations of Software Engineering (ESEC/FSE), Szeged, Hungary.
5. Waqas M, Ahmed SU, Tahir MA, Wu J, Qureshi R. Exploring multiple instance learning (MIL): A brief survey. Expert Systems with Applications. 2024 Sep 15;250:123893.
6. Khan, M. A., Khan, S. U. R., Rehman, H. U., Aladhadh, S., & Lin, D. (2025). Robust InceptionV3 with Novel EYENET Weights for Di-EYENET Ocular Surface Imaging Dataset: Integrating Chain Foraging and Cyclone Aging Techniques. International Journal of Computational Intelligence Systems, 18(1), 204.
7. Betz, S., mite, D., Fricker, S., Moss, A., Afzal, W., Svahnberg, M., Wohlin, C., Börstler, J., & Gorschek, T. (2013, 9-9 Oct. 2013). An Evolutionary Perspective on Socio-Technical Congruence: The Rubber Band Effect. 2013 3rd International Workshop on Replication in Empirical Software Engineering Research
8. Bird, C., Murphy, B., Nagappan, N. & Zimmermann, T. (2011). *Empirical software engineering at microsoft research*. Proceedings of the ACM 2011 conference on Computer supported cooperative work, ACM, 143-150.
9. Waqas M, Bandyopadhyay R, Showkatian E, Muneer A, Zafar A, Alvarez FR, Marin MC, Li W, Jaffray D, Haymaker C, Heymach J. The Next Layer: Augmenting Foundation Models with Structure-Preserving and Attention-Guided Learning for Local Patches to Global Context Awareness in Computational Pathology. arXiv preprint arXiv:2508.19914. 2025 Aug 27.
10. Bolici, F., Howison, J., & Crowston, K. (2009). *Coordination without discussion? Socio-technical congruence and stigmergy in free and open source software projects* Proceedings of the 31st International Conference on Software Engineering (ICSE), Vancouver, Canada.
11. Calefato, F., Lanubile, F. & Novielli, N. (2017).A preliminary analysis on the effects of propensity to trust in distributed

software development. Global Software Engineering (ICGSE), 2017 IEEE 12th International Conference on, IEEE, 56-60.

12. Cataldo, M. & Herbsleb, J. D. (2008).Communication networks in geographically distributed software development. Proceedings of the 2008 ACM conference on Computer supported cooperative work. San Diego, CA, USA: ACM.
13. Khan, S. U. R., Asif, S., Zhao, M., Zou, W., Li, Y., & Li, X. (2025). Optimized deep learning model for comprehensive medical image analysis across multiple modalities. Neurocomputing, 619, 129182.
14. Waqas M, Tahir MA, Khan SA. Robust bag classification approach for multi-instance learning via subspace fuzzy clustering. Expert Systems with Applications. 2023 Mar 15;214:119113.
15. Cataldo, M., & Herbsleb, J. D. (2013). Coordination Breakdowns and Their Impact on Development Productivity and Software Failures. *IEEE Transactions on Software Engineering*, *39*(3), 343-360. https://doi.org/10.1109/TSE.2012.32
16. Khan, S. U. R., Asif, S., Zhao, M., Zou, W., & Li, Y. (2025). Optimize brain tumor multiclass classification with manta ray foraging and improved residual block techniques. Multimedia Systems, 31(1), 1-27.
17. Asif Raza, Salahuddin, Ghazanfar Ali, Muhammad Hanif Soomro, Saima Batool, "Analyzing the Impact of Artificial Intelligence on Shaping Consumer Demand in E-Commerce: A Critical Review", International Journal of Information Engineering and Electronic Business(IJIEEB), Vol.17, No.5, pp. 42-61, 2025.
18. Cataldo, M., Herbsleb, J. D., & Carley, K. M. (2008). *Socio-technical congruence: A framework for assessing the impact of technical and work dependencies on software development productivity* Proceedings of the Second ACM-IEEE International Symposium on Empirical Software Engineering and Measurement, Kaiserslautern, Germany.
19. Cataldo, M., Mockus, A., Roberts, J. A., & Herbsleb, J. D. (2009). Software Dependencies, Work Dependencies, and Their Impact on Failures. *IEEE Transactions on Software Engineering*, *35*(6), 864-878.
20. Cataldo, M., Wagstrom, P. A., Herbsleb, J. D., & Carley, K. M. (2006). *Identification of Coordination Requirements: Implications for the design of collaboration and awareness tools* Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW), Banff, Alberta, Canada.
21. Khan, S. U. R., Asif, S., & Bilal, O. (2025). Ensemble Architecture of Vision Transformer and CNNs for Breast Cancer Tumor Detection From Mammograms. International Journal of Imaging Systems and Technology, 35(3), e70090.
22. Choudhary, S., Bogart, C., Rose, C., & Herbsleb, J. (2020). Using Productive Collaboration Bursts to Analyze Open Source Collaboration Effectiveness. IEEE International Conference on Software Analysis, Evolution and

Reengineering (SANER), London, Canada.

23. Khan, S. U. R. (2025). Multi-level feature fusion network for kidney disease detection. Computers in Biology and Medicine, 191, 110214.
24. Waqas M, Tahir MA, Qureshi R. Deep Gaussian mixture model based instance relevance estimation for multiple instance learning applications. Applied intelligence. 2023 May;53(9):10310-25.
25. Bilal, Omair, Arash Hekmat, Inzamam Shahzad, Asif Raza, and Saif Ur Rehman Khan. "Boosting Machine Learning Accuracy for Cardiac Disease Prediction: The Role of Advanced Feature Engineering and Model Optimization." The Review of Socionetwork Strategies (2025): 1-30.
26. Colfer, L., & Baldwin, C. Y. (2010). *The mirroring hypothesis: theory, evidence and exceptions* (SSRN Scholarly Paper ID 1539592).
27. Bilal, O., Hekmat, A., & Khan, S. U. R. (2025). Automated cervical cancer cell diagnosis via grid search-optimized multi-CNN ensemble networks. Network Modeling Analysis in Health Informatics and Bioinformatics, 14(1), 67.
28. Conway, M. E. (1968). How do committees invent? *Datamation*, *14*, 28-31.
29. Khan, S. U. R., Asim, M. N., Vollmer, S., & Dengel, A. (2025). Robust & precise knowledge distillation-based novel context-aware predictor for disease detection in brain and gastrointestinal. arXiv preprint arXiv:2505.06381.
30. Raza, Asif, Inzamam Shahzad, Muhammad Salahuddin, and Sadia Latif. "Satellite Imagery Employed to Analyze the Extent of Urban Land Transformation in The Punjab District of Pakistan." Journal of Palestine Ahliya University for Research and Studies 4, no. 2 (2025): 17-36.
31. Cruzes, D. S., & Dybå, T. (2011). Research synthesis in software engineering: A tertiary study. *Information and Software Technology*, *53*(5), 440-455.
32. Yang, H., Khan, S. U. R., Bilal, O., Chen, C., & Zhao, M. (2025). CEOE-Net: Chaotic Evolution Algorithm-Based Optimized Ensemble Framework Enhanced with Dual-Attention for Alzheimer's Diagnosis. Computer Modeling in Engineering & Sciences, 145(2), 2401.
33. Waqas M, Tahir MA, Al-Maadeed S, Bouridane A, Wu J. Simultaneous instance pooling and bag representation selection approach for multiple-instance learning (MIL) using vision transformer. Neural Computing and Applications. 2024 Apr;36(12):6659-80.
34. Datta, S. (2017). How does developer interaction relate to software quality? An examination of product development data. *Empirical Software Engineering*.
35. S. Ur Rehman Khan, O. Bilal, S. Mistry, N. Deb, M. Mahmud and M. Bhuyan, "KDLight: A Lightweight Knowledge Distillation Framework for Medical Image Classification," 2025 International Joint Conference on Neural Networks (IJCNN), Rome, Italy, 2025, pp. 1-8.
36. de Santana, A. M., da Silva, F. Q. B., de Miranda, R. C. G., Mascaro, A. A., Gouveia, T. B., Monteiro, C. V. F., &

Santos, A. L. M. (2013). *Relationships between communication structure and software architecture: An empirical investigation of Conway's Law at the Federal University of Pernambuco* Proceedings of the 3rd International Workshop on Replication in Empirical Software Engineering Research,

37. De Stefano, M., Iannone, E., Pecorelli, F., & Tamburri, D. A. (2022). Impacts of software community patterns on process and product: An empirical study. *Science of Computer Programming*, *214*, 102731. https://doi.org/https://doi.org/10.1016/j.scico.2021.102731
38. Khan, Saif Ur Rehman, Asif Raza, Inzamam Shahzad, and Shehzad Khan. "Subcellular Structures Classification in Fluorescence Microscopic Images." International Conference on Computing & Emerging Technologies, pp. 271-286. Cham: Springer Nature Switzerland, 2023.
39. Di Stefano, M., Pecorelli, F., Tamburri, D. A., Palomba, F., & De Lucia, A. (2020). Refactoring Recommendations Based on the Optimization of Socio-Technical Congruence. IEEE International Conference on Software Maintenance and Evolution (ICSME), Adelaide, Australia.
40. Khan, S. U. R., Rehman, H. U., & Bilal, O. (2025). AI-powered cancer diagnosis: classifying viable (live) vs non-viable (dead) cells using transfer learning. Signal, Image and Video Processing, 19(15), 1326.
41. Waqas M, Tahir MA, Qureshi R. Ensemble-based instance relevance estimation in multiple-instance learning. In2021 9th European workshop on visual information processing (EUVIP) 2021 Jun 23 (pp. 1-6). IEEE.
42. O. Bilal, S. Ur Rehman Khan, S. Mistry, N. Deb, M. Mahmud and M. Bhuyan, "Towards Efficient Pruning and Multi-Scale Feature Transformations to Uncover Medical Diseases," 2025 International Joint Conference on Neural Networks (IJCNN), Rome, Italy, 2025, pp. 1-8.
43. Asif Raza, Inzamam Shahzad, Ghazanfar Ali, and Muhammad Hanif Soomro. "Use Transfer Learning VGG16, Inception, and Reset50 to Classify IoT Challenge in Security Domain via Dataset Bench Mark." Journal of Innovative Computing and Emerging Technologies 5, no. 1 (2025).
44. Ehrlich, K., Helander, M., Valetto, G., Davies, S., & Williams, C. (2008). *An analysis of congruence gaps and their effect on distributed software development* Proceedings of the Socio-Technical Congruence Workshop at the International Conference on Software Engineering,
45. Khan, U. S., & Khan, S. U. R. (2025). Ethics by Design: A Lifecycle Framework for Trustworthy AI in Medical Imaging From Transparent Data Governance to Clinically Validated Deployment. arXiv preprint arXiv:2507.04249.
46. Waqas M, Khan Z, Anjum S, Tahir MA. Lung-Wise Tuberculosis Analysis and Automatic CT Report Generation with Hybrid Feature and Ensemble Learning. In CLEF (Working notes) 2020 Sep (pp. 1-10).

47. Gokpinar, B., Hopp, W. J., & Iravani, S. M. R. (2010). The impact of misalignment of organizational structure and product architecture on quality in complex product development. *Management Science*, *56*(3), 468-484.
48. Khan, S. U. R., Asif, S., Zhao, M., Zou, W., Li, Y., & Xiao, C. (2026). ShallowMRI: A novel lightweight CNN with novel attention mechanism for Multi brain tumor classification in MRI images. Biomedical Signal Processing and Control, 111, 108425.
49. Kitchenham, B., & Charters, S. (2007). *Guidelines for performing systematic literature reviews in software engineering*.
50. Khan, Zia, Saif Ur Rehman Khan, Omair Bilal, Asif Raza, and Ghazanfar Ali. "Optimizing Cervical Lesion Detection Using Deep Learning with Particle Swarm Optimization." In 2025 6th International Conference on Advancements in Computational Sciences (ICACS), pp. 1-7. IEEE, 2025
51. Kwan, I., & Damian, D. (2011). Extending Socio-Technical Congruence with Awareness Relationships. Proceedings of the Fourth International Workshop on Social Software Engineering, Szeged, Hungary.
52. S. U. R. Khan, A. Raza, I. Shahzad and G. Ali, "Enhancing Concrete and Pavement Crack Prediction through Hierarchical Feature Integration with VGG16 and Triple Classifier Ensemble," 2024 Horizons of Information Technology and Engineering (HITE), Lahore, Pakistan, 2024, pp. 1-6.
53. Kwan, I., Schroter, A., & Damian, D. (2011). Does socio-technical congruence have an effect on software build success? A study of coordination in a software project. *IEEE Transactions on Software Engineering*, *37*(3), 307-324.
54. Al-Khasawneh, Mahmoud Ahmad, Asif Raza, Saif Ur Rehman Khan, and Zia Khan. "Stock Market Trend Prediction Using Deep Learning Approach." Computational Economics (2024): 1-32
55. Kwan, I., Schröter, A., & Damian, D. (2009). A Weighted Congruence Measure. Workshop on Socio-Technical Congruence,
56. Li, J., Carley, K. M., & Eberlein, A. (2012). Assessing Team Performance from a Socio-Technical Congruence Perspective. International Conference on Software and System Process (ICSSP),
57. Lopez, T., Sharp, H., Wermelinger, M., Langer, M., Levine, M., Jay, C., Yu, Y., & Nuseibeh, B. (2023). Accounting for Socio-Technical Resilience in Software Engineering. IEEE/ACM International Conference on Cooperative and Human Aspects of Software Engineering (CHASE),
58. Raza, Asif , Soomro, M. H., Shahzad, I., & Batool, S. (2024). Abstractive Text Summarization for Urdu Language. Journal of Computing & Biomedical Informatics, 7(02).
59. Maheshwari, M., & Kumar, U. (2012). Alignment between social and technical capability in software development teams: An empirical study. *International Journal of Project Management*.
60. Khan, S. R., Asif Raza, Inzamam Shahzad, & Hafiz Muhammad Ijaz.

(2024). Deep transfer CNNs models performance evaluation using unbalanced histopathological breast cancer dataset. Lahore Garrison University Research Journal of Computer Science and Information Technology, 8(1).

61. Marczak, S., Kwan, I., & Damian, D. (2009). *Investigating collaboration driven by requirements in cross-functional software teams* Collaboration and Intercultural Issues on Requirements: Communication, Understanding and Soft skills (CIRCUS),
62. M. Wajid, M. K. Abid, A. Asif Raza, M. Haroon, and A. Q. Mudasar, "Flood Prediction System Using IOT & Artificial Neural Network", VFAST trans. softw. eng., vol. 12, no. 1, pp. 210–224, Mar. 2024.
63. Khan, S. U. R., & Khan, Z. (2025). Detection of Abnormal Cardiac Rhythms Using Feature Fusion Technique with Heart Sound Spectrograms. Journal of Bionic Engineering, 1-20.
64. Mauerer, W., Joblin, M., Tamburri, D. A., Paradis, C., Kazman, R., & Apel, S. (2022). In Search of Socio-Technical Congruence: A Large-Scale Longitudinal Study. *IEEE Transactions on Software Engineering*, *48*(8), 3159-3184. https://doi.org/10.1109/TSE.2021.3082074
65. O. Bilal, Asif Raza, S. ur R. Khan, and Ghazanfar Ali, "A Contemporary Secure Microservices Discovery Architecture with Service Tags for Smart City Infrastructures ", VFAST trans. softw. eng., vol. 12, no. 1, pp. 79–92, Mar. 2024
66. Nguyen, T., Wolf, T. & Damian, D. (2008). Global software development and delay: Does distance still matter? Global Software Engineering, 2008. ICGSE 2008. IEEE International Conference on, 2008. IEEE, 45-54.
67. Omoronyia, I., Ferguson, J., Roper, M., & Wood, M. (2010). A review of awareness in distributed collaborative software engineering. *Software: Practice and Experience*, *40*(12), 1107-1133. https://doi.org/10.1002/spe.1005
68. S. ur R. Khan, Asif. Raza, Muhammad Tanveer Meeran, and U. Bilhaj, "Enhancing Breast Cancer Detection through Thermal Imaging and Customized 2D CNN Classifiers", VFAST trans. softw. eng., vol. 11, no. 4, pp. 80–92, Dec. 2023.Rajapakse, R. N., & Szabo, C. (2024). Towards Multi-Class Socio-Technical Congruence: Assessing Coordination in Collaborative Software Development Settings.
69. HUSSAIN, S., Raza, A., MEERAN, M. T., IJAZ, H. M., & JAMALI, S. (2020). Domain Ontology Based Similarity and Analysis in Higher Education. IEEEP New Horizons Journal, 102(1), 11-16.
70. Mahmood, F., Abbas, K., Raza, A., Khan,M.A., & Khan, P.W. (2019). Three Dimensional Agricultural Land Modeling using Unmanned Aerial System (UAS). International Journal of Advanced Computer Science and Applications (IJACSA) [p-ISSN : 2158-107X, e-ISSN : 2156-5570], 10(1).
71. Sierra, J. M., Vizcaíno, A., Genero, M., & Piattini, M. (2018). A Systematic Mapping Study About Socio-Technical Congruence. *Information and Software Technology*, *94*, 111-129.

72. Raza, A., & Meeran, M. T. (2019). Routine of Encryption in Cognitive Radio Network. Mehran University Research Journal of Engineering and Technology [p-ISSN: 0254-7821, e-ISSN: 2413-7219], 38(3), 609-618.
73. Sosa, M. E., Eppinger, S. D., & Rowles, C. M. (2004). The misalignment of product architecture and organizational structure in complex product development. *Management Science*, *50*(12), 1674-1689.
74. Tahir, M., Khan, F., Babar, M., & Arif, F. (2016). Framework for Better Reusability in Component Based Software Engineering. *Journal of Applied Environmental and Biological Sciences*, *6*(4S), 77-81.
75. Meeran, M. T., Raza, A., & Din, M. (2018). Advancement in GSM Network to Access Cloud Services. Pakistan Journal of Engineering, Technology & Science [ISSN: 2224-2333], 7(1).
76. Valetto, G., Chulani, S., & Williams, C. (2008). *Balancing the value and risk of socio-technical congruence* Proceedings of the 30th International Conference on Software Engineering (ICSE), Leipzig, Germany.
77. Valetto, G., Helander, M., Ehrlich, K., Chulani, S., Wegman, M., & Williams, C. (2007). Using Software Repositories to Investigate Socio-Technical Congruence in Development Projects. Proceedings of the 29th International Conference on Software Engineering Workshops,
78. Wang, X., Xiao, L., Yang, Y., Xu, X., & Jiang, Y. (2018). *Poster: Identifying TraIn: A Neglected Form of Socio-Technical Incongruence* ACM/IEEE 40th International Conference on Software Engineering: Companion Proceedings (ICSE),
79. Zhang, F., Khomh, F., Zou, Y., & Hassan, A. E. (2014). An empirical study of the effect of file editing patterns on software quality. *Journal of Software: Evolution and Process*, *26*(11), 996-1029.