RESEARCH ON THE INTEGRATION OF ADVANCED MATHEMATICS TEACHING AND COMPUTER MAJOR COMPETENCY CULTIVATION

Gao xia
School of Mathematics and statistics, JiNing Normal University,Ulanqab, 012000, Inner Mongolia, P. R. China
Teaching Reform and Research Project of Jining Normal University: Teaching Reform and Exploration of Public Mathematics Courses Based on Professional Orientation (JGKT2024039)
https://doi.org/10.5281/zenodo.19573245

Abstract
Advanced Mathematics is a core foundational course for computer majors. Its cultivation of logical thinking, abstract modeling, and algorithm analysis capabilities directly supports the quality of talent training for computer professionals. Under the background of education in the new era, the organic integration of Advanced Mathematics knowledge with the cultivation of computer professional abilities and the immediate realization of curriculum value has become an important direction for the reform of Advanced Mathematics courses in computer-related majors. Guided by the cultivation of computer professional abilities, this paper discusses the teaching of Advanced Mathematics from the perspectives of teaching content reconstruction, teaching method innovation and assessment method reform.

Keywords: Advanced Mathematics; computer major; professional ability; teaching reform

1 Introduction

In the era of digital economy and intelligent technology, computer technology has penetrated into all walks of life. As a core basic course for computer-related majors, Advanced Mathematics is not only a theoretical prerequisite for students to understand follow-up courses such as data structures, artificial intelligence, computer graphics, cryptography, etc., but also a key carrier for cultivating students' logical reasoning, abstract thinking and problem-solving abilities [1]. However, the current teaching of Advanced Mathematics in computer majors at colleges and universities generally suffers from the following problems: outdated content system, monotonous teaching methods, rigid assessment modes, and insufficient learning motivation among students. The fundamental reason is that students fail to perceive the connection between mathematics and their major during their studies, and cannot immediately realize the important role of mathematics in professional learning.

This study reconstructs teaching content, innovates teaching methods, and reforms assessment modes, which can effectively improve computer majors' ability in mathematical application, computational thinking and engineering practice. It aims to cultivate innovative talents with solid mathematical foundations for the computer field, and provide references for mathematics teaching reform in other engineering majors. Furthermore, it enriches the theoretical system of the integration of basic courses and professional courses in colleges and universities, offers a theoretical framework and practical approach for collaborative education between Advanced Mathematics and computer majors, and promotes the innovative development of mathematics education and teaching theories.

Foreign universities have long attached importance to the integrated teaching of mathematics and computer science. For example, the Massachusetts Institute of Technology (MIT) in the United States has incorporated a large number of programming practices and numerical computation contents into the mathematics courses of computer-related majors, emphasizing the combination of mathematical modeling and computer implementation [2]. Universities such as the University of Oxford and the University of Cambridge in the UK focus on cultivating students' computational thinking and algorithm design ability in mathematics teaching. In addition, foreign scholars began exploring the disciplinary connection between mathematics and computer science at an early stage. They clarified that mathematics courses should serve as core prerequisites for computer-related majors, and that mathematics teaching content should be optimized in light of the application demands of the computer science discipline, so as to avoid focusing merely on sorting out theoretical premises while neglecting practical application [3]. Internationally, there are also academic journals dedicated specifically to the integration of mathematics and computer science, which continuously publish relevant research findings, promote theoretical and practical innovations in the integration of the two disciplines, and emphasize that the coordinated development of mathematical theories and computer applications constitutes an important foundation for modern scientific and technological progress [4].

Many scholars in China have also explored the integration of advanced mathematics and computer science. Some studies focus on the optimization of teaching content, such as incorporating numerical computation and the application of mathematical software [5]; others are concerned with the innovation of teaching methods, including project-based learning and flipped classrooms [6]; and some emphasize the reform of assessment approaches, for example, strengthening process assessment and experimental assessment [7]. Most of these studies only concentrate on reforms in a single dimension, lacking systematic and in-depth research on integration from the three dimensions of content, methods and assessment. Furthermore, practical cases targeting computer majors in local undergraduate universities are relatively insufficient.

2 Definition of the Connotation of Advanced Mathematics, Computer Professional Competencies and Teaching Integration

(1) Advanced mathematics is a basic theoretical course for computer-related majors. It mainly includes

functions and limits, derivatives and differentials, integration, differential equations, infinite series, vector algebra and spatial analytic geometry. It serves as an important theoretical tool for students to learn computer professional courses.

(2) Computer professional competencies cover computational thinking, algorithm design and analysis ability, programming implementation ability, data processing and analysis ability, mathematical modeling ability, engineering practice ability and other aspects, which are the core qualities that computer majors must possess.

(3) Teaching integration refers to the organic integration of the knowledge system and thinking methods of advanced mathematics with the competency requirements, curriculum content and practical links of computer majors, so as to form a mutually supporting and mutually promoting teaching system and realize the coordinated development of mathematics teaching and professional competency cultivation.

3 Reconstruction of Teaching Content

Constructing a mathematics knowledge system deeply integrated with computer majors is the foundation for integrating advanced mathematics and computer professional competencies. It is necessary to break the knowledge framework of traditional mathematics courses, carry out modular and scenario-based design according to the application requirements of computer majors, and achieve precise alignment between mathematical knowledge and professional competencies [8].

3.1 Division of Teaching Modules Oriented to Computer Application Scenarios

Abandoning the traditional chapter-based content organization, this study divides advanced mathematics into the following modules by combining core computer courses and practical application scenarios:

3.1.1 Function and Limit Module

Corresponding Content: Convergence of numerical algorithms, iteration methods, function approximation, image gray value transformation, etc.

Key Explanations: Definition and properties of limits, continuity of functions, limit calculation methods, combined with computer-related problems such as convergence analysis of algorithms and stability analysis of iterative processes.

Additional Content: Numerical approximation methods of functions, simple algorithm implementation cases based on limits.

Case 1: Calculation of the Circumference Ratio π Using the Idea of Limits

When explaining "sequence limits and the idea of the cyclotomic method" in class, to help students grasp the limit concept of "infinite approximation and approaching a fixed value", a computer can be used to approximate the limit through finite iterations.

First, the algorithmic implementation of the idea of limits in a computer is given as follows:

(1) Set the circle radius r=1;

(2) Adopt regular n-sided polygons (n=3,6,12,24,48,96,…);

(3) Calculate the perimeter using the side length formula;

(4) Obtain the approximate value of π by dividing the perimeter by 2;

(5) The larger n is, the closer the result is to π.

The process of calculating the circumference ratio π through finite iterations with the help of a computer is shown in Figure 1:

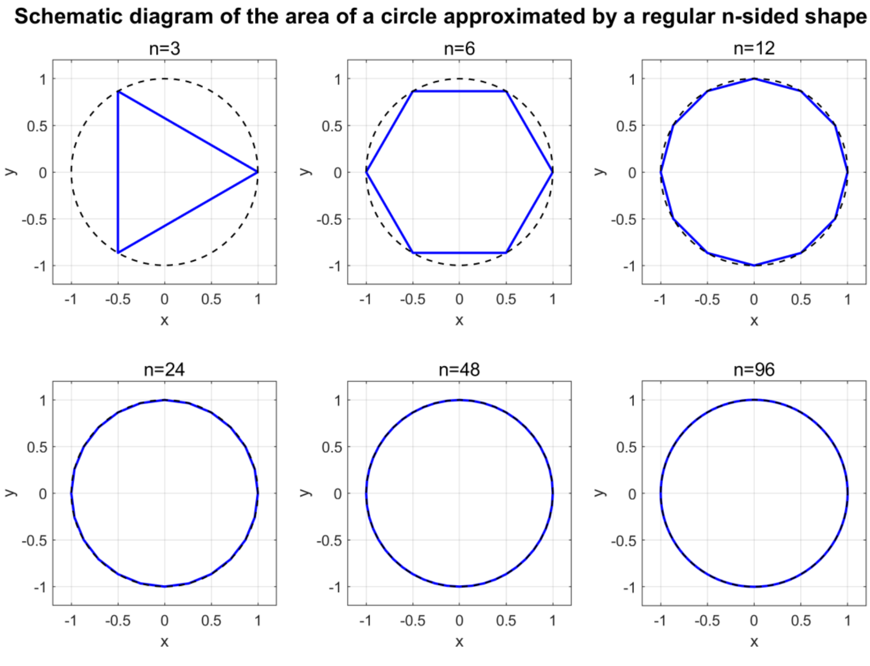

Figure 1：Schematic diagram of the area of a circle approximated by a regular n-sided shape

Finally, let the students observe the results：

n=6 π≈3.0000

n=12 π≈3.1058

n=96 π≈3.1410

n=10000 π≈3.1415926…

Such a mathematics class can not only improve students' mathematical application ability, that is, students can apply mathematical knowledge in the process of solving computer professional problems, thereby enhancing their mathematical application ability and innovative thinking. At the same time, it can also enhance students' learning motivation and interest. When students realize the importance of advanced mathematics in the study of computer majors and can connect mathematical knowledge with practical applications, their learning enthusiasm and initiative will be significantly improved, thus improving learning effects.

3.1.2 Derivative and Differential Module

Corresponding Content: Optimization algorithms (such as gradient descent), loss functions in machine learning, image edge detection, speed and acceleration calculation, etc.

Key Explanations: The definition and geometric meaning of derivatives, the calculation method of differentials, the application of higher-order derivatives, which are explained in combination with the principle of gradient descent, the derivation of loss functions, the mathematical principle of image edge detection, etc [9].

Additional Content: Derivative derivation of common optimization algorithms, implementation of simple optimization programs based on derivatives.

3.1.3 Integral Module

Corresponding Content: Numerical integration, area and volume calculation, data feature extraction, integral transformation in image processing, etc.

Key Explanations: The calculation methods of definite integrals and indefinite integrals, the applications of multiple integrals, line integrals and surface integrals, which are explained in combination with numerical integration algorithms, integral calculation of image pixels, integral feature extraction of data, etc.

Additional Content: Programming implementation of numerical integration, application cases of integrals in image processing.

3.1.4 Differential Equation Module

Corresponding Content: System modeling, dynamic process simulation, dynamic analysis of neural networks, epidemic models, etc.

Key Explanations: The solution methods and stability analysis of ordinary differential equations, which are explained in combination with simple system modeling, numerical simulation of dynamic processes, dynamic behavior analysis of neural networks, etc.

Additional Content: Numerical solution methods of differential equations, modeling and simulation cases based on differential equations.

3.1.5 Infinite Series Module

Corresponding Content: Data compression, signal processing, activation functions of neural networks, polynomial approximation, etc.

Key Explanations: The convergence of series, the applications of power series and Fourier series, which are explained in combination with the principle of data compression, spectral analysis of signals, approximation of activation functions of neural networks, etc.

Additional Content: Application cases of series in data processing and signal processing.

3.2 Integration of Computer-Related Elements and Practical Cases

In each teaching module, computer-related elements and practical cases are proactively integrated to closely link mathematical knowledge with professional computer applications.

3.2.1 Introduction of Typical Computer Cases

Typical problems in the computer field are selected as teaching cases, such as calculating the area of an image using integration, analyzing the time complexity of sorting algorithms with derivatives, and establishing infectious disease transmission models via differential equations. Through case-based teaching, students can intuitively perceive the application value of mathematical knowledge in the computer discipline. This is also consistent with relevant foreign scholars' research on the application of mathematical models in the computer field [10].

Case 2: Analyzing the Time Complexity of Sorting Algorithms Using Derivatives

In computer science, the time complexity of an algorithm describes how its running time evolves with the input size n. By analyzing the growth rates of functions using mathematical derivatives, the merits of different algorithms can be rigorously established in theory.

(1) Define the algorithm complexity function

The time complexity function of quadratic-order algorithms (bubble sort, insertion sort) is denoted as $T_1(n) = n^2$, and the time complexity function of logarithmic-order algorithms (quick sort, merge sort) is denoted as $T_2(n) = n\log n$.

(2) Analyze the growth rate based on derivatives

Differentiate the two complexity functions respectively to obtain the growth rate functions: the growth rate of quadratic-order algorithms is $T_1'(n) = 2n$, and the growth rate of logarithmic-order algorithms is $T_2'(n) = \log n + 1$.

(3) Strictly prove the superiority of algorithms

When the input size n is sufficiently large, it is obvious that $2n \gg \log n + 1$ holds, meaning $T_1(n)$ grows much faster than $T_2(n)$.

Thus the conclusion can be drawn: in scenarios with large data sizes, quadratic-order algorithms (bubble sort / insertion sort) are less efficient, while logarithmic-order algorithms (quick sort / merge sort) are more efficient.

3.2.2 Strengthen Computational Thinking and Algorithmic Ideas

In teaching, emphasis is placed on cultivating students' computational thinking, guiding them to transform mathematical problems into algorithmic problems, design reasonable algorithmic steps, and implement them through programming. For instance, when explaining the extreme values of functions, students are guided to design algorithms for solving function extrema and implement them with code.

3.3 Optimize Content Structure and Strengthen Practical Orientation

(1) Remove overly abstract, complex theoretical proofs that have little relevance to the computer discipline, such as proofs of some complicated integral theorems and profound theoretical proofs of infinite series. Shift the focus of teaching to the application and practice of knowledge.

(2) Supplement numerical computation methods, including numerical integration, numerical differentiation, numerical solutions to systems of linear equations, etc., as these contents form the foundation of numerical computation for computer majors. Add basics of mathematical modeling to guide students in using mathematical knowledge to establish mathematical models for computer-related problems, as well as solve and analyze them. Include training in the use of mathematical software and programming tools such as Python, MATLAB, GeoGebra, etc., to improve students' ability to apply mathematical software and their programming skills.

4 Innovation in Teaching Methods

Innovation in teaching methods is the key to integrating advanced mathematics with computer professional competencies. It is necessary to change the traditional single teaching mode of "teachers lecture and students listen", adopt diversified teaching methods, emphasize the combination of theory and practice, and stimulate students' enthusiasm and initiative in learning.

4.1 Problem-Based Learning (PBL): Guiding Mathematics Learning with Real-World Computer Problems

Problem-based learning is driven by real-world problems in the computer field, guiding students to learn and explore mathematical knowledge around these problems. The specific implementation steps are as follows:

(1) **Problem Introduction**: At the beginning of each class, a practical problem related to the computer major is raised to arouse students' interest and thinking. For example, when teaching definite integrals, the question "How to calculate the area of a digital image?" can be posed; when teaching derivatives, the question "How to optimize the loss function of a machine learning model?" can be proposed.

(2) **Problem Analysis**: Guide students to analyze the problem, clarify the mathematical knowledge and computer skills involved in the problem, and define learning objectives.

(3) **Knowledge Inquiry**: Organize students to learn relevant mathematical knowledge centered on the problem, and guide them to apply mathematical knowledge to solve the problem.

(4) **Solution Design**: Encourage students to discuss in groups and design problem-solving schemes, including mathematical modeling, algorithm design, programming implementation, etc.

(5) **Achievement Presentation and Evaluation**: Each group presents its solution; the teacher comments and evaluates the schemes, and summarizes experiences and lessons.

Through problem-driven teaching, students can learn mathematical knowledge in the process of solving practical problems, and improve their mathematical application ability and problem-solving ability.

4.2 Integration of Theory and Practice: Synchronous Mathematical Derivation and Programming Implementation

Integration of theory and practice refers to the organic combination of mathematical knowledge derivation and computer program realization in classroom teaching, so as to achieve an integrated teaching model of "learning mathematics, applying mathematics, and programming". The specific implementation methods are as follows:

(1) **Synchronous Classroom Teaching**: When explaining the derivation of mathematical formulas or theorems, the corresponding computer program implementation is introduced simultaneously. For example, when explaining the principle of gradient descent, derive the calculation method of the gradient while writing simple Python code examples to implement the gradient descent algorithm, allowing students to intuitively perceive the connection between mathematical derivation and program implementation.

(2) **Mathematical Visualization**: Use mathematical software (such as GeoGebra, Matplotlib, etc.) to visualize abstract mathematical concepts, formulas and theorems, helping students understand mathematical knowledge. For instance, when explaining function graphs, draw function images and dynamically demonstrate the change process of functions to enable students to intuitively understand function properties; when explaining the geometric meaning of integrals, dynamically demonstrate the process of dividing and summing integral regions to help students understand the concept of definite integrals.

(3) **After-Class Practical Expansion**: Assign programming practice tasks related to classroom content, allowing students to verify mathematical knowledge through programming after class, so as to deepen their understanding and application of knowledge. For example, after explaining numerical integration, assign tasks for students to implement different numerical integration methods using Python and compare the calculation results.

4.3 Project-Based Learning (PBL): Integrating Mini-Projects Throughout the Course

Project-based learning uses one or more mini computer projects as a carrier to integrate knowledge and skills from advanced mathematics into the project implementation process. Students learn mathematical concepts and improve professional competencies while completing the projects. The specific implementation steps are as follows:

(1) **Project Design**: Based on the teaching content of advanced mathematics, design mini-projects relevant to the computer science major. Examples include: Simple image processing project based on integration ，Function optimization program based on derivatives ，Dynamic system simulation project based on differential equations.

(2) **Grouping**: Divide students into several groups of 3–5 members, with clear roles and responsibilities assigned to each student.

(3) **Project Implementation**: Each group conducts requirements analysis, mathematical modeling, algorithm design, programming, testing, and optimization according to project requirements. The instructor provides guidance and answers questions throughout the process.

(4) **Project Acceptance and Evaluation**: Each group submits deliverables including project reports, source code, demonstration videos, etc. The instructor organizes project presentations and evaluates the outcomes.

Through project-based learning, students can systematically apply advanced mathematics knowledge to solve computer science problems, and enhance their teamwork, project management, and engineering practice abilities.

4.4 Information-Based Teaching: Enhancing Teaching Effect with Intelligent Tools

Full use shall be made of information technology to strongly support the integrated teaching of Advanced Mathematics and the computer specialty.

(1) **Use of online teaching platforms**

Online platforms such as Chaoxing Learning Platform and Rain Classroom are adopted to realize the sharing of teaching resources, interaction of teaching activities, and monitoring of the learning process. Teachers can upload courseware, operation videos of mathematical software, programming cases, assignments and experimental tasks on the platform. Students can watch teaching videos, finish homework, participate in discussions, and submit experimental reports via the platform [11].

(2) **Use of intelligent teaching systems**

Intelligent teaching systems are utilized to provide students with personalized learning guidance and feedback. According to students' learning progress, the system can automatically recommend customized learning materials and exercises, diagnose learning difficulties, and offer targeted solutions.

(3) **Use of mathematical software and programming tools**

Students are encouraged to use mathematical and programming tools including Python, MATLAB and GeoGebra for mathematical learning and practice. Teachers can demonstrate the operation of such software in class and guide students to perform mathematical calculations, graph plotting, and program implementation, so as to improve students' information literacy and practical abilities.

5 Reform of Assessment Methods: Establishing a Diversified Competency Evaluation System

Assessment methods act as the guide for teaching. Scientific and reasonable assessment plays a vital role in guiding student learning, evaluating teaching effectiveness, and achieving training objectives. In response to the teaching goal of integrating advanced mathematics with computer professional competencies, it is necessary to reform traditional assessment methods and establish a diversified competency evaluation system.

5.1 Increasing the Proportion of Process Assessment

The proportion of process assessment will be increased to 50% or above to comprehensively evaluate students' learning process and competency development. Process assessment mainly includes the following components:

(1) Daily Assignment Assessment

Diversified assignments will be assigned, including basic mathematical problems, programming implementation tasks, case analysis questions, and mathematical modeling problems. Through assignments, students' mastery of mathematical knowledge, programming skills, application ability, and innovative thinking will be evaluated.

(2) Class Performance Assessment

Attention will be paid to students' classroom participation, question responses, performance in group discussions, project presentations, etc., so as to assess their learning initiative, communication skills, and teamwork ability.

(3) Experiment Assessment

Special mathematical experiment sessions will be arranged, requiring students to implement mathematical algorithms through programming, verify mathematical conclusions, and solve practical computer problems. Experiment assessment includes experiment reports, operational performance, and result analysis, evaluating students' practical operation ability, data analysis ability, and problem-solving ability [12].

(4) Study Notes and Reflection Assessment

Students will be required to write study notes and learning reflections, summarizing key and difficult points, insights, and areas for improvement, so as to evaluate their autonomous learning ability and reflective thinking.

5.2 Reform of the Content and Form of Summative Assessment

Summative assessment adopts a combination of closed-book written examination and open-book assessment, focusing on evaluating students' comprehensive application ability and innovative ability.

(1) **Optimization of closed-book written examination content**

In the closed-book written examination, purely computational questions will be reduced, while application-oriented and comprehensive questions will be increased. For example, mathematical modeling problems, algorithm analysis problems, programming design problems under computer scenarios will be added to examine students' ability to use mathematical knowledge and computer professional knowledge to solve problems collaboratively.

(2) **Design of open-book assessment content**

An open-book assessment session will be set up, in forms such as mathematical modeling papers, project design reports, comprehensive case analysis reports, etc. Students may refer to materials and use mathematical software and programming tools to complete assessment tasks, which mainly assess their autonomous learning ability, information retrieval ability, innovation ability and engineering practice ability.

Advanced Mathematics is the foundation for computer science students to learn professional courses such as data structures, algorithm analysis, artificial intelligence, and machine learning. It plays an irreplaceable role in cultivating students' logical reasoning, abstract modeling, numerical calculation and systematic thinking.Through reconstructing teaching content, innovating teaching methods and reforming assessment modes, we will deeply integrate curriculum teaching with the specialty, guide students to focus on classroom learning, and improve the quality of classroom teaching.

References:

1. Xie Xuejun, Li Jing. Research on Teaching Reform of Advanced Mathematics Based on the Cultivation of Mathematical Thinking Ability under the Background of Curriculum Ideology and Politics. Journal of Yichun University, 2022: 122-125.
2. Dori Y, Meyer A. Team Problem Solving and Project-Based Learning: Design and Assessment of a Multimodal Approach to Teaching Mathematics for Computer Science[R]. MIT Registrar, 2015.
3. Ralston A. The Role of Mathematics in the Computer Science Curriculum[J]. ACM Digital Library, 1985, 28(3): 243-250.
4. Wamiliana M A. Integra: Journal of Integrated Mathematics and Computer Science[J]. Integra JIMCS, 2026, 3(1): 1-15.
5. Wei Nana. Exploration on the Reform of Advanced Mathematics Teaching Methods and the Cultivation of Mathematical Modeling Ideas. Modern Vocational Education, 2021: 210-211.
6. Zhan Lingchao. Research on Online + Offline Multi-dimensional Interactive Teaching Innovation of Discrete Mathematics Course Based on OBE Concept. Computer Knowledge and Technology, 2023: 167-169.
7. Liu Huicai. Exploration on the Application of Mathematical Modeling Ideas in Advanced Mathematics Teaching Reform. Industrial & Science Tribune, 2022: 134-135.
8. Liang Jing. Some Thoughts on the Teaching Reform of Calculus Classroom. Science & Technology Vision, 2021: 42-43.
9. Yu Ziye. Mathematical Foundation of Deep Learning, Algorithm Models and Practice[M]. 2023.
10. Fujita T. Note for Soft Multi Expert Graph and Multisoft Multi Expert Graph[J]. Integra JIMCS, 2026, 3(1): 6-15.
11. Zheng Xuejing, Huang Donglan, Wang Yiqin. Research on the Reconstruction of Advanced Mathematics Teaching Model Based on "Internet +". Journal of Fujian Institute of Education, 2022: 69-72.
12. Dong Ge. Exploration on Mathematics Experimental Teaching Combined with Medical Majors. Modern Business & Trade Industry, 2021: 138-140.