A CSCL Model for Educational Multiplayer Games

E. Bachu and M. Bernard
Department of Computing and Information Technology, University of the West Indies, Trinidad and Tobago

Abstract - Teaching computer programming to novices has always posed a major challenge to educators. Many novices acquire basic programming skills but they are unable to utilize them in a meaningful way to solve non-routine problems. As a result, they do not achieve any level of programming fluency. Collaboration has been identified as a useful tool to help overcome this problem if utilized at the right times during the programming/problem solving process. This paper presents a model for a synchronous gaming CSCL (Computer Supported Collaborative Learning) application aimed at improving problem-solving skill. The model addresses some major issues in CSCL such as unequal participation among group members, increasing argumentative discussion and promoting positive interdependence and individual accountability. A game, Collaborative Online Problem Solving (COPS), which was developed using this model, is presented. COPS aims to improve the problem-solving skill of novice programmers, by requiring them to collaboratively build program flowcharts.

Keywords: CSCL, collaborative learning, problem solving, programming, educational game, metacognition

1 Introduction

Learning to program is a difficult process for many students; this is reflected in the high failure rates in introductory programming courses around the world. In Trinidad and Tobago, and by extension the Caribbean region, secondary school students write the Caribbean Secondary Education Certificate (CSEC) Information Technology exam. This syllabus contains two sections on problem solving and programming in Pascal and annual reports produced by CXC (Caribbean Examination Council) have continuously indicated the difficulties faced by students with these two sections. This is especially worrying since programming lies at the core of information technology and increasing attention has been given to producing students who are considered fluent in information technology, most recently highlighted in [1]. In [2] and [3], the authors identify low problem solving skill as one of the main reasons for this high failure rate. The results presented in [4] also highlight the designing of a program to solve a particular task as a major difficulty faced by novice programmers. Computer Supported Collaborative Learning (CSCL) has been identified as one of the most promising innovations to improve teaching and learning with the help of modern information and communication technology [5] since it aims to enhance learning by combining computer support and collaborative learning [6]. This paper proposes the design of an online multiplayer game, COPS (Collaborative Online Problem Solving), which utilizes CSCL principles to assist students with their problem solving and program design by strengthening their problem solving ability. Most of the successful CSCL systems have employed the use of existing technologies such as forums, whiteboards, learning managements systems and shared workspaces asynchronously but COPS encourages synchronous collaboration between learners. COPS is developed based on a model which we introduce for a synchronous gaming CSCL application. It addresses major issues in CSCL such as unequal participation among group members, increasing argumentative discussion, promoting positive interdependence and individual accountability.

2 Background Review

2.1 Collaborative Learning and Programming

Collaborative learning is an instruction method in which students work in groups toward a common academic goal [7]. Some of the major achievements of collaborative learning as described by [8] include (1) Motivation since the students are driven by a reward or goal and they recognize that they can only achieve their goals if the other members of the group succeed. (2) Social Cohesion implies that the students may actually care about the other members of their group. (3) Development since each member will be exposed to the different abilities and viewpoints of the individuals in the group and (4) Cognitive Elaboration since each member of the group will be required to explain their solutions in a social context and they benefit from having to provide their explanations.

In [9], the following criteria are presented for tasks which are deemed applicable to collaborative learning:

- The task is complex or conceptual
- Problem Solving is desired
- Divergent thinking or creativity is desired
- Mastery or retention is important
- Quality of performance is expected
- Higher level reasoning strategies and critical thinking are needed
All of the above criteria clearly apply to computer programming and real world applications of programming are indeed collaborative. In [10], the following five critical attributes for successful collaborative learning are given:

1. Common Task or learning activity
2. Small group learning
3. Co-operative behavior
4. Positive Interdependence
5. Individual Accountability and responsibility

The author in [11] used the above framework to analyze the pair programming pedagogy and concludes that pair programming is a model for collaborative learning. Pair programming research has had many successes including higher quality programs being written [12], decreased time to complete programs [12], and improved performance on exams.

An experiment conducted in [13] also concluded that collaboration is an important pedagogy to use in teaching computer science and in performing java programming, their experiment investigated the influence of collaboration through pair programming and groups of three or four students. Collaboration was deemed to be most important in the ‘brainstorming and formulating the problem’ step of the programming process. This result follows with research that the major cause of students’ failure in introductory programming is the lack of basic problem solving skill [2]. The author in [14] describes three basic attributes which are required by a successful problem solver; basic skill / cognition which can be thought of as individual learning objectives; metaskill / metacognition which refer to knowing when and how to use the basic skills and motivation.

2.2 Computer Supported Collaborative Learning

Computer supported collaborative learning (CSCL) aims to improve the collaborative learning experience by utilizing the rapidly evolving technology available to students in the classroom. Originally, collaborative learning was mainly adopted in classroom based environments which required face to face interaction between students and lecturers but web based implementations of CSCL eliminates the need for this physical interaction. Although there has been an abundance of research in collaboration and programming in the pair programming pedagogy, there is still the opportunity for exploration of CSCL and programming. The turtle graph system [15] uses collaboration to help teach recursion and has shown considerable success. A study involving the KnowCat system [19] concluded that some students’ metacognitive skills increased after using a CSCL environment and that the use of CSCL systems can enhance the development of metacognitive learning processes. Findings from [4] show that novice programmers prefer practical programming sessions and collaborating with lecturers or other students and this indicates that a CSCL application of programming can be very successful.

2.3 Using Games to Teach Programming

In [17], the authors argue that traditional teaching of computer science education is not well suited to millennial students and they suggest that collaborative educational games can make the experience better for both students and educators. A work in progress attempt at a CSCL video game is given in [18] and the authors in [19] agree that the principles of CSCL and problem solving can be applied to multi-player games. Most recently, [20] agrees that online multiplayer games are an attractive and useful avenue for developing educational games. The use of games to teach introductory computer science and computer programming is well documented [21]. A game environment is described as one in which the concepts that emerge from interacting with it are created by the goal [22]; the authors recognize the highly abstract nature of core programming concepts and suggests that games which are successful at teaching programming are those which causes the learner to develop and understand concepts from the content of the game as a consequence of its system and interface. Traditional learning approaches using computer support have always utilized user or learner centered design since the learner is identified as the main component but since we are dealing with a group of students and not an individual, [23] recommends that we utilize task or activity centered design principles and this is easily achieved by the use of a game which is designed around completing the objectives of the game.

3 Game Design

3.1 Overview

In [3], the authors present a six step model which describes the computer programming / problem solving process; formulating the problem, planning the solution, designing the solution, translation, testing and delivery. The first three steps are those which present the toughest task for novices since it requires the problem solving ability which they lack. Two common tools which are used during these steps are pseudocode and flowcharts. Pseudocode is a notation for programming which uses a combination of semi-structured programming structures and verbal instructions. Flowcharts are a visual representation of program flow using a combination of arrows and symbols to represent the actions and sequence of the program. An experiment conducted in [24] overwhelmingly indicated that students preferred flowcharts to pseudocode for understanding algorithms.

A web-based multiplayer game, COPS (Collaborative Online Problem Solving) has been developed which would allow students to collaboratively solve flowchart puzzle based problems. Flowcharts were chosen because they depict the sequence of the program as well as the actions; these actions are the basic skills / cognition which programmers require and the sequence of the flowchart represents the metaskill / metacognition which show how the basic skills are used in a
meaningful way to solve a problem. However, the use of pseudocode is very important in helping students translate their solution into program code and as such pseudocode was utilized throughout the game to provide a guide for the players while solving the problem. Fig. 1 gives a screenshot of the COPS prototype. The area in the middle represents the playing area where players build the flowchart using the pieces from the container in the bottom right. The target users of COPS are secondary school students (ages 13-17) and as such, a web based implementation was chosen because of the increasing familiarity of adolescents with online multiplayer games such as World of Warcraft. This implementation would also allow the students to collaborate outside of the classroom where they may be more comfortable and express themselves more freely. Like other online multiplayer games, COPS forces synchronous communication between players [25] which is essential for knowledge construction in CSCL.

3.2 Gameplay

COPS requires a group of between two to four players/students. The group will be required to create a flowchart to solve a given problem within a specified time. The game will have three levels; beginner, intermediate and expert. At the beginner level, the students will be given all the pieces of the flowchart (symbol and text within symbol) required to solve the problem and they simply have to build the flowchart like a jigsaw puzzle placing the pieces in the correct order. At the intermediate level, the students will be given the flowchart in incorrect order like a scrambled picture and they will be required to re-arrange the pieces to form the correct flowchart. At the expert level, the student will be given the flowchart symbols and the text for the symbols separately and the students will be required to match the text to the correct symbol and build the flowchart. Additionally, the students will be given extra symbols and text which do not form part of the solution.

4 CSCL Gaming Application Model

COPS has been developed using a model which we propose that takes into account the research found in the literature about CSCL. The model identifies the attributes of the game design that should be incorporated to make a successful CSCL environment for teaching problem solving and program design. A notable feature of this model, shown in Fig. 2 is the encouraging of collaboration between the learners, as opposed to cooperation. Using this model, each learner does not work on individual parts of the problem; instead the entire group works on the problem synchronously. The model describes the necessary input components for the game: leaner task, leaner groups, and game design and characteristics. A scripted game cycle is then given in which the group is forced to collaborate after a user action which leads to group actions. The game then updates to provide system feedback in response to the user and group actions and also guide the players towards achieving the task. Finally the game must provide output to both the learner and the educator. The learner should have accomplished some level of mastery of the skills learnt throughout the game. The educator will be provided with data regarding the individual and group performance, collaboration and participation. This data is important to allow the educator to refine the learning process to continuously produce better results. We will discuss this model from the perspective of the COPS multiplayer game.

4.1 Learning Task

Each task in COPS targets a specific skill such as read/write, selection or iteration. This approach is necessary to ensure that the students master these skills so they will be able to utilize them in a more meaningful way to solve other non – routine problems. For example, in the beginner level, the students will be instructed on which iteration structure to use such as ‘while’ or ‘for’ loop but in the advanced levels, they will simply be asked to solve the problem.

4.2 Learner Groups

Each group contains between two to four students who are familiar with each other and can interchangeably play within different groups. The players are expected to be novices who are at the same programming competency level. All members of the group will be working on the same puzzle simultaneously. The game has a chat facility to help facilitate communication. Puzzles have traditionally been attempted collaboratively and real world programming is a collaborative process.

4.3 Participation and Argument

In [26], the importance of equal participation by all members of the group and argumentative discussion in the collaborative learning process is highlighted. The lack of dialogue between the members of the group in CSCL is documented in [27] and we expect to see these problems since it is prevalent in group work where certain members of the group take full responsibility and other members do not participate. These problems were addressed in the game through multiple features. Firstly, the turn based design of COPS ensures that each member of the group participates in the problem solving process, when a user makes a move; the other members of the group are polled by the game asking if everyone agrees.
Create a flowchart which prompts the user to enter a number and prints the square of that number.

**Player Two**
- Agree
- Disagree
- Vote

**Poll Group**

Figure 1. Screenshot of COPS

---

**Scripted Game Cycle**

- **Game design and characteristics**
  - Engender or embody educational content
  - Motivate students to learn the educational content
  - Should not overwhelm educational content
  - Promote positive interdependence
  - Encourage Argumentative discussion
  - Encourage Equal participation
  - Promote Individual Accountability

- **User actions**

- **Group collaboration**
- **System feedback**

- **Learner groups**
  - Small groups, ideally between two and four members.
  - Familiar with each other.
  - All members at same competency level.

- **Learning task**
  - Suitable for group work.
  - Not abstract or ambiguous.
  - Target a specific skill.

- **Learner outcomes**
  - Mastery of specific skills
  - Meta cognition – knowing how to use specific skills to solve non-routine problems
  - Game statistics and rewards

- **Educator outcomes**
  - Player Participation statistics related to both quality and quantity of participation
  - Student performance on specific tasks

Figure 2. Model describing a gaming application of CSCL.
If the poll receives a positive result, the move is allowed else the game rejects the move and the game moves to the next player. To avoid cases of split votes, the player who made the move automatically receives a higher weighted positive vote than the other players. Additionally, when it is a player’s turn, they can choose to undo previous moves before making their own move. The turn based design also lets students know that in order to solve the puzzle and win the game, every member of the team must succeed as well; this is referred to as positive interdependence. This forces the group to care about the decisions and learning process of each other.

This component of COPS which polls the members of the group will not only encourage greater participation by all group members but it will also encourage argumentative discussion between the players using the game’s chat facility and the undoing of previous players’ moves. Each player must convince their group members of their decision if the entire team is to succeed therefore; each member of the group becomes accountable for their decisions. In [29], the authors recognize that each member of the group’s participation can be measured by counting the number of chat entries made but the epistemic value of the member’s contribution cannot be easily obtained from the chat logs. The epistemic dimension is not concerned with the quantity of participation but the content and value of the member’s discussion related to solving the puzzle. The result of the player polls and the tracking of undoing of previous player’s moves (both correct and incorrect undoing) provide a better opportunity to measure the epistemic contribution. COPS keeps track of this data and it can be used by educators to improve the learning of students.

4.4 Coordination and Guidance

In [28], the author recognizes the importance of proper coordination in CSCL environments to ensure that the individual efforts of the group members contribute to the learning task. This coordination is usually provided by the technology itself acting as a mediator by providing chat facilities and shared workspaces but COPS takes a more interactive approach. It is expected that the entire group may get stuck at a certain step and this can be dangerous to the learning process. To help prevent this scenario, pseudocode matching the flowchart solution being constructed will be automatically generated; the students will be able to view the pseudocode version of their solution and more easily recognize where they have gone wrong in their solution. Another aspect of the game which will guide the learner’s process is an accuracy indicator for the puzzle solution; when the group has placed a portion of the puzzle in the correct sequence, that portion will change color and players will not be able to alter any of the pieces within that portion.

4.5 Motivation and Positive Interdependence

The authors in [29] highlight the importance of motivation in computer programming courses due the uniquely demanding requirements of learning to program and motivation has also been identified as crucial to a problem solver. The use of games to provide motivation in learning and programming is well documented in [2] and therefore the game based design of COPS will provide motivation to students. In COPS, the players are only rewarded when the entire group succeeds and all members of the group are rewarded equally. This will motivate the individual members to do their best and also encourage and help the other members of the group to do their best; this also promotes positive interdependence. Additionally in COPS, the group will be provided with a target number of moves in which to complete the puzzle and if they do, the entire group will receive bonus points. This feature of the game is intended to motivate the players to collaborate more with their peers to ensure that the best possible moves are made throughout the game. Additionally, COPS maintains the individual score for each player for all games played (possibly with different groups). The current highest score is displayed so players can set themselves the goal of becoming or maintaining the lead in the game. This encourages them to play the game often, thereby achieving greater learning.

5 Future Works

The model presented has satisfied the attributes of successful CSCL and is tailored towards improving problem solving skill. COPS has been designed to improve the problem solving ability of novice programmers but innovative ways must be developed to evaluate students after the collaboration has concluded. Another single player game is being developed which will evaluate students based on the learning tasks used throughout COPS. This game will also be designed to help students master their coding skill within a specific programming language. However, it will still be necessary to have formal evaluation exercises such as classroom examinations to measure the student’s improvement against their expected competency level.

6 Conclusion

Low problem solving skill has been identified as one of the main contributors to the difficulties faced by novice programmers. Collaboration, if utilized at the right times during the programming/problem solving process can help solve this problem as evident in research performed in pair programming. CSCL provides an avenue for enhancing this collaborative learning through the use of technology. In this paper, a model integrating the various elements of CSCL was
presented. The multiplayer game, COPS, that was described adheres to the principles of collaborative learning and provides the cognition, metacognition and most importantly the motivation which is required by successful problem solvers in the programming domain. The various attributes of COPS were designed to force collaboration within a group in an effort to solve a problem while motivating them to ensure that the entire group succeeds since their own success lies within the success of the group. The use of games like COPS has become necessary to appeal to millennial students and help them become better problem solvers and programmers. The CSCL model presented can be applied to any subject domain in which collaborative learning is applicable and problem solving is desired.

7 References


