
Themenheft Nr. 42: Optimierung in der Medienpädagogik.
Forschungsperspektiven im Anschluss an den 27. Kongress der DGfE
Herausgegeben von Patrick Bettinger, Klaus Rummler und Karsten D. Wolf

Der Hochschullehrgang Coding und Robotik für Lehrkräfte an der Pädagogischen Hochschule Burgenland

Konzeption, Implementation und erste Ergebnisse einer Begleitevaluation

Thomas Leitgeb, Alexander Zimmermann und Wolfram Rollett

Zusammenfassung

Um an einer zunehmend digitalisierten Welt beruflich und gesellschaftlich teilnehmen zu können, bedarf es umfangreicher und tiefer Kenntnisse bezüglich informatischer Konzepte. Vor diesem Hintergrund erscheint eine Erweiterung des herkömmlichen schulischen Curriculums um die systematische Implementierung von Computational Thinking (CT) – also dem Lösen von Problemen bzw. Aufgaben mit Hilfe algorithmischer Lösungsstrategien unter Rückgriff auf grundlegende Konzepte der Informatik (Wing 2006, 33) – sinnvoll und notwendig. Weltweit wurde dies bereits in vielen Ländern umgesetzt (Bocconi u. a. 2016). Der Lehrkräfteprofessionalisierung wird für die erfolgreiche Implementation von CT in der Schule eine besondere Bedeutung zugesprochen (Pollak und Ebner 2019). Im österreichischen Burgenland wurde zur Förderung von CT für die Primarstufe eine integrative Lehrplanverankerung vorgenommen und für die Sekundarstufe I das Wahlpflichtfach Coding und Robotik eingeführt (Leitgeb 2019). Um die Lehrkräfte entsprechend fort- bzw. weiterbilden zu können, wurde an der Pädagogischen Hochschule Burgenland ein zweijähriger Hochschullehrgang (HLG) Coding und Robotik entwickelt und implementiert. Im vorliegenden Beitrag werden der theoretische Rahmen und Hintergrund, die theoretische Konzeption, die inhaltliche und didaktische Ausgestaltung sowie die Ziele dieses HLG vorgestellt. Ferner werden Ergebnisse einer nach dem ersten Studienjahr durchgeführten Begleitevaluation präsentiert.

The University Course Coding and Robotics for Teachers at the University of Education Burgenland

Abstract

Professional and social participation in an increasingly digitized world requires extensive and in-depth knowledge of computer science concepts. It seems both sensible and necessary to expand the conventional school curriculum to include the systematic implementation of Computational Thinking (CT), i. e. solving problems or tasks with the help of algorithmic

solution strategies, drawing on fundamental computer science concepts (Wing 2006, 33). This has already been implemented in many countries worldwide (Bocconi et al. 2016). Professionalization of teachers is considered particularly important for successful implementation (Pollak and Ebner 2019). In the Austrian province of Burgenland, an integrative curriculum has been implemented to promote CT for the primary level, and the elective subject Coding and Robotics has been introduced for secondary level 1 (Leitgeb 2019). In order to provide further training for teachers, a two-year university course in Coding and Robotics has been developed and implemented at the Burgenland University of Education. This paper presents the theoretical framework, background and conception, the contentual and didactic design and the goals of the course, together with results of an evaluation conducted after the first year of study.

1. Einleitung

In unserer zunehmend digitalisierten Welt verändern Anwendungen digitaler Medien aller Art das gesellschaftliche, berufliche und private Leben zusehends und damit auch jenen Kompetenzkanon, der für eine gelungene Teilhabe in wichtigen Lebensbereichen notwendig ist (Schuhr 2015, 226; Eguchi 2014, 5). Für die Gestaltung schulischer Curricula stellt sich die Frage, wie die dafür erforderlichen Kompetenzen angebahnt, gefördert und vertieft werden können. Ein diesbezüglich vielversprechender Ansatz besteht darin, Kompetenzen in den Blick zu nehmen, die unter dem Begriff *Computational Thinking* (CT) subsumiert werden (Eickelmann, Vahrenhold, und Labusch 2019, 367). Nach Aho (2012, 834) ist unter CT die Gesamtheit von Denkprozessen zu verstehen, die genutzt werden, um Probleme sowie Verfahren zu deren Lösung so zu modellieren, dass eine algorithmische Verarbeitung von Problemstellungen ermöglicht wird. Somit geht CT weit über das bloße Erlernen des Umgangs mit Hard- und Software hinaus (Eickelmann u. a. 2019, 367), ist jedoch im Wesentlichen nicht von konkreten (z. B. computerbasierten) Anwendungen abhängig. Bei der Erarbeitung von CT im Unterricht werden neben informatischen Fachinhalten wichtige allgemeine Problemlösestrategien und Kenntnisse im Bereich des strukturierten Denkens vermittelt (Ilic, Haseski, und Tugtekin 2018, 136). Konzeptuell zeigt CT eine grosse Nähe zum Critical Thinking, das neben Communication, Collaboration und Creativity eine der vier 21st Century Skills darstellt, die weltweit im Fokus der Weiterentwicklung von Bildungssystemen stehen bzw. stehen sollten (OECD 2018). Viele europäische und aussereuropäische Länder wie etwa Tschechien, Finnland, Grossbritannien bzw. Südkorea, die USA und Australien haben CT in der einen oder anderen Form bereits curricular im Schulsystem verankert (Eickelmann u. a. 2019, 368; Shi 2018, 32; Angeli u. a. 2016, 48). Für die erfolgreiche Implementation hat sich dabei die Lehrkräfteaus-, -fort- und -weiterbildung als wichtige Voraussetzung erwiesen (Orvalho 2017; Hsu, Chang, und Hung 2018, 307). Gleichzeitig werden

mangelnde Fachkenntnisse seitens der Lehrkräfte als wichtigster Grund für die in weiten Teilen noch immer lückenhafte bzw. sogar fehlende Beschäftigung mit CT in der Schule angeführt. Zudem lässt sich feststellen, dass es häufig an entsprechenden Fort- und Weiterbildungsangeboten fehlt (Pollak und Ebner 2019, 7). Dies ist umso bedauerlicher, da die Abstraktheit und Komplexität von CT, die grosse Bandbreite an benötigten Fachinhalten und schliesslich die damit einhergehenden methodisch-didaktischen Erfordernisse eine eigens dafür konzipierte Lehrkräfteprofessionalisierung erfordern.

2. Computational Thinking in der Schule

Seit den 2010er Jahren wird CT im Bildungsbereich international zunehmend berücksichtigt; auch in wissenschaftlicher Hinsicht (Eickelmann u. a. 2019; Ilic u. a. 2018). So wurde CT z. B. 2014 in Grossbritannien, Dänemark und Irland in das Schulcurriculum aufgenommen (Balanskat und Engelhart 2015, 24). In Grossbritannien wurde dabei auf die sechs Aspekte für die operationale Umsetzung von CT im Schulunterricht der Computer Science Teachers Association & International Society for Technology in Education (2011) Bezug genommen:

- «Formulating problems in a way that enables us to use a computer and other tools to help solve them;
- Logically organizing and analyzing data;
- Representing data through abstractions such as models and simulations;
- Automating solutions through algorithmic thinking (a series of ordered steps);
- Identifying, analyzing, and implementing possible solutions with the goal of achieving the most efficient and effective combination of steps and resources;
- Generalizing and transferring this problem solving process to a wide variety of problems.» (ebd.)

Die angeführten Inhalte weisen ein hohes Abstraktionsniveau auf, so dass sich die Frage einer für die Schule praktikablen und schülerorientierten Gestaltung und Umsetzung eines entsprechenden Curriculums stellt (Sentance und Csizmadia 2015, 1). Zur Vermittlung von CT hat sich deshalb in den letzten Jahren die Verbindung mit Educational Robotics (ER) mehr und mehr etabliert (Fislake 2020, o. S.). Hintergrund dafür ist, dass sich mit Hilfe von ER die oftmals abstrakt-komplexen Konzepte von CT auf relativ einfache und naheliegende Weise in der Schule anschaulich bzw. haptisch erfahrbar und mithin auch leichter begreifbar machen lassen (Eguchi 2014, 6). Der Ansatz, ER als Medium für die Vermittlung von CT zu nutzen, findet auch in der entsprechenden Fachliteratur zunehmend Eingang (García-Valcárcel-Muñoz-Repiso und Caballero-González 2019, 70). ER wird in der einschlägigen Literatur als Schnittstelle zwischen den drei Fachdisziplinen Robotik, Mensch-Roboter-Interaktion und Bildung

angesehen (Angel-Fernandez und Vincze 2018, 2). Im Unterricht wird ER dabei in verschiedener Weise eingesetzt: Erstens als Lernobjekt, d. h., als eigener Lerninhalt mit Fokus auf die Konstruktion und Programmierung von Lernrobotern (Karampinis 2018); zweitens als Lernmittel, also als Medium zur Vermittlung von Inhalten (Angel-Fernandez und Vincze 2018, 2; Alimisis und Kynigos 2009, 17) und drittens als Aktivierungs- und Motivierungsmassnahme zur Unterstützung des Lernens (Eguchi 2014, 7; Genser und Leitgeb 2020). Mehrere Studien bescheinigen der Aufnahme von CT und ER in die Schulausbildung bzw. den Unterricht einen positiven Effekt sowohl auf die Entwicklung kognitiver Fähigkeiten (Ilic u. a. 2018) als auch speziell auf die Weiterentwicklung von Denkprozessen der Schülerinnen und Schüler bzw. deren Befähigung, Probleme zu lösen. (Scherer, Siddiq, und Viveros 2018).¹

In der Literatur werden drei Möglichkeiten der schulischen Vermittlung von CT beschrieben (Eickelmann u. a. 2019, 368; Ilic u. a. 2018):

- im lehrplanübergreifenden Unterricht: Bei diesem Ansatz wird CT als Thema des fächerübergreifenden Unterrichts verstanden, wobei die jeweiligen Fächer auf unterschiedliche Art und Weise zum Erwerb von CT beitragen sollen. Diese Möglichkeit wird weltweit praktiziert (Grover, Pea, und Cooper 2015, 34).
- im Informatikunterricht: Dies bietet sich insofern an, als in dessen Rahmen wichtige mit CT zusammenhängende Themen besprochen werden. Eine solche Vermittlungsart wird etwa in Grossbritannien praktiziert (vgl. Department of Education 2013).
- als eigenes Unterrichtsfach: CT wird hierbei als eine Schlüsselkompetenz im Bereich neuer Technologien angesehen. So erprobt Dänemark *Computational thinking* seit 2018 als eigenes Fach in Modellschulen (Eickelmann u. a. 2019, 369; Undervisningsministeriet 2018). Im österreichischen Burgenland wird seit 2018 CT als Wahlpflichtfach *Coding und Robotik* in der Sekundarstufe I angeboten (Leitgeb 2018; Leitgeb 2019).

Durch den Einbezug von CT in den Schulunterricht werden auch bedeutsame Kompetenzen anderer Unterrichtsfächer gefördert, wie z. B. Kreativität, das sprachliche Vermögen beim Formulieren von Problemen, mathematische Fähigkeiten bei deren Modellierung und der Datenauswertung, naturwissenschaftliche Kenntnisse bei der Exploration von Sachverhalten durch Bedingungsvariation sowie technisches Verständnis durch den zielgerichteten Einsatz von Technik und digitalen Medien bei der Bearbeitung anspruchsvoller Aufgabenstellungen (siehe Ma 2006, 41; Scherer, Siddiq, und Viveros 2018, 12-21; Shute, Sun, und Asbell-Clarke 2017, 4). Somit hat die schulische Auseinandersetzung mit CT auch einen hohen fächerübergreifenden und mithin fächerverbindenden Wert.

1 Zur Förderung mathematischer Argumentierfähigkeiten siehe (Zimmermann 2020).

Vor diesem Hintergrund wurde von der Pädagogischen Hochschule Burgenland ein Lehrplan für CT verbunden mit ER für die burgenländischen Pflichtschulen der Primar- und Sekundarstufe I entwickelt (Leitgeb 2018, 32). Während für die Primarstufe eine integrative Vermittlung im bestehenden Fächerkanon vorgesehen ist, wurde für die Sekundarstufe I ein curricular entsprechend ausgerichtetes Wahlpflichtfach *Coding und Robotik* eingeführt. Ziel ist es dabei, Schülern und Schülerinnen grundlegende, aber oftmals auch abstrakte und komplexe Konzepte von CT mit Hilfe haptisch erfahrbarer Lernmaterialien leichter verständlich zu machen. Ausserdem wurden und werden kompetenzorientierte Jahrespläne sowie für die einzelnen Schultypen und -stufen eigens entwickelte und angepasste Lehr- und Lernmaterialien entwickelt. Unterstützend bietet die Pädagogische Hochschule Burgenland eine fachspezifische und -didaktische Beratung, Begleitung und Evaluation bei der schulischen Organisations-, Personal- und Unterrichtsentwicklung an. Seit dem Schuljahr 2018/19 nehmen über 40 burgenländische Schulen dieses Angebot wahr.

3. Theoretischer Rahmen für den Hochschullehrgang (HLG) Coding und Robotik

Nicht spezifisch geschultes Lehrpersonal ist im Unterricht mit der Vermittlung von Kompetenzen für CT in vielfacher Hinsicht überfordert (Hsu, Chang, und Hung 2018, 298). Für eine erfolgreiche Implementation von CT im schulischen Curriculum sind daher spezielle institutionell durchgeführte Lehrkräfteaus-, -fort- und -weiterbildungen notwendig (Pollak und Ebner 2019, 7), welche die folgenden Inhaltsbereiche abdecken sollten:

- theoretische und technische Fachinhalte,
- Wissen über Lernprozesse und die Leistungsbeurteilung (Csizmadia u. a. 2015, 16),
- geeignete methodisch-didaktische Vorgehensweisen (Denning 2017) sowie geeignete Aufgaben und Beispiele (Futschek 2016),
- systematisches Wissen zur Vermittlung von CT in verschiedenen Altersgruppen bzw. Schulstufen (Li u. a. 2020, 5),
- das Hinterfragen des Einsatzes und der Wirkung technologischer Systeme (Wing 2006), wie z. B. einen kritischen und mündigen Umgang mit digitalen Medien und zu Auswirkungen der Digitalisierung auf die Gesellschaft und die Umwelt,
- das Suchen von Fehlern in Systemen (Wing 2006) und
- eine Auseinandersetzung mit spezifischen Themen zu Sicherheit und Datenschutz (Czerkowski 2015).

Ein geeigneter Ausgangspunkt für die Entwicklung eines alle diese Bereiche berücksichtigenden Kompetenzmodells für die institutionelle Weiterbildung ist das COACTIV-Modell (Kunter, Klusmann, und Baumert 2009; Baumert und Kunter 2011).

Es fasst bereits bestehende Konzepte zur Lehrkräfteprofessionalisierung (Shulman 1986; Bromme 1997) zusammen und lässt sich auf unterschiedliche Bildungsbereiche anwenden. Dabei werden vier «Aspekte professioneller Kompetenz» unterschieden: das Professionswissen, die Überzeugungen und Werthaltungen, die Motivation sowie die Selbstregulation. Um das COACTIV-Modell für den HLG Coding und Robotik heranziehen zu können, wurde es strukturell erweitert und modifiziert (s. Abbildung 1). So wurden die sogenannten «Kompetenzfacetten» des Kompetenzbereichs Professionswissen an die Inhalte des HLG angepasst. Ferner wurden der Kompetenzbereich Überzeugungen/Werthaltungen um den Aspekt der Einstellungen zur Digitalisierung, der Bereich Motivation um jenen der positiven Erfahrungen aus dem HLG bzw. den darauf aufbauenden Unterrichtsversuchen und schliesslich der Bereich Selbstregulation um jenen der digitalen Unterstützung erweitert.

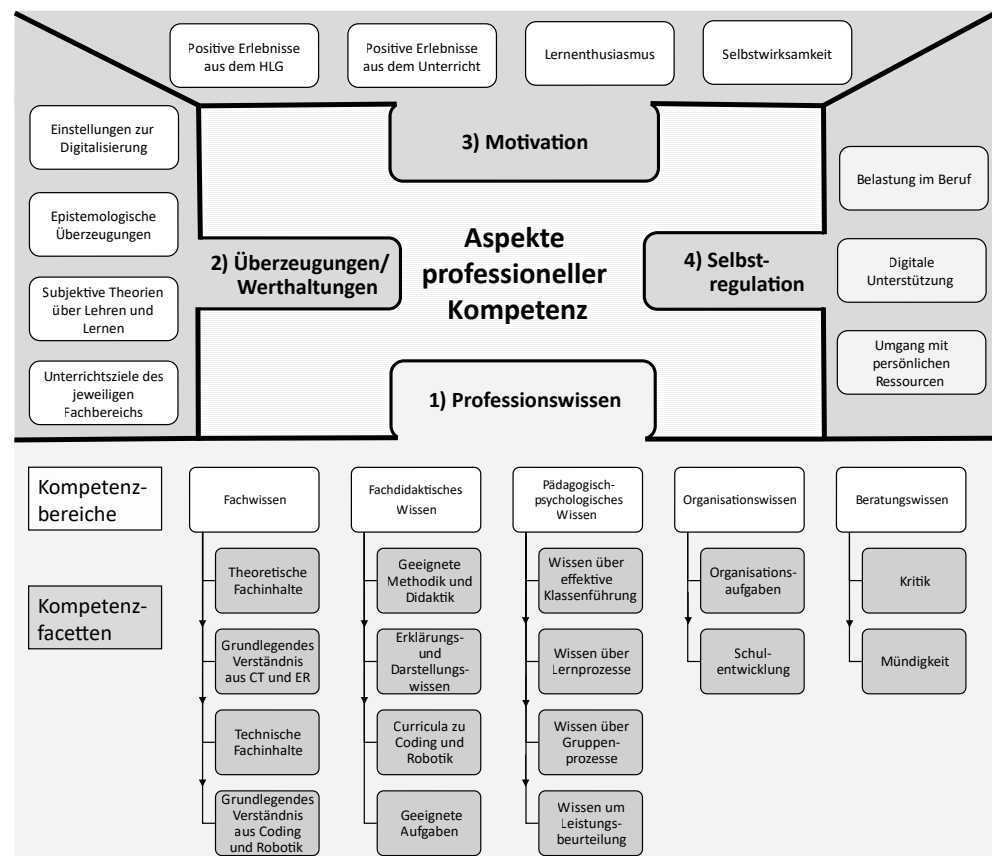


Abb. 1.: Theoretischer Rahmen des HLG Coding und Robotik gemäss dem strukturell erweiterten und modifizierten Kompetenzmodell COACTIV (nach Baumert und Kunter 2011, 32; Kunter, Klusmann, und Baumert 2009, 155).

Bei dem für den HLG *Coding und Robotik* besonders relevanten Aspekt des Professionswissens werden die Bereiche Fachwissen, fachdidaktisches Wissen, pädagogisch-psychologisches Wissen sowie Beratungs- und Organisationswissen unterschieden (Baumert und Kunter 2011, 32). Auf den HLG *Coding und Robotik* bezogen vermittelt das behandelte Fachwissen ausgehend von den theoretischen Inhalten ein grundlegendes Verständnis von CT mit ER als Problemlösestrategie, Lernwerkzeug und -objekt. Basierend auf geeigneten methodisch-didaktischen Vorgehensweisen und unter Berücksichtigung der curricularen Bedingungen seitens der unterschiedlichen Schulstufen und -arten, zielt der Erwerb von fachdidaktischem Wissen auf geeignete Aufgaben und Beispiele im Unterricht ab. Das zu vermittelnde pädagogisch-psychologische Wissen bezieht sich auf Besonderheiten des Unterrichtsfachs Coding und Robotik in Bezug auf Kompetenzen der effektiven Klassenführung, der Gestaltung von Lern- und Gruppenprozessen sowie der Leistungsbeurteilung. Das Organisationswissen betrifft die Organisation des Fachs bzw. lehrplanübergreifende oder integrative Vermittlung im schulischen Rahmen und umfasst auch weitere mit der Digitalisierung einhergehende schulische Herausforderungen, wie beispielsweise die Raumbedarfs- und Ressourcenplanung. Zusätzlich erfordern die im Schulkontext vermehrt auftretenden Beratungssituationen mündiges Beratungswissen hinsichtlich der Chancen, Grenzen und Risiken von Coding und Robotik im Besonderen und der Digitalisierung im Allgemeinen.

Für die Vermittlung des Professionswissens wird im HLG eine Modifikation des Think-Make-Improve-Modells (TMI-Modell) von Martinez und Stager (2013) verwendet. Bei diesem Modell wird der Problemlöseprozess ausgehend von verschiedenen «Design-Modells» (Resnick 2007; Boehm 1988) auf drei Phasen reduziert. Die Think-Phase beinhaltet die Problemstellung, die Identifikation und das Erlernen des nötigen Fachwissens, wodurch auch das Vorwissen aktiviert werden soll. In der Make-Phase wird das erarbeitete und erlernte Wissen einschliesslich der Strategien und Methoden zielgerichtet angewendet. In der Improve-Phase wird das Ergebnis dieser Anwendung kritisch reflektiert, um es zu verbessern, nötigenfalls auch zu verwerfen, wobei auch letzteres zu einem Erkenntnisgewinn führen kann. Durch die Form der Frage- und Aufgabenstellungen wird selbstgesteuertes Lernen in den Mittelpunkt gerückt. Ein weiterer wichtiger didaktischer Aspekt des HLG ist die Entwicklung und Produktion von Lehr- und Lernmaterial durch die studierenden Lehrkräfte für den eigenen Unterricht, um Aspekte des inneren Antriebs für den persönlichen Lernprozess nutzbar zu machen (Martinez und Stager 2013, 32). Der spielerische Aspekt von ER hilft zusätzlich dabei, den inneren Antrieb der studierenden Lehrkräfte zu wecken und zu stimulieren. Zudem wurde das TMI-Modell um den wichtigen Gesichtspunkt der Präsentation des eigenen Lernprodukts erweitert und so zum Think-Make-Improve-Present-Modell (TMIP-Modell) weiterentwickelt (s. Abbildung 2). Das TMIP-Modell berücksichtigt, dass eine dem Inhalt angemessene Präsentationsleistung

eine analytische, systematische und didaktisch aufbereitete Zusammenschau über sämtliche erarbeitete Lerninhalte erfordert. D. h., dass die einzelnen Lerninhalte zueinander in Beziehung gesetzt werden müssen, was zur Festigung und Vertiefung der Kenntnisse über die jeweilige Materie beiträgt. Durch den Aspekt der Präsentation soll jeder Studierende sowohl das Endprodukt als auch den Weg zu seiner Erreichung einer Reflexion unterziehen. Die Präsentation schliesst den Lernprozess mit einer Vorstellung des Endproduktes, seiner theoretischen wie praktischen Genese und einer fachlichen Diskussion in der Lerngruppe ab.

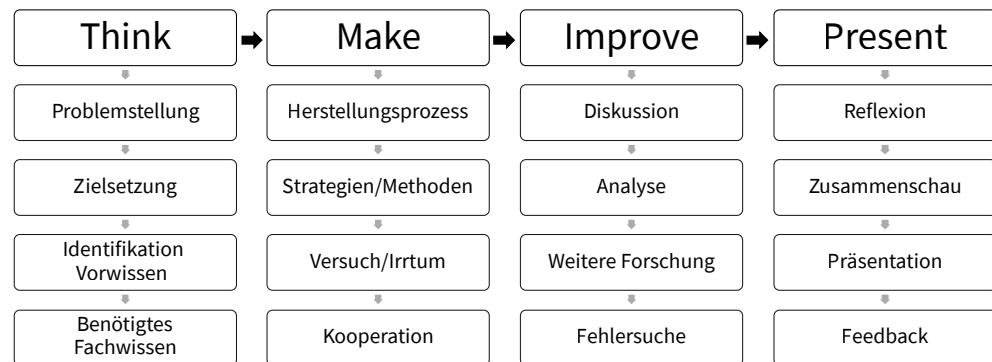


Abb. 2.: Das Think-Make-Improve-Present-Modell.

4. Der HLG Coding und Robotik

Da im österreichischen Burgenland Coding und Robotik seit dem Schuljahr 2018/19 Teil des schulischen Curriculums ist (Leitgeb 2018), wurde die Einrichtung einer entsprechenden institutionellen Weiterbildung in Form eines HLG erforderlich. An der Pädagogischen Hochschule Burgenland wurde daher der viersemestrige und 30 ECTS umfassende HLG Coding und Robotik entwickelt und im Herbst 2019 in deren Weiterbildungsangebot aufgenommen. Ziel des HLG ist es, die studierenden Lehrkräfte dazu zu befähigen, ihren Schülerinnen und Schülern einen modernen Unterricht in Coding und Robotik auf Höhe des fachlichen Diskurses zu ermöglichen. Dazu vermittelt der HLG fachwissenschaftliche und fachdidaktisch-methodische Kompetenzen (a) bezüglich des Programmierens und physischer Robotersysteme, (b) hinsichtlich einer differenzierten Reflexion über gesellschaftliche Entwicklungen im Rahmen der Automatisierung und Digitalisierung sowie (c) für die Planung, Durchführung und Evaluation von Unterrichtsszenarien mit physischen Robotersystemen und Algorithmen (vgl. Pädagogische Hochschule 2019, 4). Für die Planung und Umsetzung des HLG wurden bereits vorhandene CT-Schulcurricula herangezogen (Grossbritannien: vgl. Department of Education 2013; Burgenland: vgl. Leitgeb 2019). Aufbauend auf den dort thematisierten fachspezifischen und -didaktischen Kompetenzen wurden die in der Weiterbildung zu vermittelnden Kompetenzen bzw. Metakompetenzen

abgeleitet. Hochschuldidaktisch wird dabei dem erfolgreichen Transfer der Fachinhalte mittels geeigneter Aufgaben und Beispiele in die Unterrichtspraxis der Lehrkräfte, der Selbstregulation, der Motivation und den persönlichen Zielen besondere Aufmerksamkeit zuteil. So wird zu jeder fachinhaltlichen Lehrveranstaltung des HLG eine entsprechende fachdidaktische Übung angeboten, um die neu erlernten Inhalte bzw. neu erworbenen Kompetenzen möglichst unmittelbar für den Schuleinsatz methodisch-didaktisch aufzubereiten und zu vertiefen. Die Lehrkräfte werden u. a. mit Formen von Seamless-Learning und Distance-Learning mittels Videokonferenzen, Kleingruppentreffen und Lernvideos unterstützt. Eine vom Lehrgangsteam betreute digitale Lernplattform unterstützt diesen Prozess und ermöglicht den studierenden Lehrkräften auch eine effiziente Vernetzung untereinander.

Jedes der vier Semester des HLG Coding und Robotik besteht aus einem Modul:

- Im ersten Semester wird eine Einführung in die Methodik und Didaktik der Informatik gegeben (7 ECTS). Es werden Kompetenzen in den Bereichen theoretische Informatik, Algorithmen und Datenstrukturen sowie Betriebssysteme vermittelt, um in der Schule selbständig Projekte im Fach Coding und Robotik konzeptionieren, umsetzen und evaluieren zu können.
- Im zweiten Semester wird das Modul *Einführung in die Programmierung und Mediengestaltung* (6 ECTS) angeboten. Es dient der Aneignung von Kompetenzen hinsichtlich einer Auswahl bekannter audiovisueller Kommunikations- und Kollaborationstools sowie deren Einsatz im pädagogischen Umfeld. Diese Kompetenzen sollen dazu dienen, neue technologische Entwicklungen im Hinblick auf pädagogische Chancen, Grenzen und Risiken in der Primar- und Sekundarstufe I sowie auf persönliche und gesellschaftliche Auswirkungen hin einschätzen zu können.
- Für das dritte Semester ist das Modul *Vertiefende methodische und didaktische Aspekte von Coding und Robotik und deren schulpraktische Umsetzung mit Plattformen und Lernrobotern* (9 ECTS) vorgesehen. Die Inhalte dieses Moduls sind «Making, Tinkering, Engineering» (Martinez und Stager 2013) sowie die Schnittstelle zwischen Elektronik und Mechanik, also die Mechatronik. Im Mittelpunkt stehen die Entwicklung und Vertiefung jener Lernprozesse, die das Problemlösen mittels Programmierung und des Einsatzes mechanischer und elektronischer Bauteile mit besonderem Augenmerk auf das eigene Tun ermöglichen (z. B. das Designen, das Verbessern und der Bau eigener Plattformen einschliesslich Programmierung).
- Der HLG endet im vierten und letzten Semester mit dem Modul Coding und Robotik unter dem Aspekt der Schulentwicklung (8 ECTS). Es dient zur Vertiefung der Kompetenzen bezüglich grafischer Programmieroberflächen und zur Einführung in die textbasierte Programmierung. Darüber hinaus erwerben die Lehrkräfte fachliches Wissen und Fertigkeiten im Hinblick auf die digitale Schulorganisation

bzw. -entwicklung, um die damit einhergehenden Prozesse identifizieren, analysieren und optimieren sowie informatische Innovationen in der Schule implementieren zu können. Ferner werden weiterführende Kompetenzen zur Erstellung eigener Projektarbeiten und damit einhergehender Differenzierungen und Individualisierungen vermittelt (vgl. Pädagogische Hochschule Burgenland 2019).

Jede Lehrveranstaltung des HLG wird mit einer Note abgeschlossen, wobei die Benotung mittels der in Österreich üblichen fünfstufigen Notenskala (Sehr gut, Gut, Befriedigend, Genügend, Nicht genügend) erfolgt. Die Prüfungsleistungen bestehen aus der Konzeption und Herstellung von Unterrichtsmaterialien sowie aus der Dokumentation, Evaluation und Reflexion deren Einsatzes im eigenen Unterricht. Ferner werden lernbegleitend auch kurze Wissens- und Kompetenztests eingesetzt. An schriftlich zu verfassenden Arbeiten werden zwei kleinere Projektarbeiten und eine 4.500 Wörter umfassende Abschlussarbeit, die vor einer Prüfungskommission zu präsentieren ist, verlangt.

5. Begleitevaluation: Ergebnisse nach dem ersten Studienjahr

Um datengestützte Hinweise zu erhalten, inwieweit der im Wintersemester 2019 gestartete HLG die in ihn gesetzten Erwartungen erfüllt, erfolgt eine Begleitevaluation des Studiengangs. Die Studierenden der ersten Kohorte werden dabei studienbegleitend in einer Vollerhebung zum Ende eines jeden Studienjahres befragt. Aktuell liegen jene Daten vor, die im Juli 2020 zum Ende des ersten Studienjahres erhoben wurden. Zentrales Ziel der Begleitevaluation ist es, eine aussagekräftige Rückmeldung der Studierenden zu der von ihnen wahrgenommenen Qualität des Studienangebotes zu erhalten. Neben der Frage, wie zufrieden die Studierenden mit verschiedenen Aspekten des HLG sind, wird auch jener Frage nachgegangen, inwieweit die Studierenden eine Entwicklung in den Kompetenzen wahrnehmen, die der Lehrkonzeption des Studiengangs zugrunde liegen. Die Rückmeldungen der Studierenden werden genutzt, um das aktuelle Studienangebot zu optimieren und den HLG weiterzuentwickeln. Bei der Bewertung der Ergebnisse ist allerdings zu beachten, dass aufgrund der in Österreich bundesweiten Massnahmen gegen die Verbreitung des Coronavirus das zweite Semester zur Gänze in Form von Online-Lehre abgehalten wurde. Die Befragung wurde mittels Microsoft Forms durchgeführt. An der Evaluationsstudie nahmen 19 der insgesamt 20 Studierenden teil (neun Frauen, zehn Männer). Vier Studierende gaben ein Alter von bis zu 30 Jahren an, elf eines zwischen 31 und 40 Jahren und vier eines von über 40 Jahren. Sechs Studierende haben eine Berufserfahrung von bis zu fünf Jahren, vier eine von sechs bis zehn Jahren und neun eine von mehr als zehn Jahren. Zwölf Studierende unterrichten als Hauptfach Mathematik, fünf Deutsch und zwei Englisch.

Der eingesetzte Fragebogen umfasst 52 Items, die sich wie folgt aufteilen: Allgemeine Studienzufriedenheit (8 Items, «Wie zufrieden sind Sie rückblickend bis jetzt mit dem Hochschullehrgang?»), die Organisation des Studiengangs (6 Items, «Ich werde durch das Coding-und-Robotik-Team zu den Fragen des Studiums kompetent beraten»), Theorie-Praxis-Bezug (4 Items, «Theorie- und Praxiselemente sind gut verknüpft»), Umfang und Qualität der Arbeitsaufträge (6 Items, «Das Feedback der Lehrenden zu meinen Studienleistungen ist für mich nachvollziehbar und hilfreich»), Einschätzung der Kompetenzen der Lehrenden (5 Items, «Die Lehrenden sind fachinhaltlich und fachdidaktisch kompetent») und die persönliche Studiensituation (5 Items, «Der Schwierigkeitsgrad der Lehrveranstaltungen ist angemessen»). Für die Einschätzung der eigenen Kompetenzentwicklung (16 Items) wurden die Kompetenzziele der ersten beiden Semester herangezogen (s. Tabelle 1). Um die persönliche Belastung durch das Studium einschätzen zu können, wurden die studierenden Lehrkräfte zudem mittels zweier Einzelitems danach gefragt, ob der Arbeitsaufwand im Selbststudium höher war als zu Studienbeginn vermutet, und ob ihre Familie oder ihr Beruf unter der zeitlichen Belastung durch das HLG-Studium gelitten hat. Alle Items waren mit einer vierstufigen Likert-Skala vorgegeben (1 «trifft nicht zu», 2 «trifft eher nicht zu», 3 «trifft eher zu», 4 «trifft zu»). Für die Auswertung wurden die Punkte der Itemantworten für jede Skala aufsummiert und durch die Anzahl der Items geteilt. Die Datenauswertung erfolgte mit SPSS 26.

Ergebnisse

Der Studiengang wurde insgesamt über alle Items hinweg sehr positiv bewertet, nämlich hinsichtlich der Allgemeinen Studienzufriedenheit (MW = 3,85; s = 0,24), der Organisation des Studiengangs (MW = 3,97; s = 0,16), des Theorie-Praxis-Bezugs (MW = 3,85; s = 0,34), des Umfangs und der Qualität der Arbeitsaufträge (MW = 3,90; s = 0,21), der Einschätzung der Kompetenzen der Lehrenden (MW = 3,91; s = 0,24) und der persönlichen Studiensituation (MW = 3,76; s = 0,27). Der Schwierigkeitsgrad der Lehrveranstaltungen sowie der Aufwand zur Erbringung der Studienleistungen erschienen als angemessen. Die Kommunikation bezüglich der Leistungsanforderungen wurde als transparent wahrgenommen. Zudem wurde bescheinigt, dass die eingesetzten Lernmaterialien auch für die Arbeit im häuslichen Umfeld hilfreich sind. Die Aussage, dass der Arbeitsaufwand für das Selbststudium höher als erwartet war, bewerteten 7 bzw. 3 Studierende als «eher zutreffend» bzw. «zutreffend». Gleichzeitig beurteilten 6 bzw. 10 der studierenden Lehrkräfte die Aussage, dass Familie und Beruf unter den zeitlichen Belastungen leiden würden, als «nicht» bzw. «eher nicht zutreffend». Nur drei Studierende gaben an, dass dies («eher») der Fall gewesen sei. Ferner nahmen die Studierenden bei sich im Studium eine ausgeprägte Kompetenzentwicklung wahr (MW = 3,79; s = 0,31; siehe Tabelle 1). Dies war über alle 16 Teilkompetenzen hinweg sehr konsistent und einhellig der Fall.

Für die Skala zur eigenen Kompetenzentwicklung wurden vertiefende inferenzstatistische Auswertungen mittels U- und Kruskal-Wallis-Tests durchgeführt. Zwischen männlichen und weiblichen Studierenden waren keine signifikanten Unterschiede in den mittleren Tendenzen festzustellen. Dies war auch bei den Vergleichen hinsichtlich der Berufserfahrung der Fall. Berichtenswert ist, dass die Lehrkräfte mit den Hauptfächern Deutsch oder Englisch ihren Kompetenzzuwachs signifikant höher einschätzten als die Mathematiklehrkräfte (Cohen's $d = 0,396$, mittlerer Effekt nach Cohen 1988).

Kompetenzzuwachs (16 Items)	Trifft zu (4)	Trifft eher zu (3)	Trifft eher nicht zu (2)	Trifft nicht zu (1)	MW	s
Kompetenzzuwachs bezüglich:						
Programmieren in unterschiedlichen Entwicklungsumgebungen (Scratch, Blockly, Make Code, ...)	15	4	0	0	3,79	0,42
Kenntnis über grundlegende Begriffe und Methoden der Robotik	15	4	0	0	3,79	0,42
Kenntnis über grundlegende Begriffe und Methoden von Computational Thinking	15	4	0	0	3,79	0,42
Fähigkeit zum Einsatz von Lernrobotern im Unterricht (z. B. BeeBot, LEGO WeDo)	16	3	0	0	3,84	0,38
Fähigkeit zum Einsatz grundlegender Konzepte von Computational Thinking in der Schule	15	3	1	0	3,74	0,56
Fähigkeit zur kritischen Reflexion pädagogischer Handlungen im Kontext von Coding und Robotik	16	3	0	0	3,84	0,37
Fähigkeit zur Identifizierung und Benennung von Problemen aus dem Themenbereich Coding und Robotik	15	4	0	0	3,79	0,54
Fähigkeit, angemessene Probleme aus dem Themenbereich Coding und Robotik in der Gruppe zu analysieren und zu lösen	16	3	0	0	3,84	0,38
Kenntnis über grundlegende Aspekte bezüglich Algorithmen	14	4	1	0	3,68	0,58
Fähigkeit zur fachspezifischen Aufbereitung grundlegender Aspekte bzgl. Algorithmen für den Unterricht	14	4	1	0	3,68	0,58
Didaktisches Grundwissen und Methodenkenntnis, um für SuS entwicklungsfördernde Lernsituationen für den Bereich Coding und Robotik zu schaffen.	16	3	0	0	3,84	0,38
Fähigkeit, Informationen digital aufzubereiten (z. B. Erstellen und Bearbeiten von Lehr- und Lernvideos) und auf einem Lern-Management-System zugänglich zu machen	17	2	0	0	3,89	0,31

Kompetenzzuwachs (16 Items)	Trifft zu (4)	Trifft eher zu (3)	Trifft eher nicht zu (2)	Trifft nicht zu (1)	MW	s
Fähigkeit zur Erstellung angemessener Aufgaben aus dem Themenbereich Coding und Robotik für den Unterricht	16	3	0	0	3,84	0,38
Fähigkeit zum Einsatz selbsterstellter Lehr- und Lernmaterialien zum Themenbereich Coding und Robotik in der Praxis	17	3	0	0	3,89	0,31
Fähigkeit zur Reflexion der DSGVO im eigenen Tun	15	2	2	0	3,63	0,68
Kenntnis über rechtliche und sicherheitsrelevante Aspekte der Informatik	15	3	1	0	3,74	0,56

Tab. 1.: Einschätzung der Studierenden zum eigenen Kompetenzzuwachs im ersten Studienjahr (Häufigkeiten, Mittelwerte und Streuungen).

6. Zusammenfassung

Die zunehmende Digitalisierung verändert das gesellschaftliche, berufliche und private Leben grundlegend. Von dieser Veränderung ist auch die Schule in hohem Masse betroffen, ist es doch ihre Aufgabe, junge Menschen auf die damit verbundenen neuen Herausforderungen der Lebenswirklichkeit vorzubereiten. Ein diesbezüglich vielversprechender Ansatz ist es, Kompetenzen curricular in den Blick zu nehmen, die unter *Computational Thinking* (CT) subsumiert werden (Wing 2008; Eickelmann u. a. 2019, 367). Die schulische Vermittlung der oftmals sehr komplexen und abstrakten Inhalte von CT bedarf aber einer geeigneten didaktisch-pädagogischen Aufbereitung, wobei sich hierfür eine enge Verbindung mit Educational Robotics (ER) anbietet (Eguchi 2014; García-Valcárcel Muñoz-Repiso und Caballero-González 2019, 65). Die Implementation eines entsprechenden schulischen Curriculums erfordert eine angemessene fachwissenschaftliche und -didaktische Lehrkräfteaus-, -fort- und -weiterbildung im Bereich CT und ER. Zu diesem Zweck wurde an der Pädagogischen Hochschule Burgenland der viersemestrige und 30 ECTS umfassende Hochschullehrgang (HLG) Coding und Robotik für Lehrkräfte entwickelt und eingeführt. Als theoretischer Rahmen für die konzeptionelle Gestaltung dieses HLG wurde das COACTIV-Modell von Baumert und Kunter (2011) bzw. Kunter, Klusmann und Baumert (2009) herangezogen, allerdings in einer strukturell erweiterten und an die fachlichen Erfordernisse des HLG angepassten Form (s. Abbildung 1). Das daraus resultierende Modell wurde für die Entwicklung und Konzeptionierung des HLG, aber auch für die inhaltliche Gestaltung der Lehrveranstaltungen genutzt. Auf diese Weise wurde auf Kompetenzebene eine einheitliche und enge Vernetzung der einzelnen Lehrveranstaltungen erreicht. Als methodisch-didaktische Grundlage der Lehrveranstaltungen

wurde aufbauend auf dem Think-Make-Improve-Modell (TMI) von Martinez und Stager (2013) das Think-Make-Improve-Present-Modell (TMIP) entwickelt (s. Abbildung 2). Durch die Present-Phase werden die studierenden Lehrkräfte im HLG zu einer professionellen Reflexion und Zusammenschau der CT-spezifischen fachwissenschaftlichen und -didaktischen Lerninhalte angeregt.

Zum Ende des ersten Studienjahres der ersten Kohorte wurde eine erste Zwischenevaluation des HLG durchgeführt. Die Ergebnisse zeigen insgesamt ein positives Bild, angefangen bei der hohen allgemeinen Studienzufriedenheit über die Organisation des Studiengangs, den Theorie-Praxis-Bezug, den Umfang und die Qualität der Arbeitsaufträge, die persönliche Studiensituation bis hin zur Einschätzung der Kompetenz der HLG-Lehrenden. Jedoch hatten einige Studierende einen geringeren Arbeitsaufwand für das Selbststudium erwartet. Besonders positiv hervorzuheben ist, dass die studierenden Lehrkräfte individuell über alle im Curriculum behandelten Teilkompetenzen hinweg einen ausgeprägten Kompetenzzuwachs wahrnahmen (s. Tabelle 1). Dies ist zum einen ein Beleg dafür, dass seitens der studierenden Lehrkräfte wichtige Kompetenzen für eine angemessene schulische Auseinandersetzung mit CT und ER vor Aufnahme des Studiums tatsächlich fehlten, zum anderen, dass diese in Form eines HLG entwickelt werden können. Ferner zeigt die Implementierung des HLG Coding und Robotik, wie wichtig die Wahl eines geeigneten, tragfähigen theoretischen Rahmens für die inhaltliche und didaktische Konzeptionierung und Umsetzung eines HLG ist.

Literatur

- Aho, Alfred V. 2012. «Computation and computational thinking». *Computer Journal*, 55(7): 833-835. <https://doi.org/10.1093/comjnl/bxs074>.
- Alimisis, Dimitris, und Chronis Kynigos. 2009. «Constructionism and robotics in education». In *Teacher Education on Robotic-Enhanced Constructivist Pedagogical Methods*, 11-26. Athens: ASPETE. http://roboesl.eu/wp-content/uploads/2017/08/chapter_1.pdf.
- Angel-Fernandez, Julian M., Markus Vincze. 2018. «Towards a formal definition of Educational Robotics». In *Proceedings of the Austrian Robotics Workshop 2018*, 37-42. Innsbruck: University Press. <https://doi.org/10.15203/3187-22-1-08>.
- Angeli, Charoula, Joke Voogt, Andrew Fluck, Mary Webb, Margaret Cox, Joyce Malyn-Smith, und Jason Zagami. 2016. «A K-6 Computational Thinking Curriculum Framework: Implications for Teacher Knowledge». *Educational Technology & Society*, 19(3): 47-57. <https://www.jstor.org/stable/jeductechsoci.19.3.47>.
- Balaskat, Anja, und Katja Engelhart. 2015. «Computing our future: Computing programming and coding – Priorities, school curricula and initiatives across Europe». Brüssel: European Schoolnet. http://www.eun.org/documents/411753/817341/Computing+our+future_final_2015.pdf/d3780a64-1081-4488-8549-6033200e3c03.

- Baumert, Jürgen, und Mareike Kunter. 2011. «Das mathematikspezifische Wissen von Lehrkräften, kognitive Aktivierung im Unterricht und Lernfortschritte von Schülerinnen und Schülern». In *Professionelle Kompetenz von Lehrkräften. Ergebnisse des Forschungsprogramms COACTIV*, herausgegeben von Mareike Kunter, Jürgen Baumert, Werner Blum, Uta Klusmann, Stefan Krauss, und Michael Neubrand. 163-192. Münster: Waxmann.
- Bocconi, Stefania, Augusto Chioccariello, Giuliana Dettori, Anusca Ferrari, und Katja Engelhardt. 2016. «Developing computational thinking in compulsory education – Implications for policy and practice». Herausgegeben von Panagiotis Kampylis und Yves Punie. Luxembourg: Publications Office of the European Union. <https://doi.org/10.2791/792158>.
- Boehm, Barry. 1988. «A spiral model of software development and enhancement». *Computer* 21(5): 61-72. <https://doi.org/10.1109/2.59>.
- Bromme, Rainer. 1997. «Kompetenzen, Funktionen und unterrichtliches Handeln des Lehrers». In *Psychologie des Unterrichts und der Schule*, herausgegeben von Franz E. Weinert. Enzyklopädie der Psychologie, Pädagogische Psychologie Bd. 3. 177–212. Göttingen, Bern: Hogrefe.
- Cohen, Jacob. 1988. *Statistical power analysis for the behavioral sciences*. Hillsdale, N. J.: L. Erlbaum Associates.
- Computer Science Teachers Association & International Society for Technology in Education. 2011. *Operational Definition for Computational Thinking*. <https://cdn.iste.org/www-root/ct-documents/computational-thinking-operational-definitionflyer.pdf?sfvrsn=2>.
- Csizmadia, Andrew, Paul Curzon, Mark Dorling, Simon Humphreys, Thomas Ng, Cynthia Selby, und John Woollard. 2015. *Computational thinking – a guide for teachers*. Computing At School. <https://community.computingatschool.org.uk/files/8550/original.pdf>.
- DeE – Department of Education UK. 2013. «Computing programmes of study: key stages 1, 2, 3 and 4». https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/239067/SECONDARY_national_curriculum_-_Computing.pdf.
- Denning, Peter J. 2009. «The profession of IT Beyond computational thinking». *Communications of the ACM*, 52(6): 28-30. <https://doi.org/10.1145/1516046.1516054>.
- Eguchi, Amy. 2014. «Educational robotics for promoting 21 century skills». *Journal of Automation, Mobile Robotics & Intelligent Systems*, 8(1): 5-11. https://doi.org/10.14313/JAMRIS_1-2014/1.
- Eickelmann, Birgit, Vahrenhold, Jan, und Amelie Labusch. 2019. «Der Kompetenzbereich Computational Thinking: erste Ergebnisse des Zusatzmoduls für Deutschland im internationalen Vergleich». In *ICILS 2018 #Deutschland. Computer- und informationsbezogene Kompetenzen von Schülerinnen und Schülern im zweiten internationalen Vergleich und Kompetenzen im Bereich Computational Thinking*, herausgegeben von Birgit Eickelmann, Wilfried Bos, Julia Gerick, Frank Goldhammer, Heike Schaumburg, Knut Schwippert, Martin Senkbeil, und Jan Vahrenhold, 367-398. Münster, New York: Waxmann. <http://nbn-resolving.org/urn:nbn:de:0111-pedocs-183309>.

- Fislake, Martin. 2021. «Educational Robotics between Coding und Engineering Education». In *Using Educational Robotics to Facilitate Student Learning*, herausgegeben von Stamatios Papadakis und Michail Kalogiannakis, 107-139. Hershey: IGI Global. <https://doi.org/10.4018/978-1-7998-6717-3.ch004>.
- Genser, Sebastian, und Thomas Leitgeb. 2020. «Ein gelungener Übergang von der Primarstufe in die Sekundarstufe I – die VS und MS St. Michael zeigen es vor». *phpublico – Zeitschrift für Bildung und Erziehung*, 4: 30-36. https://www.ph-burgenland.at/fileadmin/user_upload/Forschung/Forschungsergebnisse/ph_publico/phpublico_2020_04.pdf.
- García-Valcárcel-Muñoz-Repiso, Ana, und Yen-Air Caballero-González. 2019. «Robotics to develop computational thinking in early Childhood Education». *Comunicar – Media Education Research Journal* 28: 63-72. <https://doi.org/10.3916/C59-2019-06>.
- Hsu, Ting-Chia, Shao-Chen Chang, und Yu-Ting Hung 2018. «How to Learn and How to Teach Computational Thinking». *Computers and Education*, 126: 296-310. <https://doi.org/10.1016/j.compedu.2018.07.004>.
- Ilic, Ulas, Halil Ibrahim Haseski, und Ufuk Tugtekin. 2018. «Publication Trends over 10 years of computational Thinking Research». *Contemporary Educational Technology*, 9(2): 131-153. <https://doi.org/10.30935/cet.414798>.
- Karampinis, Tassos. 2018. «Robotics-Based Learning Interventions and Experiences from our Implementations in the RobESL Framework». *International Journal of Smart Education and Urban Society*, 9(1): 1-13. <https://doi.org/10.4018/IJSEUS.2018010102>.
- Kunter, Mareike, Uta Klusmann, und Jürgen Baumert. 2009. «Professionelle Kompetenz von Mathematiklehrkräften. Das COACTIV-Modell». In *Lehrprofessionalität – Bedingungen, Genese, Wirkungen und ihre Messung*, herausgegeben von Olga Zlatkin-Troitschanskaia, Klaus Beck, Detlef Sembill, Reinhold Nickolaus, und Regina Mulder, 153-165. Weinheim: Beltz.
- Leitgeb, Thomas. 2018. «Coding und Robotik für alle». *OCG Journal*, 43(1): 32-33. <https://www.ocg.at/sites/ocg.at/files/medien/pdfs/OCG-Journal1801.pdf>.
- Leitgeb, Thomas. 2019. «Digital-unterstützte Lehrkräftefortbildung am Beispiel Coding und Robotik». Master Thesis, University of applied Sciences Burgenland. https://www.researchgate.net/publication/336578633_Digital_inkludierte_Lehrerprofessionalisierung_am_Bei-spiel_von_Coding_und_Robotik_ii.
- Li, Yeping, Alan H. Schoenfeld, Andrea A. diSessa, Arthur C. Graesser, Lisa C. Benson, Lyn D. English, und Richard A. Duschl. 2020. «Computational Thinking is more about thinking than computing». *Journal for STEM Education Research*, 5(18): 1-8. <https://doi.org/10.1007/s41979-020-00030-2>.
- Ma, Hsen-Hsing. 2009. «The effect size of variables associated with creativity: A meta analysis». *Creativity Research Journal*, 21: 30-42. <https://doi.org/10.1080/10400410802633400>.
- Martinez, Sylvia, und Gary Stager. 2013. *Invent to learn – Making, Tinkering and Engineering in the classroom*. Torrance: Constructing Modern Knowledge Press.
- Orvalho, Joao. 2017. «Computational Thinking for Teacher Education». Paper presented at the Scratch2017BDX Opening, Inspiring, Connecting. 6. Bordeaux, France. <https://hal.inria.fr/hal-01555455>.

- Pädagogische Hochschule Burgenland. 2019. *Curriculum Hochschulelehrgang Coding und Robotik*. https://www.ph-burgenland.at/fileadmin/user_upload/Studium/Hochschulelehrgaenge/HLG_Coding_und_Robotik.pdf.
- Pollak, Michael, und Martin Ebner. 2019. «The Missing Link to Computational Thinking». *Future Internet*. 11(12): 263. <https://doi.org/10.3390/fi11120263>.
- Resnick, Mitchel. 2007. «All I Really Need to Know (About Creative Thinking) I Learned (By Studying How Children Learn) in Kindergarten». *ACM Creativity & Cognition conference*, Washington DC, June 2007. <https://web.media.mit.edu/~mres/papers/kindergarten-learning-approach.pdf>.
- Sentance, Sue, und Andrew Csizmadia. 2015. «Teachers' perspectives on successful strategies for teaching computing in school». In *IFIP TC3 Working Conference. A New Culture of Learning: Computing and Next Generations*, 243-252. Vilnius. http://www.ifip2015.mii.vu.lt/file/repository/IFIP_Proceedings.pdf.
- Scherer Ronny, Fazilat Siddiq, und Bárbara Sánchez Viveros. 2018. «The Cognitive Benefits of Learning Computer Programming: A Meta Analysis of Transfer Effects». *Journal of Psychology*, 111(5): 764-792. <https://doi.org/10.1037/edu0000314>.
- Schuhr, Jan C. 2015. «Recht, Technik, Roboter». *Rechtstheorie*, 46(2): 225-261. <https://doi.org/10.3790/rth.46.2.225>.
- Shi, Wenchong. 2018. «Summary of global research and practice of Computational Thinking». *CEA – Computer Engineering and Applications* 54(4): 31-35. <http://cea.ceaj.org/EN/Y2018/V54/I4/31>.
- Shulman, Lee S. 1986. «Those who understand: Knowledge growth in teaching». *Educational Researcher* 15(2): 4-14. <https://doi.org/10.3102/0013189X015002004>.
- Shute, Valerie J., Chen Sun, und Jodi Asbell-Clarke. 2017. «Demystifying computational thinking». *Educational Research Review*, 22: 142-158. <https://doi.org/10.1016/j.edu-rev.2017.09.003>.
- Undervisningsministeriet. 2018. *Computational tankegang*. <https://www.emu.dk/grundskole/teknologiforstaelse>.
- Wing, Jeannette M. 2006. «Computational Thinking». *Communication of the ACM*, 49: 33-35. <https://doi.org/10.1145/1118178.1118215>.
- Zimmermann, Alexander. 2020. «Mathematisches Argumentieren im Schulunterricht fördern». *R&E-Source, Online Journal for Research and Education*, 13: 1-10. <https://journal.ph-noe.ac.at/index.php/resource/article/view/781>.