June 2024

Model-Based Systems Engineering for Engineering Education Systems Simulation

Pallavi Singh
University of South Florida

Model-Based Systems Engineering for Engineering Education Systems Simulation

by

Pallavi Singh

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
Department of Electrical Engineering
College of Engineering
University of South Florida

Major Professor: Wilfrido A. Moreno, Ph.D.
Ismail Uysal, Ph.D.
Chung Seop Jeong, Ph.D.
Autar Kaw, Ph.D.
Dhinesh Balaji Radhakrishnan, Ph.D.

Date of Approval:
June 21, 2024

Keywords: Multi-criteria Team Formation, Synthetic data, Bayesian network, Education Data Mining

Dedication

I would like to thank the Lord God Almighty for enabling me to complete my Ph.D. journey. I dedicate this dissertation to my father, Sunil Kumar, who has always been zealous about education and always inspired us about the importance of education in life. To my mom, Mira Kumari, for her understanding, support, and patience throughout these years.

To friends Prerna Patil, Akash Mashalkar, Rudens Mara, Natheer Nimrat, Carleen and Dan Murphy, Jennifer, Ale Myrna, and Toy Coleman for their love, food, coffee, and heartwarming conversation.

Acknowledgments

I am deeply thankful to my major advisor, Dr. Wilfrido Moreno, for inspiring, coaching, assisting, continuously guiding, and supporting me throughout this journey. I am grateful for the time and support that he has invested in me.

I would like to thank my committee members, Dr. Ismail Uysal and Dr. Chung Seop Jeong, for the courses they taught and their invaluable feedback during the research proposal and candidacy preparation stages. Their contributions have been instrumental. Additionally, I extend my thanks to Dr. Autar Kaw and Dr. Dhinesh Balaji Radhakrishnan for serving as valuable references in engineering education research during the early stages of my study. Their constructive criticism and guidance have significantly shaped my dissertation research, directing it toward novelty.

I would like to thank Dr. Alexandro Castellanos for his wisdom, energy, and unwavering commitment to education, especially for women engineers. His influence has been invaluable, setting an example of how to find joy in one's work.

I am very thankful to the Electrical Engineering Department at USF for providing the financial support for me to complete the graduate program. I would also like to thank Dr. Nancy Diaz for allowing me to collaborate with her on an engineering education project. It has been a great opportunity for my growth and learning.

Lastly, I would like to thank Dr. Grisselle Centeno for helping me structure my dissertation document. I am also grateful to my colleague, Liliana Villavicencio Lopez, for her continuous guidance and support over many years of my career. Liliana has been a much-needed colleague and friend, a constant source of motivation and encouragement, and has always celebrated my achievements. Additionally, I want to thank Shraddha Pandey, who has been a dedicated mentor, providing valuable guidance throughout my Ph.D. journey.

Table of Contents

List of Tables

List of Figures

Abstract

As the engineering education system continuously evolves to meet the demands of modern industry and society, there is a need for a methodology that would manage and resolve the complexities inherent in engineering educational systems. Model-based Systems Engineering (MBSE) is a structured approach to system design that utilizes models across all stages of the system's life cycle and supports requirements management, design, analysis, verification, and validation processes. While MBSE has been applied successfully in industries like defense, aerospace, and automotive, its application in engineering educational systems remains unexplored. This dissertation develops a dynamic model of the university-level engineering education system using an MBSE framework, which is named "Engineering Learning Analytic Systems (ELAS)". ELAS is a human-centered entity involving students, educators, administrators, and industry partners. It aims to improve student learning and performance across all levels by focusing on enhancing professional competencies like communication and teamwork. This is achieved through the development of a multi-criteria team formation algorithm for grouping students in capstone projects and engineering courses. The research also addresses challenges in model simulation and evaluation due to the lack of datasets available for engineering education system research, which is addressed by developing a generative synthetic data model using Bayesian network and Gibb sampling. The research concludes by highlighting the transformative potential of Model-Based Systems Engineering (MBSE) in engineering education, illustrating the necessity of integrating 21st-century tools such as MBSE, System Simulation, Artificial Intelligence, and Machine Learning to address 21st-century challenges. It demonstrates how systems engineering tools and frameworks can facilitate a more adaptable, efficient, and student-centered approach to learning.

Chapter 1: Introduction

Engineering education plays a crucial role in technological and industrial advancement, operating as a dynamic system that adapts to the needs of societal demands. Within this system, key players such as students, teachers, administrators, and industry partners engage in interactions that form a complex network, significantly shaping the educational framework. This intricate web of relationships supports the entire educational ecosystem, driving its functionality and development.

The complexity of engineering education arises from nonlinear relationships among its variables, necessitating multi-level analysis [26]. Classroom learning, often studied as a complex system [46][45], involves various factors, from curriculum design and delivery to students' knowledge assimilation—all interacting in intricate ways. Figure 1.1 illustrates the interconnected factors influencing engineering education. These factors include policymakers and society, subject matter experts, external interfaces, teaching and learning styles, knowledge mapping, industry-university collaboration, and the development of 21st-century skills.

Policymakers and society play a critical role in shaping the educational landscape by establishing regulations, standards, and expectations to which institutions must adhere. These policies can influence the availability of resources, the design of curricula, and the overall direction of educational initiatives, thereby affecting the quality and accessibility of engineering education.

Subject matter educational experts, including accrediting bodies like ABET, universities, state agencies, and federal agencies, contribute to the formulation and dissemination of educational standards and best practices. Their expertise ensures that the engineering curriculum remains relevant, rigorous, and aligned with industry needs. These experts also support faculty development and promote the adoption of effective teaching methodologies.

External interfaces encompass a range of social, economic, financial, and health support resources that students require to succeed. These factors can significantly influence students' ability to focus on their studies, access necessary learning materials, and maintain their overall well-being. Addressing these external factors is crucial for creating an equitable and supportive learning environment.

Teaching and learning styles are fundamental to the educational process. The interaction between human teachers and students, along with the pedagogical approaches employed, can greatly impact student engagement and learning outcomes. Effective teaching strategies, such as active learning and project-based learning, have been shown to enhance student understanding and retention of complex engineering concepts [71].

Knowledge mapping involves the systematic tracking of student learning progress and retention validation. This process helps educators identify areas where students may struggle and adjust instructional methods accordingly [46], by continually assessing and validating

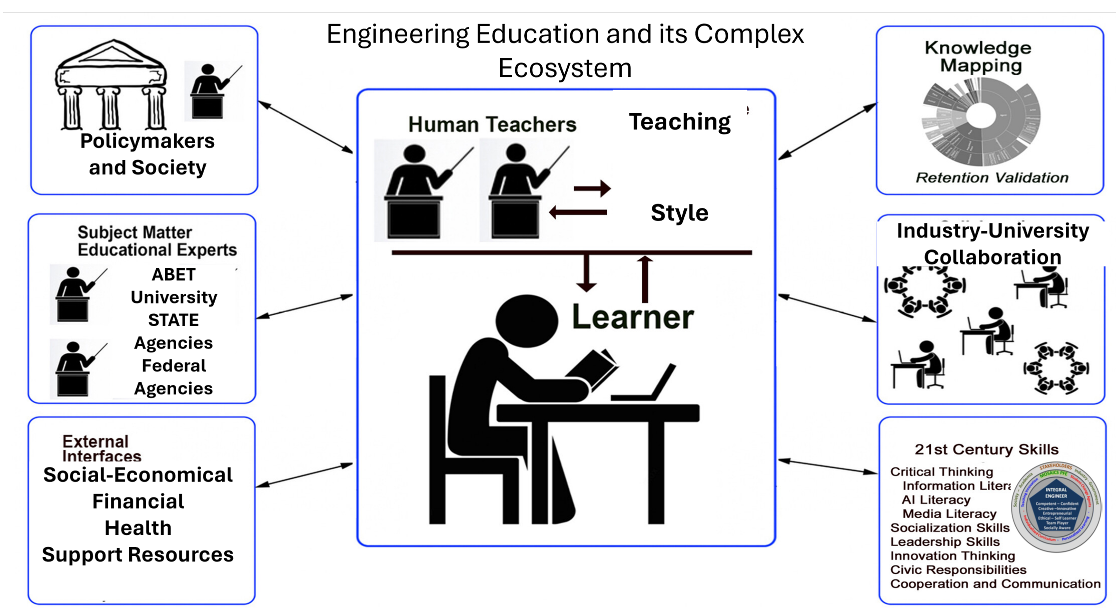


Figure 1.1 Engineering Education and its Complex Ecosystem

student knowledge, educators can ensure that learning objectives are being met and that students are prepared for professional challenges.

Industry-university collaboration is another critical component of the engineering education ecosystem. Partnerships between educational institutions and industry provide students with real-world experiences such as internships, co-ops, capstone projects, and exposure to current industry practices [46]. These collaborations help bridge the gap between theoretical knowledge and practical application, preparing students for successful careers in engineering.

Lastly, the development of 21st-century skills, such as critical thinking, information literacy, AI literacy, media literacy, socialization skills, leadership skills, innovation thinking, civic responsibilities, and communication skills, is essential for engineering students. These skills are necessary for navigating the complexities of the modern workforce and contributing to societal advancement.

Given the multi-level nature of this complex system, an effective methodology is required to navigate and understand it. This methodology must enable analysis at various levels, comprehending the relationships and dependencies, and ultimately, formulating strategies to enhance the effectiveness of Engineering Education. Therefore, the complexity of engineering education should not be viewed as a challenge to be overcome, but rather as a reality to be understood and leveraged for the betterment of all stakeholders involved, such as educational institutions, state governments, and ABET accreditation agencies.

1.1 The Need for Evolving Engineering Educational Methodologies

Engineering education research has significantly evolved over time. Although the American Society for Engineering Education (ASEE) was established over a century ago, modern research in this field began to consolidate around 2003 [61]. This shift was marked by the Journal of Engineering Education's exclusive focus on research and the creation of new engineering education departments and Ph.D. programs at various U.S. universities. These changes were driven by accreditation standards introduced in 1997 that required clear learning goals and evidence of student learning [39]. This section discusses the emerging challenges in engineering education, such as the need for integrating trans-disciplinary knowledge, the importance of professional competencies, and the rapid technological advancements impacting industry.

Currently, engineers face significant challenges due to rapid technological advances and changing workplace dynamics. Industry, government, and professional societies urge educators to equip engineering students with both technical and professional skills such as communication, teamwork, creativity, lifelong learning, and problem-solving. Despite these demands for reform, many engineering programs continue to follow an outdated model of engineering practice, which does not align with the actual demands of the field [50][88].

A holistic, multidisciplinary approach is critical. It's imperative that engineering curricula not only convey theoretical knowledge but also integrate real-world projects and collaborative efforts that reflect the current demands of the industry. This ensures that students can effectively apply their knowledge in diverse and practical settings, preparing them to tackle the different challenges they will face in their professional careers. Employers and faculty have noticed a decline in the problem-solving abilities and technical skills of engineering graduates compared to their predecessors [76]. Students themselves are often uncertain about the nature of engineering work upon graduation. This disconnect between education and practice leads to several consequences. Graduates struggle to remain relevant in an industry that increasingly outsources and downsizes. Employers bear additional training costs for new hires, and some graduates leave engineering for other fields due to uncertainty about their roles or the allure of better pay elsewhere [63][59]. Providing students with mentorship and internship opportunities enriches their educational experience, offering them a glimpse into the day-to-day realities of engineering work. These experiences are instrumental in cultivating the essential professional competencies, allowing students to emerge as well-rounded professionals.

The study conducted by Trevelyan in 2007 [84] suggests that improved engineering practices are essential for guiding students in their career decisions and enhancing their readiness for professional roles. This transformation involves a twofold strategy. Firstly, academic institutions must infuse their curricula with real-world projects and collaborations that mirror the current engineering landscape, ensuring that students can apply theoretical knowledge in practical settings. Secondly, there must be a coordinated effort to maintain an ongoing dialogue with industry leaders to keep educational objectives in sync with the evolving needs of the engineering sector. Moreover, enhancing mentorship and internship opportunities provides students with practical insights into engineering, clarifying the profession and equipping them with essential competencies like collaboration, communication, and adaptability, in addition to technical expertise.

In addition to addressing these educational strategies, there is also a need to foster a culture of continuous learning within the engineering community. As technological advancements outpace traditional learning cycles, the ability to learn and adapt becomes as crucial as foundational knowledge. Hence, engineering programs should aim to instill a mindset geared towards lifelong learning, encouraging graduates to view their education as the beginning rather than the culmination of their professional development.

1.2 Research Objective

The primary objective of this research is to develop a comprehensive framework using Model-Based Systems Engineering (MBSE) tools and methods in the design and execution of university-level engineering education system. Figure 1.2 shows the top-down view of university level engineering education system from a Model-Based System Engineering (MBSE) perspective. It begins within the problem domain, identifying stakeholder needs captured by government, industry, and accreditation agencies like ABET. These needs drive the solution domain within various departments such as Electrical Engineering, Computer Engineering, and Mechanical Engineering among others, where curriculum development is a central activity, comprising multiple tracks and courses for each department.

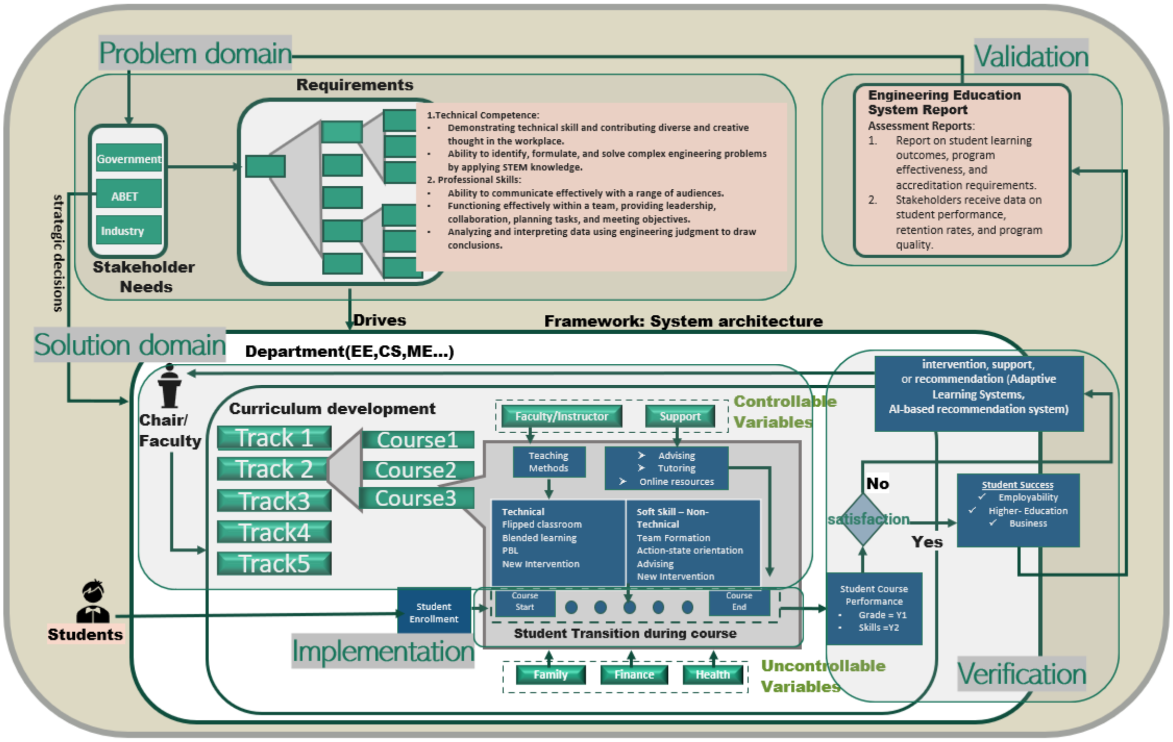


Figure 1.2 Engineering Education System Top-down Framework from a Model-Based System Engineering Perspective

This framework undertakes the complex nature of the engineering education system, characterized by its twofold nature: the human-centered entities, such as the interrelationships among students, educators, administrators, and industry partners; and the dynamics of 21st-century needs in engineering graduates, i.e., technical and professional competencies. Furthermore, the framework is tailored to the diversity of stakeholders, their needs, and the dynamic changes in the engineering education sector, it also integrates early simulation-based verification and validation of use-case scenario models for effectiveness and reliability.

In order to successfully implement the proposed framework, two byproducts need to be developed which are:

1. Development and utilization of the Generative Synthetic Data Model (GSDM) required for the MBSE model verification.

2. Development of a team formation algorithm using multi-criteria integral programming for implementation in use cases required for MBSE model validation and Graphical User Interface (GUI) to automate the proposed team formation process.

These byproducts are also shown in Figure 1.3, which illustrates the workflow of the framework from the context of Model-Based Systems Engineering (MBSE).

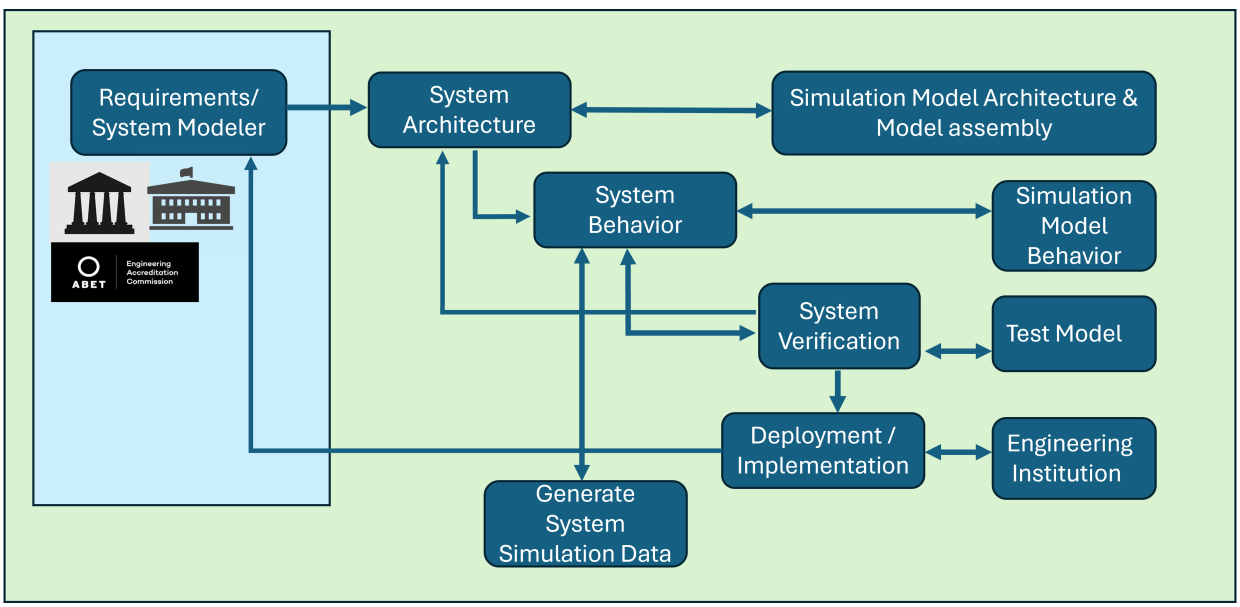


Figure 1.3 Workflow of Model-Based Systems Engineering (MBSE) Framework

The workflow process begins with system requirements, followed by the design of the system architecture, which includes high-level components, interfaces, and interactions. Subsequently, the system behavior is modeled, and system simulation data is generated to simulate system behavior. After this modeling phase, system verification for specific use case scenarios are performed and verified using requirement verification and validation. Upon successful system verification, system validation is done through the implementation of use case scenarios in real-world environments.

The expected results of the proposed research framework will allow for the execution of various use case scenarios that will enhance educational methodologies and strategies in engineering education systems following novel system engineering principles to engineering education research. While accomplishing this research, three peer-reviewed articles have been published in conference proceedings [78], and a journal transaction [77].

1.3 Contributions

This research explores the applications of Systems Engineering (SE) and Model-Based Systems Engineering (MBSE) tools such as CAMEO System Modeler (CSM) to design an Engineering Learning Analytic Systems (ELAS) model. ELAS represents a university-level engineering education system aimed at enabling transparent communication among all stakeholders, more effectively managing the complexity of the engineering education systems, and assisting higher education administrators in making better decisions, with the aim for preparing students for 21st-century engineering challenges by using 21st-century tools such as MBSE. This includes enhancing essential skills like creativity, innovation, problem-solving, collaboration, and communication through improved curriculum design and instructional materials throughout engineering programs.

1.4 Dissertation Organization

The dissertation is structured into seven comprehensive chapters, each dedicated to specific aspects of the research. Chapter 2 provides a comprehensive historical context of engineering education, identifying gaps in the current educational system affecting student performance and the evolving dynamics between educators and students. It also examines factors affecting students performance and the applications of Model-Based Systems Engineering (MBSE) in diverse fields. Chapter 3 outlines the research methodology which includes, the development of a life cycle framework and system architecture for an engineering education system, detailing MBSE system frameworks, system requirements, and the interplay of system components. Chapter 4 focuses on byproduct 1 implementation details, particularly the development of a generative synthetic data model to support MBSE simulation, and it presents an extensive section on synthetic data generation, including introduction, methods, mathematical formulations, and algorithm implementation, culminating in results, conclusions, and a discussion of the implications. Chapter 5 focuses on byproduct 2 implementation details, particularly the development of leveraging Multi-Criteria Integer Programming Optimization algorithm for effective team formation. This chapter is divided into several sub-sections, including an introduction, a detailed literature review, and an elaborate explanation of the proposed Multi-Criteria Integer Programming Team Formation Framework. The framework discussion covers various components, such as objectives, mathematical formulation, objective functions, constraints, a two-stage optimization strategy, and data simulation and validation. The results are then discussed, which include data simulation and visualization, validation of the MCIP model using simulated data, and comparative performance analysis. Chapter 5 concludes with the GUI implementation of the MCIP model, a discussion, and the conclusion. Chapter 6 is focused on MBSE Model System Verification and Result Analysis. It covers system simulation, use case scenarios, and the development and testing of the Team Formation Algorithm, with specific implementations based on diversity and project requirements. This chapter demonstrates that the proposed models and algorithms are tested and validated against real-world scenarios. Chapter 7 finalizes the dissertation with conclusions, limitations and future research. In addition, it highlights the significance of modeling the engineering education system using MBSE tools.

Chapter 2: Literature Review

This chapter explores the key components that underpin the framework introduced in Chapter 1. These components include Engineering Education Research with an emphasis on Factors Impacting Students' Performance and Students' Success, Model-Based Systems Engineering (MBSE), and its applications across different fields.

2.1 Engineering Education Research

Engineering Education Research (EER) investigates various aspects of engineering education, including teaching methodologies, curriculum design, assessment techniques, and student learning experiences. It aims to enhance the quality of engineering education by applying research findings to instructional practices. Research in engineering education is divided into five categories:

1. Engineering Learning Approaches - research on learners' knowledge and competencies [40][75].

2. Engineering Teaching Strategies- research on instructional design and teaching methods [70].

3. Engineering Assessment - research on assessment methods, instruments, and measurements to inform engineering education practice [12].

4. Engineering Epistemologies- research on what constitutes engineering thinking and knowledge within a particular context [21].

5. STEM Education Research- research on STEM education in the university [15].

This research work contributes to two of the above areas: STEM Education Research and Engineering Teaching Strategies. STEM Education Research focuses on improving educational practices in science, technology, engineering, and mathematics, aiming to create more effective and inclusive learning environments. This field investigates innovative teaching methods, curriculum design, and educational technologies that can enhance students' understanding and interest in STEM subjects. Furthermore, engineering teaching strategies delve into the specific instructional methods and pedagogical approaches that can improve learning outcomes in engineering education. This includes exploring active learning techniques, project-based learning, and other strategies that foster critical thinking and problem-solving skills.

At the intersection of STEM Education Research and Engineering Teaching Strategies lies the fundamental goal of helping students perform better and succeed. By integrating insights from both fields, educators can develop comprehensive and evidence-based frameworks that address the diverse needs of students. This holistic approach ensures that teaching strategies are not only grounded in solid research but are also tailored to the unique challenges of engineering education. In order to develop a comprehensive framework for the engineering education systems with the aim of enhancing student performance, a thorough literature review is necessary. This review identifies the factors that significantly impact student performance and success. By understanding these factors, educators and policymakers can implement targeted strategies to improve educational outcomes. Therefore, exploring the factors impacting student performance and success is crucial to the development of effective engineering education framework.

Student performance and success are influenced by various aspects/factors, such as the environment, i.e., financial challenges, school location, community and pedagogical factors, i.e., teaching methods, standardized tests, teacher-student relationship, and engagement with industry partners. The next section presents research related to each of these aspects/factors.

2.1.1 Environment Factors

- Financial: Financial resources available to the students, which play an important role in determining the extent to which they can learn and practice. According to [32] the allocation of financial resources affects student's performance in educational groups which is evidence that students respond to the availability of financial resources and incentives, along with their current financial constraints to meet their educational goals. The study suggests that policymakers and school governments should consider financial resources as an effective tool for achieving good governance and improving effectiveness in the educational system.

- School Location: Student learning is influenced by the location of their school, whether in urban or rural areas [13]. Students from rural areas often exhibit lower performance due to various factors including familial circumstances, parents' educational backgrounds, parental expectations, socioeconomic conditions, the level of qualifications of teachers, and the availability of facilities and resources. Additionally, these conditions can vary significantly from one country to another, a situation that is influenced by the respective government and policymakers [91][20] [5]. It is important to note that the learning experiences of students depend on numerous factors and are not confined to those mentioned in surveys.

- Community: According to research by Anne T. Henderson and Karen L. Mapp student success is impacted by school, family, and community. Studies suggest that there is a significant positive correlation between community involvement and student success [44]. Student education extends beyond school campuses to include experiences outside school premises, underscoring the importance of external community factors in evaluating student success.

2.1.2 Pedagogical Factors

- Standardized Tests: In the past, standardized tests were often used to evaluate students' performance and help teachers identify students' strengths and weaknesses based on test results [65]. However, this method has its advantages and disadvantages regarding students' personalities. For example, formative assessment helps authentic learning in students, promoting future success [86], while standardized tests do not assess behavioral aspects [14]. A substantial amount of research has measured academic performance, indicating that learning can improve communication skills and that parental guidance plays a significant role [79]. The methods used in these studies involve data sampling and statistical techniques. Students' performance also correlates with administrators' decisions, as they are responsible for defining the school's academic structure. They address issues such as classroom overcrowding, school funding, and community building, and work to increase the number of qualified faculty [24] [16]. These factors, although seemingly minor, significantly impact the student's experience at a university. Additionally, there are other factors that affect student performance and success.

- Teaching Methodologies: According to past surveys [56] [64]. Teaching methods involving higher student participation, for example, in cooperative problem-solving during class time and in-class assessments have a major impact on students' ability to learn compared with traditional lecture pedagogy. These methods include student group activities, pre-tests, post-tests, homework problems, and in-class formative assessments. Additionally, it is crucial to consider factors like student engagement and learning strategies. [36] explores the relationship between perceived course value, student engagement, and levels of learning. The study implies that students learning through deep-learning strategies are more engaged with the learning process and has a higher perceived value compared to the surface-learning strategies. The study suggests that deep learning has a greater impact in the perceived course value and student engagement.

- Industries Collaboration: Recent studies [58] indicate that besides resource centers and learning tools, schools and universities must focus on equipping students with the specific skills needed for sustained success and rewarding careers. In today's changing landscape, skills are increasingly prioritized over credentials. If students lack the necessary skills upon completing their degree programs, their education will have limited practical value, regardless of their academic performance. Educational systems within colleges and universities need to collaborate closely with industry to address gaps in student achievements. It is crucial for educational institutions to partner with industries and employers. This collaboration will allow college faculties to develop curricula tailored to industry needs, potentially reducing unemployment issues stemming from a mismatch between developed skill sets and industry requirements [11].

- Teacher-Student Relationship: Teacher-Student relationship is yet another factor which plays an important role in student academic performance improvement [90] [7] [87]. Studies demonstrate that teacher-student relationships correlate with student personality traits. Teachers who exhibit empathy tend to manage student behavior and academic engagement more effectively, resulting in improved student grades and attendance [74].

Based on the above literature review it can be said that many studies have worked on evaluating students' performance and noting factors that affect students' learning. However, there are no studies that model engineering education systems as complex systems and that uses complex system methodologies to analyze the challenges that engineering education systems face. This research leverages the modeling of complex systems, i.e., engineering education systems using Model-Based Systems Engineering (MBSE) methodology. MBSE have been used in many fields such as aerospace, automotive, DoD, DoT, among many others.

2.2 Model-Based Systems Engineering (MBSE)

Unlike traditional engineering approaches that rely on text-based documents and manual processes, MBSE employs digital modeling and simulation techniques to design complex systems [31]. These models provide a visual and interactive representation of system components and their interconnections, making it especially valuable for intricate systems and interfaces. By using digital models, MBSE enhances efficiency, reduces the risk of errors, improves communication among engineering teams, and ensures information consistency throughout the project's life cycle. The benefits of MBSE include better stakeholder understanding, reduced errors, early issue detection, cost and time savings, and adaptability to various project sizes and complexities. It is a versatile approach applicable across domains, supporting product development throughout the entire life cycle.

MBSE includes interactions such as analyzing user needs, specifying system requirements, creating models to represent different aspects of the system, conducting simulations and tests for verification, implementing and maintaining the system. As a cost-effective approach, MBSE allows timely exploration and documentation of system characteristics. By validating these characteristics early on, models facilitate rapid feedback on requirements and design decisions, contributing to efficient system development. Whether in aerospace, automotive, or other domains, MBSE plays a crucial role in achieving robust and reliable systems by placing models at the center of system design.

There are four pillars in Model-Based Systems Engineering (MBSE): the modeling systems language, the modeling tools, the methodology, and the Safety & Reliability [85] which are describe below:

1. Modeling Language: the modeling systems language serves as the foundation for MBSE. It provides a formal syntax and semantics for expressing system requirements, architecture, behavior, and interactions. Common languages include SysML (Systems Modeling Language) and UML (Unified Modeling Language). These languages enable

consistent communication among stakeholders and facilitate precise system representation.

2. Methodology: the methodology guides how MBSE is applied throughout the system life cycle. It encompasses processes, practices, and guidelines for model development, analysis, and validation. A robust methodology ensures systematic and effective use of models.

3. Modeling Tools: a modeling tool is essential for creating, visualizing, and managing system models. These tools allow engineers to construct diagrams, define relationships, and simulate system behavior. Examples include Enterprise Architect, MagicDraw, and Papyrus.

4. Safety & Reliability (S&R): safety and reliability are critical aspects of MBSE. Engineers must consider safety requirements, hazard analysis, and risk mitigation. Reliability modeling assesses system performance, failure rates, and maintenance strategies.

Cameo Systems Modeler excels in simulating complex systems and analyzing real-world scenarios [27]. It is a cross-platform tool, allowing engineers and stakeholders to collaborate seamlessly across different operating systems. Cameo Systems Modeler strictly adheres to the OMG SysML (Systems Modeling Language) standard. It provides a rich set of modeling elements, diagrams, and notations for representing system requirements, architecture, behavior, and interactions. The tool facilitates requirements management by linking system requirements to specific model elements. Engineers can establish traceability between requirements, design decisions, and system components. Cameo Systems Modeler supports intermediate model-based simulation. Engineers can simulate system behavior, test scenarios, and evaluate performance early in the design process. Engineers can perform parametric studies to explore how system characteristics change based on varying parameters. This capability helps in sensitivity analysis and optimization. In summary, Cameo Systems

Modeler empowers engineers to create, analyze, and validate system models, ensuring robust and efficient system development.

2.2.1 MBSE and its Application

Systems Engineering is an inter-disciplinary field that focuses on designing, integrating, and managing complex systems throughout their life cycles. It emphasizes a holistic view, considering every aspect of a system, from its inception to its decommissioning. Systems engineering involves a range of activities like requirement analysis, system design, implementation, and validation, often dealing with not just the technical aspects of a system but also logistical, human factors, and other application-driven elements.

The conceptual model serves as a comprehensive and cohesive representation of a system and its operating domain. Detailed below is the development process of the conceptual modeling process from problem space to solution space following a four-step approach:

1. Domain Model: this artifact captures the high-level components of the system and its environment. It establishes a general framework for diverse stakeholder organizations.

2. Use Case: describes what the system shall do, capturing its expected behaviors and interactions with external actors and how it will be tested.

3. Functional Model: describes how the system will accomplish its goals, the functional model breaks down use cases into greater detail. It shows activity flows and state transitions among components.

4. Structural Model: captures the specification of system structure allocating attributes and operations to system components, expanding and adding detail to the domain model.

Model-Based Systems Engineering (MBSE) is a methodology that uses graphical language to generate and record details related to system's requirements, design, analysis, verification, and validation [72]. With the advancement in technology over the past decade, computer applications have been developed to apply object-oriented software concepts to systems engineering to support the development of high-level complex systems. MBSE implies that the models are composed of an integrated set of representations. All leading MBSE tools and methodologies assume that the representations of behavior and structure are interconnected in a central repository [60]. Each descriptive element can be represented in many forms to create a variety of designs and architectural representations. Expanding upon the INCOSE definition [42], MBSE is a methodology where models are central to the specification, design, integration, verification, and validation of systems. The representations of system behavior and structure are captured along with statements of needs and verification methods. This approach makes it easier to evaluate complex descriptions as consistency is enforced, reducing errors that might not be apparent until much later in the development cycle.

Model-Based Systems Engineering (MBSE) is an evolution of classic systems engineering [43]. MBSE enhances traditional Systems Engineering processes by using models as the primary means of information exchange, rather than relying on document-based approaches. These models serve as a visual and analytical representation of the system, capturing its components, relationships, behaviors, and constraints more dynamically and interactively. MBSE allows for more efficient communication among stakeholders, facilitates better understanding and analysis of the system, and supports decision-making throughout the system's life cycle.

In different disciplines, MBSE has proven to be a valuable tool for solving complex system problems, see Figure 2.1. In aerospace engineering, for instance, MBSE facilitates the design and integration of intricate systems, such as aircraft or spacecraft, by capturing requirements, managing interfaces, and coordinating subsystems [62] [89]. Similarly, in automotive engineering, MBSE helps streamline the development of modern vehicles by modeling their various components, including mechanical, electrical, and software systems [1] [25] [57].

Moreover, MBSE finds applications in healthcare, where it aids in the design and optimization of complex medical devices and systems [52] [53]. By employing MBSE techniques, healthcare professionals can ensure that these systems meet safety regulations, accommodate different user needs, and effectively integrate with existing healthcare infrastructure [47].

In the context of engineering education, MBSE offers promising solutions to enhance the efficiency and effectiveness of educational systems. By employing modeling techniques, educators can:

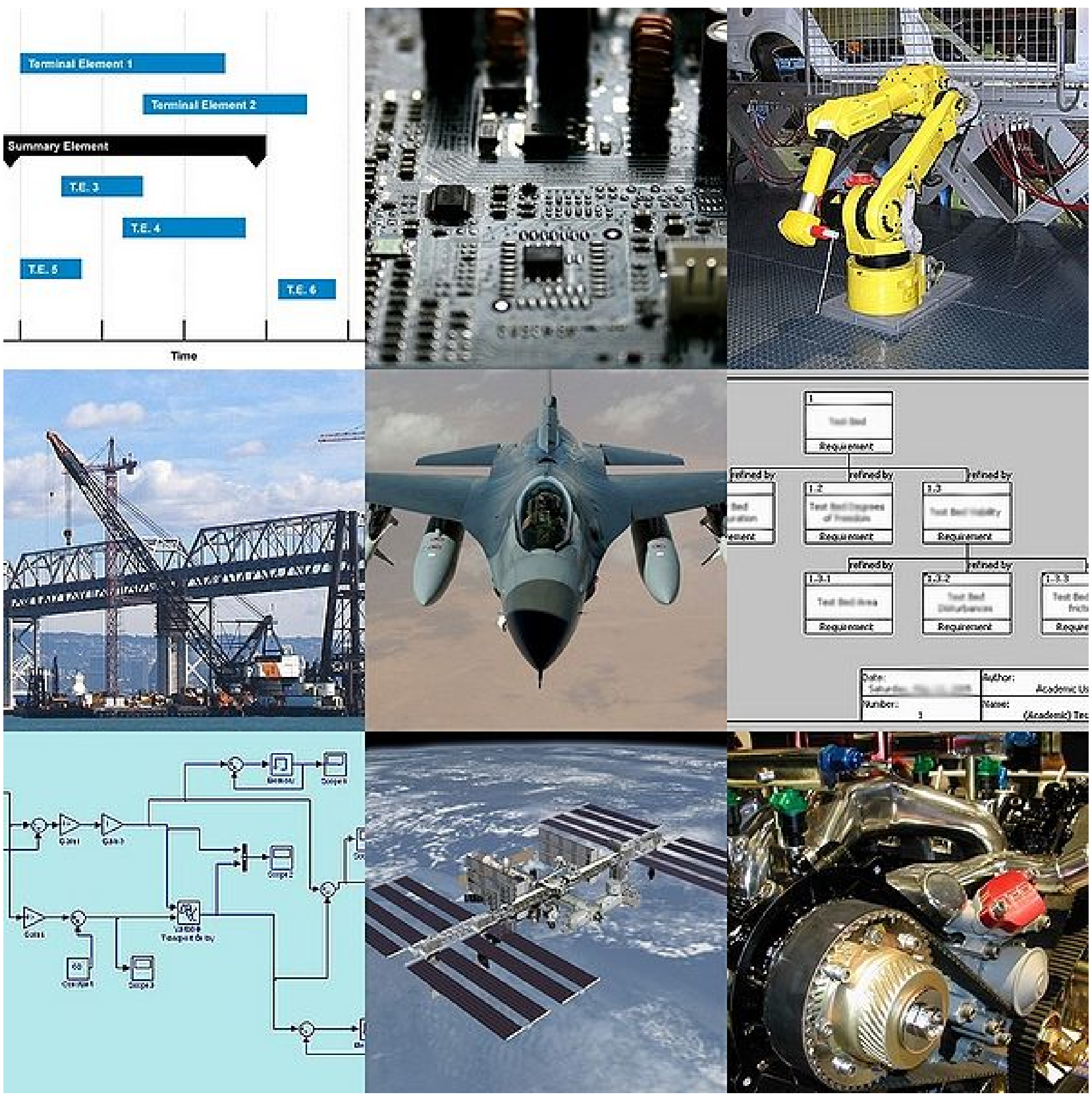


Figure 2.1 Systems Engineering Application in Various Industries

- Improve collaboration: MBSE encourages collaboration among different stakeholders in engineering education systemss, including educators, students, administrators, and industry partners. By creating a shared model, stakeholders can have a common understanding of the system and work together to improve it.

- Make data-driven decisions: by collecting and analyzing data on student performance and engagement, MBSE can help educators make informed decisions about how to improve the learning process and promote student learning and success. This can lead to more effective teaching strategies and better student outcomes.

- Improve efficiency: MBSE can help identify areas of inefficiency in the engineering education systemss, such as redundant or outdated course materials. By eliminating these inefficiencies, educators can improve the overall effectiveness of the system.

- Modify/Adapt Model: MBSE is designed to be adaptable to change, which is crucial, especially in fields like engineering education, where technology, requirements, and environments evolve rapidly.

By applying MBSE in education, particularly in engineering education, it's possible to construct and analyze detailed models of educational processes and environments. This approach enables educators and administrators to better understand and optimize learning pathways, curriculum designs, and the integration of technology in classrooms. The systematic nature of MBSE, combined with its capacity for handling complex, multifaceted systems, makes it an ideal tool for tackling the challenges of modern education systems. It allows for a more dynamic, responsive, and student-centered approach to learning, aligning educational outcomes more closely with industry requirements and student needs.

Chapter 3: Research Methodology

This chapter presents the methodology of the research which include; Framework Development Process, and MBSE for University Level Engineering Education System. Key components include understanding stakeholder needs, defining requirements, designing system architecture, analyzing system behavior, and system verification.

3.1 Framework Development Process

In the development process of the Model-Based Systems Engineering (MBSE) framework for engineering education systems, the Vee model stands out a cycle development methodology that integrates Verification throughout all phases of the Vee sequential methodology up to the Validation phase (V&V). This model originated from the need for a systematic approach to model development, in which each stage of the Vee model plays a crucial role in ensuring the system mode accuracy, relevance, and adaptability. The different stages of the Vee model are illustrated in Figure 3.1.

The V-Model emphasizes the importance of early and continuous consideration of system requirements definitions, design, verification, and validation activities throughout the development lifecycle.

At the top of the V-Model, the left side represents the initial stages of the development process. This includes activities such as requirements analysis and system design. During this phase, system requirements are gathered, analyzed, and documented. The system design is then developed based on these requirements, selecting the architecture and components necessary to fulfill them.

Moving down the V-Model, the right side represents the subsequent stages of the development process. This includes activities such as component design, implementation, and testing. The design is broken down into individual components, and each component is designed and implemented according to the system design architecture. Once implemented, rigorous testing is conducted to ensure that each component functions as intended and meets the system's requirements.

At the bottom of the V-Model, the left side represents the stages focused on system verification. This includes activities such as system integration and testing. Integration involves combining all the individual components and ensuring their proper interaction and functionality as a cohesive system. System testing is then performed to verify that the system meets the defined requirements and functions as expected.

Moving up the right side of the V-Model, the stages represent validation activities. This includes activities such as system validation and acceptance testing. Validation ensures that the system meets the intended user needs and performs its intended functions within its operational environment. Acceptance testing involves assessing the system's readiness for

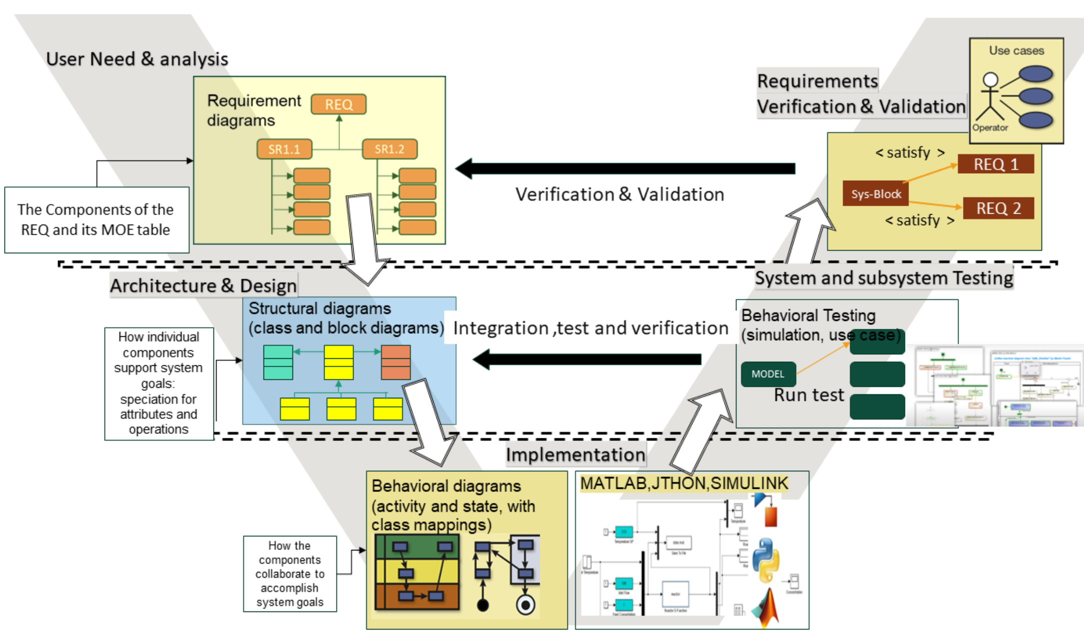


Figure 3.1 V-Model for the Development of the Engineering Learning Analytic System

deployment and its compliance with the stakeholders' expectations. The V-Model is a valuable tool in systems engineering as it emphasizes the importance of considering requirements, design, verification, and validation activities throughout the entire development process. By following this model, engineers can mitigate risks, identify issues early on, and ensure that the final system meets the desired specifications and satisfies the needs of its users.

3.2 MBSE for University Level Engineering Education System

The framework shown in Figure 3.2 is a top-down view of a university level engineering education system from a Model-Based System Engineering (MBSE) perspective. It begins by, identifying stakeholder needs captured by government, industry, and accreditation agencies such as ABET. These needs drive the university's architecture, which consists of various departments (EE, CE, ME), tracks, curricula and courses, and the students and faculty who are engaged within it, utilizing different teaching methods, support systems, and resources.

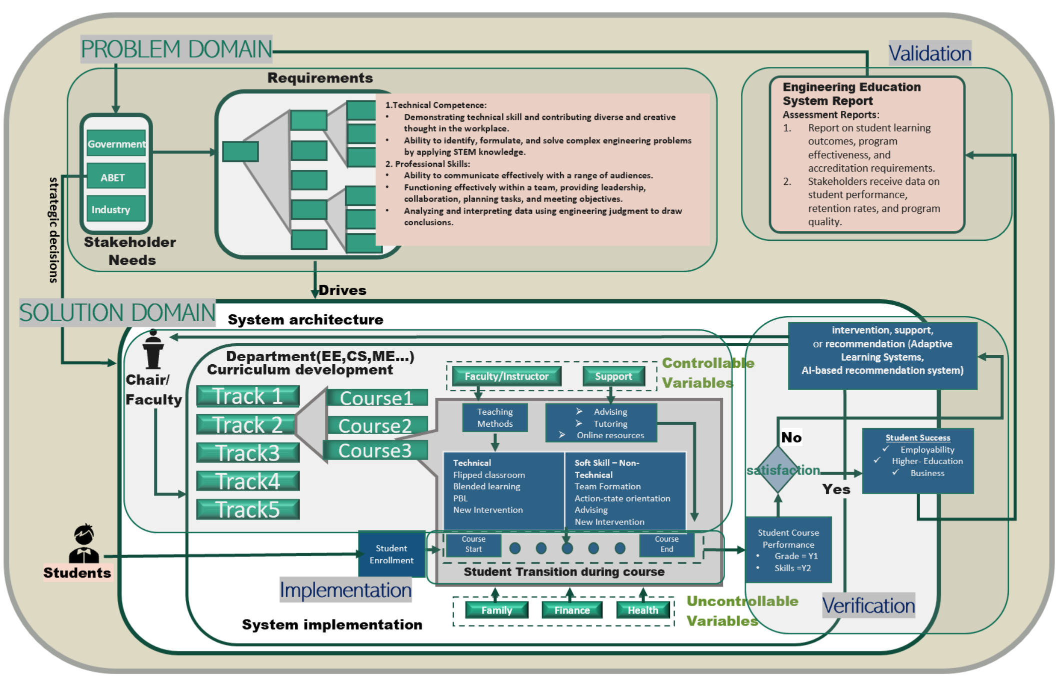


Figure 3.2 Engineering Education System Top-down Framework from a Model-Based Systems Engineering Perspective

Student performance in a particular course is further subjected to controllable variables, like teaching methods and support systems, as well as uncontrollable variables such as student family, finances, health, among others. To evaluate, student's course performance, skills must be continuously monitored, in order to be able to incorporate adaptive learning/interventions. Finally, an overall report from the university is provided to the stakeholders for feedback and support.

Figure 3.2 also represents the MBSE design framework, which is divided into two main domains: problem and solution domains. Within the problem domain, the analysis starts by collecting stakeholder needs, identifying users and external systems interacting with it, and conducting a black box analysis to operationally examine the system in various contexts. This is followed by a white-box analysis for a deeper understanding of the system [73]. Functional analysis is then conducted to comprehend system functions and outline conceptual subsystems, defining their quantifiable characteristics, known as Measures of Effectiveness (MoEs). These needs are then integrated into the SysML model. In the solution domain, which consists of system architecture, system implementation, and validation, system architecture represents various departments (EE, CE, ME), tracks, curricula and courses, and system implementation representing the students and faculty engaged, utilizing different teaching methods, support systems, and resources. System implementation consists of the development of a synthetic data model for engineering education dataset generation and Multi-criteria team formation algorithm development to be implemented as an intervention in engineering courses to improve student professional skills. This process includes a precise, cross-disciplinary logical architecture for the system that can address the problems identified via stakeholder analysis and specifying the system's requirements, structure, behavior, and parameters.

3.2.1 Stakeholder Needs

Identifying stakeholders is the initial step prior to gathering their needs. A stakeholder is an individual, group, or entity with a vested interest in a project, organization, or business [38]. These stakeholders can significantly impact or be impacted by the outcomes and decisions related to the project or business. They often have diverse interests, needs, and perspectives. The initial phase involves identifying the stakeholders associated with the engineering education system, which serves as the System of Interest (SoI). Key stakeholders in the context of engineering education systems are Educational Institutions, Government, Industry and Accreditation Board for Engineering and Technology (ABET). These stakeholders play a crucial role in the system's development, implementation, and ongoing operation. Their respective needs and concerns have been meticulously gathered from literature review articles and reports [29] [49] [67].

The ELAS framework stakeholder requirements are outlined in the Table 3.1. Each requirement is considered a building block for the system, ensuring the final product aligns with the specific needs of each stakeholder group:

3.2.2 Stakeholder Requirements

1. Government Requirements ($SH_R1 - SH_R2$): for government stakeholders, the system must adhere to performance-based funding models, ensure graduate earnings meet expected averages, maintain eligibility for financial aid, and achieve a certain job placement rate. These requirements are translated into system specifications that track and report on these metrics.

2. Educational Institution Requirements ($SH_R3 - SH_R4$): the institution's needs, including maintaining ABET accreditation and demonstrating a progression in technical skill and responsibility, are incorporated into the curriculum development process. This includes

Table 3.1 Stakeholder Requirements

ID	Stakeholder Requirement	Stakeholder Requirements Description
	GOVERNMENT	
SHR_1	Florida State University System - Performance Based Funding Model:	
$SHR_1.0$	-Bachelor's Graduates Earning	Average Bachelor's Graduates Employed Earning - Full-time must be >= $60000
$SHR_1.1$	-Eligibility for financial assistance	Students must maintain a completion rate of 67% or higher to remain eligible for financial assistance
$SHR_1.2$	-Pell-grant financial assistance	Students must complete graduation in six-years for receiving pell-grant
$SHR_1.3$	-Job placement rate or continue education	Each year students job placements rate should be > 50% or continue education [4]
SHR_2	**ABET accreditation**	
$SHR_2.0$	Demonstrate a progression in technical competence and increasing responsibility in the practice of engineering	(1) An ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics.(2) An ability to apply engineering design to produce solutions that meet specified needs
SHR_3	**Industry**	
$SHR_3.0$	Essential 21st century soft skills for engineer	1) Problem Solving:The ability to identify, analyze, and solve complex problems, 2) Communication: Effective exchange of information and ideas, 3) Collaboration: Work effectively with others towards a common goal, 4) Leadership: Guide and inspire others to achieve goals, 5) Critical Thinking: Objective analysis and evaluation of information, 6) Teamwork: Collaboration within a group to achieve objectives, 7) Adaptability Ability to adjust and thrive in changing environments [6].
SHR_4	**Society**	
$SHR_4.0$	Student Wellness Support	The institution must provide comprehensive support and resources to promote the physical, mental, and emotional well-being of students, fostering a healthy and balanced campus environment

defining learning outcomes that map ABET criteria and creating assessment tools to measure progression in the practice of engineering.

3. Industry Requirements ($SH_R5 - SH_R6$): industry requirements emphasize problem-solving abilities and essential 21st-century soft skills for engineers. The framework

must include mechanisms to evaluate and enhance these skills, with a particular focus on collaboration, leadership, and adaptability.

4. Society Requirements (SH_R7): the broader societal impact is addressed by ensuring the ELAS framework supports the institution's role in fostering a healthy and supportive student environment. This might involve integrating wellness resources and support structures within the system.

5. Translation into System Features: each requirement is translated into specific features and functionalities within the ELAS framework, see Figure 3.3. For example, SH_R1 might result in a feature that allows tracking of graduate employment outcomes, while SH_R2 could lead to the development of a financial aid eligibility tracking system within the ELAS.

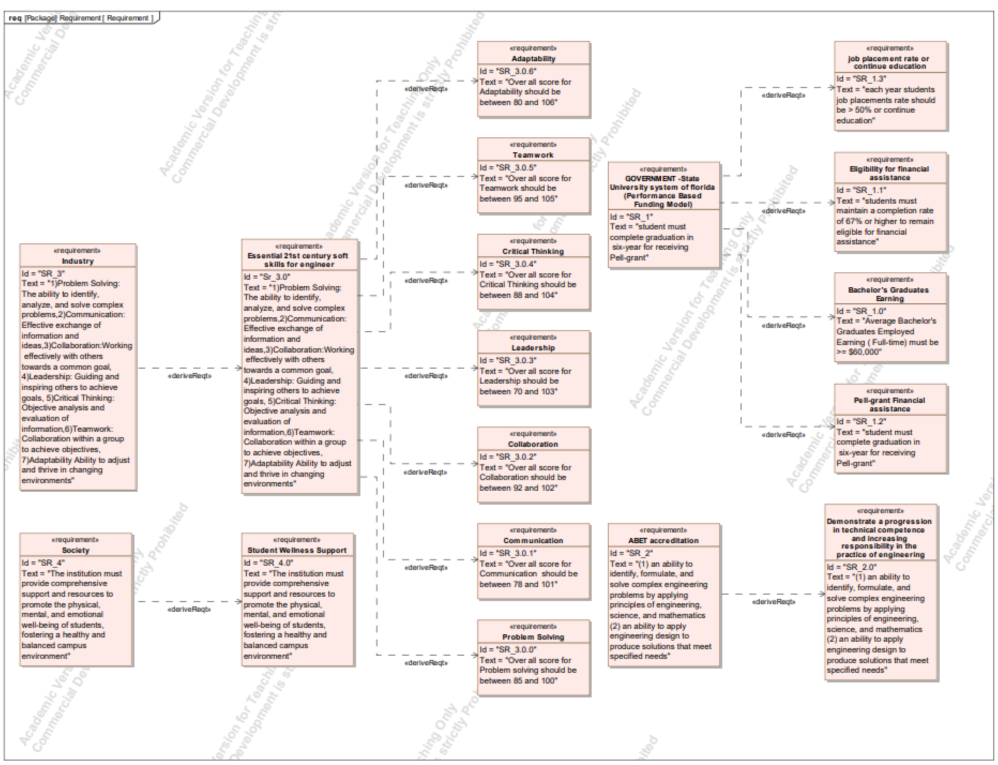


Figure 3.3 SysML Requirement Diagram Illustrating the Hierarchical Structure and Interrelationships of System Requirements for the ELAS Model

6. Requirements Verification: the defined system requirements are verified with stakeholders to ensure accuracy and completeness. This step may involve reviewing the requirements with government bodies, industry partners, faculty, and students to confirm that they reflect the stakeholders' true needs and expectations.

After the construction of the requirements table, a requirement diagram is developed using the SysML modeling language which contains both the primary and secondary needs of the stakeholders, all of which can be traced back to their origins.

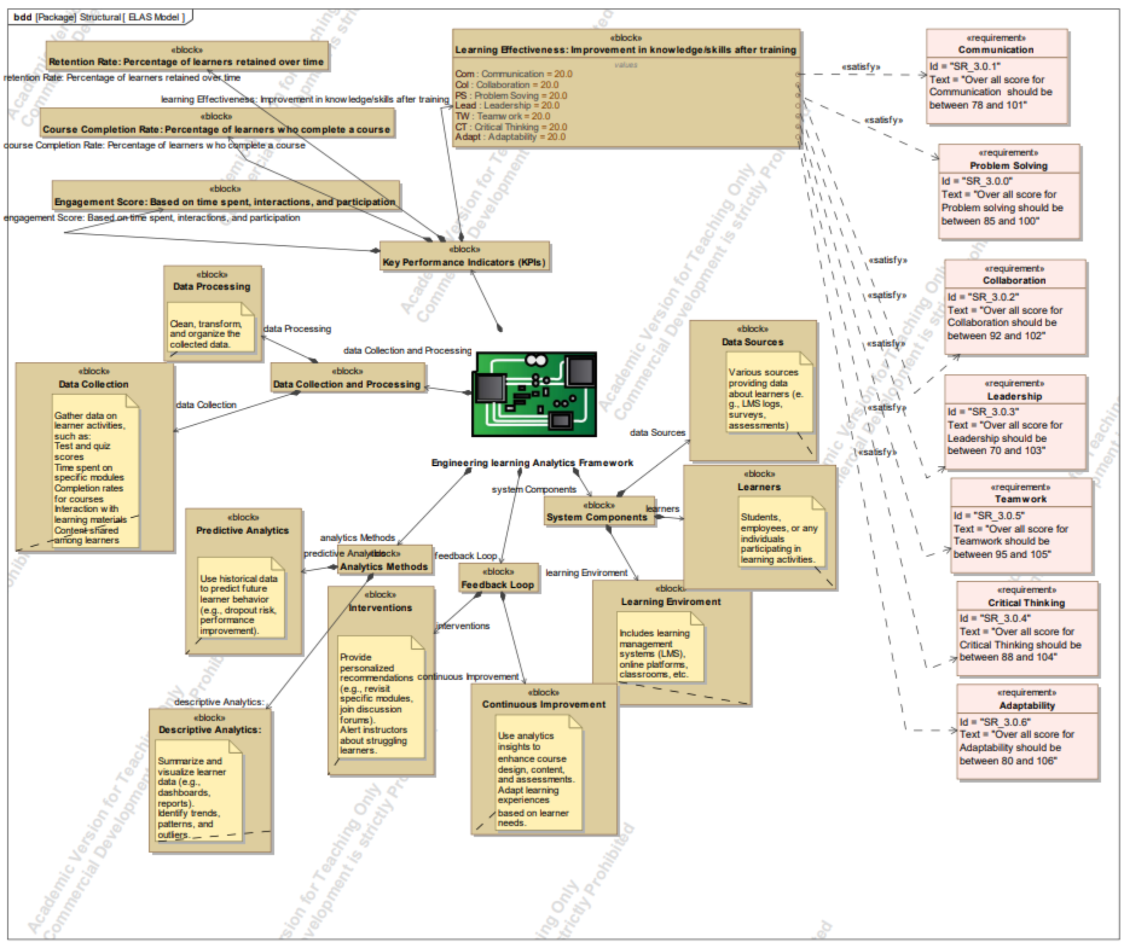


Figure 3.4 ELAS's System Architecture

3.2.3 ELAS's System Architecture

Within the MBSE framework, the Engineering Learning Analytic Systems (ELAS) structural architecture is created by assessing stakeholder needs and converting them into system requirements. It is then modeled by analyzing and codifying stakeholder needs into detailed system requirements. The model outlines the framing of the problem and provides conceptual representations of the system's interactions within its operational context, as detailed in Figure 3.4. The system architecture is organized into several components:

1. Data Collection and Processing: this is the foundation where data on learner activities, assessments, time spent on specific modules or courses, interaction with learning materials, and queries raised by students are collected from various sources, such as LMS logs, student surveys, etc.

2. Descriptive Analytics: the collected data undergoes initial analytics to summarize and visualize learner data, i.e., through dashboards, reports, identifying patterns, trends, and behaviors.

3. Predictive Analytics: this is where historical and current data are analyzed for predictive models that can forecast future learner behavior and performance, pinpointing potential areas for improvement.

4. Analytics Methods: this encompasses both predictive and descriptive analytics to feed into the system's feedback loops, enabling data-driven interventions and personalized guidance to students, such as module-specific recommendations, alert instructions for instructors, and identifying struggling learners.

5. Key Performance Indicators (KPIs): ELAS model defines specific KPIs for assessing learning effectiveness, including retention rate, course completion rate, engagement score, and improvement in knowledge/skills after training.

6. Feedback Loop: a critical component, the feedback loop, allows the system to adapt and provide interventions based on analytics. This loop informs continuous improvement processes within the ELAS.

7. Continuous Improvement: the system utilizes analytics to enhance course design, content, and assessments, adapting learning experiences based on learner needs.

8. Requirements: the system ensures all activities satisfy specific educational requirements, such as Communication, Problem Solving, Collaboration, Leadership, Teamwork, Critical Thinking, and Adaptability. These are measured by their respective scores, which are benchmarked against defined standards.

ELAS structure architecture model is a cohesive model that monitors, analyzes, and reports on student performance, sending feedback to the educational process for continuous development and alignment with industry requirements.

For the system behavior, a use case model is created shown in Figure 3.5. The use case diagram for the ELAS system architecture captures the interactions between various users such as students, research analysts, faculty/instructors, department chair and the system's functionalities. The system functionalities include data collection and integration, and data pre-processed for Learning Analytics and Data Mining. Learning Analytics processes the data to provide real-time analysis, while data mining focuses on feature selection using historical educational data to identify patterns. The insights from Learning Analytics and Data Mining feed into Dashboards and Reports, giving users actionable information. An Early Warning System uses these insights to identify students who may need additional support, facilitating Personalized Learning. Concurrently, Program/Course Assessment evaluates educational outcomes. This holistic approach allows stakeholders to make informed decisions, ensuring that the engineering course system meets educational objectives and effectively supports student development.

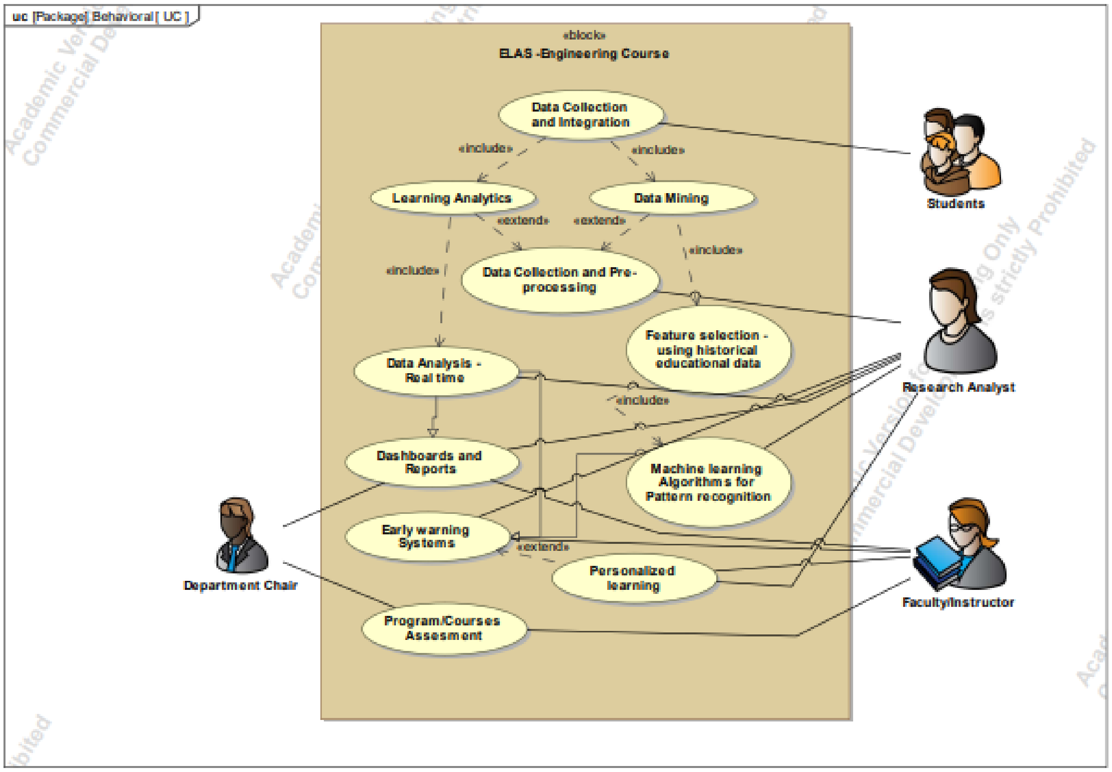


Figure 3.5 ELAS Engineering Course Use Case Model

3.2.4 System Implementation

The Implementation phase shown in Figure 3.2 is further expanded in Figure 3.6. The implementation phase is divided into two sections: Simulation and Real-world. In the Simulation section, synthetic data models are generated and tested for engineering education datasets. This phase also involves the development and testing of algorithms for team formation based on multiple criteria. Additionally, use case scenarios are developed using Cameo SysML to model real-world scenario interactions, which are then verified through SysML simulation.

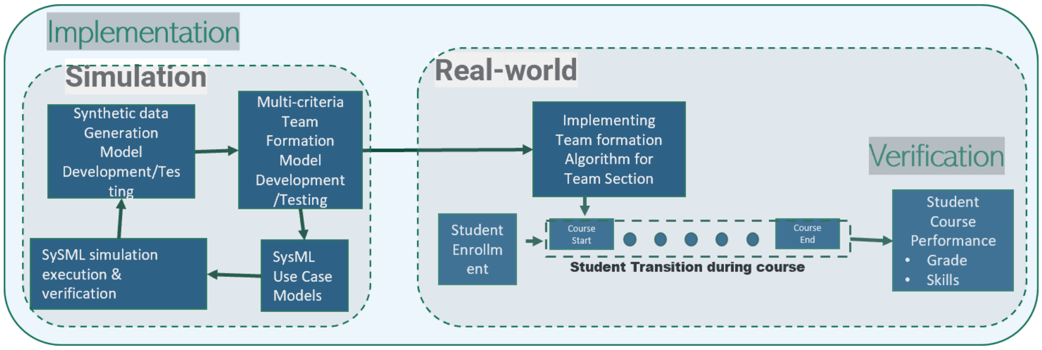


Figure 3.6 ELAS Implementation Roadmap at Glance

The multi-criteria team formation algorithm developed during the simulation phase was implemented in a real-time engineering course. Students were grouped into teams based on either diversity-based criteria or project-based requirements. At the end of the course, team performance was measured in terms of both soft skills and technical skills. The impact of the team formation algorithm on student performance is measured.

To evaluate the use case model, data is required. Hence, Chapter 4, will elaborate on the development of a generative synthetic data generation model, which will supply the necessary simulated datasets for use case model testing. This will be followed by the development of a team formation algorithm critical for the use case model implementation. The team formation algorithm is presented in Chapter 5. This team formation intervention is implemented in electrical engineering courses at the University of South Florida (USF).

3.2.5 System Verification

At the verification phase, SysML ELAS Use case model shown in Figure 3.5 is simulated for two use case scenarios that are "Equipping Students with Advanced Semiconductor Skills through Mechatronics, Robotics, and Control Laboratory tailored to Semiconductor Industry Needs" and "Emphasizing student soft skills development through various assessment methods". The simulation is further executed for various scenarios at different instances with different parameters. Detailed verification is presented in Chapter 6. The validation phase is outside the scope of this study.

Chapter 4: Implementation - Synthetic Data Generation Model for Engineering Education (Byproduct 1)

[1]The lack of relevant engineering education datasets for modeling in engineering education systems is a well-acknowledged concern. Any verified model necessitates data for use case simulation and analysis. This section presents a development of generative synthetic data models that can address data scarcity for research development i.e., to create high-quality, representative, and diverse datasets that can be used for system simulations. This model uses a Bayesian approach to generate data that closely mimics real-world scenarios. Synthetic data is used to enhance the accuracy of system simulations and system verification.

Engineering education research often requires large amounts of data that can be time-consuming and costly to collect. In response to these challenges, synthetic data generation has emerged as a pragmatic solution across various industries, allowing researchers to sidestep the complexities associated with data collection and pre-processing, thereby streamlining the focus on actual model implementation. Despite the widespread adoption of synthetic data in diverse domains, its integration into engineering education research remains relatively underexplored. Therefore, this section presents a method for generating synthetic educational data using a Bayesian approach. The method capitalizes on a Bayesian network, a probabilistic graphical model that depicts a set of variables and their conditional dependencies as a Directed Acyclic Graph (DAG). Additionally, it utilizes Gibbs sampling, a Markov Chain Monte Carlo (MCMC) algorithm that simplifies sampling from a multivariate probability

[1]The contribution of this chapter has already been published in the IEEE 3rd International Conference on Advanced Learning Technologies on Education & Research (ICALTER) [78]. Therefore, the following sections have been organized as a stand-alone chapter. The copyright permission is provided in Appendix A.

distribution. This is particularly useful when direct sampling from the joint distribution is difficult, but sampling from the conditional distribution is feasible.

4.1 Introduction

Collecting data from engineering education students is difficult due to multiple complex factors. Ethical and legal considerations demand strict compliance with privacy laws and the need for informed consent, particularly when dealing with minors [28] [41] [55] [81]. Maintaining data confidentiality and data integrity is crucial; therefore, ensuring the data's accuracy and reliability is paramount for significant analysis. Moreover, the diverse sources of student's data from academic records to digital platforms require sophisticated integration techniques to construct a cohesive dataset. Achieving a representative sample is also difficult, impacted by the voluntary nature of participation and the heterogeneity of the student body. Logistically, coordinating large-scale data collection across various educational settings demands considerable resources and careful planning. Additionally, the rapid evolution of educational practices and technologies necessitates continuous updates to data collection methodologies. An alternative to using real student data is employing synthetic data, which replicates the characteristics of actual data without containing personal information. Synthetic data is a valuable tool for researchers and educators, facilitating meaningful analysis and controlled testing. Additionally, it enables testing and analysis without compromising data privacy or quality, and it ensures representativeness, thus supporting both the integrity and applicability of the data [10] [68].

Synthetic data is artificially generated to closely resemble real-world data and; serves as a valuable resource across various domains, including machine learning, privacy protection, and software testing [30]. By creating synthetic datasets, researchers and practitioners can overcome challenges such as data scarcity, privacy concerns, and bias mitigation. These artificially generated data points enhance model training, augment existing datasets, and enable robust system simulations. Although synthetic data has limitations, its strategic use effectively bridges gaps and fosters innovation in diverse fields. It is commonly used for testing operational datasets and is widely employed in sectors such as healthcare [19], manufacturing [66], agriculture [2], and eCommerce [69]. It is adapted when real data is either unavailable or necessitates privacy preservation due to Personally Identifiable Information (PII) or compliance risks.

Businesses find synthetic data advantageous as it addresses privacy concerns, expediting product testing processes, and facilitates the training of machine learning algorithms [30]. Although data privacy regulations impose restrictions on how businesses handle sensitive information, the mitigation of privacy concerns remains a primary motive for investing in synthetic data generation methods. In scenarios where data for entirely new products is unavailable, obtaining human-annotated data has been proven to be a costly and time-consuming endeavor, therefore synthetic data becomes a viable alternative.

However, generating synthetic educational data is challenging since it involves multimodal, codependent variables. Educational datasets feature high-dimensional attributes which are related to or depend on another variable. Therefore, when constructing a generative data model, it is crucial to capture the intricate relationships among variables that may not be directly observable but significantly influence observed outcomes. These are referred to as latent factors. Latent variables uncover the hidden underlying structures within the data, potentially explaining the observed correlations among known variables. In real-world datasets, variables often exhibit interdependencies. For instance, when applying generative modeling principles to an educational dataset, the students attribute are interconnected rather than isolated factors. For example, a student's math performance may relate to their reading ability, influenced by factors like resource access or prior educational experiences. Latent variables impact multiple observed attributes within the dataset. They represent unobserved factors, such as innate abilities, socio-economic status, or instructional quality. Although not directly measured, these latent variables significantly influence observed outcomes, such as grades or test scores. A Bayesian approach effectively incorporates latent factors into models by using probabilistic techniques that account for uncertainty and the relationships between observed and unobserved variables. This method allows for the integration of prior knowledge through distributions and it uses hierarchical models to manage complex interactions influenced by latent variables. Bayesian inference techniques, such as Markov Chain Monte Carlo, estimate latent variables by calculating their posterior distributions, considering both the observed data and prior beliefs. This approach is adaptable, robust to missing data, and ideal for uncovering hidden structures within data, making it particularly useful in fields like educational research where many variables may not be directly observable [35].

In this section, the Bayesian approach is presented as a pioneering methodology for synthetic data generation model for engineering education datasets. The remainder of this section details the implementation of the proposed approach and provides a comprehensive analysis that compares the features of the original and the synthetic datasets. The findings highlight the Bayesian approach's potential as a valuable tool for synthetic data generation model in engineering education.

4.2 Method

This section introduces a framework for generating synthetic data model using a Bayesian network and Gibbs sampling. The primary goal is to closely match the true joint distribution of the original dataset with the synthetic dataset. A detailed block diagram of the proposed framework is shown in Figure 4.1.

4.2.1 Mathematical Formulation and Objective Functions

A Bayesian network is a graphical model that represents probabilistic relationships among a set of variables. Let $P(X = X_1, X_2, ...X_n)$ be the set of variables representing the features in the real dataset. The joint distribution of these variables is denoted as $P(X)$.

The goal is to generate a synthetic dataset $\hat{X}$ such that $\hat{X}$ follows a distribution $P*(X)$ that is close to the true distribution $P(X)$.

A Bayesian network is defined as $G = (V, E)$, where V is the set of nodes representing the variables X_i, and E is the set of edges representing the conditional dependencies between the variables. The joint distribution $P(X)$ can be factorized using the Bayesian network structure:

$$P(X) = \prod_{i=1}^{n} P(X_i|\text{parents}(X_i))$$

Next, the generative model involves using Gibbs sampling to iteratively sample from the conditional distributions given the rest of the variables in the Bayesian network. The Gibbs sampling update for variable X_i, is given by:

$$P(X_i|rest\ \ of\ \ variables) \propto P(X_i|Parents(X_i)) \times P(Children(X_i)|X_i) \tag{4.1}$$

Proceeding with the generation of synthetic datasets: First, initialization involves starting with an initial guess for the dataset. Second, Gibbs sampling iterations followed by sampling for each variable X_i from the conditional probability $P(X_i|rest\ of\ variables)$, then updating the dataset and repeating this process until convergence is reached. Subsequently, to evaluate the quality of the synthetic dataset, metrics such as the Kullback-Leibler divergence are

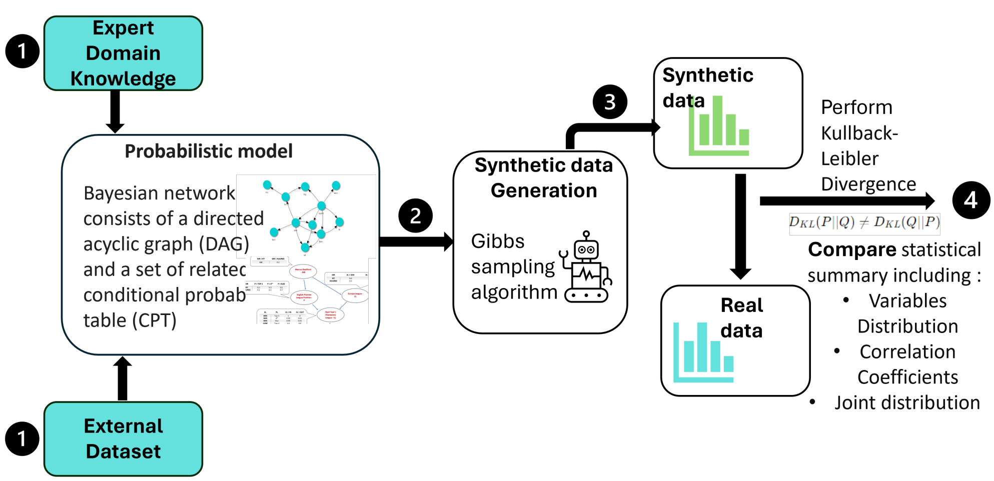


Figure 4.1 Framework for Generating Synthetic Data

used. This divergence measures the difference between the true distribution $P(X)$ and the synthetic distribution $P^*(\hat{X})$:

$$D_{\mathrm{KL}}(P(X)\,||\,P^*(\hat{X})) = \sum_{\mathbf{x}} P(\mathbf{x}) \log\left(\frac{P(\mathbf{x})}{P^*(\mathbf{x})}\right) \tag{4.2}$$

The mathematical objective function for generating synthetic datasets using Bayesian Networks and Gibbs sampling is to minimize the difference between the true distribution of the original dataset and the synthetic distribution of the synthetic dataset. This is achieved by the Kullback-Leibler (KL) divergence.

4.3 Implementation - Bayesian Network

The methodology consists of two steps. The first step involves the construction of a Bayesian model, which is built upon the domain knowledge base of engineering educational research. In the second step, relationships are established between key variables - gender, major, study time, and grades. These relationships are defined using their respective Conditional Probability Tables (CPTs), which provide a statistical representation of the dependencies between variables.

Following this, a Bayesian network structure is created. This structure visually represents the relationships between these variables, offering a clear and concise graphical representation of the complex interdependencies, see Figure 4.2. The Bayesian network structure is essentially a Directed Acyclic Graph (DAG). In this graph, a set of variables and their conditional dependencies are represented. Each node in the DAG symbolizes a variable, and each edge signifies a direct conditional dependency between the variables. This structure provides a comprehensive overview of the relationships and dependencies, aiding in the understanding and interpretation of the data.

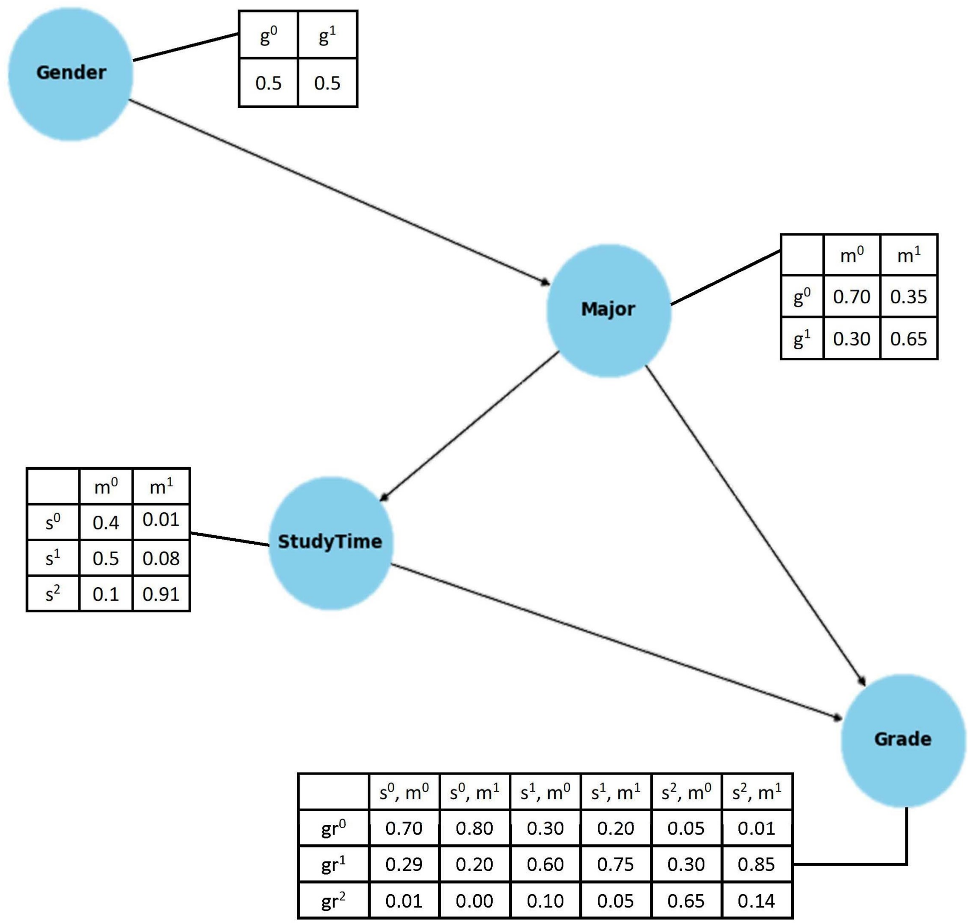


Figure 4.2 Bayesian Network: Graphical Representation Illustrating Probabilistic Dependencies and Relationships Among Variables in the Network

Synthetic data is generated using a Probabilistic Graphical Model (PGM) with Conditional Probability Distributions (CPDs). The PGM is defined by three variables: 'Gender,' 'Major,' 'StudyTime,' and 'Grade.' CPDs specify the probabilities of each variable given its each respective parents. The Gender variable has a uniform distribution, i.e., 0.5 probability for both Males and Females. The Major variable depends on Gender, and its CPD is defined accordingly. Similarly, StudyTime depends on Major, and Grade depends on both StudyTime and Major. After defining the PGM, the code initializes synthetic data with columns for 'Gender,' 'Age,' and 'Grade.' It then iteratively generates samples for these variables using Gibbs sampling, i.e., a Markov Chain Monte Carlo method used to approximate joint distributions by iteratively sampling from the conditional distributions. The loop generates random values for Gender, Age, and Grade and performs 10 Gibbs sampling iterations to update the values based on the defined CPDs. Finally, the generated sample is appended to the synthetic data.

4.4 Results

In this section, key findings from the analysis generating synthetic datasets that have the same characteristics as the original dataset are presented. Tables 4.1 and 4.2 illustrat side-by-side comparison of the dataset characteristics:

1. 'Grade' given 'StudyTime' and 'Major': the analysis of the synthetic dataset, generated based on 'Grade' given 'StudyTime' and 'Major,' reveals patterns similar to those observed in the original dataset. Notably, the distribution of grades across different study times and majors appears to be preserved, although there are variations in the frequency of occurrences. This suggests that the synthetic data successfully captures the relationships between study time, major, and grades present in the original dataset.

2. 'Major' given 'Gender': examination of the synthetic dataset in the context of 'Major' given 'Gender' indicates a reasonable replication of the original dataset's gender-major distribution. The frequencies of female and male students in both engineering and medicine majors are comparable between the original and synthetic datasets. This suggests that the synthetic data preserves the gender-major relationships observed in the original dataset.

3. 'StudyTime' given 'Major': the analysis of 'StudyTime' given 'Major' in the synthetic dataset mirrors the patterns found in the original dataset. The distribution of study times across different majors appears to be faithfully reproduced, with variations in frequency reflective of the synthetic data generation process. This implies that the

Table 4.1 Original Data Characteristics

	'Grade' given 'Study Time' and 'Major'		
Study Time	**Major**	**Grade**	**Frequency**
0-2 hrs	Engineering	60-79	61
0-2 hrs	Engineering	80-100	2
0-2 hrs	Engineering	below 60	130
0-2 hrs	Medicine	60-79	1
2-5 hrs	Medicine	below 60	2
2-5 hrs	Engineering	60-79	157
2-5 hrs	Engineering	80-100	18
2-5 hrs	Engineering	below 60	92
5-12 hrs	Medicine	60-79	29
5-12 hrs	Medicine	80-100	2
5-12 hrs	Medicine	below 60	9
0-2 hrs	Engineering	60-79	14
0-2 hrs	Engineering	80-100	43
5-12 hrs	Engineering	below 60	5
5-12 hrs	Medicine	60-79	372
5-12 hrs	Medicine	80-100	59
5-12 hrs	Medicine	below 60	4
	Gender	**Major**	**Frequency**
'Major' given 'Gender'	Female	Engineering	184
	Female	Medicine	339
	Male	Engineering	338
	Male	Medicine	139
	Study Time	**Major**	**Frequency**
'Study Time' given 'Major'	0-2 hrs	Engineering	193
	0-2 hrs	Medicine	3
	2-5 hrs	Engineering	267
	2-5 hrs	Medicine	40
	5-12 hrs	Engineering	62
	5-12 hrs	Medicine	435

synthetic data successfully emulates the study time patterns within each major as observed in the original dataset.

In summary, the synthetic dataset demonstrates an ability to replicate the essential characteristics of the original dataset across multiple dimensions. While some variations in frequencies exist, the synthetic data appear to maintain the underlying relationships and distributions present in the original dataset. These results suggest that the synthetic data generation

Table 4.2 Synthetic Data Characteristics

	'Grade' given 'Study Time' and 'Major'		
Study Time	**Major**	**Grade**	**Frequency**
0-2 hrs	Engineering	60-79	66
0-2 hrs	Engineering	below 60	158
2-5 hrs	Engineering	60-79	162
2-5 hrs	Engineering	80-100	22
2-5 hrs	Engineering	below 60	75
2-5 hrs	Medicine	60-79	29
2-5 hrs	Medicine	80-100	3
2-5 hrs	Medicine	below 60	11
5-12 hrs	Engineering	60-79	13
5-12 hrs	Engineering	80-100	32
5-12 hrs	Engineering	below 60	2
5-12 hrs	Medicine	60-79	366
5-12 hrs	Medicine	80-100	55
5-12 hrs	Medicine	below 60	6
	Gender	**Major**	**Frequency**
'Major' given 'Gender'	Female	Engineering	188
	Female	Medicine	317
	Male	Engineering	342
	Male	Medicine	153
	Study Time	**Major**	**Frequency**
'Study Time' given 'Major'	0-2 hrs	Engineering	224
	2-5 hrs	Engineering	259
	2-5 hrs	Medicine	43
	5-12 hrs	Engineering	47
	5-12 hrs	Medicine	427

process effectively captures the key features of the original data, providing a valuable tool for privacy-preserving data sharing and analysis.

4.5 Conclusion

In conclusion, a Kullback-Leibler (KL) Divergence of 0.002158 indicates a high similarity between the original and synthetic datasets, affirming the effectiveness of the synthetic data generation process in preserving the statistical integrity of the original data. This outcome validates the use of synthetic data for privacy-sensitive tasks.

Chapter 5: Implementation - Multi-Criteria Integer Programming Optimization for Team Formation (Byproduct 2)

[2]This chapter introduces a novel application of Multi-Criteria Integer Programming (MCIP) to address the intricate task of team formation. Unlike traditional single-objective optimization methods, the study designs a comprehensive framework that models various factors, including skill levels, backgrounds, and personality traits. The objective function optimizes within-team diversity while minimizing conflict levels and variance in diversity between teams. The approach involves a two-stage optimization process: first segmenting the population into sub-groups using a weighted heterogeneous multivariate K-means algorithm, followed by applying a surrogate optimization technique within these sub-groups. The study considers explicit constraints, including potential interpersonal conflicts, an aspect often overlooked in previous research. Study results demonstrate the model's robustness across simulation scenarios with varying data heterogeneity levels. Additionally, the study bridges critical gaps in existing literature by providing a theory-backed, empirically validated framework for advanced team formation. Beyond the theoretical implications, it also offers practical guidance for implementing conflict-aware, sophisticated team formation strategies in real-world contexts. This advancement lays the groundwork for future research to explore and enhance this model, ultimately leading to more advanced and efficient team formation strategies.

[2]The contribution of this chapter has already been published in the IEEE Transactions on Learning Technologies [77]. Therefore, the following sections have been organized as a stand-alone chapter. The copyright permission is provided in Appendix A.

5.1 Introduction

Team Formation Problem (TFP) is a widely recognized challenge in various fields such as operation research, computer science, management, and education [51]. The central task in TFP is to organize a group with diverse characteristics into interconnected and effective teams. Team-based Learning (TBL) [17] is a pedagogical strategy that heavily relies on the effective resolution of the TFP in educational and professional settings. In educational settings, effective team formation can enhance students' learning experiences by fostering collaborative skills and increasing engagement. Furthermore, diverse and balanced teams are known to enrich the learning experience, since members can benefit from different perspectives and experiences [80]. In professional settings, the strategic assembly of teams plays a vital role in the success of the organization. Teams are often formed to tackle specific projects, solve complex problems, or innovate new ideas. The diversity of skills, expertise, and perspectives within a teams significantly influences the quality and overall effectiveness of the team's outputs [3]. However, forming optimal teams is a complex task, which requires the consideration of multiple factors, such as individual skills, compatibility, and workload balance.

The increase in class sizes and task complexities in academic institutions and professional environments requires the development of automated and optimized methods for forming balanced and diverse teams. A well-structured team ideally represents a combination of various skills, backgrounds, and perspectives that can mutually enhance its problem-solving capacity and productivity. Conventional practices of forming teams often rely on heuristic and subjective approaches, which may be inadequate to handle the diversity and dynamics of contemporary classrooms and workplaces [23]. Traditionally formed teams, often fail to incorporate a comprehensive range of student attributes, like academic background, technical skills, and interpersonal skills, which profoundly influence the efficacy and satisfaction levels of a team. Furthermore, arbitrarily formed teams may lead to skill gaps and workload imbalances, hampering team performance and causing dissatisfaction and burnout [18].

Therefore, the deployment of automated and optimized team formation strategies can provide a solution to these challenges by considering a broad spectrum of factors and forming teams that optimize specific objectives, such as skills diversity, equal workload distribution, and interpersonal compatibility. Using optimization algorithms in TBL can enable educators and managers to harness computational power and handle task complexities, thereby creating teams that augment learning outcomes and enhance work productivity. This study aims to develop an optimization method for team formation, accounting for multiple criteria that impact team performance and satisfaction. The goals are to address the complexities inherent in team formation, specifically the need for balance and diversity while minimizing team conflict.

The approach aims to produce effective, balanced, and harmonious teams. The scope of this study primarily lies within the academic setting, focusing on team formation for collaborative learning initiatives such as TBL. Therefore, there is a necessity for team diversity in these settings, which spans skills, experiences, and perspectives, as it fosters creativity, innovation, and improved problem-solving [83]. Hence, one of the primary objectives of the optimization model is to maximize intra-group diversity, which encourages the formation of heterogeneous teams.

A secondary objective is to minimize inter-group diversity, ensuring equity and fairness across teams. It is crucial in an academic setting to avoid disparities that might lead to uneven competition or exclusionary practices. Since diversity can lead to potential conflicts arising from differences in personal values, attitudes, or cultural beliefs, the model also considers minimizing the conflict level to maintain smooth teamwork and collective productivity.

5.2 Literature Review

Team formation within academic settings has its own unique challenges and requirements. The shift towards active learning methodologies, such as TBL, has amplified the importance of proper team formation in educational contexts. Factors such as learner needs, academic performance, interpersonal skills, time availability, and even cultural backgrounds should carefully be considered [22]. Traditionally, instructors have often taken the reins in forming teams in academic environments, relying on their knowledge and judgment about the students' behaviors, skills, personalities, and academic performance [8]. This instructor guided approach, although personalized and adaptable, is qualitative, subjective, and tends to be time-consuming in larger classes. Moreover, these methods may overlook some intricate team dynamics, like the potential conflicts or synergies between students, or fail to balance the team in terms of diverse skills and backgrounds [82]. Furthermore, the complexity of the team formation process increases with the rise in class sizes and the diverse range of student attributes that need to be considered, highlighting the need for more sophisticated and scalable team formation methods in educational settings [37]. To enhance team formation for TBL, researchers have proposed various computational methods to overcome the limitations of traditional approaches. Early investigations in this field focused on enhancing team performance based on individual skills or expertise, utilizing models like linear and integer programming with single objective [37]. In addition, matching algorithms like the Stable Marriage, Gale-Shapley, and Hungarian algorithms have been employed, using mathematical models to balance team members' compatibility based on preferences, skills, and characteristics. Clustering algorithms like K-means clustering, hierarchical clustering, and density-based clustering aim to create teams by grouping individuals based on similarities in skills, interests, or other relevant criteria. Notably, more advanced methods such as Genetic Algorithms (GAs) have been utilized to optimize multiple team attributes like skills, preferences, and demographics [48]. Similarly, machine learning techniques have been used to suggest optimal team configurations based on individual attributes, and expert systems [34] have employed rules and heuristics based on domain knowledge to assist in team formation. However, these methods still face challenges, especially in scalability, robustness, and accommodating the complex nature of the problem, including factors like individual skillsets, potential for collaboration, and various constraints.

Following these computational strategies, a distinct approach known as the Maximum Diversity Grouping Problem (MDGP) has been explored to further enhance team formation in the context of TBL [33]. This method promotes team diversity by considering a multitude of factors. Regarded as an extension of the MDGP, the MDGP's objective is to distribute students into non-overlapping groups, thereby maximizing the sum of differences between each pair of individuals within the same group. As this problem has attracted significant attention, there have been extensive research initiatives and algorithmic solutions proposed to tackle the MDGP formulation [9]. However, the key limitations of the MDGP approach include challenges in scalability as the computational complexity grows exponentially with the number of objects and groups, and sensitivity to the quality of input data, which influences the quality of obtained solutions. Thus, while the MDGP presents a novel perspective on team formation, its limitations necessitate a more robust and scalable method that can comprehensively address the complex nature of TBL problem.

Therefore, this study proposes a novel methodology utilizing Multi-Criteria Integer Programming (MCIP) to address these research gaps and overcome the limitations inherent in current team formation studies. The innovative application of MCIP diverges from traditional and single-objective optimization methods by inherently considering multiple criteria simultaneously, providing a comprehensive solution to the intricate task of team formation. This study, presents several novel contributions to the field of TBL. First, the framework models a broad set of diverse factors including skill levels, background, and personality traits, addressing a significant research gap left by studies that only consider a few aspects of diversity. In addition, the framework's objective function is specifically designed to maximize within-team diversity while minimizing the conflict level and the variance in diversity levels between teams, offering a sophisticated approach to team formation. The methodology also includes modeling explicit constraints such as potential interpersonal conflicts, which previous studies have overlooked. Moreover, this study proposes a two-stage methodology that strategically divides the student population into sub-populations using a weighted heterogeneous multivariate K-means algorithm and optimizes team formation for these sub-populations through a surrogate optimization approach. This handles the MCIP problem effectively with large student populations, demonstrating an improvement over conventional method. Lastly, the framework includes rigorous model validation, demonstrating its efficacy with real-world scenarios with different levels of data heterogeneity, thus addressing the critical gap in the current literature that lacks robust validation. Hence, the proposed framework offers a comprehensive, flexible, and empirically validated approach to team formation.

5.3 Proposed Multi-Criteria Integer Programming Team Formation Framework

The multi-criteria integer programming team formation framework is shown in Figure 5.1. The chapter presents introduction and examination of MCIP-based framework aimed at addressing the complexities inherent to team formation.

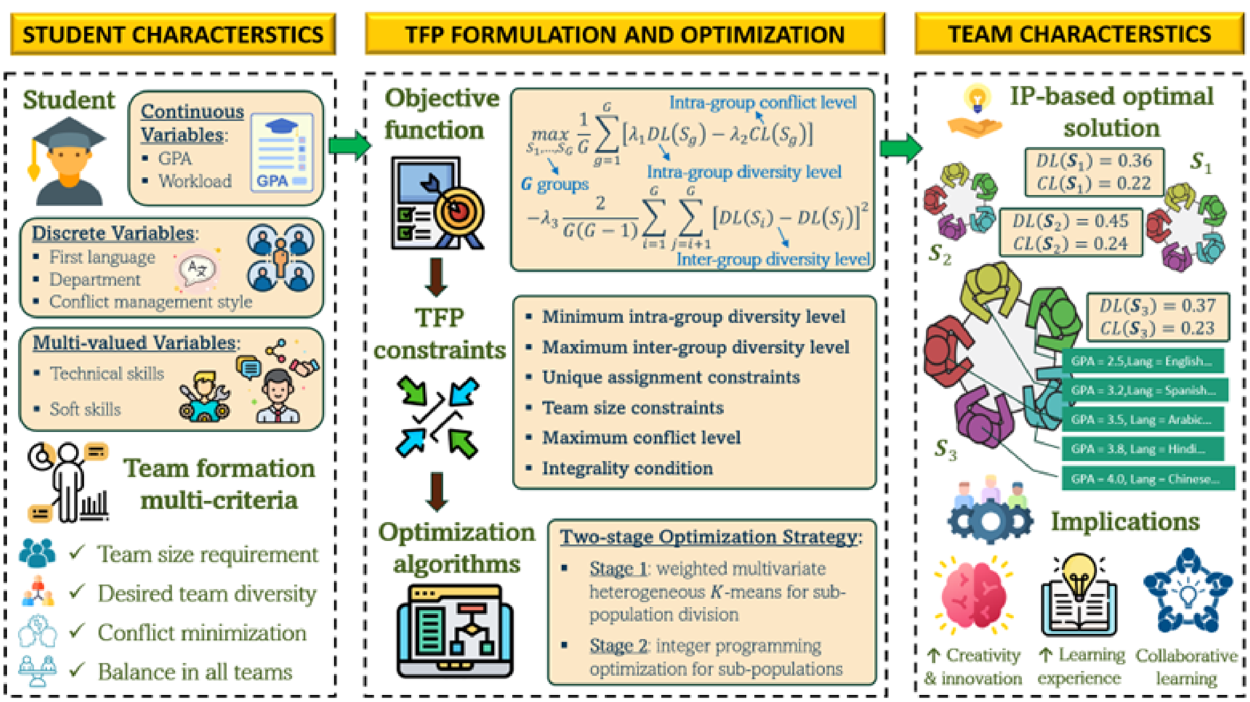


Figure 5.1 Schematic Representation of the Proposed MCIP-Based Team Formation Framework

The primary aim of the approach is to maximize the levels of intra-group diversity while minimizing the inter-group diversity and conflict levels within teams. The primary objectives of the proposed MCIP-based model are:

1. Maximizing Intra-group Diversity: the primary objective is to enhance the diversity within each team, which leads to more heterogeneous perspectives, skills, and ideas in problem-solving.

2. Minimizing Inter-group Diversity: the model aims to ensure that the variance between different groups is minimized, thus maintaining equity across teams.

3. Minimizing Conflict Level: recognizing that diversity can also give rise to potential conflicts due to differences in personal values, attitudes, cultural beliefs, etc., the model strives to minimize the level of conflict within each team.

5.3.1 Mathematical Formulation and Objective Functions

The study proposes a model based on Multi-Criteria Integer Programming (MCIP) that optimizes the assignment of N individuals to K groups. This ensures that a set of constraints are met to maintain the feasibility of team formation. Denoted are the binary decision variables $z_{ig} = 1, \forall i = 1, ..., N; g = 1, ..., G$; if the individual i is assigned to the group k, and $z_{ig} = 0$ otherwise. Therefore, the decision variables are represented by a binary matrix $Z \in \{0, 1\}^{N \times G}$. $S_g = \{s_1, ..., s_{n_g}\}$, where n_g is the total number of individuals in group g and $S_g \subseteq \{1, ..., N\}$, as the set of the individuals that belong to group g. The objective function of the proposed framework is formulated as a weighted sum of three key components: intra-group diversity level, inter-group diversity level, and intra-group conflict level. The objective function is mathematically defined as:

$$\max_{S_1,...,S_G} \frac{1}{G} \sum_{g=1}^{G} [\lambda_1 DL(S_g) - \lambda_2 CL(S_g)] - \lambda_3 \frac{2}{G(G-1)} \sum_{i=1}^{G} \sum_{j=i+1}^{G} [DL(S_i) - DL(S_j)]^2 \qquad (5.1)$$

where $DL(S_g)$ and $CL(S_g)$ denote the diversity and conflict levels of the group g respectively, and $\lambda_1, \lambda_2, \lambda_3$ are the normalized weights $(\lambda_1 + \lambda_2 + \lambda_3 = 1)$ associated with the intra-group diversity level, the conflict level, and the inter-group diversity level. Notably, this MCIP-based model can be applied to any team formation scenario that requires the balance of these components. Although the chapter later provides a focused application in student team formation as an illustrative example, the methods still hold potential for broader applications. The intra-group diversity level $DL(s_g)$ is formulated as:

$$DL(s_g) = \frac{2}{|s_g|(|s_g| - 1)} \sum_{i,j \in s_g, i<j} D_{ij} \tag{5.2}$$

where $|s_g|$ is the cardinality of the set s_k and D_{ij} is the diversity level of the individual pair i and j. This diversity level D_{ij} is computed by considering the weighted *Euclidean distance* $E(x_{c_i}, x_{c_j})$ for continuous variables x_c normalized to the range [0, 1], the weighted *Hamming distance* $H(x_{d_i}, x_{d_j})$ for discrete variables x_d, and the weighted *Jaccard distance* $J(x_{m_i}, x_{m_j})$ for multi-valued variables x_m. The notations x_{c_i}, x_{c_j}, and x_{m_i} represent the values of individual i's continuous, discrete, and multi-valued attributes, respectively. Those distance functions are:

$$E(x_{c_i}, x_{c_j}) = \sqrt{\sum_{x_c \in E_{x_c}} w_c (x_{c_i} - x_{c_j})^2} \tag{5.3}$$

$$H(x_{d_i}, x_{d_j}) = \sum_{x_d \in E_{x_d}} w_d \left| x_{d_i} - x_{d_j} \right| \tag{5.4}$$

$$H(x_{d_i}, x_{d_j}) = \sum_{x_d \in E_{x_d}} w_d \left| x_{d_i} - x_{d_j} \right| \tag{5.5}$$

$$J(x_{m_i}, x_{m_j}) = \sum_{x_m \in E_{x_m}} w_m \frac{1 - |x_{m_i} \cap x_{m_j}|}{|x_{m_i} \cup x_{m_j}|} \tag{5.6}$$

Here, $1(x_{d_i} \neq x_{d_j})$ is an indicator function that equals 1 if the discrete values of individuals i and j differ, and 0 otherwise. Next, D_{ij} is calculated as the average of these weighted distances:

$$D_{ij} = \frac{1}{3}[E(\cdot) + H(\cdot) + J(\cdot)] \tag{5.7}$$

The normalized weights w_c, w_d, and w_m represent the importance of each variable in the sets X_c, X_d, and X_m in which $\sum_{x_c \in X_c} w_c = 1$, $\sum_{x_d \in X_d} w_d = 1$, and $\sum_{x_m \in X_m} w_m = 1$. The conflict level of a group $CL(s_g)$ represents the potential for disagreement or discord within a group due to differences in attributes such as personal values, attitudes, and cultural beliefs. Similar to $DL(s_g)$, It is measured at a group level by aggregating pair-wise conflict levels among the individuals, denoted by C_{ij}, in the group:

$$CL(s_g) = \frac{2}{|s_g|(|s_g| - 1)} \sum_{i,j \in s_g, i<j} C_{ij} \tag{5.8}$$

$$C_{ij} = \frac{1}{3}[E(\cdot) + H(\cdot) + J(\cdot)] \tag{5.9}$$

where Y_c , Y_d , and Y_m represent the sets of continuous, discrete, and multi-valued conflict attributes respectively, w_{y_c}, w_{y_d}, and w_{y_m} are the weights associated with each variable in the sets Y_c , Y_d , and Y_m reflecting the conflict potential of the respective variables. The normalization constraints for weights are:

$$\sum_{c \in C} w_c y_c = 1, \sum_{d \in D} w_d y_d = 1, \text{ and } \sum_{m \in M} w_m y_m = 1. \tag{5.10}$$

5.3.2 Constraints

The constraints of the proposed IP-based framework are intended to ensure the feasibility of the team formation and to align the teams with specific requirements and preferences. The constraints are as follows:

1. Minimum intra-group diversity level: Each team should have a diversity level of at least $e_{\min}^{\text{intra}}$. Mathematically, this can be represented as:

$$DL(S_g) > e_{\min}^{\text{intra}}, \forall g = 1, ..., G. \tag{5.11}$$

2. Maximum inter-group diversity level: The squared difference in diversity levels between any two groups should be at most difference in diversity levels between any two groups should be at most θ_{max}. This maintains a balance in the diversity levels across all teams, thus ensuring fairness and equality in team composition. This is expressed as:

$$DL(S_i) - DL(S_j)]^2 \leq \theta_{max}, \forall i, j \in \{1, ..., G\} \tag{5.12}$$

3. Unique assignment constraint: Every student must be assigned to exactly one team. This is represented as:

$$\sum_{g=1}^{G} Z_{i,g} = 1, \forall i \in \{1, ..., N\} \quad (11) \tag{5.13}$$

4. Team size constraints: The number of students in each team should be within a specified range. This ensures that no team is too large or too small, which allows for effective collaboration and responsibility among team members. This is expressed as:

$$\omega_{min} \leq \sum_{i=1}^{N} Z_{i,g} = 1 \leq \omega_{max}, \forall g \in \{1, ..., G\} \tag{5.14}$$

5. Maximum conflict level: The conflict level within each team should not exceed ξ_{max}, which is given as:

$$CL(S_g) \leq \xi_{max}, \forall g \in \{1, ..., G\} \tag{5.15}$$

6. Integrality condition: The decision variables $Z_{i,g}$ are binary variables, which implies:

$$Z_{i,g} \in \{0, 1\}, \forall i \in \{1, ..., N\}, \forall g \in \{1, ..., G\} \tag{5.16}$$

5.3.3 Two-stage Optimization Strategy

The two-stage methodology strategically divides the student population into sub-populations by using a weighted heterogeneous multivariate K-means algorithm and optimizes the team formation for these sub-populations by a surrogate optimization method.

1. Student Sub-Population Division using Weighted Multivariate Heterogeneous K-Means: The first stage employs a division-and-conquer strategy to break down the large-scale TFP into smaller, more tractable sub-problems, i.e., individual student sub-populations. This approach respects the goal of maximizing intra-group diversity while providing a more structured and less complex set of problems to solve. The basis of this strategy is the "weighted multivariate heterogeneous K-means (WMH K-means)" algorithm, an advanced version of the traditional K-means algorithm. The algorithm considers the weighted combination of different variable types, allowing us to appropriately consider and incorporate the diverse attributes of the students in the clustering process. Upon convergence, the algorithm forms K clusters of students, where each student is assigned to the cluster that minimizes the diversity level of their attributes. Upon convergence, round-robin sampling creates balanced N_p sub-populations $\{P_k\}_{k=1}^{N_p}$ with sizes of N_{sub}. The resulting balanced sub-populations are optimal for the next stage of team formation optimization.

2. Surrogate Optimization Method for Sub-Population Team Formation: Following the division stage, each sub-population $P_p, p \in \{1, ..., N_p\}$ is ready for the subsequent optimization process. This stage is implemented using a Surrogate Optimization (SO) method, which is effective when the objective function evaluation is time-consuming.

The SO algorithm capitalizes on creating an approximation (surrogate) of the original problem and minimizing the surrogate within predefined bounds. The algorithm repeatedly generates trial points, constructs a surrogate model, finds an adaptive point, and updates the surrogate based on the obtained results, driving the search towards the global minimum of the problem. This iterative process continues until all trial points are within a specified minimum distance from the evaluated points, thereby providing a solution that is both efficient and effective for the TFP. Therefore, applying this method to the MCIP problem enables us to efficiently navigate the solution space and optimize the formation of diverse and balanced student teams.

To solve the optimization problem, mathematical optimization solver is employed. The solver takes the MCIP-based model and applies numerical techniques to identify the optimal or near-optimal solution. There are several commercial and open-source solvers available such as Gurobi, CPLEX, SCIP, and MATLAB. MATLAB's Optimization Toolbox provides functions for finding parameters that minimize objectives while satisfying constraints. The solver selection would largely depend on the size and complexity of the problem, and the available resources. However, surrogate optimization is just one approach in a wide array of potential methods for solving this kind of problem. Alternative methodologies could be more suitable depending on the specifics of the problem and the resources. Metaheuristic techniques such as Genetic Algorithms (GA), Simulated Annealing (SA), or Particle Swarm Optimization (PSO) could provide better solutions when dealing with different kinds of constraints or objectives. These methods operate on different principles and may offer superior performance in different problem contexts.

5.3.4 Data Simulation and Validation

The effectiveness of the surrogate optimization approach for the MCIP-based team formation method is validated by running a thorough simulation and validation process. Detailed below are steps for simulation setup and data simulation:

1. Simulation setup: The setup of the simulation involves creating variables that are representative of a diverse set of scenarios, with each scenario having a unique heterogeneity level. Two key metrics are used to control these heterogeneity levels: the *Coefficient of Variation (CV)* for continuous variables, and the *Simpson's Diversity Index (SDI)* for discrete and multi-valued variables. In this context, CV_x measures the relative variability in the continuous attributes, which is calculated as the ratio of the standard deviation σ_x to the mean μ_x of the student attribute x:

$$CV_x = \frac{\sigma_x}{\mu_x}, \quad \forall x \in X_c \cup Y_c \tag{5.17}$$

   Conversely, SDI_d quantifies the diversity in discrete and multi-valued variables d, which is defined as:

$$SDI_d = 1 - \sum_{c=1}^{C} p_{d,c}^2, \quad \forall d \in X_d \cup X_m \cup Y_d \cup Y_m \tag{5.18}$$

   where $p_{d,c}$ represents the proportion of students whose variables taking the value c, and C is the number of variable categories. In the next sections, data generation process is explained and it aligns with these metrics and how the simulated data is used for validation.

2. Data simulation

   To create a simulation scenario that closely mimics the reality of team formations, data is generated that satisfies the pre-specified CV and SDI values. For continuous variables $x \in X_c \cup Y_c$, presuppose non-negativity and follow a Gamma-distribution, chosen for its ability to closely represent the distribution of attributes like skill levels and experience in a team context. With a pre-specified CV_x, it determines the mean μ_x and standard deviation σ_x of this distribution. If μ_x is a chosen constant, then the standard deviation σ_x can be computed as: $\sigma_x = CV_x \cdot \mu_x$. To generate the

Gamma-distributed random variable, the shape parameter k_x and scale parameters θ_x are derived from CV_x as follows:

$$k_x = \frac{1}{CV_x^2}, \quad \theta_x = \mu_x CV_x^2 \tag{5.19}$$

Subsequently, calculated parameters can be used k_x and θ_x to generate the Gamma-distributed random variable, which ensures that the simulated data fits the prespecified CV_x and reflects the desired variability in team characteristics. For discrete and multi-valued variables, such as the first language and technical skills of the students, a modified stick-breaking process to simulate these attributes is employed. This method involves binary encoding of the variables, thereby creating additional binary variables for each possible value. Through M iterative simulations, it generates diverse sets of discrete and multi-valued variables, each assessed for its alignment with the target SDI_d. The simulation yielding an SDI_{opt} closest to the target is selected for use in the experiments. This process not only ensures data variability but also replicates the complexity and diversity found in real team formations. Utilizing the simulated data, the experiments will be meticulously designed to evaluate the effectiveness of the MCIP-based optimization in forming teams. The framework is applied to the generated data to closely examine the impact of various configurations and criteria on team composition and performance. The experimental design is structured to highlight the model's capability to handle varying levels of diversity and complexity, reflecting real-world team formation challenges. This direct application of simulated data enables a comprehensive analysis of the model's performance, offering insights into its practical implications and scalability in diverse settings.

3. The proposed MCIP-based TFP methods are validated using datasets, which involves applying these team formation methods to problem instances derived from simulated datasets. These problem instances are designed to reflect a wide range of realistic

scenarios, governed by three main control parameters: SDI for discrete and multi-valued variables, CV for continuous variables, and the total number of students N. To implement the methods, the simulated datasets are set as input into the MCIP-based team formation model. The model's constraints are set up to represent the unique characteristics of each problem instance. Subsequently, the MCIP-based algorithm is executed for each problem instance. The algorithm processes the defined attributes and constraints of each instance, including aspects such as team size and diversity level, and seeks to find an optimal solution. The output of this execution step is an optimal or near-optimal solution for each problem instance. These solutions represent the most effective distribution of students into teams, adhering to the constraints and objectives outlined in the problem instance. Finally, an evaluation process is undertaken to assess the performance of the MCIP-based method. This assessment focuses on the quality of the solutions, i.e., team formation outcomes derived from the MCIP-based approach, which examines the qualities of team compositions, how well the solutions satisfy the team formation criteria, and the desired diversity levels. In addition, the computational efficiency and robustness analysis are performed to investigate how well the MCIP-based approach handles variations in the scenarios. The proposed method is also benchmarked against other well-established team formation methods.

5.4 Results

This section presents the results in three distinct parts. First, it explored the data simulation process that generates scenarios mimicking realistic team formations. Second, it validated the MCIP-based team formation model using the simulated data by assessing the model's ability and the derived solution to facilitate the diverse skill sets and individual attributes while minimizing the potential conflicts. Third, the comparative performance analysis was performed to benchmark the model against other well-established team formation methods.

5.4.1 Data Simulation and Visualization

To ensure that the simulated data accurately reflects the complexity and diversity of real-world scenarios, the model takes into account various student attributes, including continuous, discrete, and multi-valued variables. In the simulation, two values of N, N = 52 and N = 104, are used to simulate small and large class scenarios. The detailed variables and their parameter settings, are designed to closely mimic real-world distributions and diversity. The simulated data for three representative variables: GPA (continuous), first language (discrete), and technical skills (multi-valued), under varying degrees of heterogeneity and N = 104 are visualized in Figure 5.2. By utilizing this simulated data, the subsequent experiments are designed to assess the MCIP-based optimization's efficacy in creating teams. By applying the framework to the generated datasets, the study critically analyzes the impact of various configurations and criteria on team composition and performance. The experimental design, anchored in the simulated scenarios, illuminates the model's capacity to address real-world team formation challenges, showcasing its versatility and potential for wide-ranging applications.

In Figure 5.2, as CV increased, GPA distribution widened, indicating greater heterogeneity. Language diversity rose from being predominantly single-language (SDI = 0.2) to balanced (SDI = 0.5) and highly varied (SDI = 0.8). Technical skills also reflected a similar trend, moving from low diversity to a diverse distribution at SDI = 0.8.

5.4.2 Validation of the MCIP Model Using Simulated Data

In the critical phase of validating the MCIP-based team formation model, the process meticulously analyzes simulated data that was generated to reflect a range of real-world team formation scenarios. Parameter settings for the MCIP model, tailored for both small (N = 52) and larger (N = 104) simulated student populations. It first segmented the simulated student populations into manageable sub-populations. This segmentation employed the WMH K-means approach, which facilitated a nuanced division of the student population based on attribute heterogeneity, effectively creating clusters that mirror the diversity and complexity found in real educational settings. Figure 5.3 illustrates the distribution of student data points across various sub-populations at different levels of attribute heterogeneity for both N = 52 and N = 104 scenarios.

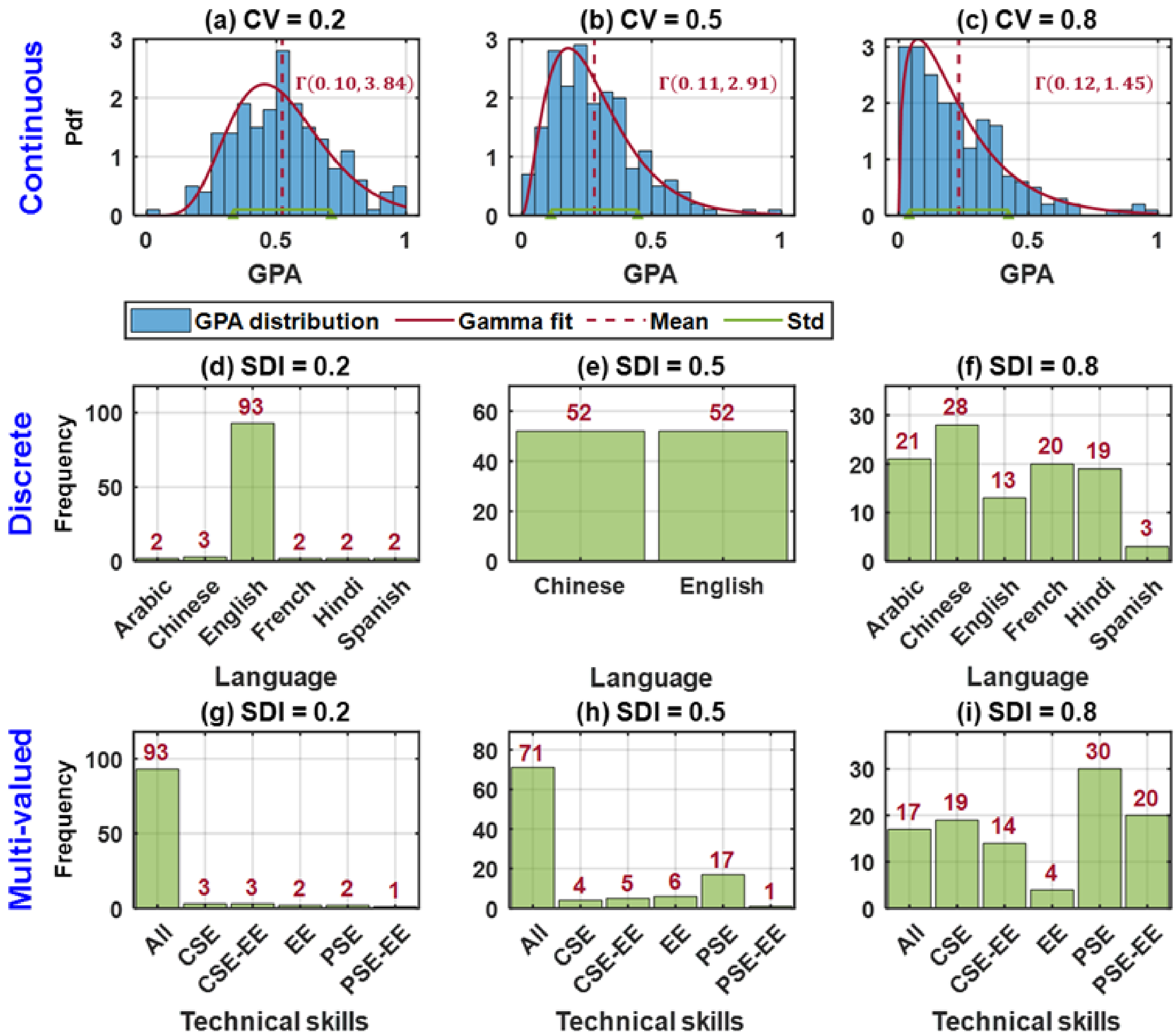


Figure 5.2 Visualization of the Simulated Data for Three Representative Variables

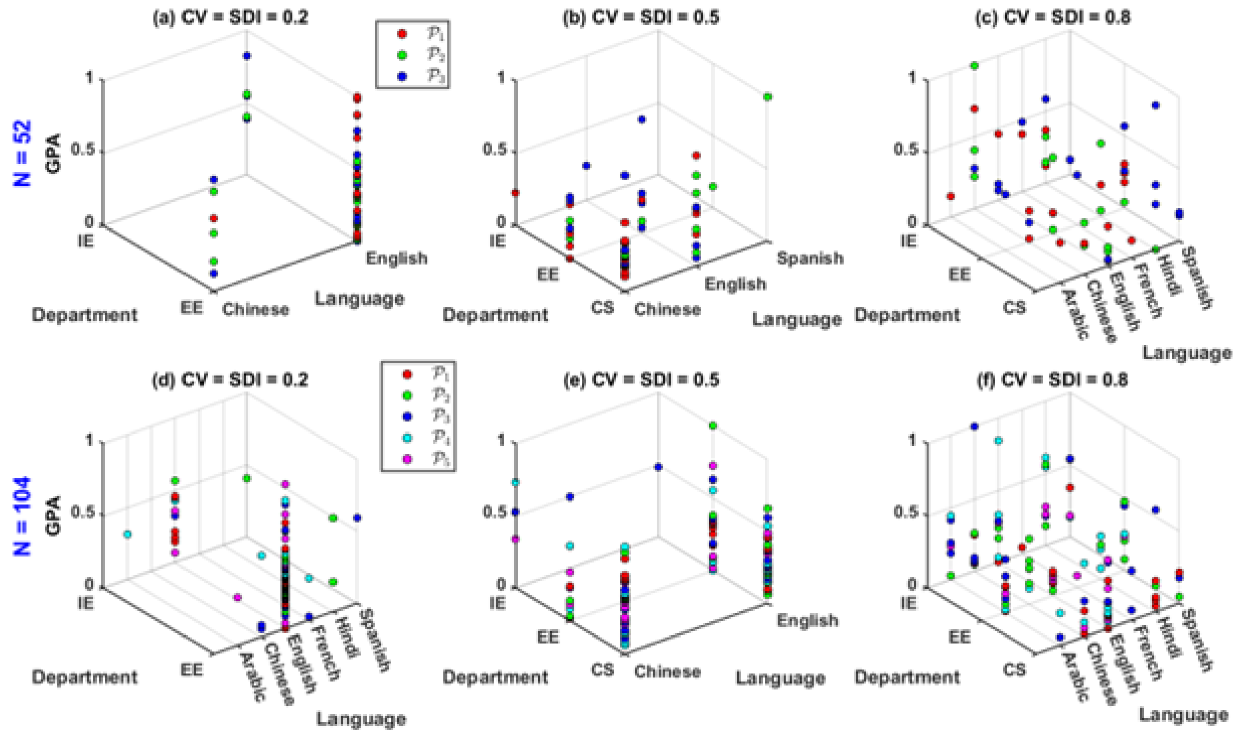


Figure 5.3 Visualization of Sub-Population Division Using WMH K-means for Different Heterogeneity Levels

For N = 52, the Figure 5.3 visualizes the sub-population divided into three distinct sub-populations, as shown in panels (a), (b), and (c). For the lower heterogeneity level of CV = SDI = 0.2 (panel a), the sub-populations appears closely grouped, indicating less diversity within sub-populations. As the heterogeneity level increases to 0.5 and then 0.8 (panels b and c), the sub-populations become increasingly scattered, illustrating a rise in internal diversity. For a larger student population size of N = 104, it partitioned into five distinct sub-populations, represented in panels (d), (e), and (f). The increasing spread of the sub-populations in these panels, correlating with the rise in heterogeneity levels (CV = SDI = 0.2, 0.5, and 0.8), confirms that the larger group has more diverse students, demanding more nuanced and careful sub-population divisions. Here, the success of the division is evident in the balanced and widespread distribution of sub-populations across various levels of heterogeneity. This effective segmentation into diverse sub-populations demonstrates the efficiency of the Weighted Multivariate Heterogeneous (WMH) K-means approach in managing student data of different sizes and diverse characteristics. This sets the stage for applying the MCIP-based team formation model effectively. With the student populations effectively segregated into well-diversified sub-populations, the next step is to facilitate the formation of effective teams within these sub-groups. This involves deploying a two-stage optimization approach that combines the WMH K-means method and the surrogate optimization method, shown in Figure 5.4.

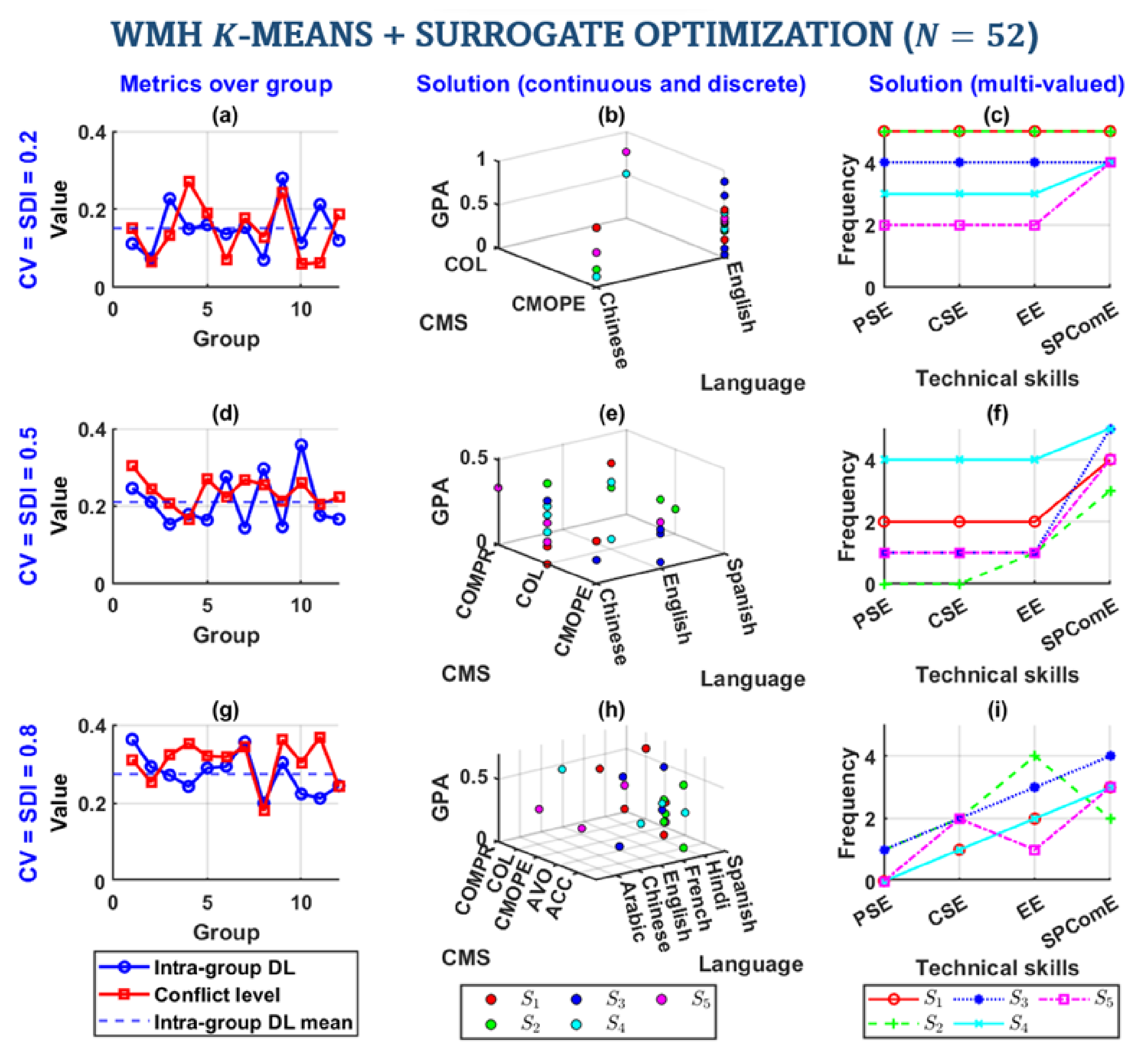


Figure 5.4 Implementation of the Two-Stage Optimization Methods

The two-stage optimization method results for three heterogeneity levels: CV = SDI = 0.2, 0.5, and 0.8, are represented by panels (a)-(c), (d)-(f), and (g)-(i) respectively. In panel (a) with the lowest heterogeneity level (CV = SDI = 0.2), the intra-group DLs ranged from 0.07 to 0.28 with a mean of 0.1512, while the conflict level spanned from 0.06 to 0.27. As the heterogeneity level increased to 0.5 (panel d), the intra-group DL values ranged from 0.144 to 0.358 with a mean of 0.2099, and the conflict level varied between 0.167 and 0.304. At the highest heterogeneity level (CV = SDI = 0.8), panel (g) revealed an even wider spread in the intra-group DL, from 0.199 to 0.362, and a mean DL of 0.2742, along with a conflict level varying between 0.179 and 0.368. This increasing spread with higher heterogeneity levels indicates that the optimization approach successfully ensures a high diversity within groups, even under different diversity levels of the student population. Panels (b), (e), and (h) highlighted the shift in group compositions as the heterogeneity levels changed from CV = SDI = 0.2, 0.5, to 0.8. At the lower heterogeneity level of CV = SDI = 0.2 (panel b), the grouping was relatively uniform, indicating that students with similar GPAs, language proficiency, and conflict management styles are likely to be grouped together. However, as it moved to higher heterogeneity levels, the solutions for these variables displayed a greater degree of variation within groups, as shown in panels (e) and (h). Panels (c), (f), and (i) provided a depiction of the solutions for the multi-valued variable of technical skills for five representative groups. The distribution of technical skills within each group varied significantly with the changes in heterogeneity level, a change clearly demonstrated as it moves from panel (c) to (i). For the lower heterogeneity level of CV = SDI = 0.2 (panel c), the distribution of technical skills within groups was more homogeneous, suggesting that students with similar skillsets are likely to be in the same group. However, at higher heterogeneity levels, the distribution of technical skills within each group was more diverse. Following the results of the two-stage optimization method, the random search approach was implemented for the same student population of N = 52. The comparisons between the random search method and the optimization method across various heterogeneity levels (CV

= SDI = 0.2, 0.5, and 0.8) indicated a slightly superior performance of the random search in terms of producing higher intra-group diversity and lower conflict levels, see Figure 5.4.

5.4.3 Comparative Performance Analysis

This section intends to evaluate the performance of the two-stage optimization method against established team formation models. The assessment was carried out using key performance indicators such as intra-group diversity level (DL), conflict level, inter-group diversity level, computation time, and the optimality gap. The optimality gap indicates how closely the benchmarking models approximate the best possible team formation solution. Particularly, the best-known solution derived from the most effective algorithm, i.e., the random search was used to estimate the optimality gap, which is the percentage difference between the best-known solution and the solution obtained from other algorithms. The performance comparison included 2 student population sizes, N = 52 and N = 104 as presented in Table 5.1. Table 5.1, presents a performance comparison of the two-stage optimization method (WHM K-means + SO) against the three well-established team formation algorithms (random search, WHM K-means, WHM K-means + genetic algorithm (GA)) serving as the benchmark to evaluate the efficacy of the proposed model. The third algorithm, WHM K-means + GA, replaces the surrogate optimization with the genetic algorithm, which is known for its capability in solving complex optimization problems, in the 2nd stage of the two-stage optimization method.

According to Table 5.1, the random search method yielded the most optimal solutions for both population sizes. However, this came at the cost of longer computation times, specifically 5028.737 $\pm$ 226.862 (secs) and 7412.777 $\pm$ 4966.39 (secs) for N = 52 and N = 104 respectively, indicating a low computational efficiency as the student population sizes increased.

Additionally, the random search approach resulted in intra-group DLs of 0.216 $\pm$ 0.065 for N = 52 and 0.204 $\pm$ 0.067 for N = 104, with corresponding conflict levels of

Table 5.1 Comparative Performance Analysis of Three Team Formation Algorithms and the Proposed Algorithm

Algorithm	**N**	**Intra-group DL**	**Conflict Level**	**Inter-group DL**	**Elapsed Time (sec)**	**Optimality Gap (%)**
Random Search	5 2 1 04	0.216, ± 0.065, 0.204, ± 0.067	0.141, ± 0.063 ,0.135, ± 0.064	0.007, ± 0.003 ,0.004, ± 0.001	5028.737, ± 226.862, 7412.777, ± 4966.39	0, ± 0, 0 ,± 0
WHM K-means	5 2 1 04	0.203, ± 0.062, 0.199, ± 0.071	0.203, ± 0.073, 0.198, ± 0.074	0.007, ± 0.003, 0.003, ± 0.001	0.801, ± 0.253, 5.083, ± 2.513	43.730, ± 3.626, 36.183, ± 5.36
WHM K-means + GA	5 2 1 04	0.204, ± 0.055, 0.194, ± 0.073	0.206, ± 0.069, 0.192, ± 0.069	0.007, ± 0.003, 0.004, ± 0.001	960.307, ± 314.207, 1376.06, ± 435.77	27.761, ± 2.97, 31.192, ± 5.079
WHM K-means + SO	5 2 1 04	0.203, ± 0.055, 0.192, ± 0.064	0.207, ± 0.057, 0.191, ± 0.062	0.007, ± 0.002, 0.004 ± 0.001	505.106, ± 66.749 ,902.833, ± 29.532	29.671, ± 5.076, 29.619, ± 3.27

0.141 ± 0.063 and 0.135 ± 0.064, respectively. This method provided minimal inter-group DLs (0.007 ± 0.003 for N = 52 and 0.004 ± 0.001 for N = 104), suggesting it can effectively balance the team diversity across different groups. In contrast, the WHM K-means algorithm reduced computation times to only 0.801 ± 0.253 (secs) for N = 52 and 5.083 ± 2.513 (secs) for N = 104, indicating superior computational speed. However, this was at the expense of an increased optimality gap — 43.730 ± 3.626% for N = 52 and 36.183 ± 5.36% for N = 104, implying a trade-off between computational speed and solution quality. This method yielded intra-group DLs of 0.203 ± 0.062 for N = 52 and 0.199 ± 0.071 for N = 104, and conflict levels of 0.203 ± 0.073 and 0.198 ± 0.074, respectively, indicating a slight increase in conflicts compared to the Random Search approach. The inter-group DLs remained low (0.007 ± 0.003 for N = 52 and 0.003 ± 0.001 for N = 104), demonstrating a robust capability to ensure balanced team diversity across groups, which was similar to the random search method.

The hybrid model, WHM K-means + GA, yielded an encouraging balance between computation time and solution quality. The optimality gap was significantly reduced compared to WHM K-means (to 27.761 ± 2.97% for N = 52 and 31.192 ± 5.079% for N = 104), suggesting that the hybrid model achieved superior solution quality. However, the algorithm did increase computation time compared to WHM K-means, but still maintained significantly lower computation times than Random Search, with average times of 960.307 ± 314.207 (secs) for N = 52 and 1376.06 ± 435.77 (secs) for N = 104, reflecting a great balance between solution quality and computational time. Interestingly, the intra-group DLs are 0.204 ± 0.055 for N = 52 and 0.194 ± 0.073 for N = 104, while the conflict levels are 0.206 ± 0.06 for N = 52 and 0.192 ± 0.069 for N = 104, implying slight improvements in team homogeneity and conflict management compared to the standalone WHM K-means. Similar to previous models, the hybrid model maintained low inter-group DLs (0.007 ± 0.003 for N = 52 and 0.004 ± 0.001 for N = 104), demonstrating its capability to foster balanced team diversity across different groups.

Finally, the proposed model, WHM K-means + SO, delivered a strong performance. It outperformed WHM K-means in terms of the optimality gap, reducing it to 29.671 ± 5.076% for N = 52 and 29.619 ± 3.27% for N = 104, thereby affirming the superior quality of its solutions. It also had significantly lower computation times than random search and WHM K-means + GA, clocking in at 505.106 ± 66.749 (secs) and 902.833 ± 29.532 (secs) for N = 52 and N = 104, respectively, underscoring its computational efficiency. This model reported intra-group DLs of 0.203 ± 0.055 for N = 52 and 0.192 ± 0.064 for N = 104, and conflict levels of 0.207 ± 0.057 and 0.191 ± 0.062, respectively. It maintained low inter-group DLs (0.007 ± 0.002 for N = 52 and 0.004 ± 0.001 for N = 104), supporting its capability to guarantee balanced diversity across teams. The detailed evaluations of all scenarios with different CV and SDI values for N = 52 and N = 104 are reported. An increased CV led to a higher intra-group diversity level and conflict level. This implies that, as the variability of continuous attributes within a team increases, the team becomes more diverse and tends to experience more internal conflicts. The impact of SDI was less consistent. Though an increased SDI led to a rise in conflict level, indicating a higher disagreement level as the diversity in variables increased, its influence on the intra-group DL was inconsistent, which suggested a more complex interaction between these factors. With regards to algorithm performance, the random search method, despite yielding optimal results, consistently had the longest elapsed times across all CV and SDI values. The proposed methods, WHMKM + GA and WHMKM + SO, offered a more balanced performance in terms of time efficiency and result optimality.

5.5 GUI Implementation of the MCIP Model

The Multi-Criteria Team Formation Model GUI is developed to offer a user-friendly Graphical User Interface (GUI) that implements the Multi-Criteria Integer Programming (MCIP) model, see Figure 5.5.

The development of the Multi-Criteria Integer Programming (MCIP) model GUI represents an interface structure and functionality significant for educational team formation:

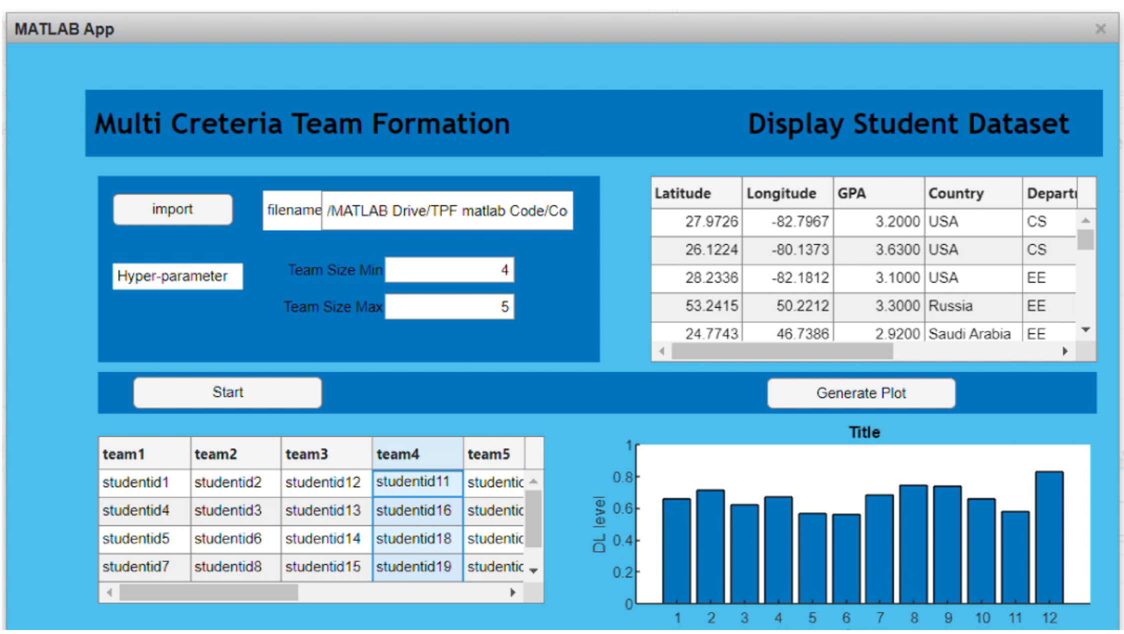


Figure 5.5 MATLAB GUI for Multi-Criteria Team Formation

1. Functionality: the system facilitates data importation, allowing the user to load a comprehensive dataset containing student profiles, including academic, geographic, and departmental information, directly from a MATLAB Drive path.

2. Hyper-parameter Customization: the developer incorporated adjustable hyper-parameters for team size constraints, enabling educators to tailor the range of team sizes in alignment with pedagogical requirements.

3. Algorithmic Processing: the activation of the 'Start' button, allows to sort and match students into teams based on the inputted multi-criteria framework.

4. Visualization of Teams: in the interface's display area, the resulting team configurations are made visible, providing algorithm's output and allowing for any necessary manual adjustments.

5. Analytical Plot Generation: a key feature of the GUI is the 'Generate Plot' function, which outputs a graphical representation of the diversity level (denoted as 'DL Level') across teams, serving as an analytical tool for assessing the effectiveness of the team formation process.

6. Usability and Educational Impact: the GUI is designed to eliminate bias and ensure balanced team formation, thus enhancing the collaborative educational environment. The tool represents a fusion of algorithmic precision and educational methodology, offering a sophisticated and accessible solution for team formation.

MCIP model GUI marks a noteworthy contribution to educational resources, providing a robust tool for optimizing team composition and dynamics. The GUI's balance of complexity and ease of use exemplifies the potential for technology to transform educational practices, offering a valuable resource for educators seeking data-driven solutions for team formation.

5.6 Discussion

This research utilized a novel two-stage optimization approach tailored for the intricate process of forming teams within heterogeneous environments, demonstrating a remarkable balance between computational efficiency and the quality of solutions. The model's effectiveness, validated across a spectrum of heterogeneity levels, showcases its ability to cultivate teams marked by elevated intra-group diversity and minimized conflict levels. This approach not only surpasses traditional methods such as random search, WHM K-means, and the hybrid WHM K-means with GA in computational speed and the narrowness of the optimality gap but also introduces a superior strategy for assembling teams. The model's success in achieving high diversity within teams while maintaining low conflict levels, especially in contrast to existing team formation strategies, underscores its potential to significantly enhance team dynamics and effectiveness. By efficiently navigating the complexities associated with diverse team compositions, this two-stage optimization method presents a groundbreaking solution to the challenges of team formation, promising substantial improvements in both educational and professional settings.

The study demonstrates the robustness, versatility, and computational efficiency of the two-stage optimization model across varying heterogeneity levels. Despite the increased problem complexity with rising heterogeneity, the model adeptly ensures diversity while maintaining low conflict levels, thus optimizing team harmony. While the random search method offers optimal results, it lacks computational speed, rendering it impractical for real-time applications. The WHM K-means model, although efficient, struggled to balance diversity and conflict. Its hybrid version, WHM K-means + GA, has demonstrated improvements, however still underperformed relative to the WHM K-means model in solution quality and computational efficiency. The study highlights the model's unique ability to maintain high diversity and low conflict levels within teams, even as the complexity of the problem escalates a testament to its advanced algorithmic design. This two-stage optimization model thereby redefines the landscape of team formation strategies, promising to significantly improve collaborative dynamics and outcomes by leveraging its sophisticated balance of diversity, conflict management, and operational efficiency.

The study has several limitations which, in turn, open avenues for future research in TBL. Firstly, the MCIP-based algorithm's effectiveness is heavily reliant on the quality and comprehensiveness of input data. Although the model managed to utilize several diverse data points, the inclusion of more detailed and varied information, such as cultural backgrounds, language proficiency, or previous project experiences, could potentially enhance the team formation process even further. Secondly, while the proposed framework has proven successful under the conditions tested, its performance in real-world domains and applications, such as corporate environments, non-profit teams, or online collaboration platforms, needs to be extensively evaluated. Thirdly, despite the model aiming to create balanced and diverse teams, it does not consider the dynamics of team interactions after the team assignment. Understanding how these teams function and adapt over time is a crucial aspect of the team formation process. Lastly, the algorithm, while efficient, is computationally intensive, especially for large-scale applications. This presents an interesting area for future research, where efforts are directed toward improving computational efficiency while maintaining or even enhancing the algorithm's accuracy and robustness. This could involve the exploration of parallel processing or the use of more efficient data structures and algorithms.

5.7 Conclusion

This study introduces a pioneering two-stage optimization strategy for team formation, combining WHM K-means with surrogate optimization tailored for Team-Based Learning (TBL). The approach effectively creates diverse teams with minimal conflict and has proven superior to traditional methods in both team performance and satisfaction. Extensive numerical analysis validates this methodology, setting new standards for team formation practices by successfully navigating the complexities of assembling balanced teams that foster innovation and collaboration.

While acknowledging limitations, the study suggests future research directions such as data enrichment and real-world application testing. This research advocates for the ongoing evolution of team formation strategies across various fields, including education and human resource management, to enhance productivity and job satisfaction in our digital, interconnected world.

Chapter 6: MBSE Model System Verification & Result Analysis

System verification involves system simulation to detect design flaws early and test various scenarios, ensuring comprehensive analysis and risk reduction before real-world deployment. This chapter includes System Simulation, Use Case Scenarios, and Team Formation Algorithm: Development and Testing for Engineering Courses.

6.1 System Simulation

System simulation involves creating computer models that mimic real-world systems. These models enable the study and analysis of complex systems without directly interacting with the physical system. Within the context of the MBSE framework for the ELAS model, system simulation plays a crucial role by simulating two distinct use-case scenarios: first, equipping students with advanced semiconductor skills through a Mechatronics, Robotics, and Control Laboratory tailored to the needs of the semiconductor industry; second, developing professional skills in students through various course assignments aimed at assessing soft skills.

Following Section 3.2, the system architecture of the ELAS model, two use case models were created, which are:

1. Equipping Students with Advanced Semiconductor Skills through Mechatronics, Robotics, and Control Laboratory tailored to Semiconductor Industry Needs.

2. Emphasizing student soft skills development through various assessment methods.

6.1.1 System Simulation - First Use Case Scenarios

The first scenario involves equipping students with advanced semiconductor skills through courses in Mechatronics, Robotics, and Control Laboratory, tailored to meet the needs of the semiconductor industry. This use case is motivated by recent initiatives and workforce demands in the semiconductor sector [54]. Building on the technical skills identified by stakeholder needs, a stakeholder requirements diagram for the Agile Design of the Mechatronics, Robotics, and Control (MRC) Lab was developed, as shown in Figure 6.1.

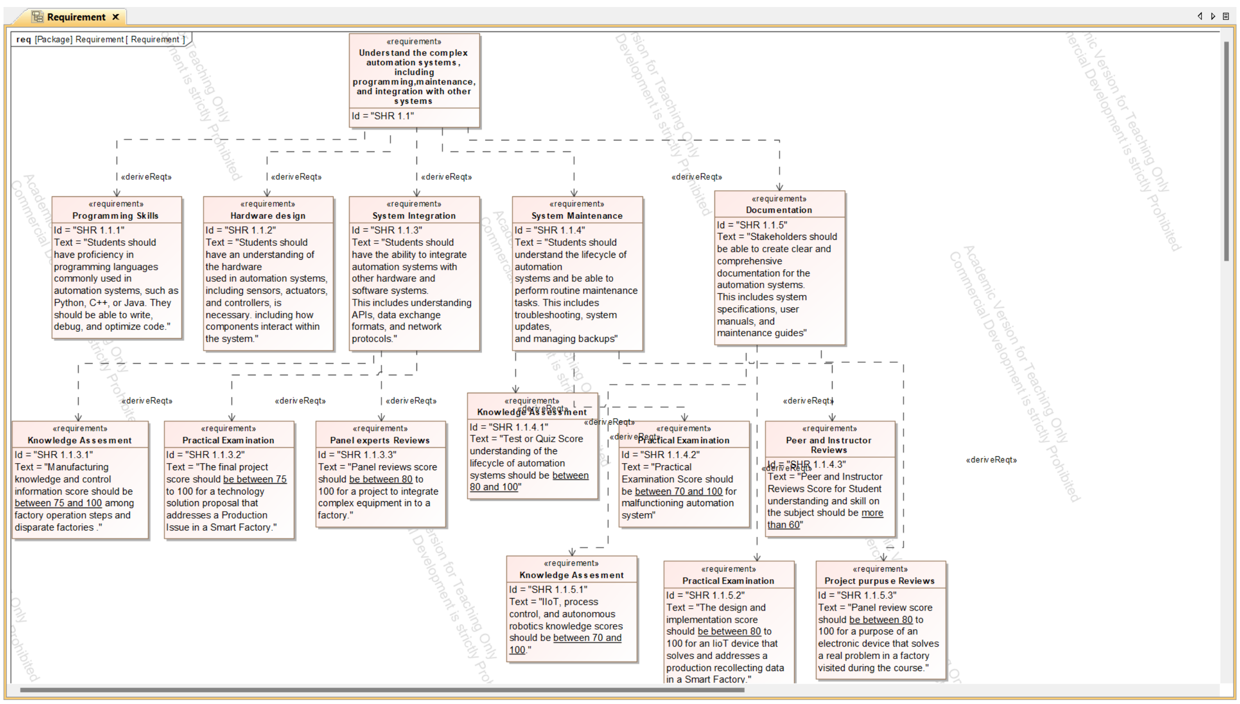


Figure 6.1 Use Case Scenario System Requirements

Based on the use case system requirements diagram, the system architecture was constructed to map different requirements with their respective functional blocks for verification, as depicted in Figure 6.2. Furthermore, to verify the use case system requirements, requirements verification and validation for various scenarios were conducted.

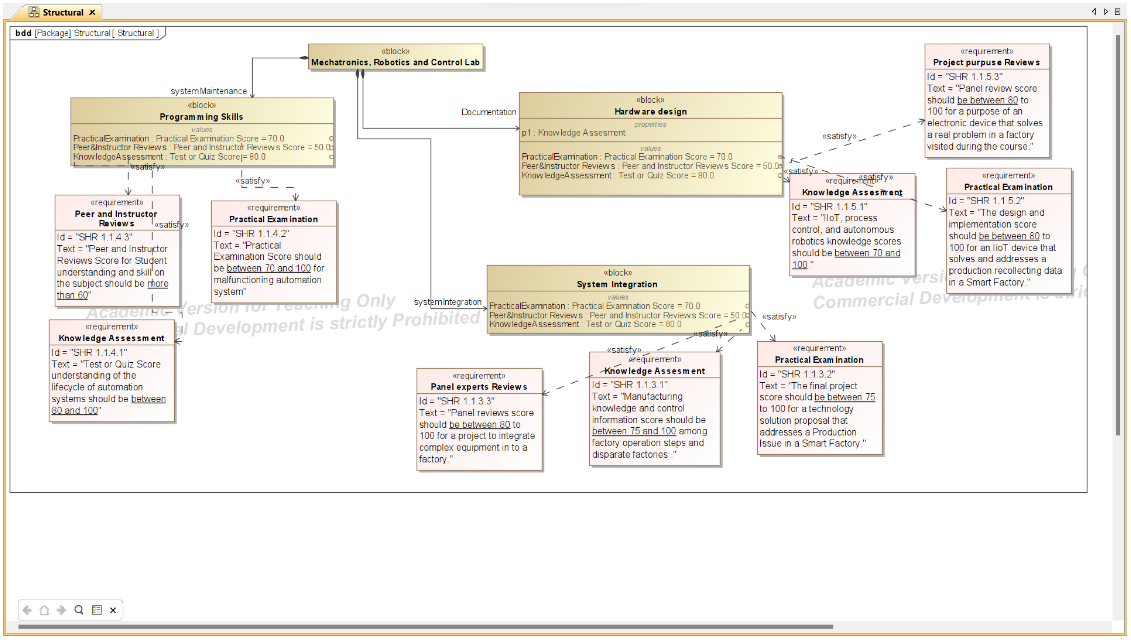
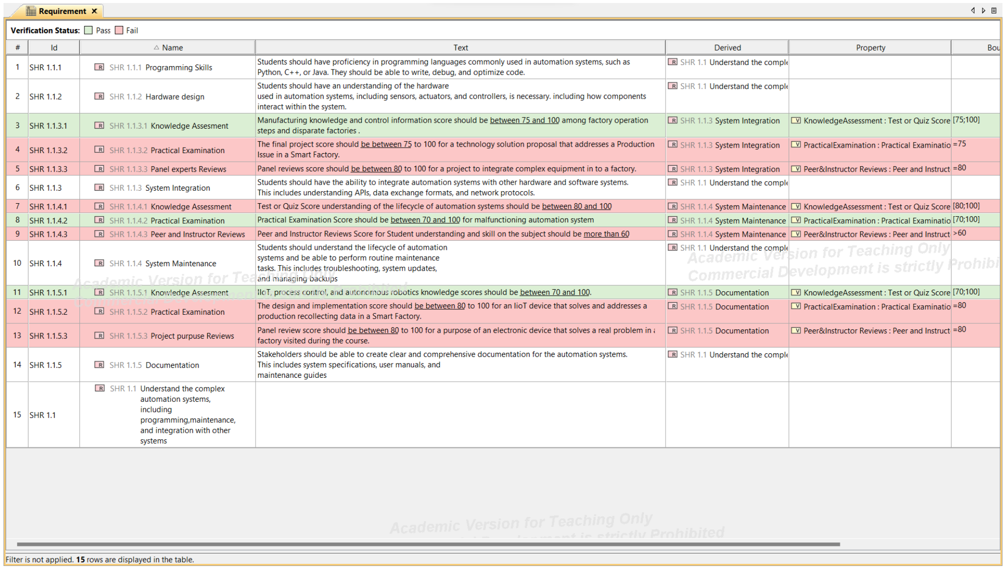


Figure 6.2 Use Case Scenario System Architecture

These simulations are based on the laboratory's learning objectives and goals, which align directly with stakeholder requirements. Figure 6.3 illustrates the results from the requirement verification and validation of Use Case Scenario at Different Environmental Variables.

6.1.2 System Simulation - Second Use Case Scenario

The second scenario focuses on enhancing student soft skills development through a variety of course assignments. This use case stems from the need to improve professional competencies among engineering students via strategic team formation. Following the team formation model outlined in Chapter 4, this scenario was implemented in an electrical engineering course, where various course assignments aimed at evaluating professional skills were developed.



Requirement

Verification Status: Pass Fail

#	Id	Name	Text	Derived	Property	Bou
1	SHR 1.1.1	SHR 1.1.1 Programming Skills	Students should have proficiency in programming languages commonly used in automation systems, such as Python, C++, or Java. They should be able to write, debug, and optimize code.	SHR 1.1 Understand the compl		
2	SHR 1.1.2	SHR 1.1.2 Hardware design	Students should have an understanding of the hardware used in automation systems, including sensors, actuators, and controllers, is necessary. including how components interact within the system.	SHR 1.1 Understand the compl		
3	SHR 1.1.3.1	SHR 1.1.3.1 Knowledge Assesment	Manufacturing knowledge and control information score should be between 75 and 100 among factory operation steps and disparate factories .	SHR 1.1.3 System Integration	KnowledgeAssessment : Test or Quiz Score	[75;100]
4	SHR 1.1.3.2	SHR 1.1.3.2 Practical Examination	The final project score should be between 75 to 100 for a technology solution proposal that addresses a Production Issue in a Smart Factory.	SHR 1.1.3 System Integration	PracticalExamination : Practical Examinatio	=75
5	SHR 1.1.3.3	SHR 1.1.3.3 Panel experts Reviews	Panel reviews score should be between 80 to 100 for a project to integrate complex equipment in to a factory.	SHR 1.1.3 System Integration	Peer&Instructor Reviews : Peer and Instruct	=80
6	SHR 1.1.3	SHR 1.1.3 System Integration	Students should have the ability to integrate automation systems with other hardware and software systems. This includes understanding APIs, data exchange formats, and network protocols.	SHR 1.1 Understand the compl		
7	SHR 1.1.4.1	SHR 1.1.4.1 Knowledge Assessment	Test or Quiz Score understanding of the lifecycle of automation systems should be between 80 and 100	SHR 1.1.4 System Maintenance	KnowledgeAssessment : Test or Quiz Score	[80;100]
8	SHR 1.1.4.2	SHR 1.1.4.2 Practical Examination	Practical Examination Score should be between 70 and 100 for malfunctioning automation system	SHR 1.1.4 System Maintenance	PracticalExamination : Practical Examinatio	[70;100]
9	SHR 1.1.4.3	SHR 1.1.4.3 Peer and Instructor Reviews	Peer and Instructor Reviews Score for Student understanding and skill on the subject should be more than 60	SHR 1.1.4 System Maintenance	Peer&Instructor Reviews : Peer and Instruct	>60
10	SHR 1.1.4	SHR 1.1.4 System Maintenance	Students should understand the lifecycle of automation systems and be able to perform routine maintenance tasks. This includes troubleshooting, system updates, and managing backups	SHR 1.1 Understand the compl		
11	SHR 1.1.5.1	SHR 1.1.5.1 Knowledge Assesment	IIoT, process control, and autonomous robotics knowledge scores should be between 70 and 100.	SHR 1.1.5 Documentation	KnowledgeAssessment : Test or Quiz Score	[70;100]
12	SHR 1.1.5.2	SHR 1.1.5.2 Practical Examination	The design and implementation score should be between 80 to 100 for an IioT device that solves and addresses a production recollecting data in a Smart Factory.	SHR 1.1.5 Documentation	PracticalExamination : Practical Examinatio	=80
13	SHR 1.1.5.3	SHR 1.1.5.3 Project purpuse Reviews	Panel review score should be between 80 to 100 for a purpose of an electronic device that solves a real problem in a factory visited during the course.	SHR 1.1.5 Documentation	Peer&Instructor Reviews : Peer and Instruct	=80
14	SHR 1.1.5	SHR 1.1.5 Documentation	Stakeholders should be able to create clear and comprehensive documentation for the automation systems. This includes system specifications, user manuals, and maintenance guides	SHR 1.1 Understand the compl		
15	SHR 1.1	SHR 1.1 Understand the complex automation systems, including programming,maintenance, and integration with other systems				

Filter is not applied. 15 rows are displayed in the table.

Figure 6.3 Requirement Verification & Validation of Use Case Scenario at Different Environmental Variables

These course assignments include group discussions, post-discussion, group projects, presentations, and self-assessments, in addition to traditional exams, assignments, and quizzes, see Figure 6.4. This scenario is used to analyze the scores of students' professional competencies at the end of the course. The scenario executes different environmental variables to determine which course assignments most effectively contribute to skill development and which areas might require more focus for individual students, see Figure 6.5. This scenario leverages the synthetic data model generated in Section 3.8, to run simulation for 'N' students for different soft skill values. Enhanced visualization of these scenarios is shown in Figure 6.6, and Figure 6.7.

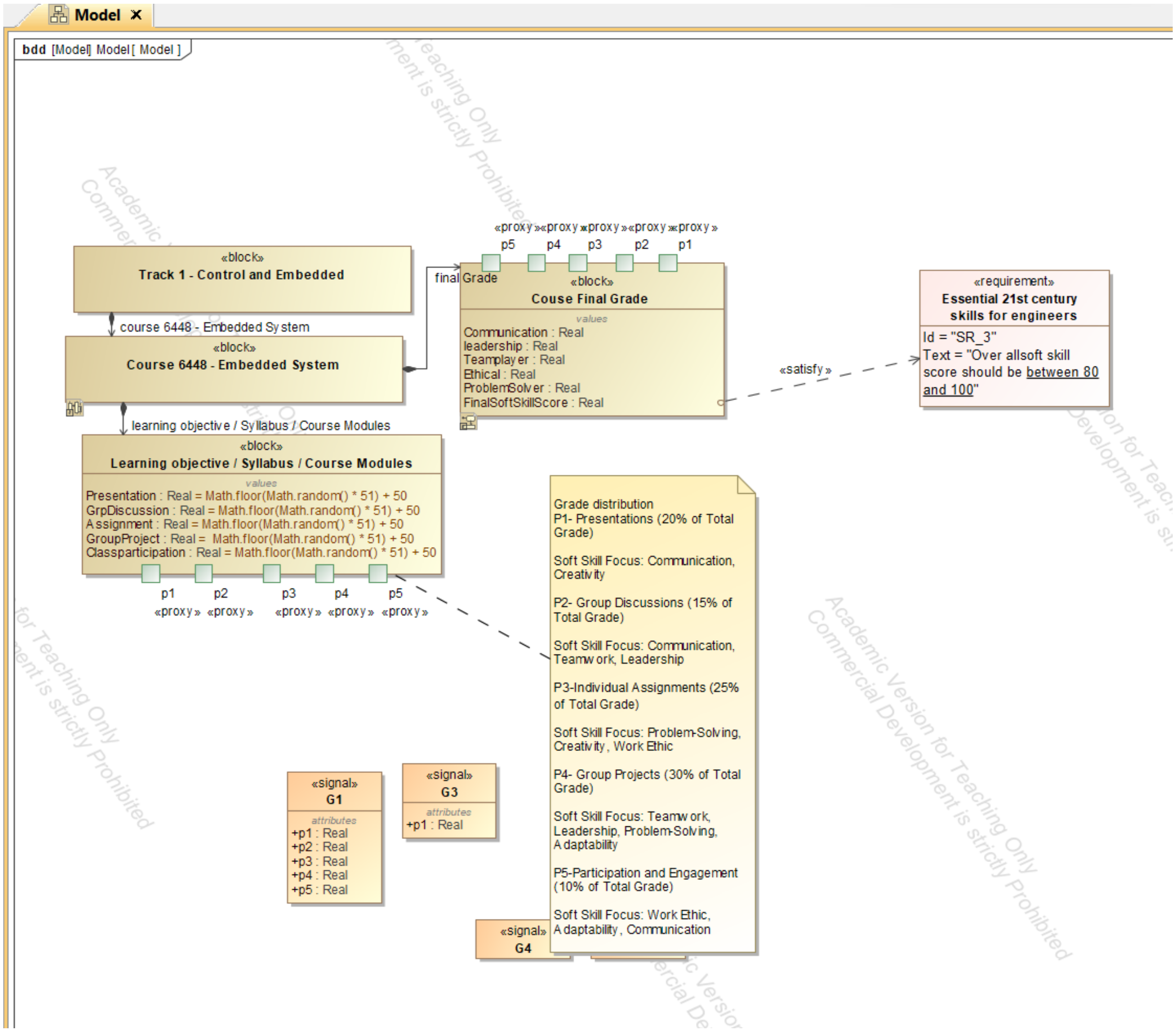


Figure 6.4 System Architecture for Activity and Sequence Diagram Representing the Course Module 'Embedded System' for 'Essential 21st Century Skills for Engineers' Use Case Model

Further, refined non-functional quantifiable needs, i.e., Measures of Effectiveness (MoEs) were identified, and are listed in Table 6.1. In conclusion, random scores for each skill, such as communication, teamwork, and adaptability, are generated. These scores vary, reflecting different levels of proficiency. This variation enables better visualization of the final score based on diverse soft skill values. In each instance the system executes a set of synthetic soft skill scores, it then calculates the overall score. This dynamic visualization

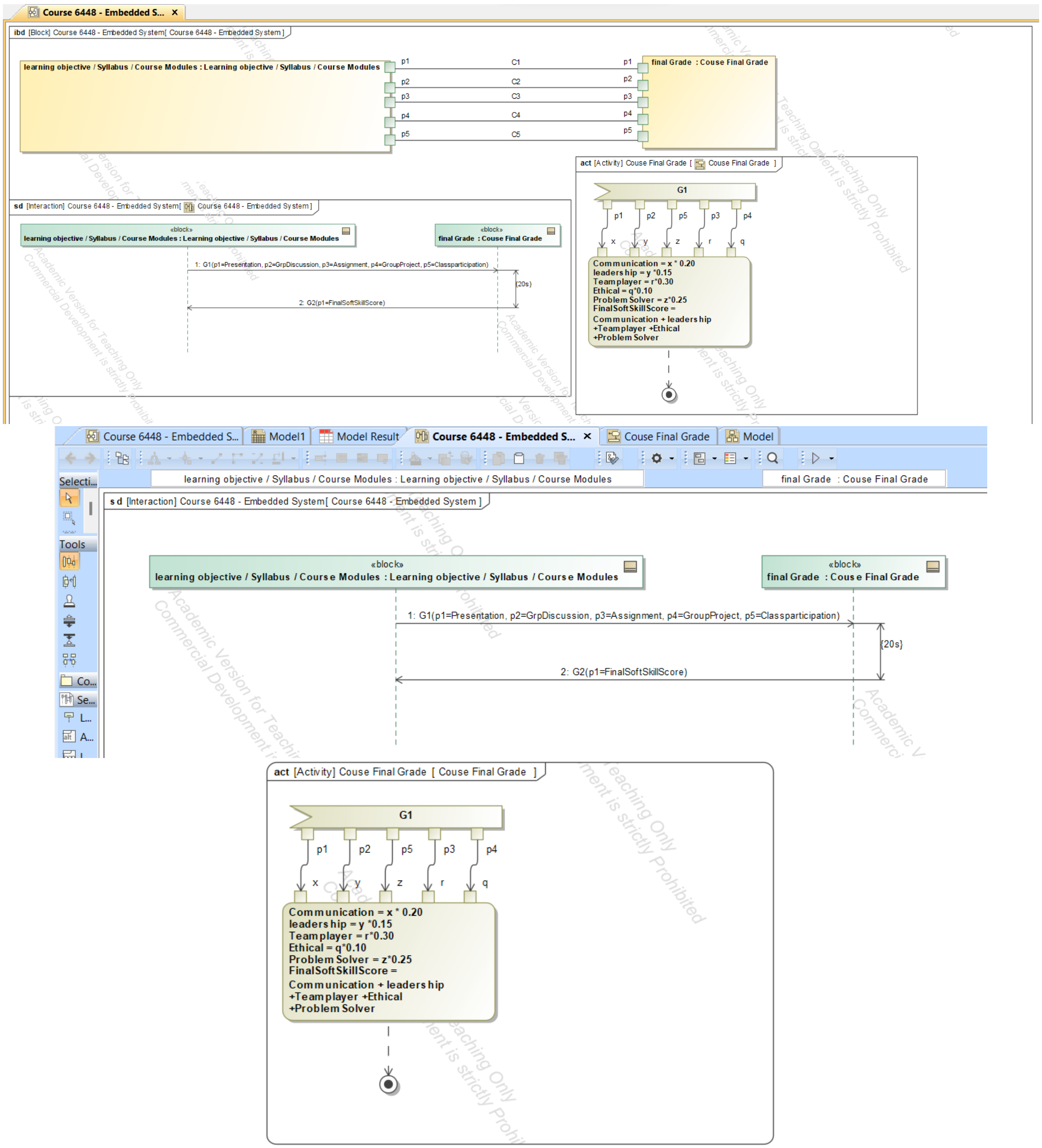


Figure 6.5 Activity and Sequence Diagram Representing the Course Module 'Embedded System'

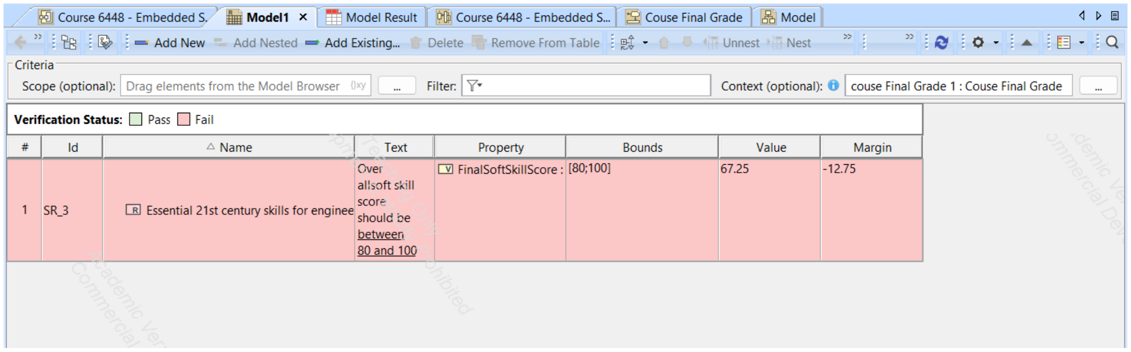


Figure 6.6 Requirement Verification of 'Essential 21st Century Skills for Engineers' Use Case Model

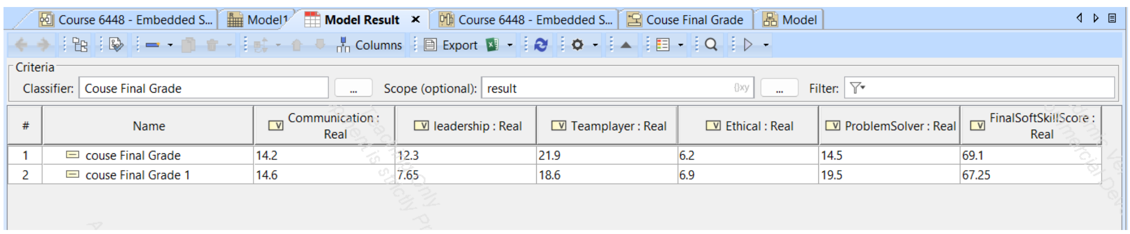


Figure 6.7 Model Test Instance of 'Essential 21st Century Skills for Engineers' Use Case Model

assists stakeholders in understanding how different skill levels impact the final outcome. Use Case Scenarios and Requirement Verification: For each use case scenario, the system was tested against specified requirements and verified to confirm that the system's behavior aligns with the intended functionality described in the use case.

Table 6.1 MOEs for Stakeholder Requirements (SHR 3.0)

Soft Skill	Evaluation Matrix	Target Score
Problem Solving	- Ability to identify and define complex problems - Analytical thinking and logical reasoning	> 85
Communication	- Ability to communication among team mates	> 78
Collaboration	- Ability to work with other teammates	> 92
Leadership	- Ability to show leadership quality in group assignment	> 70
Critical Thinking	- Ability to identify innovative solutions	> 88
Teamwork	- Ability to understand team member personality and work together	> 95
Adaptability	- Ability to adapt to the problem needs	> 80

6.2 Real-World Scenario - Team Formation Algorithm: Development and Testing in Engineering Course

The research applies the Multi-Criteria Integral Programming (MCIP) algorithm, developed in Chapter 4, for team formation in university-level engineering courses. This method is compared with previous team formation approaches to assess its effectiveness. A detailed report and statistics are presented in the next section, focusing on skill balance and demographic diversity within teams.

6.2.1 Diversity-Based Implementation

The Team Formation intervention, implemented in the engineering courses of Spring and Summer 2022, utilized the MCIP algorithm to maximize team diversity across GPA, skills, language, and country of origin. Table 6.2 displays the statistical control variables for

Table 6.2 Team Formation Intervention Control Variables - Embedded System

Course: Embedded Systems		
Control variables	**Spring 2022**	**Summer 2022**
Total no. of students	80	84
No. of Team	20	21
Team-size	3-4 students per team	3-4 students per team
Course Length	16 weeks	10 weeks
Course Type	Electives	Electives
Instructor	Dr. Castellanos	Dr. Castellanos
Gender	Male = 75, Female = 5	Male = 81, Female = 4
Education Level	Undergraduates	Undergraduates
Department	Electrical Engineering	Electrical Engineering

the two groups, namely the engineering courses of Spring and Summer 2022. The results of the intervention show improved team dynamics, effective communication skills, and enhanced learning experiences, especially in problem-solving abilities, for the group selected through multi-criteria team formation.

Figure 6.8 illustrates the comparative performance of student teams in the Embedded System courses for Spring and Summer 2022. For Spring 2022, the student teams performed higher in the first two assignments. However, a noticeable performance decline was observed in Assignment 3 and the final project. This downturn is attributed to the struggles faced by self-selected teams, primarily composed of friends, as the complexity of the assignments escalated.

On the other hand, the Summer 2022 teams initially under performed in the first two assignments. However, a significant improvement was observed in the subsequent assignments and the final project. This progress was credited to the formation of diverse teams. These teams, initially strangers, gradually developed effective collaboration and communication over time. It can be observed that the Summer 2022 teams initially faced challenges in adapting to the new environment and understanding their team members. However, as they became familiar with each other's personalities, their problem-solving abilities improved. This led to a higher performance in tackling the assignments, even those with increased dif-

Table 6.3 Team Formation Intervention Control Variables - Applied Mechatronics

Course: Applied Mechatronics		
Control variables	**Fall 2021**	**Fall 2021**
Total no. of students	48	47
No. of Team	12	12
Team-size	3-4 students per team	3-4 students per team
Course length	16 weeks	16 weeks
Course Type	Electives	Electives
Instructor	Dr. Castellanos	Dr. Castellanos
Gender	Male = 45, Female = 3	Male = 45, Female = 2
Education Level	Undergraduates	Undergraduates
Department	Electrical Engineering	Electrical Engineering

ficulty such as Assignment 3 and the final project, surpassing the performance of the Spring 2022 teams.

The same Team Formation intervention was implemented for Applied Mechatronics courses in Fall 2021 and Fall 2022, resulting in consistent improvements in both team and

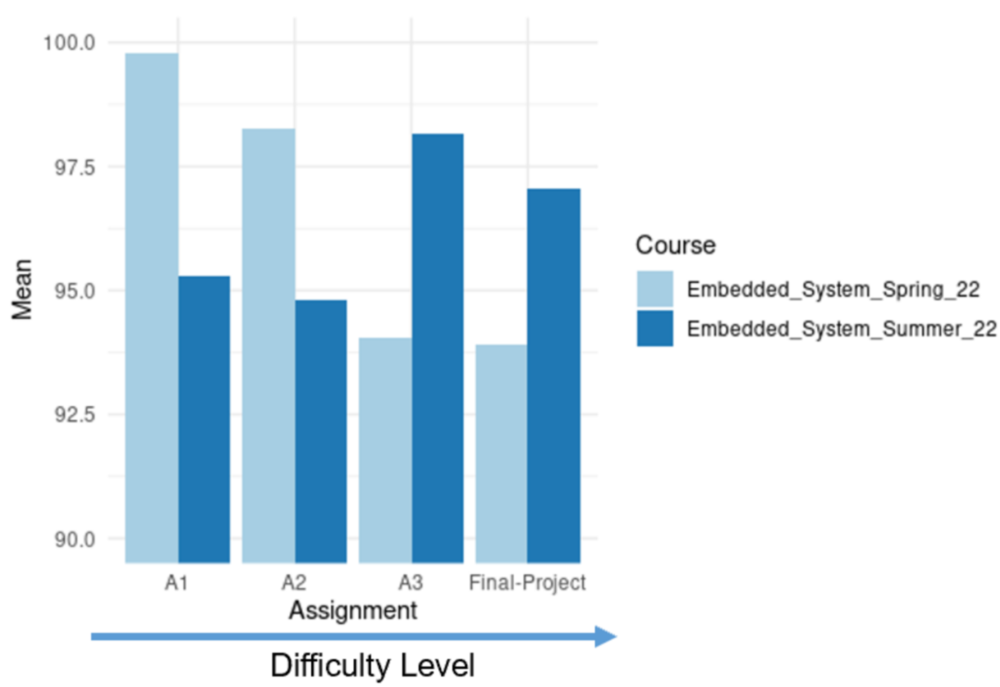


Figure 6.8 Team Formation in the Embedded System Courses – Spring vs. Summer 2022 Analysis

individual performance in terms of grades and professional skills, as shown in Figure 6.9. The control variable for this intervention is detailed in Table 6.3.

6.2.2 Project-Based Implementation

The Project-Based Team Formation (PBTF) method for Team Formation was implemented in the two-semester senior capstone design courses I and II. Prior to this PBTF intervention, teams were formed on a First-Come-First-Serve (FCFS) method, which had several challenges such as lack of skill diversity, lack of motivation and engagement, team dynamics issues, and work distribution inefficiency.

Since Fall 2022, teams have been formed based on PBTF methods which take into account variables such as project preference, skillset compatibility, time availability, team expertise, technical competencies, and demographic diversity. Each year, a survey was con-

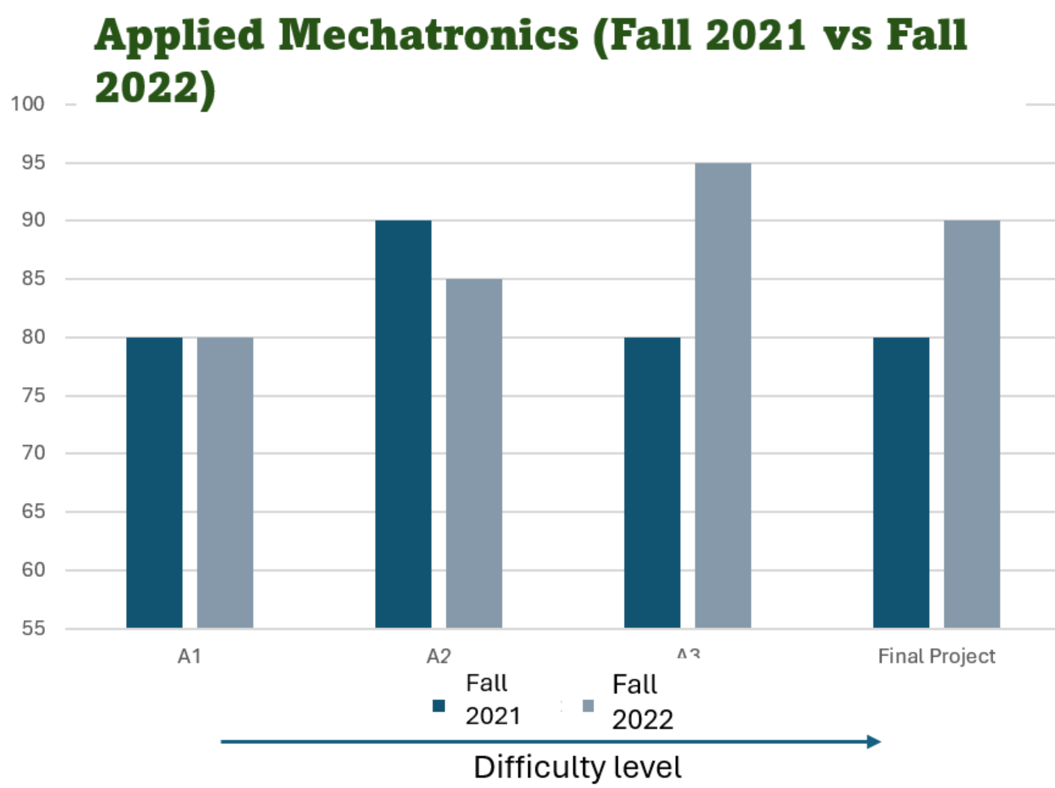


Figure 6.9 Team Formation in the Applied Mechatronics Courses – Fall 2021 vs. Fall 2022 Analysis

ducted to gather feedback from faculty members who advised the capstone design projects and the industry partners who monitored the capstone design project's life-cycle. The survey feedback was analyzed and is represented in Figure 6.10 which illustrates the performance of students selected via the First-Come-First-Serve (FCFS) method vs student selected via Project-Based Team Formation (PBTF) method. It is observable that the First-Come, First-Served (FCFS) method scores lower across various parameters such as presentation, comprehensiveness, quality of work, team dynamics, and division of work, with the exception of ethical standards. This highlights the effectiveness of the Project-Based Team Formation intervention in enhancing students' interpersonal skills and overall performance.

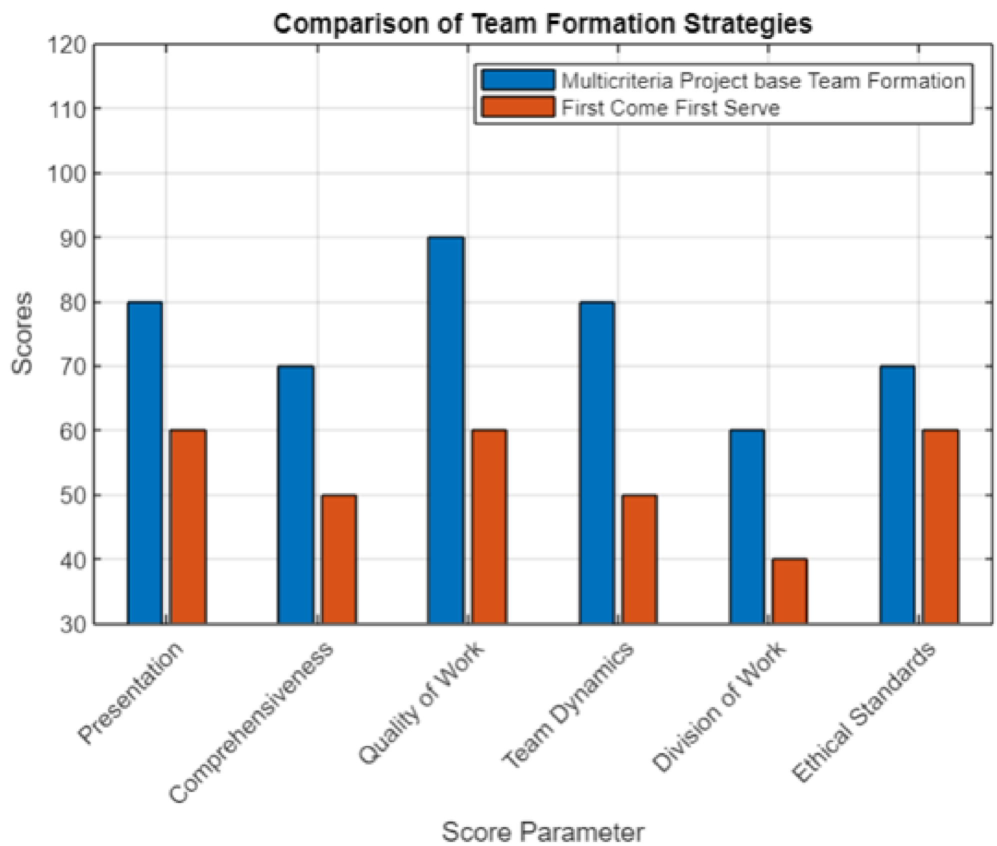


Figure 6.10 Comparative Analysis of Team Performance: Multi-Criteria vs. First-Come-First-Serve Team Formation Methods

Chapter 7: Conclusion and Future Work

7.1 Conclusion

This research has systematically demonstrated the efficacy of Model-Based Systems Engineering (MBSE) within the context of university-level engineering education. By deploying the Engineering Learning Analytic System (ELAS), this research not only bridges the theoretical constructs of MBSE with practical applications but also showcases a significant advance in addressing complex engineering educational challenges.

ELAS, designed as a human-centric model, involves key stakeholders, i.e., students, educators, and industry professionals to ensure that the educational system is aligned with actual educational needs. Through rigorous system simulations, the model effectively mirrors real-world educational settings, thereby validating its applicability and relevance in modern educational environments.

A pivotal contribution of this research is the development and implementation of innovative tools, including synthetic data models and multi-criteria team formation algorithms. These tools have been instrumental in refining the management of engineering educational data and optimizing student team formations, thereby enhancing both individual and collective educational outcomes.

In summary, the research illustrates that MBSE can be innovatively applied to engineering education research, highlighting its potential to revolutionize traditional educational methodologies by introducing precision, adaptability, and efficiency.

7.2 Future Work

While the current research have marked significant advancements in integrating Model-Based Systems Engineering (MBSE) methodology for engineering education systems, it is not without its limitations which, in turn, make the way for future research opportunities. The diverse simulation scenarios employed did not encompass all variables present in real-world settings. There are challenges with respect to model's dependency on quantifiable inputs such as student grades, internships, and income levels, which are significant but still do not capture complete student attributes like cultural background and community support. These elements are crucial for fully understanding and enhancing student experiences and performance. Hence, future enhancements of the ELAS model should incorporate these attributes, particularly focusing on the experiences of underrepresented students, this will give a more inclusive and accurate representation of the student body. Furthermore, the team formation algorithm, though effective, requires continuous refinement to adapt to evolving educational demands. The generative synthetic data model, fundamental in overcoming the lack of concrete datasets, demands ongoing validation to keep pace with the dynamic nature of educational environments.

Moving forward, the future work of the research includes, advancing the Professional Formation of Engineers (PFE) initiative through data-driven approaches. The Professional Formation of Engineers (PFE) is a new comprehensive educational framework designed to equip electrical engineering USF students with the necessary skills and competencies for their professional careers. It encompasses a series of structured courses, talent development, and goal-setting exercises that aim to the practice of engineering ethically, with impact on both local and global communities. The future scope includes implementing Data-Driven Pathways such as data analytics and machine learning algorithms to create early detection systems models from data accumulated over the past six years since the PFE courses sequence was introduced. The Team formation model shall leverage historical PFE data to create predictive analyses that function as early warning systems, identifying potential challenges and opportunities for students. A significant aspect of this development will demonstrate the role of interdisciplinary skills, internships, hands-on workshops, and leadership roles, in student career development, thereby customizing the learning experience. Additionally, future work includes conducting longitudinal studies on the impact of Model-Based Systems Engineering (MBSE) on student learning and success, as well as exploring MBSE's application in diverse educational fields beyond its current use, focusing particularly on its potential for cross-disciplinary integration. Moreover, refining and testing team formation algorithms to enhance collaborative skills and project outcomes is essential.

This dissertation represents a significant step forward in the application of Model-Based Systems Engineering to enhance the field of engineering education. While substantial progress has been made, the journey towards fully realizing the potential of MBSE in engineering education continues. The paths laid out by this research invite further exploration and promise to yield innovative solutions to enduring challenges in educational systems design.

References

[1] Arun Adiththan Adiththan, Joseph GD'Ambrosio D'Ambrosio, Azad M Madni Madni, Edwin Ordoukhanian Ordoukhanian, S Ramesh Ramesh, Prakash Peranandam Peranandam, and Padma Sundaram Sundaram. An mbse approach for development of resilient automated automotive systems. 2019.

[2] Manya Afonso and Valerio Giuffrida. Synthetic data for computer vision in agriculture. *Frontiers in Plant Science*, 14:1277073, 2023.

[3] I. Aggarwal and A. W. Woolley. Team creativity, cognition, and cognitive style diversity. *Management Science*, 65(4):1586–1599, 2019.

[4] Monika Alesnik. *A Policy Analysis of Performance-Based Funding in the State University System of Florida*. University of South Florida, 2020.

[5] Funmilola Bosede Alokan and Amos Emiloju Arijesuyo. Rural and urban differential in student's academic performance among secondary school students in ondo state, nigeria. *Journal of Educational and Social Research*, 3(3):213, 2013.

[6] Shahid Anwar. *Essential skills for the 21st-century engineer*. PhD thesis, Murdoch University, 2022.

[7] John P Barile, Dana K Donohue, Elizabeth R Anthony, Andrew M Baker, Scott R Weaver, and Christopher C Henrich. Teacher–student relationship climate and school outcomes: Implications for educational policy initiatives. *Journal of Youth and Adolescence*, 41(3):256–267, 2012.

[8] E. F. Barkley and C. H. Major. *Student Engagement Techniques: A Handbook for College Faculty*. John Wiley Sons, 2020.

[9] G. R. Bauer, J. Braimoh, A. I. Scheim, and C. Dharma. Transgender-inclusive measures of sex/gender for population surveys: Mixed-methods evaluation and recommendations. *PloS One*, 12(5):e0178043, 2017.

[10] Peter Bautista and Paul Salvador Inventado. Protecting student privacy with synthetic data from generative adversarial networks. In *International Conference on Artificial Intelligence in Education*, pages 66–70. Springer, 2021.

[11] Alice Bertoletti and Geraint Johnes. Efficiency in university-industry collaboration: an analysis of uk higher education institutions. *Scientometrics*, 126(9):7679–7714, 2021.

[12] Mary Besterfield-Sacre, Cynthia J Atman, and Larry J Shuman. Engineering student attitudes assessment. *Journal of Engineering Education*, 87(2):133–141, 1998.

[13] Melvin V Borland and Roy M Howsen. A note on student academic performance: In rural versus urban areas. *American Journal of Economics and Sociology*, 58(3):537–546, 1999.

[14] William Brescia and Jim C Fortune. Standardized testing of american indian students. *College Student Journal*, 1989.

[15] Josh Brown. The current status of stem education research. *Journal of STEM Education: Innovations and Research*, 13(5), 2012.

[16] Mary T Brownell, Eric Hirsch, and Seonjin Seo. Meeting the demand for highly qualified special education teachers during severe shortages: What should policymakers consider? *The Journal of Special Education*, 38(1):56–61, 2004.

[17] A. Burgess, J. Bleasel, I. Haq, C. Roberts, R. Garsia, T. Robertson, and C. Mellis. Team-based learning (tbl) in the medical curriculum: Better than pbl? *BMC Medical Education*, 17:1–11, 2017.

[18] M. Burisch. In search of theory: Some ruminations on the nature and etiology of burnout. pages 75–93, 2017.

[19] Richard J Chen, Ming Y Lu, Tiffany Y Chen, Drew FK Williamson, and Faisal Mahmood. Synthetic data in machine learning for medicine and healthcare. *Nature Biomedical Engineering*, 5(6):493–497, 2021.

[20] Yi-Hsin Chen. Cognitive diagnosis of mathematics performance between rural and urban students in taiwan. *Assessment in Education: Principles, Policy & Practice*, 19(2):193–209, 2012.

[21] Steen Hyldgaard Christensen, Christelle Didier, Andrew Jamison, Martin Meganck, Carl Mitcham, Byron Newberry, et al. Engineering identities, epistemologies and values. *Engineering Education and Practice in Context*, 2, 2015.

[22] S. Civitillo, L. P. Juang, and M. K. Schachner. Challenging beliefs about cultural diversity in education: A synthesis and critical review of trainings with pre-service teachers. *Educational Research Review*, 24:67–83, 2018.

[23] S. R. Clegg, T. S. Pitsis, and M. Mount. *Managing and Organizations: An Introduction to Theory and Practice*. Sage, 2021.

[24] Thomas B Corcoran. Teaching matters: How state and local policymakers can improve the quality of teachers and teaching. 2007.

[25] Joseph D'Ambrosio and Grant Soremekun. Systems engineering challenges and mbse opportunities for automotive system design. In *2017 IEEE international conference on systems, man, and cybernetics (SMC)*, pages 2075–2080. IEEE, 2017.

[26] Zoltán Dörnyei. Researching complex dynamic systems:'retrodictive qualitative modelling'in the language classroom. *Language Teaching*, 47(1):80–91, 2014.

[27] Bruce Powel Douglass and Christian von Holst. *Agile Model-Based Systems Engineering Cookbook: Improve system development by applying proven recipes for effective agile systems engineering*. Packt Publishing Ltd, 2022.

[28] Hendrik Drachsler and Wolfgang Greller. Privacy and analytics: it's a delicate issue a checklist for trusted learning analytics. In *Proceedings of the sixth international conference on learning analytics & knowledge*, pages 89–98, 2016.

[29] Joshua Earnest. Abet engineering technology criteria and competency based engineering education. In *Proceedings Frontiers in Education 35th Annual Conference*, pages F2D–7. IEEE, 2005.

[30] Khaled El Emam, Lucy Mosquera, and Richard Hoptroff. *Practical synthetic data generation: balancing privacy and the broad availability of data*. O'Reilly Media, 2020.

[31] Jeff A Estefan et al. Survey of model-based systems engineering (mbse) methodologies. *Incose MBSE Focus Group*, 25(8):1–12, 2007.

[32] Mitra Ezati and Abolghasem Naderi. The impacts of financial resource allocation mechanisms on educational performance of university departments: the case study of university of tehran. *Quarterly Journal of Research and Planning in Higher Education*, 15(2):23–54, 2009.

[33] Z.-P. Fan, Y. Chen, J. Ma, and S. J. J. o. t. O. R. S. Zeng. Erratum: A hybrid genetic algorithmic approach to the maximally diverse grouping problem. *Journal of the Operational Research Society*, 62(7):1423–1430, 2011.

[34] F. Farhadi, M. Sorkhi, S. Hashemi, and A. Hamzeh. An effective expert team formation in social networks based on skill grading. In *2011 IEEE 11th International Conference on Data Mining Workshops*, pages 366–372. IEEE, 2011.

[35] Brendan Flanagan, Rwitajit Majumdar, and Hiroaki Ogata. Fine grain synthetic educational data: challenges and limitations of collaborative learning analytics. *IEEE Access*, 10:26230–26241, 2022.

[36] Kevin S Floyd, Susan J Harrington, and Julie Santiago. The effect of engagement and perceived course value on deep and surface learning strategies. *Informing Sci. Int. J. an Emerg. Transdiscipl.*, 12:181–190, 2009.

[37] J. Frerejean, J. J. van Merriënboer, P. A. Kirschner, A. Roex, B. Aertgeerts, and M. Marcellis. Designing instruction for complex learning: 4c/id in higher education. *European Journal of Education*, 54(4):513–524, 2019.

[38] Sanford Friedenthal, Alan Moore, and Rick Steiner. *A Practical Guide to SysML: The Systems Modeling Language*. Morgan Kaufmann, 2015.

[39] Jeffrey E Froyd and Maura Borrego. Leadership insights from the national science foundation engineering education coalitions program and other large curriculum initiatives. *Journal of Leadership Studies*, 8(1), 2014.

[40] Jenni Gilleard and John D Gilleard. Developing cross-cultural communication skills. *Journal of professional issues in engineering education and practice*, 128(4):187–200, 2002.

[41] Mehmet Emre Gursoy, Ali Inan, Mehmet Ercan Nergiz, and Yucel Saygin. Privacy-preserving learning analytics: challenges and techniques. *IEEE Transactions on Learning technologies*, 10(1):68–81, 2016.

[42] Jonas Hallqvist and Jonas Larsson. Introducing mbse by using systems engineering principles. In *INCOSE International Symposium*, volume 26, pages 512–525. Wiley Online Library, 2016.

[43] Laura E Hart. Introduction to model-based system engineering (mbse) and sysml. In *Delaware Valley INCOSE Chapter Meeting*, volume 30. Ramblewood Country Club Mount Laurel, New Jersey, 2015.

[44] Anne T Henderson and Karen L Mapp. A new wave of evidence: The impact of school, family, and community connections on student achievement. annual. 2002.

[45] Jonathan C Hilpert and Jenefer Husman. Instructional improvement and student engagement in post-secondary engineering courses: the complexity underlying small effect sizes. *Educational Psychology*, 37(2):157–172, 2017.

[46] Jonathan C Hilpert and Gwen C Marchand. Complex systems research and evaluation in engineering education. In *Annual Conference & Exposition: final program and proceedings. American Society for Engineering Education*, volume 2017. NIH Public Access, 2017.

[47] Niamat Ullah Ibne Hossain, Mostafa Lutfi, Ifaz Ahmed, and Hunter Debusk. Application of system engineering modeling language (sysml) and discrete event simulation to address patient waiting time issues in healthcare. *Smart Health*, page 100403, 2023.

[48] P. Imbrie, J. Agarwal, and G. Raju. Genetic algorithm optimization of teams for heterogeneity. In *2020 IEEE Frontiers in Education Conference (FIE)*, pages 1–5. IEEE, 2020.

[49] Irum Inayat, Siti Salwah Salim, Sabrina Marczak, Maya Daneva, and Shahaboddin Shamshirband. A systematic literature review on agile requirements engineering practices and challenges. *Computers in human behavior*, 51:915–929, 2015.

[50] Leah H Jamieson and Jack R Lohmann. Creating a culture for scholarly and systematic innovation in engineering education: Ensuring us engineering has the right people with the right talent for a global society. *Washington, DC: American Society for Engineering Education*, 2009.

[51] J. Juárez, C. Santos, and C. A. Brizuela. A comprehensive review and a taxonomy proposal of team formation problems. *ACM Computing Surveys (CSUR)*, 54(7):1–33, 2021.

[52] Kalpak Kalvit. *Application of an innovative MBSE (SysML-1D) co-simulation in healthcare*. Purdue University, 2018.

[53] Iakovos Katsipis. *Health Care Management System for Diabetes Mellitus: A Model-based Systems Engineering Framework*. PhD thesis, 2015.

[54] William W Keller and Louis W Pauly. Innovation in the indian semiconductor industry: The challenge of sectoral deepening. *Business and Politics*, 11(2):1–21, 2009.

[55] Mohammad Khalil and Martin Ebner. De-identification in learning analytics. *Journal of Learning Analytics*, 3(1):129–138, 2016.

[56] Jennifer K Knight and William B Wood. Teaching more by lecturing less. *Cell biology education*, 4(4):298–310, 2005.

[57] Robert Kraus, George Papaioannou, and Arun Sivan. Application of model based system engineering (mbse) principles to an automotive driveline sub-system architecture.

[58] S Kuper. Industry relevance and its role in student success and corporate training. *Journal of Bew and Innovative Market Opportunities*, 2018.

[59] Lisa R Lattuca, Patrick T Terenzini, and J Fredericks Volkwein. Panel session-engineering change: Findings from a study of the impact of ec2000. In *Proceedings. Frontiers in Education. 36th Annual Conference*, pages 1–2. IEEE, 2006.

[60] Brian London. *A model-based systems engineering framework for concept development.* PhD thesis, Massachusetts Institute of Technology, 2012.

[61] Michael C Loui and Maura Borrego. 11 engineering education research. *The Cambridge handbook of computing education research*, page 292, 2019.

[62] Azad M Madni, Dan Erwin, and Carla C Madni. Digital twin-enabled mbse testbed for prototyping and evaluating aerospace systems: Lessons learned. In *2021 IEEE Aerospace Conference (50100)*, pages 1–8. IEEE, 2021.

[63] Holly Matusovich, Ruth Streveler, and Ron Miller. We are teaching engineering students what they need to know, aren't we? In *2009 39th IEEE Frontiers in Education Conference*, pages 1–6. IEEE, 2009.

[64] Cheryl B McCarthy. Effects of thematic-based, hands-on science teaching versus a textbook approach for students with disabilities. *Journal of Research in Science Teaching: The Official Journal of the National Association for Research in Science Teaching*, 42(3):245–263, 2005.

[65] James H McMillan. Fundamental assessment principles for teachers and school administrators. *Practical Assessment, Research, and Evaluation*, 7(1):8, 2000.

[66] Steven Moonen, Bram Vanherle, Joris de Hoog, Taoufik Bourgana, Abdellatif Bey-Temsamani, and Nick Michiels. Cad2render: A modular toolkit for gpu-accelerated photorealistic synthetic data generation for the manufacturing industry. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 583–592, 2023.

[67] Corina Pacher, Manuel Woschank, Bernd M Zunk, and Elke Gruber. Engineering education 5.0: a systematic literature review on competence-based education in the industrial engineering and management discipline. *Production & Manufacturing Research*, 12(1):2337224, 2024.

[68] Abelardo Pardo and George Siemens. Ethical and privacy principles for learning analytics. *British journal of educational technology*, 45(3):438–450, 2014.

[69] Marden Pasinato, Carlos Eduardo Mello, Marie-Aude Aufaure, and Geraldo Zimbrao. Generating synthetic data for context-aware recommender systems. In *2013 BRICS Congress on Computational Intelligence and 11th Brazilian Congress on Computational Intelligence*, pages 563–567. IEEE, 2013.

[70] Arun S Patil and Zenon J Pudlowski. Instructional design strategies for interactive web-based tutorials and laboratory procedures in engineering education. *World Transactions on Engineering and Technology Education*, 2(1):107–110, 2003.

[71] Helena JM Pennings and Tim Mainhard. Analyzing teacher–student interactions with state space grids. *Complex dynamical systems in education: Concepts, methods and applications*, pages 233–271, 2016.

[72] Ana Luísa Ramos, José Vasconcelos Ferreira, and Jaume Barceló. Model-based systems engineering: An emerging approach for modern systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(1):101–111, 2011.

[73] Dean Revell, David C Gross, Gong Zhou, and Adrian Hernandez. Applying a mbse methodology in small scale technology development. In *2023 IEEE International Systems Conference (SysCon)*, pages 1–6. IEEE, 2023.

[74] Debora L Roorda, Helma MY Koomen, Jantine L Spilt, and Frans J Oort. The influence of affective teacher–student relationships on students' school engagement and achievement: A meta-analytic approach. *Review of educational research*, 81(4):493–529, 2011.

[75] Siegfried Rouvrais, Janet Ormrod, Gabrielle Landrac, Julien Mallet, J-M Gilliot, André Thépaut, and Pierre Tremenbert. A mixed project-based learning framework: preparing and developing student competencies in a french grande ecole. *European Journal of Engineering Education*, 31(1):83–93, 2006.

[76] Hal Salzman and Leonard Lynn. Engineering and engineering skills: What's really needed for global competitiveness. 2010.

[77] Pallavi Singh, Phat K Huynh, Dang Nguyen, Trung Q Le, and Wilfrido Moreno. Leveraging multi-criteria integer programming optimization for effective team formation. *IEEE Transactions on Learning Technologies*, 2024.

[78] Pallavi Singh, Ayisha Necholi, and Wilfrido Moreno. Synthetic data generation for engineering education: A bayesian approach. In *2023 IEEE 3rd International Conference on Advanced Learning Technologies on Education & Research (ICALTER)*, pages 1–4. IEEE, 2023.

[79] SP Singh, Savita Malik, and Priya Singh. Research paper factors affecting academic performance of students. *Indian Journal of Research*, 5(4):176–178, 2016.

[80] K. Srikanth, S. Harvey, and R. Peterson. A dynamic perspective on diverse teams: Moving from the dual-process model to a dynamic coordination-based model of diverse team performance. *Academy of Management Annals*, 10(1):453–493, 2016.

[81] Christina M Steiner, Michael D Kickmeier-Rust, and Dietrich Albert. Lea in private: A privacy and data protection framework for a learning analytics toolbox. *Journal of Learning Analytics*, 3(1):66–90, 2016.

[82] K. H. D. Tang. Personality traits, teamwork competencies, and academic performance among first-year engineering students. *Higher Education, Skills and Work-Based Learning*, 11(2):367–385, 2021.

[83] T. Tang, V. Vezzani, and V. Eriksson. Developing critical thinking, collective creativity skills, and problem solving through playful design jams. *Thinking Skills and Creativity*, 37:100696, 2020.

[84] James Trevelyan. Technical coordination in engineering practice. *Journal of Engineering Education*, 96(3):191–204, 2007.

[85] Mariana Villalabeitia Arenas. High-level requirements for conceptual design of bridge deflection measurement system using model-based systems engineering. 2023.

[86] Louis Volante. Teaching to the test: What every educator and policy-maker should know. *Canadian Journal of Educational Administration and Policy*, 2004.

[87] Ming-Te Wang, Maureen Brinkworth, and Jacquelynne Eccles. Moderating effects of teacher–student relationship in adolescent trajectories of emotional and behavioral adjustment. *Developmental psychology*, 49(4):690, 2013.

[88] Gregg M Warnick. Global competence: Its importance for engineers working in a global environment. In *2011 ASEE Annual Conference & Exposition*, pages 22–748, 2011.

[89] Wang Wenyue, Hou Junjie, Mao Yinxuan, Lu Zhiang, et al. Application and development of mbse in aerospace. In *Journal of Physics: Conference Series*, volume 2235, page 012021. IOP Publishing, 2022.

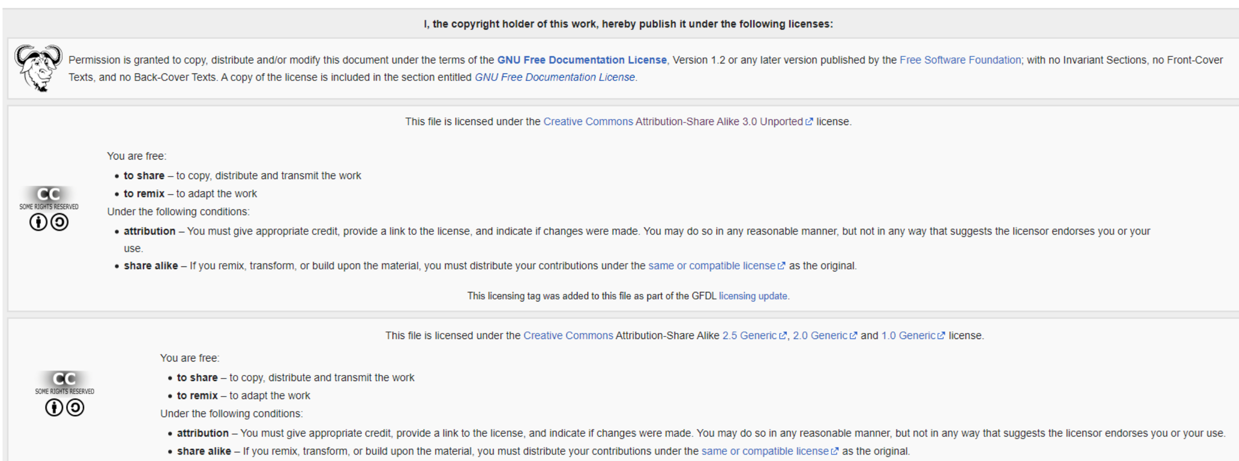
[90] Theo Wubbels, MIEKE Brekelmans, PERRY den Brok, LINDY Wijsman, T Mainhard, and J Van Tartwijk. Teacher-student relationships and classroom management. *Handbook of classroom management*, pages 363–386, 2014.

[91] Deidra J Young. A comparison of student performance in western australian schools: Rural and urban differences. *The Australian Educational Researcher*, 21(2):87–105, 1994.

Appendix A: Copyright Permissions

This Appendix lists copyright clearances and permissions for all figures and the edited manuscript included in Chapter 4, and Chapter 5 of this dissertation. The figures are listed in order, including all copyright clearances and permissions.

Figure 2.1 is an image in the public domain, released under the GNU Free Documentation License, and was originally by the author Betelgeuse, URL:https://commons.wikimedia.org/wiki/File:Systems_engineering_application_projects_collage.jpg.



I, the copyright holder of this work, hereby publish it under the following licenses:

Permission is granted to copy, distribute and/or modify this document under the terms of the **GNU Free Documentation License**, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled *GNU Free Documentation License*.

This file is licensed under the Creative Commons Attribution-Share Alike 3.0 Unported license.

You are free:

- **to share** – to copy, distribute and transmit the work
- **to remix** – to adapt the work

Under the following conditions:

- **attribution** – You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.
- **share alike** – If you remix, transform, or build upon the material, you must distribute your contributions under the same or compatible license as the original.

This licensing tag was added to this file as part of the GFDL licensing update.

This file is licensed under the Creative Commons Attribution-Share Alike 2.5 Generic, 2.0 Generic and 1.0 Generic license.

You are free:

- **to share** – to copy, distribute and transmit the work
- **to remix** – to adapt the work

Under the following conditions:

- **attribution** – You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.
- **share alike** – If you remix, transform, or build upon the material, you must distribute your contributions under the same or compatible license as the original.

Chapter 4 includes an updated version of a published manuscript "Synthetic Data Generation for Engineering Education: A Bayesian Approach" by the author and co-author (Ayisha Necholi, Wilfrido Moreno). This manuscript is available through open access, and the corresponding IEEE copyright and consent form is included below.

Chapter 5 includes a revised edition of the published manuscript titled "Leveraging Multi-Criteria Integer Programming Optimization for Effective Team Formation," authored by Pallavi Singh and co-author by Phat K. Huynh, Dang Nguyen, Trung Q. Le, and Wilfrido

CCC | RightsLink

Sign in/Register

IEEE Requesting permission to reuse content from an IEEE publication

Leveraging Multi-Criteria Integer Programming Optimization for Effective Team Formation

Author: Pallavi Singh

Publication: IEEE Transactions on Learning Technologies

Publisher: IEEE

Date: 16 May 2024

Copyright © 2024, IEEE

Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

BACK

CLOSE WINDOW

Moreno. This manuscript is available through open access, and the corresponding IEEE copyright and consent form is included below.

IEEE COPYRIGHT AND CONSENT FORM

To ensure uniformity of treatment among all contributors, other forms may not be substituted for this form, nor may any wording of the form be changed. This form is intended for original material submitted to the IEEE and must accompany any such material in order to be published by the IEEE. Please read the form carefully and keep a copy for your files.

Synthetic Data Generation for Engineering Education: A Bayesian Approach

Pallavi Singh,Ayisha Necholi,Wilfrido Moreno

2023 IEEE 3rd International Conference on Advanced Learning Technologies on Education & Research (ICALTER)

U.S. GOVERNMENT EMPLOYEE CERTIFICATION AND TRANSFER OF INTERNATIONAL COPYRIGHT

This will certify that all authors of the Work are U.S. Government employees and prepared the Work on a subject within the scope of their official duties. As such, the Work is not subject to U.S. copyright protection. In order to enable IEEE to claim and protect copyright rights in international jurisdictions, the undersigned hereby assigns to the IEEE all such international rights under copyright that may exist in and to the above Work, and any revised or expanded derivative works submitted to the IEEE by the undersigned based on the Work.

GENERAL TERMS

1. The undersigned represents that he/she has the power and authority to make and execute this form.
2. The undersigned agrees to indemnify and hold harmless the IEEE from any damage or expense that may arise in the event of a breach of any of the warranties set forth above.
3. The undersigned agrees that publication with IEEE is subject to the policies and procedures of the IEEE PSPB Operations Manual.
4. In the event the above work is not accepted and published by the IEEE or is withdrawn by the author(s) before acceptance by the IEEE, the foregoing copyright transfer shall be null and void. In this case, IEEE will retain a copy of the manuscript for internal administrative/record-keeping purposes.
5. For jointly authored Works, all joint authors should sign, or one of the authors should sign as authorized agent for the others.
6. The author hereby warrants that the Work and Presentation (collectively, the "Materials") are original and that he/she is the author of the Materials. To the extent the Materials incorporate text passages, figures, data or other material from the works of others, the author has obtained any necessary permissions. Where necessary, the author has obtained all third party permissions and consents to grant the license above and has provided copies of such permissions and consents to IEEE

You have indicated that you DO wish to have video/audio recordings made of your conference presentation under terms and conditions set forth in "Consent and Release."

CONSENT AND RELEASE

1. In the event the author makes a presentation based upon the Work at a conference hosted or sponsored in whole or in part by the IEEE, the author, in consideration for his/her participation in the conference, hereby grants the IEEE the unlimited, worldwide, irrevocable permission to use, distribute, publish, license, exhibit, record, digitize, broadcast, reproduce and archive, in any format or medium, whether now known or hereafter developed: (a) his/her presentation and comments at the conference; (b) any written materials or multimedia files used in connection with his/her presentation; and (c) any recorded interviews of him/her (collectively, the "Presentation"). The permission granted includes the transcription and reproduction of the Presentation for inclusion in products sold or distributed by IEEE and live or recorded broadcast of the Presentation during or after the conference.
2. In connection with the permission granted in Section 1, the author hereby grants IEEE the unlimited, worldwide, irrevocable right to use his/her name, picture, likeness, voice and biographical information as part of the advertisement, distribution and sale of products incorporating the Work or Presentation, and releases IEEE from any claim based on right of privacy or publicity.

BY TYPING IN YOUR FULL NAME BELOW AND CLICKING THE SUBMIT BUTTON, YOU CERTIFY THAT SUCH ACTION CONSTITUTES YOUR ELECTRONIC SIGNATURE TO THIS FORM IN ACCORDANCE WITH UNITED STATES LAW, WHICH AUTHORIZES ELECTRONIC SIGNATURE BY AUTHENTICATED REQUEST FROM A USER OVER THE INTERNET AS A VALID SUBSTITUTE FOR A WRITTEN SIGNATURE.

[redacted] [redacted]

Signature Date (dd-mm-yyyy)

Information for Authors

AUTHOR RESPONSIBILITIES

The IEEE distributes its technical publications throughout the world and wants to ensure that the material submitted to its publications is properly available to the readership of those publications. Authors must ensure that their Work meets the requirements as stated in section 8.2.1 of the IEEE PSPB Operations Manual, including provisions covering originality, authorship, author responsibilities and author misconduct. More information on IEEE's publishing policies may be found at http://www.ieee.org/publications_standards/publications/rights/authorrightsresponsibilities.html Authors are advised especially of IEEE PSPB Operations Manual section 8.2.1.B12: "It is the responsibility of the authors, not the IEEE, to determine whether disclosure of their material requires the prior consent of other parties and, if so, to obtain it." Authors are also advised of IEEE PSPB Operations Manual section 8.1.1B: "Statements and opinions given in work published by the IEEE are the expression of the authors."

RETAINED RIGHTS/TERMS AND CONDITIONS

- Authors/employers retain all proprietary rights in any process, procedure, or article of manufacture described in the Work.
- Authors/employers may reproduce or authorize others to reproduce the Work, material extracted verbatim from the Work, or derivative works for the author's personal use or for company use, provided that the source and the IEEE copyright notice are indicated, the copies are not used in any way that implies IEEE endorsement of a product or service of any employer, and the copies themselves are not offered for sale.
- Although authors are permitted to re-use all or portions of the Work in other works, this does not include granting third-party requests for reprinting, republishing, or other types of re-use.The IEEE Intellectual Property Rights office must handle all such

third-party requests.
- Authors whose work was performed under a grant from a government funding agency are free to fulfill any deposit mandates from that funding agency.

AUTHOR ONLINE USE

- **Personal Servers**. Authors and/or their employers shall have the right to post the accepted version of IEEE-copyrighted articles on their own personal servers or the servers of their institutions or employers without permission from IEEE, provided that the posted version includes a prominently displayed IEEE copyright notice and, when published, a full citation to the original IEEE publication, including a link to the article abstract in IEEE Xplore. Authors shall not post the final, published versions of their papers.
- **Classroom or Internal Training Use.** An author is expressly permitted to post any portion of the accepted version of his/her own IEEE-copyrighted articles on the author's personal web site or the servers of the author's institution or company in connection with the author's teaching, training, or work responsibilities, provided that the appropriate copyright, credit, and reuse notices appear prominently with the posted material. Examples of permitted uses are lecture materials, course packs, e-reserves, conference presentations, or in-house training courses.
- **Electronic Preprints.** Before submitting an article to an IEEE publication, authors frequently post their manuscripts to their own web site, their employer's site, or to another server that invites constructive comment from colleagues. Upon submission of an article to IEEE, an author is required to transfer copyright in the article to IEEE, and the author must update any previously posted version of the article with a prominently displayed IEEE copyright notice. Upon publication of an article by the IEEE, the author must replace any previously posted electronic versions of the article with either (1) the full citation to the IEEE work with a Digital Object Identifier (DOI) or link to the article abstract in IEEE Xplore, or (2) the accepted version only (not the IEEE-published version), including the IEEE copyright notice and full citation, with a link to the final, published article in IEEE Xplore.

Questions about the submission of the form or manuscript must be sent to the publication's editor.
Please direct all questions about IEEE copyright policy to:
IEEE Intellectual Property Rights Office, copyrights@ieee.org, +1-732-562-3966

About the Author

Pallavi Singh received a Bachelor's degree in Electronics and Communication Engineering from Guru Nanak Dev Engineering College (GNDEC), Bidar, India, in 2016 and a Master's degree in Electrical Engineering from University of South Florida, Tampa, FL, USA, in 2019. Since earning her bachelor's degree, Pallavi has worked in various roles including data science engineer, embedded systems engineer, computer vision engineer, systems engineer, and project manager. Pallavi is currently a Ph.D. candidate in Electrical Engineering from the University of South Florida and is expected to graduate in the summer of 2024. Her main areas of research center around Machine Learning applications for Engineering Education, with the goal of enhancing students' performance and learning. This includes Digital Engineering, Model-Based Systems Engineering (MBSE), and Educational Data Mining. In addition, She is interested in research application of Machine Learning across diverse sectors like automotive, healthcare, education, and manufacturing.