

# DEVELOPING SECONDARY EDUCATION COMPUTER SCIENCE TEACHERS' TECHNOLOGICAL PEDAGOGICAL CONTENT KNOWLEDGE

*Charoula Angeli, PhD*

University of Cyprus, Cyprus

*Ioannis Ioannou, MS, PhD Candidate*

University of Cyprus and Ministry of Education and Culture, Cyprus

---

## Abstract

In the study herein the authors adopted the framework of technological pedagogical content knowledge (TPCK) and the approach of technology mapping in order to teach secondary education computer science teachers how to teach with technology. During a 15-hour teacher professional development program, teachers learned how to think about the educational affordances of different computer tools, and how to use them to make computer science content more understandable to learners. In addition, teachers learned how to think iteratively about technology, content, and pedagogy in order to design learning activities appropriate for learners' conceptual ecology. The study presents good examples of TPCK in practice by demonstrating teachers' actual instructional artifacts as these emerged through their participation in the teacher professional development program, as well as their evaluations of the program.

---

**Keywords:** Teacher professional development, Computer science teachers, Technological Pedagogical Content Knowledge, Technology Mapping

## Introduction

In an era where new digital technologies are gradually revolutionizing all aspects of daily life, teachers are called once more to respond to the needs of the society and find ways to integrate these technologies in their classroom teaching. The challenge to respond competently to this call is highly demanding as teachers are expected to think beyond the box and learn new things in order to transform their existing classroom practices. A factor that greatly influences teachers' efforts and dispositions to teach with technology is directly related to their participation

in teacher professional development programs (Mouza, 2009). Therefore, ongoing research efforts regarding teachers' professional development in the educational uses of technology are fully warranted.

In accordance with this line of argument, the authors herein describe their effort toward designing and implementing a teacher professional development program for the educational uses of technology in teaching computer science content. The teaching of computer science has been traditionally teacher-centered, ignoring for the most part the interrelations among subject matter, pedagogy, and learners' alternative conceptions about the subject matter of computer science (Gal-Ezer, Vilner, & Zur, 2003; ACM K-12 Task Force Curriculum Committee, 2003; Hazzan, Lapidot, & Ragonis, 2001; Tucker, Deek, Jones, McGowan, Stephenson, & Verno, 2003). During the last few years, an effort has been systematically undertaken in many countries worldwide for the purpose of improving computer science education through the systematic integration of educational technologies (Kadijevich, Angeli, & Shulte, 2013). The emphasis is on approaching the teaching of computer science topics in learner-centered ways taking into consideration learners' misconceptions or difficulties in understanding computer science content, as well as teachers' difficulties in making the computer science content more teachable to the students.

Accordingly, the authors herein adopted the framework of Technological Pedagogical Content Knowledge (TPCK) and the approach of Technology Mapping (TM) (Angeli & Valanides, 2005, 2009, 2013) for the purpose of designing and implementing a 15-hour teacher professional development program for secondary education computer science teachers. The main research purpose of the study is to show good examples of TPCK in practice by demonstrating teachers' actual instructional lessons as these emerged through their participation in the teacher professional development program. Therefore, the authors herein provide examples of learning activities that the teachers designed with the use of educational technologies following the principles of the TPCK framework and the guidelines of TM, and also report on the teachers' evaluations of the specific teacher professional development program.

### **Teacher professional development in the educational uses of technology:**

Educating teachers on how to effectively integrate technology in classroom practices for the purpose of improving education and reforming curricula has been one of the main and continual goals of national and international school reform efforts in various countries (International Society for Technology in Education, 2002; Kozma & Anderson, 2002; Pelgrum, 2001; NCEE, 2007; Mouza, 2009). A variety of approaches have been adopted over the years to prepare and support teachers in integrating

technology in classroom practices, although the results have not always been positive (Harrison et al., 2003; Rodrigues, 2003; Ertmer & Ottenbreit-Leftwich, 2010). A review of the literature also indicates that for the most part teacher professional development has focused on learning how to use various computer tools (NCEE, 2007). The main focus of the learning-how-to-use-tools approach has been on technical skills, and, despite the fact that basic computing skills constitute the cornerstone of computer literacy there have been serious reactions to this approach. The opposition is mainly based on the argument that skills-based courses are not enough for preparing teachers how to teach with technology or how to integrate technology in classroom instruction, simply because they are usually taught in isolation from a pedagogical context (Selinger, 2001; Grossman, Wilson, & Shulman, 1989; Mishra & Koehler, 2006).

The failure of teacher professional development programs to adequately prepare teachers to integrate technology in teaching and learning can be also attributed to various other factors (Cuban, 2000). According to Margerum-Lays and Marx (2003), one major contributing factor is the lack of a conceptual framework to systematically guide the integration of technology into teachers' practices. In view of recognizing that the knowledge base of the teaching profession is not adequately developed to effectively guide the integration of technology in teaching and learning, about a decade ago, researchers around the world set out to develop a conceptual framework to guide teachers' cognition about technology integration. Overwhelmingly, researchers agreed that teachers needed to develop a body of knowledge that has been referred to in the literature as Technological Pedagogical Content Knowledge.

### **Technological Pedagogical Content Knowledge (TPCK):**

TPCK has been conceptualized as an extension of Shulman's (1986, 1987) Pedagogical Content Knowledge (PCK). PCK identifies the distinctive bodies of knowledge for teaching, and constitutes a special amalgam of content, pedagogy, learners, and context (Shulman, 1986). Shulman's (1987) conceptualization of PCK goes beyond teachers' knowledge of subject matter and pedagogy per se, and encompasses the dimension of how to teach and transform content into forms or representations comprehensible to learners, taking always into consideration learners' content-related difficulties. In the literature, there are two theoretical models about the conceptualization of TPCK - the integrative model proposed by Mishra and Koehler (2006) shown in Figure 1, and the transformative model proposed by Angeli and Valanides (2009) shown in Figure 2. The integrative view conceptualizes TPCK as an integrative body of knowledge defined by the intersections between content and pedagogy, content and technology, and

pedagogy and technology. In the transformative model, content, pedagogy, learners, technology, and context are regarded as significant contributors to the development of TPCK, which is regarded as a unique body of knowledge.

In more detail, the integrative view of TPCK, as depicted in Figure 1, is represented in terms of three intersecting circles, one for each distinct knowledge base, namely, content, pedagogy, and technology (Mishra & Koehler, 2006), while its subcomponents, i.e., technological content knowledge (TCK), technological pedagogical knowledge (TPK), and pedagogical content knowledge (PCK) are also clearly depicted in the figure. Empirical work by Mishra and Koehler and other researchers who adopted the integrative view of TPCK (e.g., Harris & Hofer, 2011; Schmidt, Baran, Thompson, Mishra, Koehler, & Shin, 2009; Chai, Koh, Tsai, & Tan, 2011) focused on identifying and measuring instances of TPCK's subcomponents, for example, TPK and TCK. So far, empirical findings from this line of research have been rather discouraging, because of the difficulty to clearly define the boundaries of the different TPCK sub-components (Archambault & Crippen, 2009; Graham, 2011; Voogt, Fisser, Pareja Roblin, Tondeur, & van Braak, 2012).

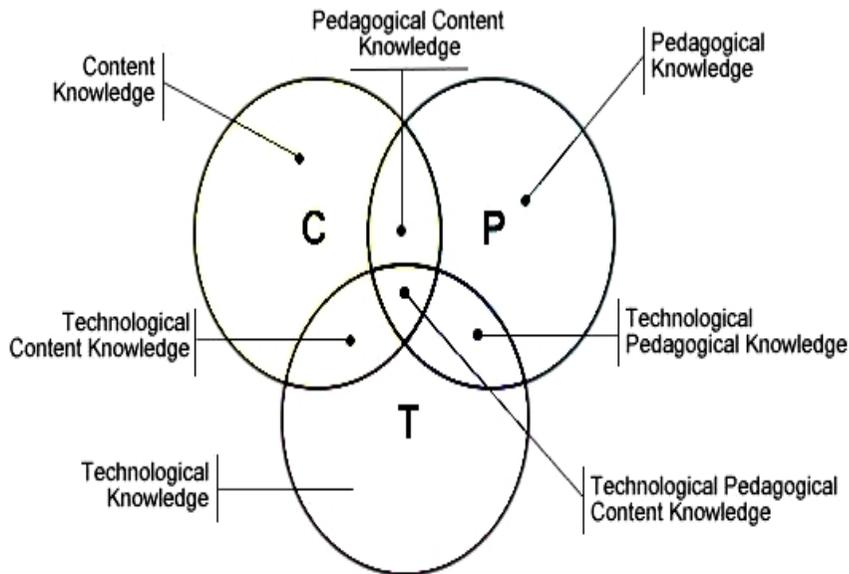


Figure 1. Integrative view of TPCK (adopted from Mishra & Koehler, 2006).

The transformative view of TPCK, as shown in Figure 2, is conceptualized in terms of five distinct knowledge bases, namely, content knowledge, pedagogical knowledge, knowledge of learners, knowledge of

educational context, and ICT knowledge (Angeli & Valanides, 2005, 2009). Based on the results of empirical investigations, Valanides and Angeli (2008a, 2008b) concluded that TPCK is a distinct body of knowledge that goes beyond mere integration or accumulation of the constituent knowledge bases, toward transformation of these contributing knowledge bases into something new and unique.

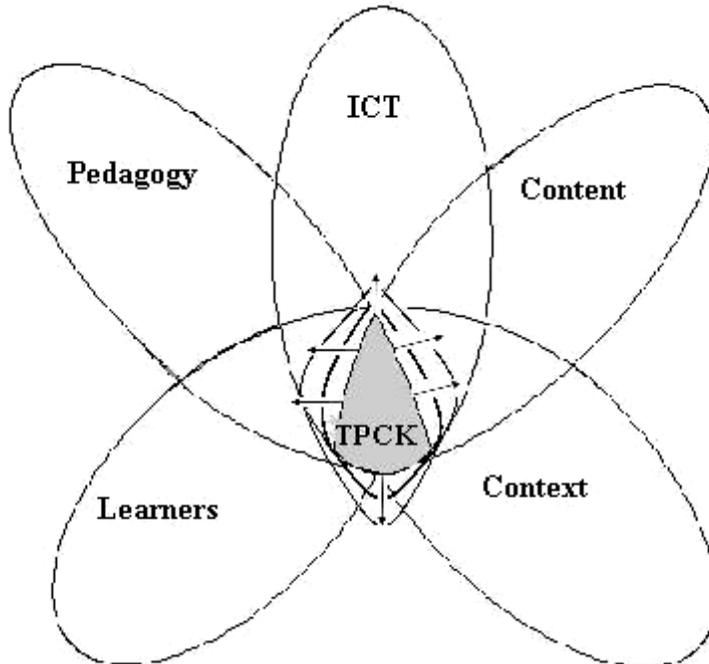


Figure 2. Transformative view of TPCK (adopted from Angeli & Valanides, 2009).

TPCK as a transformative body of knowledge is defined as knowledge about how to transform content and pedagogy with Information and Communication Technology (ICT) for specific learners in specific contexts and in ways that signify the added value of ICT (Angeli & Valanides, 2009). As illustrated in Figure 2, there are a number of individual knowledge bases that contribute to the development of TPCK; however, as it was found in a series of empirical studies, growth in the individual contributing knowledge bases alone, without specific instruction targeting exclusively the development of TPCK, does not result in TPCK growth (Angeli & Valanides, 2005; Angeli, 2005; Valanides & Angeli, 2006, 2008a, 2008b). Angeli and Valanides (2009) also proposed that TPCK, as a unique body of knowledge, is better understood in terms of competencies that teachers need to develop in order to be able to teach adequately with technology. A conceptualization of TPCK in terms of competencies has led to more robust and reliable ways of assessing learners' TPCK, bypassing measurement difficulties of the nature that researchers who adopted the

integrative view of TPCK reported in their studies (Archambault & Barnett, 2010; Cox & Graham, 2009; Graham, 2011; Niess, 2011). It is for this reason, that the Authors herein adopted the transformative view of TPCK as the theoretical framework of their study and concentrated on designing and implementing a teacher professional development program that focused on developing computer science teachers' TPCK competencies, such as:

1. Identify topics to be taught with technology in ways that signify the added value of technology tools, such as, topics that students cannot easily comprehend, or that teachers face difficulties teaching or presenting effectively in class. These topics may include abstract concepts (i.e., cells, molecules) that need to be visualized, phenomena from the physical and social sciences that need to be animated (i.e., water cycle, the law of supply and demand), complex systems (i.e., ecosystems, organizations) in which certain factors function systemically and need to be simulated or modeled, and topics that require multimodal transformations (i.e., textual, iconic, and auditory), such as, phonics and language learning.
2. Identify appropriate representations for transforming the content to be taught into forms that are pedagogically powerful and difficult to support by traditional means. These include interactive representations, dynamic transformation of data, dynamic processing of data, multiple simultaneous representations of data, and multimodal representations of data.
3. Identify teaching tactics, which are difficult or impossible to implement by other means, such as, the application of ideas in contexts that are not experienced in real life. For example, exploration and discovery in virtual worlds, virtual visits (i.e., virtual museums), testing of hypotheses, simulations, complex decision-making, modeling, long distance communication and collaboration with experts, long distance communication and collaboration with peers, personalized learning, adaptive learning, and context-sensitive feedback.
4. Select tools with appropriate affordances to support 2 and 3 above.
5. Infuse computer activities with appropriate learner-centered strategies in the classroom. This includes any strategy that puts the learner at the center of the learning process to express a point of view, observe, explore, inquire, think, reflect, discover, and problem solve.

### **Technology Mapping as an approach for developing teachers' TPCK:**

According to Mishra and Koehler (2003) what stands between reality and the vision of teachers using technology in the classroom is not what teachers need to learn about technology, but how they are supposed to learn it in order to become competent to teach with technology. Teaching teachers how to use computer tools does not guarantee their pedagogical uses in the classroom. For example, training teachers to learn how to use software

packages “not only makes their knowledge too specific to be applied broadly, but it also becomes quickly outdated. Technology is changing so fast that any method that attempts to keep teachers up to date on the latest software, hardware, and terminology is doomed to create knowledge that is out of date every couple of years” (Mishra & Koehler, 2003, p. 102). Angeli and Valanides (2009, 2013) argued that Technology Mapping (TM) can potentially be an effective approach for developing the knowledge that teachers need to have in order to effectively teach with technology, namely, TPCK.

TM, as shown in Figure 3, was first introduced as an approach for developing teachers’ TPCK in 2009 (Angeli & Valanides, 2009). TM was proposed as an instructional design approach for mapping tool affordances onto content and pedagogy in powerful and transformative ways, enabling teachers to develop complex and interrelated ideas between the affordances of technology and their pedagogical content knowledge. Angeli and Valanides (2009) argued that TM can engage learners in a process of developing technological solutions to pedagogical problems by aligning teachers’ PCK with knowledge about the affordances and constraints of various computer-based technologies. Mapping refers to the process of establishing connections or linkages among the affordances of a tool, content, and pedagogy in relation to learners’ content-related difficulties.

TM is a dynamic, situated and personal design process, which greatly diverges from traditional instructional design practices as teachers’ instructional design decisions are guided by a body of knowledge that is highly situated in the context of their real classroom experiences (Moallem, 1998). As shown in Figure 3, context is an overarching factor in the process of designing learning with technology. The process of designing technology-enhanced learning is influenced by certain context-related factors, such as, teachers’ beliefs about how students learn, teachers’ practical experiences about what can and what cannot work in a real classroom, teachers’ views about the role of technology in teaching and learning, teachers’ adopted instructional practices, school’s vision and educational goals. These context-related factors influence teachers’ thinking about how technology is integrated in the classroom (Niess, 2005, 2011; Ottenbreit-Leftwich, Glazewski, Newby, & Ertmer, 2010; Abbitt, 2011). For example, if a teacher has deep-rooted beliefs in teacher-centered learning, then technology integration will most likely be teacher-directed (i.e., the teacher uses the technology to deliver information to students) and not learner-directed (i.e., the students use the technology to construct/represent meaning about something).

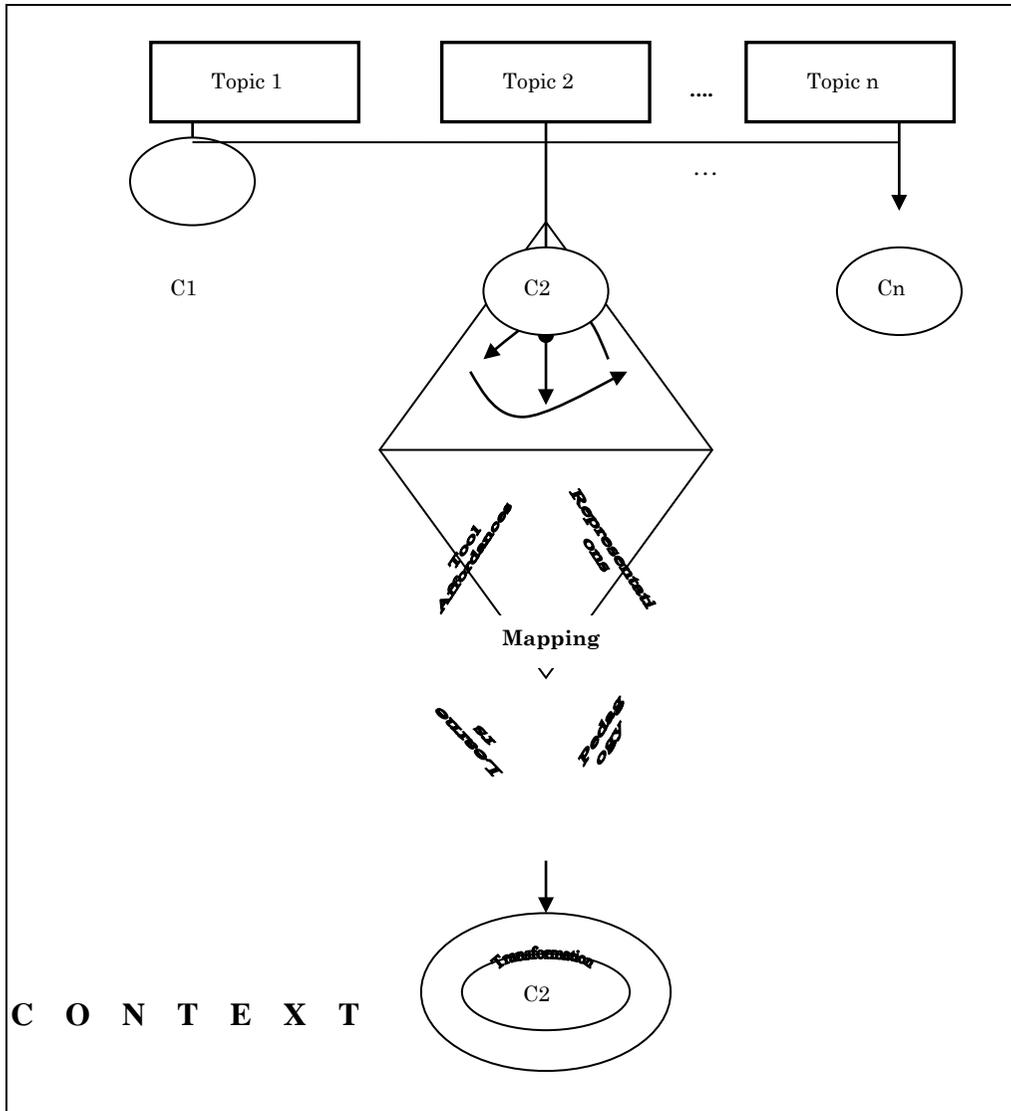


Figure 3. Technology Mapping (adopted from Angeli & Valanides, 2009).

TM allows teachers to bring experiences from their classrooms into the design process, and, specifically, experiences that are related to teachers' PCK, that is, teachers' understandings of their students' alternative conceptions and learning difficulties in relation to certain curriculum topics, as well as teachers' understandings of their own difficulties in making a specific content teachable and easily learnable for their students. According to the TM process depicted in Figure 3, teacher educators ask pre-service or in-service teachers to think about a content domain as well as particular topics within the domain, and, based on their experiences, to indicate their difficulties in making the most challenging aspects of the topics teachable to

students, in connection with students' content-related difficulties. Subsequently, for each topic, teachers associate relevant content (represented as C1, C2,...Cn in Figure 3) and tentative objectives based on learners' related alternative conceptions that need to be addressed. Then teachers are engaged in iterative decision making in order to think how to go about transforming the content with technology into representations that are more understandable to learners. In doing so, teachers need to first decide how tools can be used to transform the content into powerful representations (upper part of the diamond), how to tailor these representations for the specific needs of their students, and how to use technology in innovative ways to transform existing pedagogical practices in their respective classrooms (lower part of the diamond). Thus, at the heart of this iterative decision making is the notion of technology affordances.

Affordances are properties of the relationship between an agent and its physical environment. These properties allow and facilitate specific types of interaction. Gibson (1979, 1982) defined affordances as all action possibilities latent in the environment, objectively measurable and independent of the individual's experience, knowledge, culture, or ability to recognize them. Norman's (1988, 1990) conceptualization of affordances diverges from Gibson's conceptualization in that Norman defines an affordance as something of both actual and perceived properties. When actual and perceived properties are combined, an affordance emerges as a relationship that holds between the object and the individual that is acting on the object (Norman, 1990). From the literature on teachers' understandings of technology affordances, it is evident that (a) teachers do not distinguish between the technical functions of technology and the educational affordances of technology (Valanides & Angeli, 2006, 2008a, 2008b; Angeli & Valanides, 2005; Angeli, 2005), (b) teachers are not always aware of the cognitive processes involved in using the affordances of a particular technology (Yoon, Ho, & Hedberg, 2005), and (c) teachers' formation of mental models of technology affordances largely depends upon their training and their professional development (Krauskopf, Zahn, & Hesse, 2012). Based on these findings, going from knowing how to use a tool to knowing how to teach with a tool, or going from knowing about the technical functions of technology to perceiving the educational affordances of technology, does not occur automatically. Therefore, it becomes imperative that teacher educators make this process explicit during teacher training. A tactic that was used in previous research (Author) with good results, and also employed in the current study is that of aligning the educational affordances of a computer tool with its technical features. To further illustrate this point, the authors present Table 1 as it was used for the purposes of the current study. Table 1 makes explicit the connections between the educational

affordances of MS Excel and its technical features. The information provided in Table 1 is quite useful, as it provides teachers with a systematic and organized way to think about tools and their educational affordances. Similar tables were prepared for the tools Powerpoint and Mindomo, but due to space limitation they are not included here.

Table 1. Educational affordances of MS Excel and technical functions

Educational affordances (sequenced from simple to complex)	affordances	Technical functions
1. Excel as a tool for organizing data.		File – New/Open/Close/Save/Save as/ Page setup/Print area/Print preview/Print/Send to. Edit – Cut/Copy/Paste/Fill/Clear/Delete/Delete sheet/Move or copy sheet/Find/Replace. Insert – Cells/Rows/Columns/Worksheet/Chart Pictures. Format – Cells/Row/Column/Sheet/Style. Review – Spelling and Grammar/Protect Sheet. Data – Sort/Text to columns/Group and outline.
2. Excel as a tool for providing context-sensitive feedback.		Insert – Function / SUM / IF Data – Data Tools/Data Validation/Setting ... /Drop down list.
3. Excel as a tool for performing calculations.		View – Formula bar. Insert – Function / SUM / IF
4. Excel as a modeling tool.		All of the above as needed.

### **The teacher professional development program - Design and implementation:**

During the period between the first week of February and the second week of March in 2014, a 15-hour teacher professional development (TPD) program, entitled “Contemporary Teaching Approaches with the Use of Technology for Teaching Computer Science Topics,” was taught for the purpose of teaching secondary education computer science teachers how to teach with technology in learner-centered ways. The program was co-designed by all authors herein and was taught by the second author. The TPD program consisted of five three-hour seminars. The participants were 13 secondary education computer science teachers; eight women and five men. Four of the participants had five to eight years of teaching experience, eight of them had 10 to 15 years of teaching experience, and one of them was a veteran with 18 years of teaching experience. Six of the participants had teaching experience only with middle school (grades 7-9) students, one of them had taught only high school (grades 10-12) students, and the rest of them had teaching experience with both middle and high school students.

The TPD program consisted of three phases. Phase I put emphasis on principles of learner-centered teaching, targeting the development of participants’ pedagogical skills through live lesson demonstrations by the

trainer in the form of microteaching experiences of three computer science lessons. Phase II of the program focused explicitly on TPCK as a framework for guiding the use of educational technologies in computer science teaching and learning, and TM as an approach for designing technology-enhanced teaching and learning. Phase III of the program consisted of a series of microteaching sessions during which each participant designed and taught a 20-min computer science lesson with the use of educational technology.

Initially, in Phase I, participating teachers were asked to complete an online questionnaire in order to answer a question about what topics from the secondary education computer science curriculum, according always to their subjective appraisal, are difficult to be taught by the teachers and difficult to be understood by the learners. Consequently, the teacher trainer taught a lesson about the Central Processing Unit (CPU) in three different ways following learning principles from behaviourism, cognitivism, and constructivism, respectively. The intention was to teach teachers about learning theories by immersing them first in teaching lessons that they could directly experience, and, afterwards, based on this experience, to discuss with them the tenets of each learning theory, and the influence of each theory on instructional design. The decision to teach about the CPU was intentional, because the related content is regarded difficult to teach and learn. This difficulty arises from the abstract nature of the content in terms of the intrinsic difficulty in understanding that the CPU is a complex set of electronic circuitry, and that it consists of a number of components that work together in some way and interact with memory (Author).

The first lesson was taught based on the tenets of behaviourism. The teacher trainer presented, in a teacher-centered approach, all relevant content information using a Powerpoint presentation. The mode of instruction was primarily based on the transmission model of teaching, and participants had to listen to the teacher trainer, take notes, answer the trainer's questions, and practice.

The second lesson was designed based on the principles of cognitivism. At the beginning of the lesson, the teacher trainer showed the electronic circuitry of the CPU using real hardware equipment, called attention to the different hardware components, such as, RAM, ROM, and VGA card, and explained their role. Then, he presented the participants with a semi-completed concept map, which contained some information about the CPU that he created using a concept mapping software, and asked them to work in groups of two or three (four groups were formed) in order to search the Internet and find relevant information for completing the concept map. Each group had to provide information related to the components of the CPU, such as Registers, Arithmetic Logical Unit, Control Unit, and CPU

clock. When all groups finished, a representative from each group presented their findings.

The third lesson was designed based on the tenets of constructivism, and relied heavily on the use of MS Excel for the purpose of simulating the functioning of the CPU in terms of the complex interactions among the different components, as well as the interactions with memory. Initially, the participants opened an MS Excel worksheet and had to demonstrate, by answering a number of questions, that they had (prior) knowledge of how data are converted to binary numbers, as well as knowledge about how data are stored temporarily in RAM before moving to the CPU for processing, which then return to RAM in the form of information to be eventually displayed on an output device like the monitor. After that, the participants were given a simulation to interact with (in MS Excel) that simulated in detail the interactions among the different CPU components and memory. As expected, the simulation created some cognitive conflict in those cases where initial conceptions differed in terms of how the participants thought the CPU was functioning. Then, learners were divided into three groups and worked collaboratively in order to study more carefully the functioning of the CPU using the simulation in MS Excel. Each group of participants interacted with all CPU components, such as the Registers, Arithmetic Logical Unit, and Control Unit, and at the end, a representative from each group presented and explained to the whole class the role of one CPU component, as instructed by the teacher trainer.

A discussion followed about the pros and cons of the design of each lesson, and implications about the educational uses of technology according to the tenets of each learning theory were discussed in depth and great detail.

During the second phase of the seminar, the instructors presented and explained the framework of TPCK as well as the approach of TM. It was pointed out that the TPCK framework, in its present form, constitutes a domain-general framework that describes the contribution of several knowledge bases to the development of a specific body of knowledge that teachers need to have in order to be able to teach with technology. The TPCK competencies were taught explicitly using as a reference point the integration of technology in each one of the three lessons that were taught during the first phase of the TPD program. The teacher trainer aimed at providing a balance between theory and practice so that teachers could develop a theory-based rationale about their teaching moving away from trial and error instructional decisions to more informed decisions based on theory.

The third phase of the TPD program was conducted in three consecutive meetings of three hours each. All participants were asked to consult first with the teacher trainer in order to jointly decide upon an appropriate topic from the secondary education computer science curriculum

to teach with technology. Then, they had to design and develop all related instructional materials and teach a 20-min lesson during the seminar. At the end of each lesson, a classroom discussion followed with comments and suggestions for possible improvement of the lesson. Finally, at the end of the TPD program, the teachers evaluated the content, the structure, and the teaching strategies of the program, and suggested recommendations for improvement.

### Outcomes of the TPD program:

The teachers' views about the topics from the secondary education computer science curriculum that are difficult to teach and learn are presented in Table 2. As shown in Table 2, the teachers identified 13 topics and explained why they regarded these topics difficult to teach and learn.

Table 2. Topics from the computer science curriculum

	Topics	Why the topic is difficult to teach and learn
1	Computer architecture	Difficult to represent the overall architecture of the computer and show how all components communicate together.
2	Repeated structure in programming	Difficult to understand how the variables for the repetition structure in programming are set, and what role they play.
3	Identifiers (variables and constants) in programming	Difficult to use and declare identifiers (variables and constants) in programming.
4	Data, Processing, Information	Difficult to understand the differences among the three concepts. While these difficulties are not visible early on, they strongly manifest themselves during the teaching of some other computer science topics, such as, algorithms, spreadsheets and databases.
5	Communication protocol	Difficult to teach how two different digital devices communicate between them.
6	Introduction to algorithms	Difficult to write an algorithm.
7	Representation of data in computer language	Difficult to teach the relationship between electric circuitry and computer machine language.
8	World Wide Web (www) and Internet	Difficult to understand the differences between the Internet and WWW.
9	Bubble sort algorithm	Difficult to teach how to use two different counters for sorting a table. There is also a complexity related to teaching the procedure for exchanging the values of two variables with the use of an intermediate variable.
10	Operating systems	Difficult to teach how operating systems work and how the hardware communicates with the software.
11	Functions and procedures	Difficult to understand the differences between functions and procedures.
12	Repetition structures	Difficult to decide which loop structure to use.

	(While..Do Repeat..Until)	or	
13	Main memory (RAM) and secondary memory		Difficult to explain the differences between the two types of memory, as there is a difficulty in representing them accurately.

Later, each teacher undertook one of the topics shown in Table 2, after consultation with the teacher trainer, and taught it in a 20-min microteaching session during the last phase of the TPD program. Due to space limitations, in the section below, the authors present the technology-enhanced lessons for only three of these topics, namely: (a) Computer architecture, (b) Introduction to algorithms, and (c) Bubble sort algorithm.

### **Topic 1- Computer architecture:**

The teaching of computer architecture usually requires two 45-min teaching periods in eighth grade. The difficulty in teaching and understanding this topic is attributed to the fact that the architecture of the computer consists of several internal hardware components that communicate amongst them in some way that is invisible to the learner. The learning objectives of this lesson were defined as follows: (a) Report and explain the term computer architecture; and (b) Recognize the main parts inside a computer and explain their role. At the beginning of the 20-min microteaching lesson, the teacher showed a short video about the basic hardware components of the computer. Afterwards, the teacher discussed with the students (role played by the other teachers who participated in the TPD program) what was meant by the term computer architecture, and used as a metaphor an example from daily life to further explain the term, namely, the architecture of a house. For the technology-enhanced learning activities, the teacher used Mindomo, a concept mapping tool that allows users to insert images, video, audio, note fields, and to create hyperlinks and bookmarks. Additionally, the tool allows users to work collaboratively in order to create a joint concept map. Consequently, the teacher sent an online request to students via email inviting them to access an incomplete Mindomo concept map that was created specifically for the purposes of the lesson. Students worked in dyads and engaged in activities that included an exploration of the web via different links, in order to find relevant information for completing the concept map. Each group undertook a different task. For example, Group A dealt with power supply, the motherboard, and CPU. Group B dealt with RAM and ROM, Group C with expansion slots, Group D with expansion cards, and Group E with connected slots. When all groups finished their work, a completed concept map was created collaboratively. Each group presented to the class their part of the task. In essence, the collaborative online activity with Mindomo enabled the students to create a virtual

motherboard that they could use afterwards as a guide in order to assemble a real one.

At the end of the lesson, a debriefing session followed about the instructional design of the lesson and different points of view were expressed regarding the integration of Mindomo in the teaching of the lesson.

## **Topic 2 - Introduction to algorithms:**

Learning how to write an algorithm constitutes a highly important topic for the field of computer science and the secondary education computer science curriculum as well. Algorithms are sets of instructions written in a step by step fashion that are followed in order to solve a problem. Algorithms can be expressed in the form of a high-level computer language, pseudo-code, verbal descriptions, and flowcharts. In the field of computer science, it is imperative for students to learn how to write algorithms, before learning how to write code using a computer programming language. Algorithms are usually taught in ninth grade and they are considered prerequisite knowledge for learning how to program. An approach that is usually used to teach students how to develop algorithms, prior to learning how to code computer programs, is flowcharting. A flowchart is a type of diagram that represents an algorithm, workflow or process, showing the steps as boxes of various kinds, and their order by connecting them with arrows. This diagrammatic representation illustrates a solution to a given problem.

The learning objectives of this lesson were defined as follows: (a) Define the cycle of Data-Process-Information in a problem; (b) Recognize the necessity and usefulness of flowcharts; (c) Recognize the different shapes of boxes that can be used for creating the flowchart of an algorithm; and (d) Create the flowchart for an algorithm. For the purpose of developing the technology-enhanced activities, MS Excel was used. Table 1 shows all information related to the educational affordances of MS Excel and its technical features. The information displayed in Table 1 was used as a guide for facilitating the construction of teachers' mental models concerning the educational utilization of spreadsheets in teaching and learning.

In more detail, at the beginning of the lesson, students (role played by the teachers in the program) were asked to open and execute a flowchart in Algo. Students were then instructed to interact with the flowchart in order to input some data and observe the result. Then, a discussion followed about the value of flowcharts in learning how to develop algorithms. Students were asked to continue interacting with the flowchart in Algo in order to discover the differences among the terms data, process, and information. Subsequently, students were instructed to work individually on a number of different activities in MS Excel. The activity shown in Figure 4 dealt with the cycle of data processing (i.e., Input Data, Process Data, Output Data)

and explicated the idea that a correct algorithm follows this cycle. Students were asked to apply the cycle of data processing by selecting an appropriate step. Upon a correct answer, students received positive feedback. Then, the students proceeded with two more MS Excel activities for more practice.

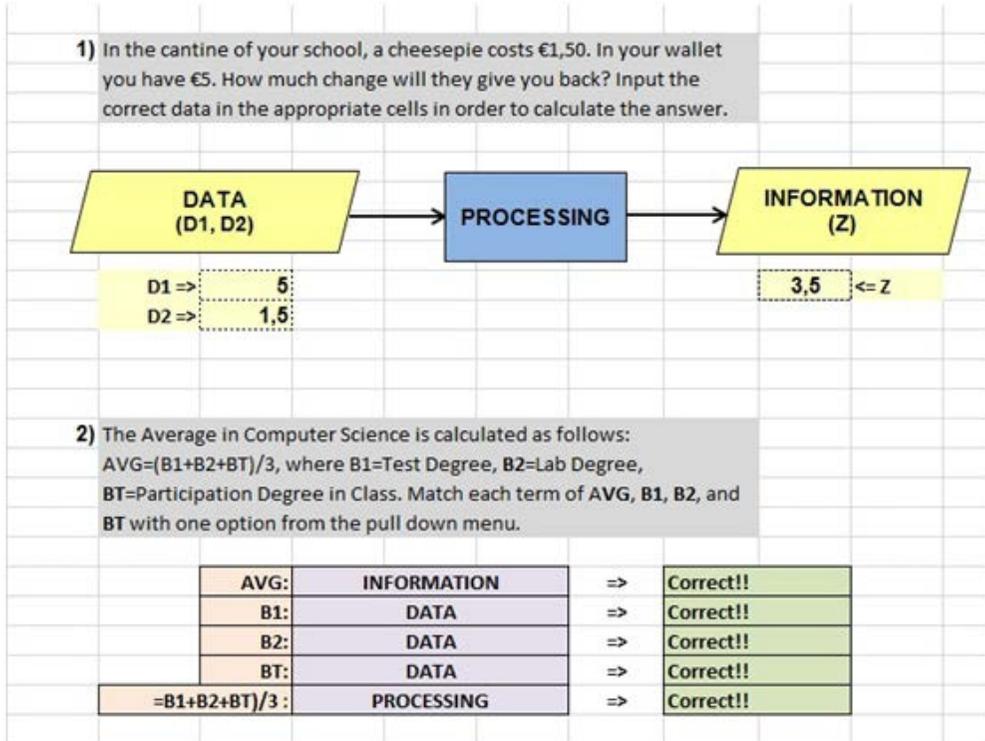


Figure 4. Data-Process-Output cycle.

### Topic 3 - Bubble sort algorithm:

One well known algorithm in programming is the Bubble sort algorithm. It is a sorting algorithm that is used when there is a need to sort a two-dimensional array. Bubble sort is an algorithm about how to compare and exchange pairs of adjacent elements. The smallest element is always moved to the left of the array. The difficulty in teaching and understanding the algorithm is attributed to the fact that in order to successfully perform a bubble sort one needs to use two counters, and consequently the need for a double loop creates an additional degree of difficulty. Students usually face difficulties in understanding the values of the two counters, e.g.,  $i$  and  $j$ . There is also a complexity related to teaching the actual procedure for exchanging the values of two variables with the use of an intermediate (temporary) variable. Bubble sort is usually taught in twelfth grade. The learning objectives of the lesson were defined as follows: (a) Describe the steps that are executed during the Bubble sort algorithm; (b) Demonstrate the

use of two different counters in order to be able to make the comparisons of all values in the cells; and (c) Write the actual code for the Bubble sort algorithm. Specifically, the teacher at the beginning of the lesson showed a short video about Bubble sort, and how it actually works. Then a discussion followed about the necessity to sort a two-dimensional array. Then the teacher, through the use of animated Powerpoint presentations (see activities shown in Figures 5 and 6), demonstrated increasingly all of the steps that are involved in Bubble sort. This way, students were able to see the swapping between cells and the series of comparisons that necessarily needed to be performed before a value was moved to its final position on the left side of the array.

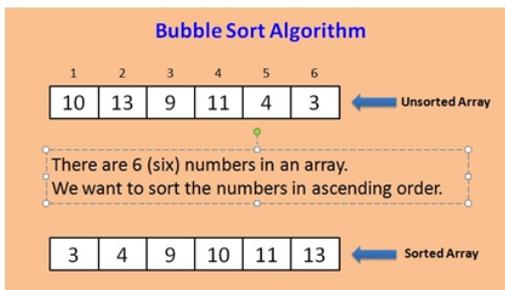


Figure 5. Bubble sort.

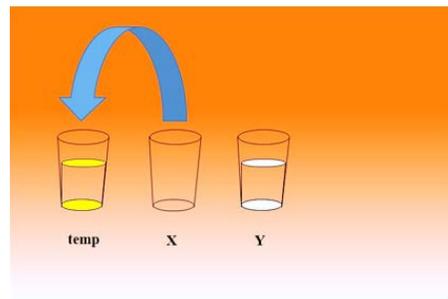


Figure 6. Using a temporary variable during Bubble sort

### Evaluation of the TPD program:

At the end of the 15-hour TPD program, all 13 participating teachers evaluated anonymously the program through the use of an online questionnaire. The questionnaire included questions about the content and the structure of the program, and questions about the pedagogical practices employed by the teacher trainer. The outcomes of the evaluation were overwhelmingly positive for the design and the implementation of the TPD program. All thirteen teachers expressed their gratitude and satisfaction for the opportunity they were given to participate in the program, and felt that the framework of TPCK as well as the approach of TM equipped them with knowledge and skills about how to rethink the teaching of difficult computer science topics. In addition, all teachers expressed their positive remarks about the approach that was followed to teach them about the educational affordances of tools, and found especially useful the table format (e.g., Table 1) that was followed in order to think in a systematic way about the educational utilization of the tools. Participants were also very keen on how the seminar was taught using primarily the microteaching approach. First, they commented on the three lessons that the teacher trainer taught during

the first phase of the program. They felt they learned a lot about the practicalities of the learning theories and their applicability in classroom teaching in terms of specific teaching strategies and tactics. Then, they commented on how important was to observe other teachers teach in the program and share ideas about how to teach with computer tools. They noted that sharing lesson designs and materials for 13 different computer science topics (shown in Table 2) was extremely beneficial since they now had at their availability materials that they could adopt or adapt in their own teaching.

Succinctly, according to the participants' evaluative comments, the strengths of the TPD program were the following: (a) The continuous effort of the teacher trainer to make explicit the relationship between the technical features of the tools and their educational affordances; (b) The selection of topics that were chosen for the trainer led seminars as well as those for the participant-teachers' microteaching sessions; (c) First-hand knowledge in experiencing the added value of educational technology in teaching difficult and abstract computer science topics; (d) The repetitive demonstration of TM by the teacher trainer and the fact that the teachers themselves practiced the process extensively during the 15-hour program; (e) Theory-driven instructional design based on the TPACK framework and the approach of TM; and, (f) Extensive discussions and reflection about the microteaching sessions as carried out by the teacher trainer and the teachers themselves.

One suggestion for improvement expressed by the participating teachers is to include a fourth phase in future TPD programs in order to have each teacher apply what he or she learned in a real classroom setting. Undoubtedly, extending the TPD program to include a fourth phase will most likely add to the authenticity and the viability of the training in real classroom contexts, and can be the focus of future research efforts.

## **Conclusion**

The authors recognize that the study described herein focused primarily on the cognitive domain of learning; that is on using technology to transform existing classroom practices in order to solve existing pedagogical problems that were directly related to the difficulty of the teacher to teach the content, or the difficulty of the learner to understand the content. They, however, recognize that the difficulties in teaching or understanding a particular content might not always be cognitive in nature. Therefore, it is the authors' conviction that in future research studies it will be valuable and promising to invest research time, effort, and resources for the purpose of examining TPACK and TM in conjunction with various facets of both the cognitive and the affective domains of learning, as these are exemplified by different content domains. This research direction will be beneficial as it will

allow the researchers to consider the specificity of TPCK within domains where affect plays an important role in teaching and learning, such as, for example, the fine arts (i.e., music, drama, and dance). Efforts toward this line of research will help the research community to further clarify the construct of TPCK and thereafter to design more effective teacher professional development programs for the development of teachers' TPCK.

### References:

Abbitt, J. (2011). An investigation of the relationship between self-efficacy beliefs about technology integration and technological pedagogical content knowledge (TPACK) among preservice teachers. *Journal of Digital Learning in Teacher Education*, 27(4), 134-143.

ACM K-12 Task Force Curriculum Committee. (2003). *A model curriculum for K-12 computer science*. New York: Computer Science Teacher Association.

Angeli, C. (2005). Transforming a teacher education method course through technology: Effects on pre-service teachers' technology competency. *Computers & Education*, 45(4), 383-398.

Angeli, C., & Valanides, N. (2005). Pre-service teachers as ICT designers: An instructional design model based on an expanded view of pedagogical content knowledge. *Journal of Computer-Assisted Learning*, 21(4), 292-302.

Angeli, C., & Valanides, N. (2009). Epistemological and methodological issues for the conceptualization, development, and assessment of ICT-TPCK: Advances in technological pedagogical content knowledge (TPCK). *Computers & Education*, 52(1), 154-168.

Angeli, C., & Valanides, N. (2013). Technology Mapping: An approach for developing Technological Pedagogical Content Knowledge. *Journal of Educational Computing Research*, 48(2), 199-221.

Archambault, L., & Crippen, K. (2009). Examining TPACK among K-12 online distance educators in the United States. *Contemporary Issues in Technology and Teacher Education*, 9(1), 71-88.

Archambault, L. M., & Barnett, J. H. (2010). Revisiting technological pedagogical content knowledge: Exploring the TPACK framework. *Computers & Education*, 55(4), 1656-1662.

Chai, C. S., Koh, J. H. L., Tsai, C.-C., & Tan, L. L. W. (2011). Modeling primary school pre-service teachers' technological pedagogical content knowledge (TPACK) for meaningful learning with information and communication technology (ICT). *Computers & Education*, 57(1), 1184-1193.

Cox, S., & Graham, C. R. (2009). Diagramming TPACK in practice: Using an elaborated model of the TPACK framework to analyze and depict teacher knowledge. *TechTrends*, 53(5), 60-69.

- Cuban, L. (2000). *So much high-tech money invested, so little use and change in practice: How come?* Paper presented for the Council of Chief State School Officers' Annual Technology Leadership Conference, Washington, D.C., USA.
- Ertmer, P. A., & Ottenbreit-Leftwich, A. T. (2010). Teacher technology change: How knowledge, confidence, beliefs, and culture intersect. *Journal of Research on Technology in Education*, 42(3), 22-43.
- Gal-Ezer, J., Vilner, T., & Zur, E. (2003). *Characteristics of students who failed (or succeeded) the introductory CS course*. Paper presented at the FIEE Conference, Boulder, CO. Retrieved July 26, 2010 from <http://fie-conference.org/fie2003/index.htm>
- Gibson, J. J. (1979). *The ecological approach to visual perception*. Boston, MA: Houghton Mifflin.
- Gibson, J. J. (1982). *Reasons for realism: Selected essays of James J. Gibson*. Hillsdale, NJ: Lawrence Erlbaum Associates.
- Graham, C. R. (2011). Theoretical considerations for understanding technological pedagogical content knowledge (TPACK). *Computers & Education*, 57(3), 1953-1960.
- Grossman, P. L., Wilson, S. M., & Shulman, L. (1989). Teachers of substance: Subject matter knowledge for teaching. In M. C. Reynolds (Ed.), *Knowledge base for the beginning teacher* (pp. 23-36). Oxford: Pergamon Press.
- Harris, J. B., & Hofer, M. J. (2011). Technological pedagogical content knowledge in action: A descriptive study of secondary teachers' curriculum-based, technology-related instructional planning. *Journal of Research on Technology in Education*, 43(3), 211-229.
- Harrison, C., Comber, C., Fisher, T., Haw, K., Lewin, C., Lunzer, E., McFarlane, A., Mavers, D., Scrimshaw, P., Somekh, B., & Watling, R. (2003). *ImpaCT2: The impact of information and communication technologies on pupil learning and attainment*. Available from: <http://www.becta.org.uk/research/impact2>
- Hazzan O., Lapidot, T., & Ragonis, N. (2011). *Guide to teaching computer science: An activity-based approach*. London: Springer.
- International Society for Technology in Education. (2002). *National Educational Technology Standards for Teachers: Preparing Teachers to Use Technology*. Danvers, MA: ISTE.
- Kadijevich, D. M., Angeli, C., & Schulte, C. (2013). *Improving computer science education*. New York: Routledge.
- Kozma, R., & Anderson. R. E. (2002). Qualitative case studies of innovative pedagogical practices using ICT. *Journal of Computer Assisted Learning*, 18, 387-394.

- Krauskopf, K., Zahn, C., & Hesse, F. W. (2012). Leveraging the affordances of Youtube: The role of pedagogical knowledge and mental models of technology functions for lesson planning with technology. *Computers & Education*, 58(4), 1194-1206.
- Margerum-Lays, J., & Marx, R. W. (2003). Teacher knowledge of educational technology: A case study of student/mentor teacher pairs. In Y. Zhao (Ed.), *What should teachers know about technology? Perspectives and practices* (pp. 123-159). Greenwich, CO: Information Age Publishing.
- Mishra, P., & Koehler, M. (2003). NOT “WHAT” BUT “HOW”: Becoming design-wise about educational technology. In Y. Zhao (Ed.), *What should teachers know about technology? Perspectives and practices* (pp. 99-122). Greenwich, CO: Information Age Publishing.
- Mishra, P., & Koehler, M. J. (2006). Technological pedagogical content knowledge: A new framework for teacher knowledge. *Teachers College Record*, 108(6), 1017–1054.
- Moallem, M. (1998). An expert teacher’s thinking and teaching and instructional design models and principles: An ethnographic study. *Educational Technology Research and Development*, 46, 37–64.
- Mouza, C. (2009). Does research-based professional development make a difference? A longitudinal investigation of teacher learning in technology integration. *Teachers College Record*, 111(5), 1195-1241.
- National Center for Educational Evaluation and Regional Assistance (NCEE). (2007). *Effectiveness of reading and mathematics software programs: Findings from the first student cohort*. Washington, DC: US Department of Education, Institute for Education Sciences.
- Niess, M. L. (2005). Preparing teachers to teach science and mathematics with technology: Developing a technology pedagogical content knowledge. *Teaching and Teacher Education*, 21, 509–523.
- Niess, M. L. (2011). Investigating TPACK: Knowledge growth in teaching with technology. *Journal of Educational Computing Research*, 44(3), 299-317.
- Norman, D. A. (1988). *The psychology of everyday things*. New York: Basic Books.
- Norman, D. A. (1990). *The design of everyday things*. New York: Doubleday.
- Ottenbreit-Leftwich, A., Glazewski, K., Newby, T., & Ertmer, P. (2010). Teacher value beliefs associated with using technology: Addressing professional and student needs. *Computers & Education*, 55, 1321–1335.
- Pelgrum, W. (2001). Obstacles to the integration of ICT in education: Results from a worldwide educational assessment. *Computers & Education*, 37, 163-178.

- Rodrigues, S. (2003). Experiences from the partnership in primary science project: Teacher professional development involving ICT and science pedagogical content knowledge. *Science Education International*, 14(2), 2-11.
- Schmidt, D. A., Baran, E., Thompson, A. D., Mishra, P., Koehler, M. J., & Shin, T. S. (2009). Technological pedagogical content knowledge (TPACK): The development and validation of an assessment instrument for preservice teachers. *Journal of Research on Technology in Education*, 42(2), 123–149.
- Selinger, M. (2001). Learning information and communications technology skills and the subject context of the learning. *Journal of Information Technology for Teacher Education*, 10, 1&2, 143-154.
- Shulman, L. S. (1986). Those who understand: Knowledge growth in teaching. *Educational Researcher*, 15(2), 4-14.
- Shulman, L. S. (1987). Knowledge and teaching: Foundations of the new reform. *Harvard Educational Review*, 57(1), 1-22.
- Tucker, A. B., Deek, F., Jones, J., McGowan, D., Stephenson, C., & Verno, A. (2003). *A model curriculum for K-12 computer science*. New York: ACM/Computer Science Teachers Association.
- Yoon, F. S., Ho, J., & Hedberg, J. G. (2005). Teacher understandings of technology affordances and their impact on the design of engaging learning experiences. *Educational Media International*, 42(4), 297-316.
- Valanides, N., & Angeli, C. (2006). Preparing pre-service elementary teachers to teach science through computer models. *Contemporary Issues in Technology and Teacher Education - Science*, 6(1), 87–98.
- Valanides, N., & Angeli, C. (2008a). Learning and teaching about scientific models with a computer modeling tool. *Computers in Human Behavior*, 24(2), 220–233.
- Valanides, N., & Angeli, C. (2008b). Professional development for computer-enhanced learning: A case study with science teachers. *Research in Science and Technological Education*, 26(1), 3–12.
- Voogt, J., Fisser, P., Pareja Roblin, N., Tondeur, J., & van Braak, J. (2012). Technological pedagogical content knowledge - a review of the literature. *Journal of Computer-Assisted Learning*, 29(2), 109-121.