Computing Education Research in Finland

Lauri Malmi, Arto Hellas, Petri Ihantola, Ville Isomöttönen, Ilkka Jormanainen, Terhi Kilamo, Antti Knutas, Ari Korhonen, Mikko-Jussi Laakso, Sonsoles López-Pernas, Timo Poranen, Tapio Salakoski, and Jarkko Suhonen

1 Introduction

In this chapter, we present first an overview of the educational system and computer science education in Finland, followed by scientometric analysis of CER publications with Finnish authors. Thereafter we present briefly work carried out in Finnish research groups and finally reflect on the factors behind the intensive work in CER in Finland.

L. Malmi (✉) · A. Hellas · A. Korhonen
Aalto University, Espoo, Finland
e-mail: Lauri.Malmi@aalto.fi; Arto.Hellas@aalto.fi; Ari.Korhonen@aalto.fi

P. Ihantola
University of Helsinki, Helsinki, Finland
e-mail: petri.ihantola@helsinki.fi

V. Isomöttönen
University of Jyväskylä, Jyväskylä, Finland
e-mail: ville.isomottonen@jyu.fi

I. Jormanainen · J. Suhonen · S. López-Pernas
University of Eastern Finland, Joensuu, Finland
e-mail: ilkka.jormanainen@uef.fi; sonsoles.lopez@uef.fi; jarkko.suhonen@uef.fi

T. Kilamo · T. Poranen
Tampere University, Tampere, Finland
e-mail: terhi.kilamo@tuni.fi; timo.poranen@tuni.fi

A. Knutas
LUT University, Lappeenranta, Finland
e-mail: antti.knutas@lut.fi

M.-J. Laakso · T. Salakoski
University of Turku, Turku, Finland
e-mail: milaak@utu.fi; tapio.salakoski@utu.fi

M. Apiola et al. (eds.), *Past, Present and Future of Computing Education Research*,
https://doi.org/10.1007/978-3-031-25336-2_16

1.1 Finnish Educational System

Finland is located in northern Europe and has a population of 5.5 million people. It is a member of the European Union (EU) and associated with Nordic countries together with Sweden, Norway, Denmark and Iceland.

Finnish educational system includes pre-school education for 6-years old children and comprehensive school (grades 1–9), followed by either high school (grades 10–12) or vocational education. Currently, students are requested to continue their studies in secondary education after comprehensive school until 18 years of age and the goal is that everyone should get either a high school or vocational school degree. Tertiary education covers two branches. Research universities provide Bachelor's, Master's and Doctoral education, with target studying time (3 + 2 + 4 years) correspondingly. Universities of applied sciences provide more practically oriented degrees in a large number of different professions (4 years) which roughly correspond to Bachelor's level degrees in research universities.

All teacher education is given in research universities. Pre-school teachers must have at least a Bachelor's degree in educational sciences, while primary school teachers responsible for a class of pupils must have at least a Master's degree in educational sciences. Subject teachers in primary school (grades 1–6), i.e., teachers responsible for teaching a particular subject such as mathematics or arts, must have at least 60 ECTS[1] worth of studies on the subject they are teaching. In lower secondary school (grades 7–9), subject teachers are required to have a master's degree (not necessarily from educational sciences) including at least 120 ECTS worth of studies in their main teaching subject and at least 60 ECTS worth of studies from other subjects they teach. They also must have completed at least 60 ECTS worth of pedagogical studies in their teacher education specialization track. Similar pedagogical studies are also required for teachers in universities of applied sciences but not in research universities, where requirements for pedagogical studies vary but typically are much smaller.

Students are admitted to tertiary education based on their national level matriculation exam results (high school track) or vocational degree or based on a field-specific entrance examination or a combination of the previous ones. Students from universities of applied sciences can continue to master's level studies in research universities within a competitive admission process.

Bachelor level education in universities is widely given in Finnish and partially in Swedish, the other official language of Finland. There are few bachelor level programs where education is given fully in English and they are targeted to international students and immigrants with no sufficient command of Finnish or Swedish. On the other hand, on master's level education, programs provided in English are much more common. Many universities of applied sciences also provide targeted programs in English to recruit good international students.

[1] European Credit Transfer System. One ECTS means roughly 26 h of work.

One leading principle in the Finnish educational system has been that education is free. There are no tuition fees. Only quite recently international students coming from non-EU countries have been requested to pay tuition fees. Generally, the programs also provide scholarships options to waive the fees partially or even wholly.

1.2 Computer Science Education in Finland

Computer science education in Finnish universities began in the 1960s when the first professorships were established. Currently, computer science and/or information systems programs are available in almost all universities. While learning programming has been a natural part of computer science programs, programming courses have also been widely taught for CS minors and as service courses. As a consequence, for a very long time, teachers have faced the challenge that the introductory programming courses are large ranging from several dozens of students to courses with 1000+ students. While the course sizes naturally vary among the universities, a common challenge has been the very limited number of faculty members as teachers. The main approaches to address this challenge have been using large numbers of teaching assistants, mostly BSc level students, to instruct younger students, and building in-house tools to support programming education, or adopting such software from other universities. Commercial solutions from companies either in Finland or elsewhere have been used on a very limited scale.

Development of in-house learning tools has generally been initiated by active teachers of large courses either as their own work, based on student projects or funded by small educational development grants provided by computer science departments or universities. This development work started actively in University of Helsinki, Helsinki University of Technology and University of Joensuu in the 1990s and a few years later in several other universities independently from each other. The tools were tailored for addressing local educational challenges in basic programming and data structures and algorithms courses. The corresponding pedagogical reforms were carried out from the same perspective.

This extensive effort in developing education was the seed for initiating research in several sites. Already in the 1990s, the first experience reports were published in educational development conferences organized in Finland, such as Hypermedia in Vaasa 1993 and 1994. The international perspective was adopted by the pioneers when ACM Innovations and Technology in Computer Science Education Conference (ITiCSE) was organized in Uppsala, Sweden in 1997 and in Helsinki in 2000 (chaired by prof. Jorma Tarhio). Moreover, the first Program Visualization Workshop was organized in 2000 by professor Erkki Sutinen in Porvoo. The pioneering professors launched the Koli Calling conference in 2001 as a swap meeting for Finnish computer science teachers. A few years later, the conference took steps towards an international research conference.

Concerning education in the K-12 level, however, Finland has not been among the pioneers globally, and computer science has never been an independent school subject. Information and Communication Technology (ICT) skills was introduced as a voluntary subject to the 8th and 9th grade of comprehensive school in the late 1980s. Pupils who chose the subject –depending on the teacher– also had the opportunity to learn programming (e.g. with Pascal). In the early 1990s, ICT was to be integrated in other subjects, which was further emphasized by the Ministry of Education in the early 2000s. This effort did not fully succeed [18]. However, due to the relative freedom of school teachers to organize additional voluntary courses, some pupils had the opportunity to learn programming despite the integration efforts [75]. Only in 2016, the school curriculum was finally revised to include computational thinking and basics of programming. These are most often implemented in the context of mathematics education.

Computer science teacher training has been organized at many universities as part of their teacher training programs. However, this sub area has not been very popular due to the availability of very few teacher positions in schools in computer science. Therefore, some programs have even been discontinued.

2 Finnish CER Community: Scientometric Analysis

In this section, we present findings from our scientometric analysis of CER in Finland. A subset of the dataset described in chapter "Scientometrics: A Concise Introduction and a Detailed Methodology for Mapping the Scientific Field of Computing Education Research" of this book [92] (containing CER worldwide) has been created by including only those papers in which at least one of the authors had a Finnish affiliation at the moment of publication. We have interpreted the affiliations as Finnish if the author has self-given an affiliation that matches a Finnish university or other institute. Thus, Finnish authors visiting foreign institutes and using their affiliation there are excluded. Correspondingly, foreign visitors in Finnish universities are excluded if they are using their home affiliation. In some cases, authors have not given a clearcut affiliation. These are excluded. The total number of articles included for analysis is 535.

We must also emphasize, as explained in chapter "Scientometrics: A Concise Introduction and a Detailed Methodology for Mapping the Scientific Field of Computing Education Research", that the metadata collection of CER articles was performed in Scopus. Scopus does not include all publication years of venues where CER papers are regularly published. Moreover, the keyword search used for finding CER papers in other publication venues is limited to the keywords we used. Therefore, it is understandable that the total publication and citation counts of specific authors are lower than what one could find, for example, from their Google Scholar page. We, however, believe that our data from Scopus corresponds well enough to theoretically complete data, if such were available, because we report

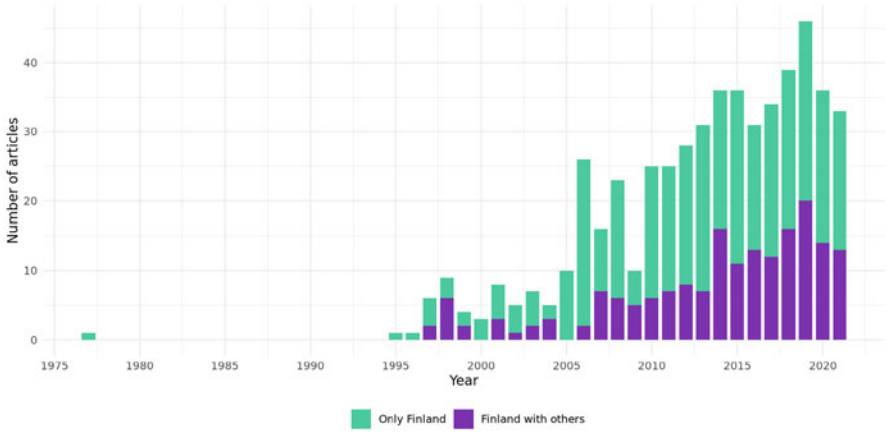

Fig. 1 Evolution of publications with Finnish affiliation. Green color indicates papers with Finnish affiliation only, and cyan color indicates papers with also international authors

mainly data concerning the most active researchers. They have been working in CER for many years, and their collaboration networks have evolved over the years.

Figure 1 presents the growth of the number of papers with Finnish affiliation in our data pool. The oldest paper [49] from 1977 discusses education from a systems' approach, presenting it as a data communication process. Thereafter, there is a long pause and only in the late 1990s a continuous stream of papers begins to appear with rapid growth reaching the level of 30–40 papers annually around 2015. It is notable that international collaboration emerged very early, and it still has a very strong role. In most years, roughly 25–40% of papers also have international authors. Note that the data pool also includes papers where the main work, including data collection and analysis, has been carried out outside of Finland, but some Finnish researchers have been participating in them as co-authors.

The ten most productive authors and their publication history are shown in Fig. 2. From those in the list, pioneers in the field are Erkki Sutinen (published since 1997), Lauri Malmi, Ari Korhonen (since 2000) and Jarkko Suhonen (since 2001). Only Malmi and Sutinen in the list are professors whose own PhD was from another area of computer science. All others are PhD graduates in CER. In total, at least 50 PhD theses have been completed in Finnish research teams during the last 20 years.[2]

Finnish authors have built their own collaboration networks, which are shown in Fig. 3. The size of the circle indicates the total publication activity and the width of connecting lines indicates the number of joint publications between the authors calculated using fractional counting. In fractional counting, instead of

[2] Many more have been completed in the groups in other, closely related areas, such as engineering education research, educational technology or ICT4D. We counted only those ones in CER. We, however, acknowledge that the borderline of what is included in CER or not is not always obvious.

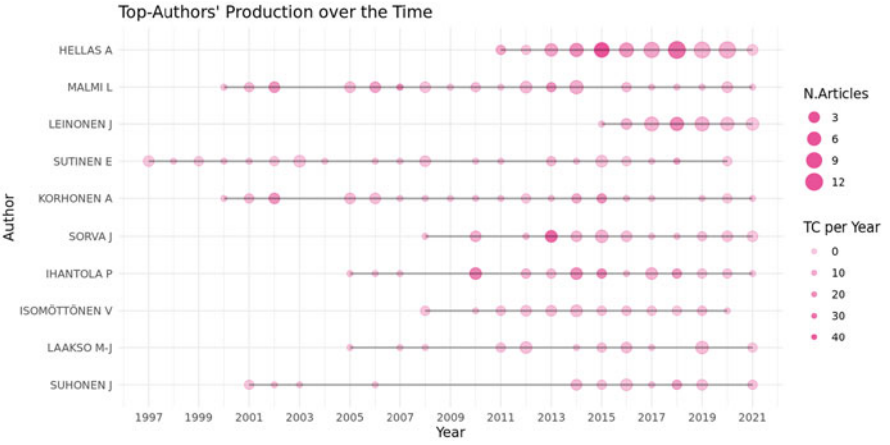

Fig. 2 Most productive authors with Finnish affiliation. The sizes of circles indicate the number of papers published by the author in a specific year. The color indicates the citations for the authors in a specific year. Data is from Scopus

counting each publication as "one" between each pair of co-authors, the count is divided by the number of co-authors in the paper (see chapter "Scientometrics: A Concise Introduction and a Detailed Methodology for Mapping the Scientific Field of Computing Education Research" for a more detailed explanation). Many of the stronger links reflect the supervisor-PhD student/postdoc relation, but this is no general rule. Colors indicate communities who have done more work together in terms of joint papers. Note that the communities are identified by an algorithm, and they are not disjoint; thus people can be a part of several communities—coloring cannot fully visualize this.

For example, Hellas and Leinonen have had very strong collaboration, as well as Malmi and Korhonen, Korhonen and Karavirta, Sutinen and Suhonen, Laakso and Apiola. Due to overlapping edges and nodes in the graph layout, some collaboration is not visible, or might give a somewhat misleading image. For example, strong collaboration between Ihantola and Karavirta is partially hidden behind the edge between Karavirta and Korhonen. Moreover, Sorva and Sirkiä have much collaboration, but this is not connected with Sheard.

On the top, in pink and orange, we can see the University of Eastern Finland (UEF) team with Sutinen, Suhonen, Tedre, Jormanainen, Toivonen and Oyrlele as the key people. On the top right, there is the University of Turku (UTU) team with Laakso, Apiola and Salakoski as the main people. On the other hand, Sutinen has moved from UEF to UTU building new collaborations there. In the center, there is the Aalto University team in gray and pink with Malmi, Sorva, Korhonen and Kinnunen (as main Finnish authors). Below this, there is the wide circle of Hellas who has a very large network with Leinonen, Luukkainen, and Ihantola forming the core of researchers at University of Helsinki. Low left in yellow is Lappeenranta University of Technology group (Knutas, Ikonen, Kasurinen, et al.) and on the left,

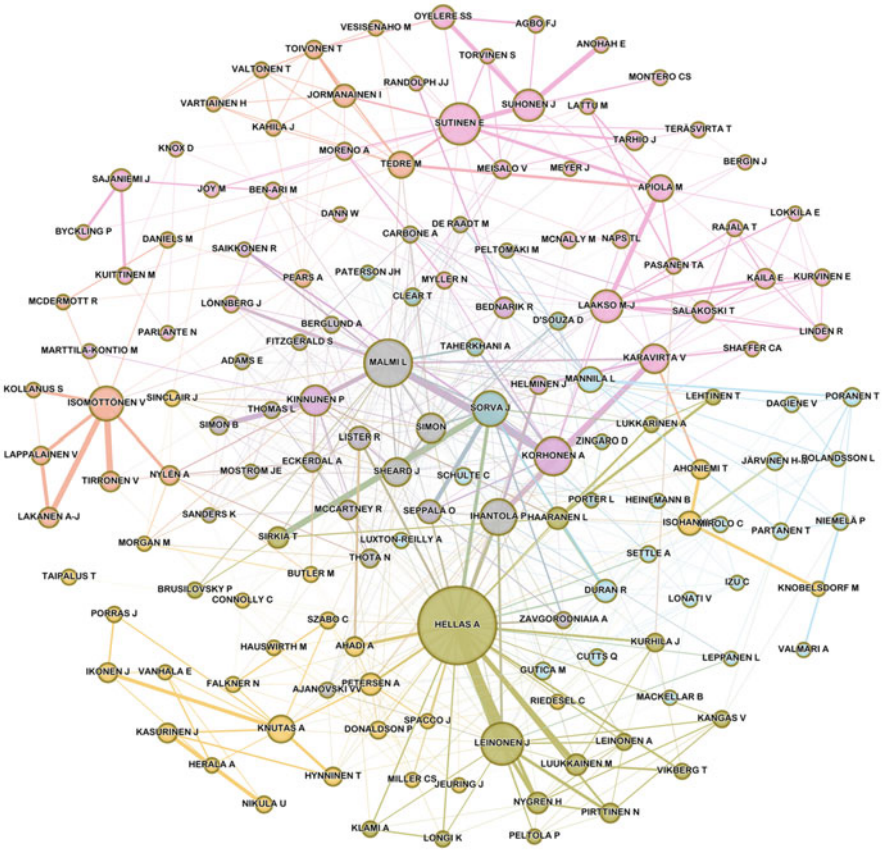

Fig. 3 Collaboration network of authors with Finnish affiliation

in light orange, there is University of Jyväskylä team with Isomöttönen as the core person.

There is a large number of foreign authors in the network, which partially confuses the picture, but at the same time demonstrates the international collaboration network among Finnish authors. It is also natural that the communities evolve, as some people change their affiliation. For example, Sutinen worked a long time in UEF and then continued his career in UTU; Ihantola has worked at Aalto University, University of Tampere and finally at University of Helsinki; Kinnunen has moved from Aalto University to University of Helsinki, and Hellas has moved from University of Helsinki to Aalto.

When considering the most popular publication venues among authors with Finnish affiliation, the two clear top venues are Koli Calling and ITiCSE, followed by ICER and SIGCSE. Table 1 presents the ten most popular venues. In total, papers

Table 1 Most popular publication venues among Finnish authors (in Scopus)

Venue	Papers
Koli Calling International Conference on Computing Education Research	158
Innovation and Technology in Computer Science Education, ITiCSE	124
International Conference on Computing Education Research, ICER	36
ACM Technical Symposium on Computer Science Education, SIGCSE	32
Computer Science Education	20
ACM Transactions On Computing Education	16
Frontiers in Education Conference, FIE	14
ACM SIGCSE Bulletin	11
Australasian Computing Education Conference, ACE	7
International Conference on Computer Supported Education, CSEDU	7

had been published in 90 different conferences and journals, including 59 venues with only a single paper.

Despite the fact that Koli Calling is always organized in Finland, it is a very international conference with participants and submissions coming from numerous countries globally. We discuss its history and character more below.

3 Koli Calling Conference

One of the landmarks in Finnish CER was launching the Koli Calling conference in 2001, with professors Erkki Sutinen and Tapio Salakoski being the initiators. Lauri Malmi soon joined the team. They were conference chairs for the first 5 years.

Koli is a high hill in Eastern Finland within a national park with a wonderful view to Lake Pielinen. The initial name of the conference in Finnish was Kolin Kolistelut, where the latter word means a rattling noise. The selected name indicated that the purpose was to shake existing practices of teaching computing and to invent something new. Indeed, for the first 3 years, the conference was a swap meeting for Finnish computer science teachers and only few foreign people attended it. Even the language of discussion changed between Finnish and English depending on whether foreign people were present in the session or not.

In 2004, the program committee agreed that the conference should take a different profile seeking to solicit research papers internationally. The program committee was extended with more international scholars, and the call for papers revised to solicit papers on two tracks: Research papers and Discussion papers. The latter were shorter and targeted to present novel educational innovations for the conference audience. In the following years, the call for papers was further elaborated to better respect the richness of work carried out in the field. Thus, new submission types were added, including system papers for describing novel educational software tools and theoretical papers for theoretical discourse. Moreover, in some years, a separate

Call for Tools was published with the idea that the submission should also include relevant software which could be evaluated, too, and not just the paper describing it. All these activities reflect the nature of the conference as a versatile venue for presenting research and discussing new developments. Over the years, a large share of Koli Calling participants have been PhD students who have presented their early work first as posters, demonstrations or discussion papers and later on presented solid research papers at Koli.

One of the basic characteristics of Koli Calling has been its location, in the middle of a national park. Staying in an isolated hotel and the small size of the conference (around 50 participants) has created excellent opportunities for networking. The conference begins with Thursday evening dinner, followed by two full days of presentations and discussions, with typically the closing session on Sunday morning. On Saturday afternoon, there is a break and the Koli Walk for visiting the national park (there is often snow on ground which is quite spectacular for many foreign visitors). In the evenings, there is an opportunity for attending the Koli sauna session or visiting Koli Spa. All these activities provide ample opportunities for meeting colleagues informally. Moreover, as nobody leaves for visiting elsewhere for restaurants and sightseeing, it is practically possible to discuss with everyone during the conference.

Koli has gained a reputation of one of the leading conferences in CER, among SIGCSE, ITiCSE, ICER, and ACE. While the share of Finnish participants has naturally always been large, the majority of participants are international, especially from Europe. There are, however, frequently many participants from Australasia, and increasingly also from the US. During the pandemic in 2020 and 2021, the conference was organized only virtually, which extended its size to 100 participants, many from the US.

For more information about the conference and a scientometric analysis of its publications, see [7].

4 CER in Finnish Universities

In this section, we present the development and main focus areas in the major research groups in Finnish universities.

4.1 Aalto University

The roots of CER at Aalto University originate from the educational development activities in 1980s and 1990s at Helsinki University of Technology, TKK.[3] The basic programming courses targeted to the whole university were very large ranging from

[3] Aalto University was launched in 2010 as a merger of Helsinki University of Technology, Helsinki School of Economics and University of Art and Design.

a few hundred to over thousand students. The courses had lectures, weekly exercises, programming projects and an exam. The weekly exercises were not graded; their model solutions were presented in large exercise groups in lecture halls, where teaching assistants also gave some guidance. One or two programming projects per course were submitted for manual grading. Exams were on paper only. In addition to such traditional teaching methods, online guidance for projects was widely used already in the 1980s, implemented with course-specific Unix newsgroups, where students could ask questions.

Lauri Malmi started to work as a lecturer in 1986, and soon became interested in improving pedagogical approaches to teaching programming. There was a burning problem: how to manage four large courses annually with roughly 2000 enrolled students with one lecturer and only a small number of BSc level teaching assistants working a few hours a week to give guidance and grade projects. While new learning resources and new pedagogical approaches were developed, grading and giving personal feedback on weekly exercises turned out to be infeasible with these human resources. Unfortunately, recruiting more teaching personnel was not an option either. Hence, the solution was to build and use software to support education.

The first educational technology project was launched as a capstone project in 1990. A student team implemented the tool called TRAKLA that automated the assessment of *algorithm simulation exercises* [39, 68] on a data structures and algorithms course. Students received the assignments and submitted their solutions in a predefined text format by email, which TRAKLA server checked. In these partially compulsory exercises, students presented in high level of abstraction how a given algorithm and a set of operations change a given data structure.

Launching the tool in spring 1991 reduced course grading workload hugely. Moreover, the final exam results also improved. Encouraged with this, a paper presenting the system was submitted to HyperMedia in Vaasa conference in 1993 [39]. In this conference, Malmi met Edmund Burke and learned about the Ceilidh tool for automatic assessment of programming submissions [16] that had been developed at University of Nottingham, UK. Based on Malmi's recommendation, the tool was adopted in the basic programming course at TKK. Ceilidh was used the first time in 1994 and made a huge change in the course. Now, it was possible to set up weekly compulsory exercises which were graded automatically, and teaching assistants' work could then be directed much more into giving guidance, instead of grading. Moreover, students could resubmit their solutions after getting feedback.

Finnish Ministry of Education launched in the mid 1990s a program to support the quality of university education. National Centers of Excellence in Education were selected every third year based on competitive applications. A team of highly devoted junior teachers and researchers who had convened regularly to discuss how CS education could be improved managed to prepare successful applications to these calls. The department's basic education section gained the national level status for 2001-2003, and this was revised for the second period 2004–2006, and after a compulsory hiatus for the next round the whole department received the status again for 2010–2012. This provided substantial funding for the team. Moreover, the

funding from the Ministry was not typical project funding tied to the project plan goals, but it was more like an award to freely improve education further.

Computer Science Education Research Group, COMPSER, was formed in 2000. In addition, Malmi was promoted to associate professor in 2001, which increased the academic independence of the group. Ari Korhonen was his first PhD student, who had already started developing TRAKLA further in his MSc thesis a few years earlier. His PhD research, completed in 2003, focused on development and evaluation of *visual algorithm simulation exercises* in TRAKLA2 [68]. Päivi Kinnunen started her PhD studies first by investigating problem-based learning in programming education and thereafter CS1 students' dropout problem [60, 61]. Several talented MSc students, who have later on gained substantial visibility in the CER field, Juha Sorva, Otto Seppälä, Petri Ihantola, and Ville Karavirta joined the team in early 2000s. They had been working earlier as teaching assistants on programming courses or summer trainees and started to work with various new software projects, and soon were involved in writing papers already when studying for their Master's degree or doing their MSc thesis. They all continued for doctoral studies after completing their master's thesis, resulting in many doctoral theses a few years later.

The same model of recruiting talented students early in master's level studies or at the latest when starting the MSc thesis project has continued and turned out to be a very successful practice, resulting in a large number of doctoral theses. The main research areas in the theses have focused on program and algorithm visualization [53, 90, 108, 125, 128], automatic assessment [33, 40, 124, 135] and games and gamification [9, 26, 29].

As part of the research, multiple software tools were developed, including several versions of the TRAKLA concept [39, 69, 96, 109], teacher's algorithm simulation tool MatrixPro [55], visualization tool for concurrent programs, Atropos [91], program simulation tool UUhistle [129], Parsons problem framework jsParsons [41], ACOS content server [127], JSVee and Kelmu visualization tools [126], and Rubyric manual grading support tool [8]. Most of them have been used for several years in large programming courses, which has enabled collecting and analyzing lots of data of their impact on students' learning results and studying process, as well as their understanding of programming concepts. Naturally, many of these software are now outdated due to being implemented in dated technologies, or as a natural result of course development when they are not needed any more. On the other hand, some tools have persisted in use. TRAKLA2 exercises have been re-implemented with Javascript library jSav [56], and the A+-learning environment [54] has been in continuous use at the department since 2013 and is now used in dozens of courses. It has also been adopted at University of Tampere. Rubyric is still being used, after 10 years, to support manual grading of project reports and submissions.

COMPSER changed its name to the Learning + Technology research group, LeTech, as some research activities extended to more general education technologies, and engineering education research. Current research themes in the group cover teaching/learning event-driven programming, automatic generation of questions from students' programs, students' misconceptions on algorithms,

motivational factors in affecting learning programming, interactive tutoring for debugging, learning analytics, as well as automatic assessment in mathematics and tools for supporting learning to write academic English.

COMPSER/LeTech members have been very active in national collaboration. Several Koli Calling program chairs have had their background in COMPSER/LeTech. Malmi and Korhonen have also coordinated important national networking projects, which are discussed more below.

International collaboration started early after being inspired by participating in ITiCSE conferences in Uppsala 1997 and Helsinki 2000. Malmi and Korhonen participated in ITiCSE working groups focusing on evaluation of the impact on algorithm visualization in 2002 and 2003, and thus built valuable contacts with international researchers working in this area. Much of this continued also in active participation of Program Visualization Workshops, a series of biannual small international workshops organized in 2000-tale, initiated by prof. Sutinen at University of Joensuu in 2000. Very many of COMPSER/LeTech members have participated in ITiCSE working groups thereafter, which has supported their own international networking. Often the working group reports were finally included among their doctoral thesis publications.

4.2 University of Helsinki

The department of Computer Science at the University of Helsinki (UH) has a long tradition in developing and utilizing educational technologies and practices to improve teaching, as well as in evaluating tools and practices developed by others. This tradition has been mostly grassroot level activity, driven by individual teachers and professors. Teachers at UH—in addition to developing tools and teaching—often studied the effect of these tools and teaching on students' learning. Results of these studies and experiments have typically been shared as reports and presentations at department-level teaching days or at university-level events. Teaching has also been valued, evidenced both through teaching-related annual awards both from the department and university levels, as well as through funding based on gaining the status of a center of excellence in education from the Ministry of Education in 2001–2003.

CER has been acknowledged at the department at least since the early 1990s. However, despite the fact that the fifth ITiCSE conference was organized at UH in 2000, presenting the work at CER venues was relatively rare at the beginning, when contrasted with the amount of work that took place at the department. Such work was often published elsewhere. For example, Eliot and Jeliot systems that piloted pedagogical algorithm animations were published at conferences related to Computer Graphics and Visualization [80] and Visual Languages [25]. The similar observation holds e.g. for intelligent tutoring systems, intelligent learning materials, and systems with social navigation support [72, 73]. The emergence of the Koli

Calling conference provided a home for some of the work in the early 2000s [58, 59] and only later on CER papers were published more in classic CER venues.

The first doctoral dissertation in CER from the Department of Computer Science (2003) was "Considering Individual Differences in Computer-Supported Special and Elementary Education" [71] that focused on how interactive learning environments can adapt to special needs. It has been followed by a handful of theses focusing on aspects such as supporting creativity in teaching programming [6], pedagogies and tools for teaching programming [32], and data analytics from programming environments [85]. Beyond the theses, CER researchers have studied approaches to teaching programming [74, 148] and good software engineering practices [70, 94]. As a part of this work, researchers have also looked into approaches to increase student engagement and support peer learning [150] as well as on building automated assessment approaches that provide stepwise help to students learning programming [149]. While the previous examples focus mostly on bachelor's level education, researchers have also looked into supporting students' working on capstone projects and beyond [23, 95].

This work has also led to building open online courses in programming [75], which in turn has led to studies on using open online courses as a way to recruit students into computer science studies [87]. Furthering this work, the Department of Computer Science has also created a MOOC-platform[4] that currently has millions of users from across the world. Moreover, developing the tools and platforms for supporting online learning has enabled automatic data collection and data-driven approaches to investigate the learning processes [149].

In particular, CER researchers have looked into approaches to identifying at-risk students and their challenges [2, 88], understanding students' help-seeking strategies [103], studying how learners use online materials and whether adjustments to contents such as images or progress visualizations helps learners [27, 42], understanding how code is written and who is writing it [86, 89], understanding characteristics of students [21, 116], and more broadly modeling students' learning [37]; this, in a sense, links back to the intelligent tutoring system-related studies conducted at the department in the late 1990s and early 2000s [72, 73].

The good track record in research driven development of learning software was recognized also at the university level when the University of Helsinki MOOC center was established late 2020 to carry out research around online learning and to extend the technology and related best practices built around computing education to other disciplines. The new center was positioned in the CS department and the head of the unit, professor Petri Ihantola, was selected from the faculty of Educational Sciences, where computing education is also developed. As an example, the faculty hosts the Innokas Network[5] with focus on K-12 education and teacher education.

[4] https://www.mooc.fi.

[5] https://www.innokas.fi/en.

4.3 University of Jyväskylä

University of Jyväskylä (JYU) has a long history in developing tools to support teaching and learning programming, especially by lecturer Vesa Lappalainen; however, little of this work has been published. JYU has also over a two-decade history of educating computer science subject teachers. Both these traditions contributed to a situation that a door was open for CER. Professor Kärkkäinen, who was in charge of subject teacher education, supervised two education-related dissertations [35, 45], of which Isomöttönen's thesis [45] was in CER. During this time, CER-related dialog started to grow in the faculty.

The background of the group can also be said to be based on accidents. Ville Isomöttönen who now leads the group started his PhD work with a computer science music topic. However, he changed the topic into project-based learning in software engineering after receiving an acceptance on a project-course themed conference paper. The change resulted from a collaboration invitation by a colleague Sami Kollanus to work within CER which was his side topic. The key point of this turn was the first publications in Koli Calling, ITiCSE, and CSEE&T conferences in 2008.

Over the years, the pioneers persuaded others to attend, which has led to research in multiple topics and completing several dissertations. The CER group has studied functional programming education that emphasizes students' self-direction [48, 140], interest development in programming education for K-12 outreach activities [82], as well as multidisciplinarity and students' view of industry collaboration [31]. More recently, a dissertation was completed on the topic of SQL education [136].

Examples of recent research themes on programming education include motivation, identity, creativity, and interest development during programming courses, as well as exams as a learning experience. Project-based learning is studied from the perspectives of reflective learning [112], justice [47], and status processes. Research on database education has continued [137], whereas a more general theme has addressed study difficulties and related interventions among CS students [36]. Many other themes (e.g., developing theoretical frameworks to explain the challenges of teaching a particular area, flexible delivery, infographics for reflective learning, and multi-purpose educational technology) have been recently addressed when attempting to introduce new persons to the group or to initialize shared research topics. The group has slightly emphasized qualitative approaches in research.

On the side of the research, programming-related course teachers have developed and taken into full use several novel software products, e.g., a unit testing tool that can be effortlessly integrated into introductory programming materials [83], an automatic assessment tool of ICT skills [81], and a tool for learning Haskell. The current prominent example is the TIM (The interactive Material) teaching and learning platform, which integrates a high number of functionalities—all that teachers need—into a single learning management system [46, 141]. The system

has served also in wider contexts, for example, to support national level university entrance examinations.

JYU group has built international collaboration with UpCERG group in Uppsala University, Sweden, resulting in multiple joint research articles (e.g., [46]) and visits. A major starting point for this collaboration was discussions (e.g., between Anders Berglund and Isomöttönen) during Koli Calling conferences after which collaborations at personal levels ensued. After a couple of less active Covid years, this collaboration was recently revitalized for more project-based learning studies in which critical incident technique (CIT) provides a framework for exploring reflective learning. Additionally, the group is currently collaborating with Eindhoven and Leiden universities on database education.

The CER group at JYU is currently an acknowledged research group of IT faculty, while not yet in the position of main research divisions. Thus, JYU has not initiated a professorship in CER. Doctoral theses in the group are now supervised based on docentships of the senior researchers in the group. Finally, it is worth noting that educational technology and subject teacher education lines also conduct important educational research in the faculty. However, they are geared towards other publication forums outside CER.

4.4 University of Joensuu/University of Eastern Finland

The first research in the CER field at the Department of Computer Science, University of Joensuu[6] can be traced back to the end of the 1990s. Prof. Martti Penttonen supervised the doctoral dissertation of Marja Kopponen, titled "CAI in CS", in 1997, the first CER dissertation from the department [67]. Dr. Kopponen continued to publish work in CER together with Prof. Jorma Sajaniemi mainly regarding computer-aided lecturing technologies [120]. At the beginning of the 2000s, Sajaniemi's research group focused on cognitive science aspects of computing education, especially on the *roles of variables* and program animation in computing education [17, 24, 110, 121, 122]. The group also worked on eye tracking research [104], which later on extended beyond computing education to the medical field, mainly by research and development work of Associate Professor Roman Bednarik.

Research in CER was expanded when Prof. Erkki Sutinen joined the department in the late 1990s, and he was responsible for coordinating the computer science teacher education studies. He also formed the edTech research group and started several new CER initiatives. The first of these was ViSCoS (Virtual Studies of Computer Science) online studies, which offered university-level computing studies to high school students in the North-Karelia region in Finland [28, 133]. The design

[6] Later on the Department of Computer Science and Statistics at the University of Joensuu (2006-2010) and the School of Computing at the University of Eastern Finland (2010-current).

aspects of ViSCoS studies were the focus of the first doctoral dissertation in the edTech research group, by Jarkko Suhonen [132].

The second initiative was Kids' Club—a technology-rich after-school club environment. The club environment accelerated the group's research, specifically on educational robotics and programming education. The Kids' Club also participated in international robotics competitions, especially RoboCupJunior, with good success. Club activities sparked ideas for in-service teacher training with educational robotics and other state-of-art technologies, and gained support from two externally funded projects (2003–2007, European Social Fund) focusing on the development of technology and computing education at schools. The projects, RoboCupJunior activities in Finland, and in-service teacher training were some of the building blocks for a Finnish network of school teachers, later known as the Innokas network. This network also influenced on the Finnish curriculum reform for primary and secondary schools by defining what "Computing at Schools" could be in Finland. The club environment formed a basis for two doctoral dissertations [50, 151]. The third CER research initiative was contextualized computing education [57, 146]. The edTech group had many years of intensive collaboration with the Tumaini University, Tanzania to implement a locally relevant bachelor's study program in Information Technology [147]. Moreover, several individual doctoral students' research topics have been connected to contextual computing education, for example, [1, 66, 101, 113].

The fourth significant line of CER research was related to the Jeliot program visualization tool, which was originally developed at the University of Helsinki. New features and related research focused especially on collaborative visualization and conflictive program animation [14, 99, 102]. The development of Jeliot continued till the end of 2000s, until it started fading after a more than a decade of work [15]. However, thereafter it has still been a part of individual doctoral students' research work [30].

The size and impact of the edTech research group increased considerably after the mid-2000s when IMPDET[7] online doctoral studies, a joint initiative between the edTech research group and education researchers [134] was launched and gained an important role in the group's activities. While the research topics of IMPDET students have been diverse and mainly related to educational technology and ICT for development, there have also been doctoral dissertation topics connected to CER, for example, in improving assessment processes of information system studies [10].

A significant change in the edTech research group happened when Prof. Sutinen left to the University of Turku in 2015, which forced the group to renew its operations. The senior researchers, Jarkko Suhonen, Ilkka Jormanainen, and Calkin Suero Montero started to supervise doctoral students and apply for external funding independently. Prof. Markku Tukiainen took over Sutinen's existing research projects and initiatives, including IMPDET studies with almost 50 enrolled doctoral students. Prof. Matti Tedre and Dr. Mohammed Saqr joined the research group

[7] https://www.impdet.org.

in 2017 from Stockholm University, which strengthened the group's activities considerably.

The changes in the core personnel brought new research focus areas, such as maker pedagogy [142], learning analytics [93], VR/AR in computing education [1], computational thinking education [51, 84] and machine learning/artificial intelligence education [100, 138, 139]. The new research topics also intertwined with earlier research, such as exploring teachers' preconceptions of teaching machine learning in the African context [123].[8] Finally, collaboration with education researchers inside the University of Eastern Finland was re-established, especially related to machine learning/artificial intelligence education and computational thinking education [144].

The edTech research group has not been focusing purely on CER topics, but the group's research interests have been very diverse, including educational technology, ICT for development, natural language processing, business informatics, and text analysis methods. New research topics have emerged, for example, when new doctoral students and faculty members joined the group's activities. Specifically, Prof. Sutinen expanded the group's research work into new areas, instead of focusing on one or even few narrowly specified research topics. The group's openness to accept a wide range of topics also seemed to attract new people with varying backgrounds to join the group. Moreover, the wide spectrum of research interests enabled acquiring funding from various sources.

The two groups, Prof. Sajaniemi's group, and edTech have had quite different methodological profiles. The former employed mainly empirical-quantitative research approaches, while the edTech group has been using a diverse mix of research approaches, quantitative, qualitative, action research, design science research and many others. Besides pure research interests, the work at edTech has also been motivated by creating completely new study opportunities for computing education for different target groups (examples: ViSCoS, contextual bachelor's degree studies in Tanzania, Kids' Club and its spin-offs, maker movement pedagogy and robotics [130, 131]).

The diversity of research topics and approaches also have some drawbacks. Research topics of CER doctoral students have been sometimes too separated, which has led to inefficient use of available resources, and in some cases, the research efforts have not deepened beyond "proof-of-concept" type of research. Moreover, collaborative work between the two research groups could have been stronger, especially on program visualization. However, the CER research at the University of Eastern Finland is currently very active and, for example, new doctoral students with CER interests are joining the group constantly.

[8] The name of the group was also changed from edTech to Technologies for Learning and Development.

4.5 Tampere University

Tampere University was created in 2019 when University of Tampere and Tampere University of Technology were merged. As their groups have a long independent history, we present them separately.

4.5.1 University of Tampere

The first Scandinavian computer science professorship was established in 1965 at University of Tampere [114]. This was also the start of university level computer science education in Finland. Professor Reino Kurki-Suonio was nominated to this position, and in 1980, he moved to the Tampere University of Technology. Although programming was already part of the first curriculum, it took four decades before computational thinking related research started. The seeds were sown in 1990s when activities related to the International Olympiad in Informatics (IOI) began.

Informatics Olympiad is one of the international science olympiads, such as International Olympiad in Mathematics, Chemistry and Physics. In IOI, high school students solve programming tasks that require exceptional algorithmic thinking skills. The Finnish team participated in the IOI the first time in 1992 [117] and in 1998, Finland also started participating in Baltic Olympiad in Informatics. During the first 3 years, leading the team was the responsibility of the University of Helsinki, after which it was circulated to the University of Tampere for the next 4 years. Since then, the team lead has been circulated between Universities of Helsinki, Turku, and Tampere. The University of Tampere organized the IOI contest in 2001; there were 272 contestants from 74 countries [111]. Finland's contest success has been relatively good when considering its population. If all medals (gold, silver and bronze) are counted, Finland is currently ranked 28th among all participating countries [43].

During IOI'2008 contest journey, professor Valentina Dagiené from University of Vilnius, Lithuania, proposed that Finland could also organize the Bebras challenge. Bebras is an international initiative aiming to promote Informatics and computational thinking among school students at all ages [13]. University of Tampere started to develop its own contest system, and the first national contest was organized in 2010 with 1472 participants from primary and secondary schools. The contest system has been used also in Sweden and Slovenia. The number of participants has increased since the beginning, and in 2021 it was about 4900. Finland has collaborated actively with Sweden, and this has helped Finland to organize the contest in both official languages, Finnish and Swedish. Currently Finland is using France's contest system, and there are plans to use the ViLLE system developed at University of Turku in 2022 (see below). Bebras challenge produces data on how pupils are solving tasks requiring computational thinking skill, and this data has been used in research with Lithuania and Sweden [19, 20].

Bebras contest has brought many contacts to primary and secondary school teachers, and this has yielded projects resulting in research on programming learning resources and MOOCs to primary and secondary school teachers [107, 115].

Finally, as many other universities, also University of Tampere developed its own learning management system, WEb Teaching Organizer (WETO) in the early 2000s, to help to organise mass courses with peer reviews and automatic assessment [106].

4.5.2 Tampere University of Technology

In Tampere University of Technology (TUT), computer science education was initially in the 1970s given under electrical engineering. The first professorship in Computing Systems was established in 1980. Professor Reino Kurki-Suonio from University of Tampere was appointed to this position which he held until his retirement in 2002. The degree program in information technology started in 1985 and finally information technology got its own department in the university in 1993.

Research in CER started gradually from the establishment of the computer science department. The first master's thesis in this area was completed in 1998 [4]. The research and development group for Programming Education, EDGE, led by professor Hannu-Matti Järvinen was established soon after the first thesis and by 2003 an EU project was running in the team.

In addition to learning programming, the main focus area at TUT has been new learning tools and how to best utilize them in computing education. The use of automatic grading, grading feedback, program visualization and peer review have been among the research topics, which has resulted in three doctoral dissertations [3, 5, 44] in addition to one in computational thinking [105]. Notably, Lahtinen et al.'s paper, "A Study Of The Difficulties Of Novice Programmers" [79] has the highest citation count in CER in Finland.

In addition to its own learning technology development, TUT has had active collaboration with Helsinki University of Technology/Aalto University. For example, the rubric-based evaluation tool Aloha, initially developed at Tampere, was further developed at Aalto under the name Rubyric. This collaboration has carried on to the present day, when Tampere University uses and co-develops the learning management system A+ [54] initially developed at Aalto.

When the two universities in Tampere were merged, a challenge emerged: how to harmonize the tool development and usage in the new Tampere University, when both partner universities had their own tools. Luckily at the same time, a national network project, The Intelligent Systems and Content Creation project, was initiated, which helped to resolve these issues (see Sect. 5 for more information).

In 2020, professor Hannu-Matti Järvinen established a new education research group which unites researchers in mathematics education and computing education. Its current main research themes include flipped learning, computational thinking and learning tools.

4.6 University of Turku and Åbo Akademi University

4.6.1 The Dawn of CER at the University of Turku

At the University of Turku (UTU), interest and efforts in developing CS teaching were substantially increased when Jorma Boberg and Tapio Salakoski joined the CS department in the mid 1980s. In 1993, Open University CS education was started, calling for a new, multimodal and partially virtual approach taking advantage of modern educational technologies. Very soon it was discovered that contemporary digital pedagogy could not handle increasing numbers of students, and pedagogical research did not focus on scaling up teaching. Challenges dealing with students' difficulties in learning computational thinking and programming posed the first CER questions. The focus remained, however, in developing one's own teaching.

In the late 1990s, Salakoski started as a fixed-term CS professor in bioinformatics. Nevertheless, he maintained his interest also in CER and learned about similar development efforts by Erkki Sutinen at University of Joensuu and by Lauri Malmi at Helsinki University of Technology. The founding of Koli Calling Conference in 2001 by the three professors marked a shift from mere professional development of CS teaching to more serious CER.

This development was accompanied by Salakoski's initiative of setting up CS teacher education at UTU. One of the very first CS Open University students, Mikko-Jussi Laakso, was recruited as a teacher for a new CS course in digital educational technology. A course project work by students Erkki Kaila and Teemu Rajala supervised by Laakso and Salakoski resulted in the first version of ViLLE programming visualization environment in 2004 [119]. The same group later developed a new version of a new more comprehensive ViLLE collaborative education tool in 2010, aimed at supporting teachers facing growing numbers of students with digital tools [78]. The group began to study the impact of technological interventions such as automated assessment and immediate feedback in research settings. In addition to ViLLE, they used the newly developed TRAKLA2 system in collaboration with Malmi and Korhonen at TKK.

At the UTU Faculty of Education, professor Erno Lehtinen was a pioneer in educational technology and technology education. His interest in learning mathematics and computing led him to collaboration with Salakoski already in the 1990s. The collaboration started with jointly supervised MSc theses and has lasted ever since. Even today, they have a joint major 6-year research project Growing Mind funded by the Strategic Research Council at the Academy of Finland.

The first PhD in CER at UTU graduated in 2010, when Laakso received his degree under Salakoski's supervision [77]. Their continued collaboration has resulted in several other PhDs: [52, 76, 145]. The focus of the Salakoski-Laakso group has been the use of automated assessment and immediate personalized feedback in supporting the learning of programming, mathematics, and computational thinking. The ViLLE system has also expanded to a general learning platform used in teaching many subjects, and automated assessment has grown to more

comprehensive learning analytics. Methodology-wise, the role of data and machine learning in analytics has increased. In addition to providing for teachers longitudinal analysis of individual students' learning results, these methods support developing tools for knowledge management and business intelligence for decision makers.

While the majority of Salakoski's scientific work has been in bioinformatics and natural language processing according to the field of his professorship, he has also continued CER work. Another CER-related professorship in interaction design was established in 2016, when Erkki Sutinen moved to Turku. Instead of CER, his main work in Turku has now focused on interactive game design and digital humanities, especially digital theology.

4.6.2 The Centre for Learning Analytics

In 2019, Mikko-Jussi Laakso began as an associate professor in learning analytics as the first explicit professorship in CER. He started as the leader of the newly established Centre for Learning Analytics at the Department of Computing. The main efforts were directed towards building a data-based ecosystem for learning and teaching, especially focusing on diagnostics of learning difficulties of mathematical and computational thinking in secondary education, supporting teaching interventions, large scale e-assessment, and knowledge management.

As a result, the ViLLE system has been widely adopted in lower and upper secondary education in Finland, the current penetration being 60% of the schools on a national level. In addition, international collaboration has increased substantially in Europe, USA, Middle East, and Asia. Annually, 500 million ViLLE exercises are being submitted and assessed. The Centre and the ViLLE system have been recognized on several occasions, also globally; they recently received the UNESCO King Hamad Bin Isa Al-Khalifa Prize for the Use of ICT in Education 2021 [143].

The Centre was given the status of an independent unit and moved to the Faculty of Science in 2022. The decision was motivated by the national and international learning loss in mathematical science subjects, along with the decreasing trend in interest of the youth towards university level studies in science and science teacher education. Also the COVID-19 pandemic catalyzed digitalization at all levels of education. The work at the Centre has been funded by the Ministry of Education, Academy of Finland, several foundations and municipalities, as well as the EU. Currently, the Centre has about 30 employees. In 2018, a spin-off company Eduten Ltd was established for the valorization and internationalization of ViLLE technology. Eduten has operations in more than 50 countries. In 2022, it received the UNICEF EdTech Award as the winner of the global Extreme Tech Challenge 2022 competition for EdTech startups.

4.6.3 CER at Åbo Akademi University

Åbo Akademi University is another university in Turku giving education in Swedish, the other official language in Finland. There the central people for CER were Ralph-Johan Back, a professor of software engineering (now retired), and Linda Mannila née Grandell, who received her PhD focusing teaching mathematics and programming in 2009 [97]. Working in adjacent premises in Turku, Back and Salakoski created a joint research group with Mannila and Mia Peltomäki, a senior mathematics teacher, CS teaching pioneer, and a PhD student at UTU.

Their group focused on the logical thinking behind both mathematical derivations and programming. The objective was to make mathematical inference visible and explicit; a strict logical formalism enabled applying automated theorem proving for automated verification and immediate individual feedback on students' work in high-school mathematics. They used refinement calculus, a software verification formalism, for describing mathematical inference as structured derivations [11, 12]. They also studied invariant based programming (PhD Johannes Eriksson 2010 [22]) and the difficulties in learning the first programming language [98].

The flagship project of the group was EU-funded E-Math, a collaborative effort in high-school mathematics education with the cities of Turku, Stockholm, and Tallinn in 2007-2013. As a result of the project, a spin-off company Four Ferries Ltd was established, offering interactive math textbooks and other learning material for upper secondary education.

4.7 Lappeenranta University of Technology

Lappeenranta University of Technology (LUT) does not have a specific education research center. Rather, several persons from the faculty contribute to the CER community as a part of their teaching development activities or PhD studies on adjacent topics.

There have been several distinct research themes. LUT participated in some of the pioneering Finnish work on hackathons (later on named Code Camps) since 2008 [118]. These events were instrumental in building industry cooperation around the technology cluster at the campus region. They have added internship opportunities and supported learning practical skills about recent technologies, as well as created opportunities for research collaborations. For example, as an offshoot from the hackathons emerged research into collaborative learning: How could students effectively collaborate in programming teams? To support these efforts, research efforts focused on teamwork analysis [63], content delivery methods [34] and designs (e.g. gamification) to support effective teamwork [65].

A more recent line of research has focused on text mining and analysis of student feedback. This has been divided into two lines of research: Processes and tools for text mining [38], and its impact on, for example, curriculum design [62].

The final line of research from LUT has been software engineering and computer science education networks [64]. LUT has coordinated the Pathways to PhDs in Software Engineering project that mapped and coordinated software engineering education at PhD level. While bachelor's and master's education has been standardized by the Bologna Process in EU, PhD level education has considerable variance.

5 National Level Collaborative Activities Related to Computing Education

The Virtual University of Finland (VUF) was a collaborative network of Finnish universities, which started its operations in 2001. Despite its name, it was not an actual university, but an umbrella organization. It was a collection of university discipline networks that build multidisciplinary nationwide networks of activities. The aim was to promote the use of information and communication technologies and to develop cooperation among universities in various fields. In 2001–2006, the activities were funded by the Ministry of Education. Starting in 2007 the universities were supposed to be responsible for the funding, which did not realize well and basically led to closing VUF at the end of 2010. However, several projects which started during these years still remain active and get funding from several sources including the Ministry of Education.

The Basic Programming Education Network (BPEN) was one of the virtual university networks (2006–2008) in the field of mathematics and science. The purpose of the network was to promote the dissemination of specialized tools and materials used in basic programming courses in Finnish universities and to promote the networking of teachers in this area. The aim was to establish a high-quality and economical approach to teaching and research that relies heavily on the use of information and communication technologies, which also utilizes the latest research data in the field.

The network worked in close co-operation with the Computer Science Teaching SIG (CSTSIG), a thematic group within the Finnish Society for Computer Science. The aim of the theme group is to bring together teachers in the field and researchers interested in learning technology. The purpose of the joint network was to promote, e.g., collaboration between teachers, exchanging teaching and learning materials, introducing ICT-based tools to support teaching and learning computing and taking a stand on current societal issues in the field of teaching computing. The concrete form of activities included courses, network meetings, and seminars, which were regularly attended by about 40–50 teachers, researchers or other people who were interested in the network's activities. The seminars included workshops presenting a variety of ICT-based solutions that had been developed in Finnish universities to support teaching and learning computing. At the time, it was typical to bundle the tools on a USB stick, but share and update the content online. In addition, the network organized other opportunities to meet, for example, doctoral students

whose dissertation topic focused on learning technology or teaching computer science.

BPEN also had good international relations which enabled bringing together users and tool developers from many other countries. Many scholars visited Finland during the years and gave courses, for example, on automatic assessment tools, software visualization, and how to teach programming in general. BPEN was a three year project that ended in 2008; however, the teachers and researchers from many universities continued networking, which also led to close collaboration in CER and supported many doctoral research projects.

ÄlyOppi project (The Intelligent Systems and Content Creation project, 2018–2021) had roots which are heavily tied in the aforementioned networks. It was funded by the Ministry of Education and its goals were to develop new and improve existing online learning materials and environments for university level use in computer science, mathematics and physics. In computer science, which was the largest subproject, the aim was to develop tools for automatic assessment, visualizations and simulation, as well as improve existing tailored (for computer science education) learning environments in universities, and support their integration with each other. Thus, tools developed at different institutes could be used elsewhere and interactive learning resources, based on the tools, could be used in wider settings. The project significantly strengthened the network of CER people in Finland. It also organized a series of webinars presenting the results and accomplishments of the project for a wider audience in tertiary education.

In recent years, the Ministry of Education has strongly supported extending opportunities for life-long learning. A major networked project FITech is a flagship in computing education, which seeks to open widely university level computing courses for people in working life who wish to upgrade their skills in computing.[9] Similarly, a smaller project Digital Education for All, focused on opening first and second year computing courses and learning resources as MOOCs to people who would like to learn computing, but have not enrolled to universities. Moreover, sufficiently good performance in these open courses would allow them to continue their studies in formal computing degree programs in universities without the need to pass an entrance examination. While these projects are not research oriented as such, they allow collecting much data and experiences which can be investigated in CER.

6 Discussion

CER is flourishing in Finland with several research groups working actively in the area, as presented above. An interesting question emerges: what may be the background behind this phenomenon.

[9] https://fitech.io/en/.

6.1 Pioneering Teachers

The roots of many CER groups emerge from solving challenges in university level education. A common problem in many institutes has been the shortage of teaching resources when compared with the number of students enrolling in introductory programming courses. This challenge has often been addressed by the extensive use of BSc level students as teaching assistants or tutors (while graduate students as teaching assistants may seem a more proper solution, they are more often recruited as teaching assistants for advanced courses). Another solution has been the development of tools to help teaching, learning and assessment, e.g., ViLLE in Turku, Jeliot in UEF and TRAKLA, Ceilidh and A+ in Aalto/TKK. These software projects have been driven by enthusiastic pioneers, typically junior teachers in charge of the courses. However, from early on, many groups have built much activity in school level education, too, such as the Kids CLub in UEF, IOI and Bebras contests in Tampere, and building the ViLLE collaborative learning environment in Turku. Interestingly, these K-12 level activities are not follow-ups of developments of the national level school curricula, because computational thinking and basics of programming have been included in school curriculum only from 2016 onwards. Previously, programming education has been organized in schools only as voluntary optional subjects.

However, much of this work can be considered more of computing education development than research. A highly important factor supporting the turn towards research have been the pioneering professors, who have had a strong personal interest in developing education and building their own research area associated with it, regardless of whether their primary research area or teaching responsibility has been some other area in computing or education. While faculty members generally have more or less academic freedom to choose their research foci, professors have stronger shoulders to push their own agenda forward despite their formal research areas. For example, Salakoski in Turku has worked mainly in bioinformatics but at the same time conceived, encouraged and supported work in CER. In Tampere, Järvinen, while working in software technology research, was also very interested in supporting education and thus provided the support for junior researchers and teachers to carry out research in CER. In Jyväskylä, Kärkkäinen was associated with teacher education, and supporting CER has been an easy extension. In the same way, in UEF, Sutinen was assigned the responsibility of teacher education and soon extended his work into a wide variation of themes in computing education, educational technologies, as well as ICT4D. At TKK/Aalto, Malmi was responsible for basic programming education and found there a fruitful symbiosis of carrying out pedagogical development and researching the impact of the implemented educational innovations. Thus, he could avoid the potential tension of carrying out research in some other computer science topics while having a heavy teaching load elsewhere. At University of Helsinki, the challenge was different. While there was a lot of educational development activity, there was no professor with a similar strong interest in the area before Ihantola got a tenure track professorship in Learning

Analytics in 2018. Before that, the solution was a grass-root level approach, led by Jaakko Kurhila and later on Matti Luukkainen and Arto Hellas, where a research group was formed to legitimize the activities of both tenured teachers and aspiring researchers.

Considering the most productive authors in Figs. 2 and 3, the pioneers and their PhD graduates are well presented. Among the list of people are two pioneering professors, Sutinen and Malmi and several of their early PhD graduates: Suhonen was Sutinen's student and Ihantola, Korhonen and Sorva are Malmi's students. Isomöttönen was the pioneering lecturer at Jyväskylä and Laakso was the first PhD graduate in Turku.

All this builds the big picture that individual people who are highly interested in improving education form the core of the success in Finland. None of them were initially nominated as professors with a research area CER, but they have used their academic freedom to target their main work in the field, or give active support for junior teachers or PhD students who wish to work in CER. The academic freedom and strong impact of pioneering individuals is also likely a reason for the richness of activities and research topics in Finnish groups. Each of the teams has clearly a unique profile. At TKK/Aalto, there is a long tradition of developing tools to support programming and data structures and algorithms courses, as well as researching programming education more widely. At University of Turku, ViLLE system that was initially developed to support programming education, has later on been extended to full scale learning environment supporting multiple disciplines and forms the core data collection tool for the Centre of Learning analytics. At UEF, Sajaniemi's research focused on cognitive aspects of learning programming. In addition, research in the edTech group in Joensuu around the Jeliot programming visualization tool became a long-term research track. However, soon the scope of activities in edTech widened very much, covering activities for broadening participation, distance education, robotics, contextualized CS education for developing countries and many more. Lappeenranta has a strong focus on software engineering education, while Tampere has focused on programming education and supporting computational thinking in K-12 level contests. At University of Helsinki, there has been much work in learning analytics and software engineering education, and Jyväskylä has had more emphasis in qualitative research in multiple topics.

6.2 Networking and Recruitment

One of the strengths in Finland is actually the small size of the country, which enables easy networking between people who are interested in CER. From early on, the Koli Calling conference became a venue where most of the active people in the field convened annually. Moreover, as the conference fee was kept low to support international visitors, this—combined with low domestic travel costs—allowed teams to send PhD students to the conference with a poster only or simply as visitors. Many seniors also attend the conference regularly regardless of whether

they have a paper presentation there or not, just to meet people. This has built a strong network in Finland where people know each other. Moreover, the small size of the conference and its format has strongly supported PhD students to familiarize with international colleagues, too. Further support has been provided by the major national level education development projects which were discussed above in Sect. 5. There is a win-win situation. As people know each other, it is easy to build even national level consortia which can apply such network funding when appropriate funding calls are available. On the other hand, the network activities bring new people into the field.

Two conferences have had a major role in international networking in the early years. The ITiCSE conference was organized in Uppsala, Sweden in 1997 and in Helsinki in 2000. For the pioneers, these events gave a good spark for international collaboration. Moreover, over many years, successful experiences from participating ITiCSE working groups have further extended international networks, not only for seniors but for many PhD students. All these factors have contributed to the significant share of publications with authors from Finland and other countries since the early 2000s, as can be seen in Fig. 1.

Another strength in many groups has been the relative ease of recruiting students in an early phase in their MSc or even BSc level studies. Many of them have been working as teaching assistants on basic computing courses, because they have been interested in helping their peers. They are therefore well aware of challenges that younger students have, e.g., in learning programming and can generate new ideas for improvement when they are already working together with the course teacher. Many of these students are also interested in developing software in an academic environment where they can work to solve real problems and there is more freedom to choose their goals compared with working in internships in companies. Moreover, getting their name as an author in a publication which concerns the tool they have been implementing, is certainly motivating for those who have an interest in research. Often their master's thesis project may result not just a new publication but also a stepping stone for their PhD research. From the team's point of view, recruiting students early is also relevant because many CS students aim at an industrial career. Recruiting new students even for a MSc thesis project may be difficult, as an industry MSc thesis project is often the gate for a working position. The situation is different if a student has already been integrated in a research group, and considers the academic career, or at least doctoral studies, as an interesting option.

Another opportunity to recruit new people to CER is persuading teacher colleagues, who have done great work in developing their courses, to collaborate and write joint papers. However, this has not been an easy task. From their perspective, the most relevant research topic is naturally their own course and teaching, and other topics might be less attractive. Thus, the effort in attracting them to CER for a longer term may be difficult.

6.3 Challenges in Funding Research

Despite these positive aspects, there is no lack of challenges. The work in CER is often not valued as relevant or high quality research among people working in "real" computer science. Some people consider that CER should be carried out by educational scientists at departments of education instead of computer science departments. The only way to mitigate this challenge has been to build academic credibility in the field in terms of writing good quality papers, completing PhD theses which are assessed with the normal academic procedures, as well as organizing and participating in international conferences, i.e., working as any other academic disciplines do.

However, the above tension has made getting research funding for CER projects a major challenge in Finland, as well as in most other countries. In Finland, the key academic research funding institution is Academy of Finland. If a CER proposal is submitted to its council of technology and natural sciences, many proposals have been evaluated with low grades, because their research goals are not considered relevant or interesting enough for computing sciences. On the other hand, if a proposal has been submitted to the council of social sciences, educational scientists may consider that the team is not competent enough for such research. Naturally collaborative proposals where CER people and researchers from social sciences work together as a consortium, have somewhat better chances. However, the competition for funding is generally very high and therefore many CER funding applications have been targeted to other venues. One option has been EU funding, where many different funding instruments exist. The challenge there is building a good consortium which is large and versatile enough to match the general criteria of EU projects. Institutes from several countries around Europe should be included, but CER is not an active research area in many EU countries. Moreover, managing the project application and the actual project, if funded, is often very complex and laborious.

At TKK/Aalto, a central factor in the beginning was gaining the Center of Excellence on Education (CEE) status which provided significant long term funding for the department, of which a considerable share could be used to start work with new educational innovations. This long term funding enabled creating scientific results, which could be used to support future applications. Unfortunately, the Ministry of Education decided to cease the CEE program some 10 years later, and only the University of Helsinki team managed to get this kind of funding, in addition to TKK/Aalto.

When gaining major research funding has been difficult, other sources have been explored. Many foundations fund research, but typically for a short time only from a few man months to a man year. Some departments or universities provide their own funded PhD student positions, in most cases based on competitive applications. Some of those resources have been successfully applied to CER PhD students. The Ministry of Education and some NGOs have provided funding for developing education in universities, which have been used for networking activities, as written

above, but also for work which strongly supports research, such as implementing new educational software. One important NGO funding organization has been the Technology Industries of Finland Centennial Foundation. Finland is a high tech country and there is a constant shortage of competent people, especially in the IT sector. The foundation has supported projects in which education is developed in various fields of technology, and some major projects have been successfully applied to support work in CER. While such funding is basically funding for development of education or educational software, it can greatly support research. Developing novel software requires much work, but when the new tools are ready and being used in real courses, it is relatively straightforward to design studies which can lead to good publications.

Similar types of funding opportunities have occasionally been available from the Ministry of Education, too. For example, the above mentioned networking projects received such funding, and the latter, ÄlyOppi project also focused mainly on developing software and new learning resources based on the available tools. It is likely that such funding either from the Ministry or from universities themselves will increase, as life-long learning is a growing area for universities and it needs advanced technologies to support it. Covid-19 pandemic caused a major step towards wider availability of blended and online learning and this development will continue. People in working life cannot attend campus teaching for longer times, and they must be supported with online resources and facilities, a challenge addressed in the above mentioned FITech project. From the perspective of CER, such projects provide rich opportunities for extending research from programming education to education of advanced CS topics.

Problems in gaining funding have likely had an effect on the diversity of topics which have been researched. As major funding is rarely available, PhD students are funded from smaller projects which may not match well for collaborating with their peers. This has led in several cases, e.g. in UEF and Aalto to a situation where research is too much individualized, and PhD students work less as a team than would be beneficial for all.

7 Future

The future of CER in Finland seems quite promising. Current group leaders in CER have been either professors or lecturers, depending on the university. Some of the professors work mainly in CER themselves while others have their main focus elsewhere despite their interest in CER. For full-time lecturers especially in large computing degree programmes, CER work offers an attractive option to combine their educational duties with academic qualification. It is particularly delightful that recently several young people with a solid background in CER have been appointed as professors (Matti Tedre at UEF, Mikko-Jussi Laakso at UTU, Petri Ihantola at University of Helsinki, and Antti Knutas at LUT). Even though the main research area of their office may not be specifically CER but some close by area, such as

learning analytics, they can target much of their energy in CER, bringing continuity to the field. It remains to be seen whether professorships with CER as their main research area will be founded in the future.

Moreover, the actors in the field are very well networked. Collaboration is easy, if some new funding is possible to gain. This network has also greatly supported the Koli Calling conference. Interestingly, the conference does not have any formal organization behind. The university of Eastern-Finland CER team has organized it well for 20 years, and each year some Finnish CER researcher is the other program chair while the other one is an international chair. This brings much continuity with fairly low effort. But it is also a strong evidence of Finnish "talkoohenki", which denotes joint free work for a common goal.

References

1. Friday Joseph Agbo. *Co-designing a smart learning environment to facilitate computational thinking education in the Nigerian context*. PhD thesis, University of Eastern Finland, 2022.
2. Alireza Ahadi, Raymond Lister, Heikki Haapala, and Arto Vihavainen. Exploring machine learning methods to automatically identify students in need of assistance. In *Proceedings of the Eleventh Annual International Conference on International Computing Education Research*, pages 121–130, 2015.
3. Tuukka Ahoniemi. *Efficient use of teaching technologies with programming education*. PhD thesis, Tampere University of Technology, 2015.
4. Kirsti Ala-Mutka. Tietokoneavusteinen ohjelmoinnin opetus. Master's thesis, Tampere University of Technology, 1998. In Finnish.
5. Kirsti Ala-Mutka. *Automatic assessment tools in learning and teaching programming*. PhD thesis, Tampere University of Technology, 2005.
6. Mikko Apiola. *Creativity-supporting learning environments: Two case studies on teaching programming*. PhD thesis, University of Helsinki, 2013.
7. Mikko Apiola, Sonsoles Lopez-Pernas, Mohammed Saqr, Arnold Pears, Mats Daniels, Lauri Malmi, and Matti Tedre. From a national meeting to an international conference: A scientometric case study of a Finnish computing education conference. *IEEE Access*, 2022.
8. Tapio Auvinen. Rubyric. In *Proceedings of the 11th Koli Calling International Conference on Computing Education Research*, pages 102–106, 2011.
9. Tapio Auvinen. *Educational technologies for supporting self-regulated learning in online learning environments*. PhD thesis, Aalto University, 2015.
10. Rosalina Babo. *Improving individual and collaborative e-assessment through multiple-choice questions and WebAVALIA - A new assessment strategy implemented at a Portuguese university*. PhD thesis, University of Eastern Finland, 2020.
11. Ralph-Johan Back, Linda Mannila, Mia Peltomäki, and Tapio Salakoski. Improving mathematics and programming education – the IMPEd initiative. *Proceedings of the Seventh Baltic Sea Conference on Computing Education Research, Australian Computer Society*, 88:161–170, 2007.
12. Ralph-Johan Back and Joakim Wright. *Refinement Calculus - A Systematic Introduction*. Springer, 1998.
13. Bebras international challenge on informatics and computational thinking. https://www.bebras.org/. Accessed: 2022-04-07.
14. Roman Bednarik, Andėrs Moreno, and Niko Myller. Program visualization for programming education - case of Jeliot3. *Association for Computing Machinery New Zealand Bulletin*, 2(2), 2006.

15. Moti Ben-Ari, Roman Bednarik, Ronit Ben-Bassat Levy, Gil Ebel, Andèrs Moreno, Niko Myller, and Erkki Sutinen. A decade of research and development on program animation: The Jeliot experience. *Journal of Visual Languages and Computing*, 22:375–384, 2011.
16. Steve Benford, Edmund Burke, Eric Foxley, Neil Gutteridge, and Abdullah Mohd Zin. Early experiences of computer-aided assessment and administration when teaching computer programming. *ALT-J*, 1(2):55–70, 1993.
17. Pauli Byckling and Jorma Sajaniemi. Roles of variables and programming skills improvement. In *Proceedings of the 37th SIGCSE Technical Symposium on Computer Science Education*, pages 413–417, 2006.
18. European Commission. Survey of schools: ICT in education. benchmarking access, use and attitudes to technology in Europe's schools, 2013.
19. Valentina Dagiene, Linda Mannila, Timo Poranen, Lennart Rolandsson, and Pär Söderhjelm. Students' performance on programming-related tasks in an informatics contest in Finland, Sweden and Lithuania. In *Proceedings of the 2014 Conference on Innovation & Technology in Computer Science Education*, pages 153–158, 2014.
20. Valentina Dagiene, Linda Mannila, Timo Poranen, Lennart Rolandsson, and Gabriele Stupuriene. Reasoning on children's cognitive skills in an informatics contest: Findings and discoveries from Finland, Lithuania, and Sweden. In *Proceedings of the International Conference on Informatics in Schools: Situation, Evolution, and Perspectives*, pages 66–77, 2014.
21. Rodrigo Duran, Lassi Haaranen, and Arto Hellas. Gender differences in introductory programming: Comparing MOOCs and local courses. In *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*, pages 692–698, 2020.
22. Johannes Eriksson. Tool-supported invariant-based programming. *TUCS Dissertations 127. Turku Centre for Computer Science*, 2010.
23. Fabian Fagerholm, Arto Hellas, Matti Luukkainen, Kati Kyllönen, Sezin Yaman, and Hanna Mäenpää. Designing and implementing an environment for software start-up education: Patterns and anti-patterns. *Journal of Systems and Software*, 146:1–13, 2018.
24. Petri Gerdt and Jorma Sajaniemi. A web-based service for the automatic detection of roles of variables. *ACM SIGCSE Bulletin*, 38(3):178–182, 2006.
25. Jyrki Haajanen, Mikael Pesonius, Erkki Sutinen, Jorma Tarhio, Tommi Teräsvirta, and Pekka Vanninen. Animation of user algorithms on the web. In *Proceedings. 1997 IEEE Symposium on Visual Languages (Cat. No. 97TB100180)*, pages 356–363. IEEE, 1997.
26. Lassi Haaranen. *Game-related learning and exposure in computer science*. PhD thesis, Aalto University, 2019.
27. Lassi Haaranen, Petri Ihantola, Juha Sorva, and Arto Vihavainen. In search of the emotional design effect in programming. In *2015 IEEE/ACM 37th IEEE International Conference on Software Engineering*, volume 2, pages 428–434. IEEE, 2015.
28. Arto Haataja, Jarkko Suhonen, Erkki Sutinen, and Sirpa Torvinen. High school students learning computer science over the web. *Interactive Multimedia Electronic Journal of Computer-Enhanced Learning*, 3(2), 2001.
29. Lasse Hakulinen. *Gameful approaches for computer science education: From gamification to alternate reality games*. PhD thesis, Aalto University, 2015.
30. Mustafa Muhammad Hassan, Andèrs Moreno, Erkki Sutinen, and Abdul Azil. On the participatory design of Jeliot Mobile: Towards a socio-constructivist mlearning tool. In *Proceedings of the International Conference on Learning and Teaching in Computing and Engineering*, pages 120–123, 2003.
31. Juho Heikkinen. Conceptualizing the role of multidisciplinarity and student perceptions of university-industry collaboration in project-based learning. In *Jyväskylä studies in computing*, volume 264. University of Jyväskylä, 2016.
32. Arto Hellas. *Retention in introductory programming*. PhD thesis, University of Helsinki, 2017.
33. Juha Helminen. *Supporting acquisition of programming skills in introductory programming education: Environments for practicing programming and recording and analysis of exercise sessions*. PhD thesis, Aalto University, 2014.

34. Antti Herala, Erno Vanhala, Antti Knutas, and Jouni Ikonen. Teaching programming with flipped classroom method: A study from two programming courses. In *Proceedings of the 15th Koli Calling Conference on Computing Education Research*, pages 165–166, 2015.
35. Leena Hiltunen. Enhancing web course design using action research. In *Jyväskylä Studies in Computing*, volume 125. University of Jyväskylä, 2010.
36. Ville Hämäläinen and Ville Isomöttönen. What did CS students recognize as study difficulties? In *2019 IEEE Frontiers in Education Conference (FIE)*, pages 1–9. IEEE, 2019.
37. Roya Hosseini, Peter Brusilovsky, Michael Yudelson, and Arto Hellas. Stereotype modeling for problem-solving performance predictions in MOOCs and traditional courses. In *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization*, pages 76–84, 2017.
38. Maija Hujala, Antti Knutas, Timo Hynninen, and Heli Arminen. Improving the quality of teaching by utilising written student feedback: A streamlined process. *Computers & Education*, 157:103965, 2020.
39. Juha Hyvönen and Lauri Malmi. TRAKLA - a system for teaching algorithms using email and a graphical editor. In *HYPERMEDIA in Vaasa, 1993*, pages 141–147. University of Vaasa, Finland, 1993.
40. Petri Ihantola. *Automated assessment of programming assignments: visual feedback, assignment mobility, and assessment of students' testing skills*. PhD thesis, Aalto University, 2011.
41. Petri Ihantola and Ville Karavirta. Open source widget for parson's puzzles. In *Proceedings of the Fifteenth Annual conference on Innovation and Technology in Computer Science Education*, pages 302–302, 2010.
42. Kalle Ilves, Juho Leinonen, and Arto Hellas. Supporting self-regulated learning with visualizations in online learning environments. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*, pages 257–262, 2018.
43. International Olympiad in Informatics, Statistics. https://stats.ioinformatics.org/countries/. Accessed: 2022-04-07.
44. Essi Isohanni. *Visualizations in learning programming: Building a theory of student engagement*. PhD thesis, Tampere University of Technology, 2013.
45. Ville Isomöttönen. Theorizing a one-semester real customer student software project course. In *Jyväskylä Studies in Computing*, volume 140. University of Jyväskylä, 2011.
46. Ville Isomöttönen, Antti-Jussi Lakanen, and Vesa Lappalainen. Less is more! Preliminary evaluation of multi-functional document-based online learning environment. In *2019 IEEE Frontiers in Education Conference (FIE)*, pages 1–5. IEEE, 2019.
47. Ville Isomöttönen and Emmi Ritvos. Digging into group establishment: Intervention design and evaluation. *Journal of Systems and Software*, 178:110974, 2021.
48. Ville Isomöttönen and Ville Tirronen. Flipping and blending — An action research project on improving functional programming course. *ACM Transactions on Computing Education Research*, 17(1):1:1–1:35, 2017.
49. Pertti Järvinen. Notes on educational planning: a systems approach. *ACM SIGCSE Bulletin*, 9(4):57–62, 1977.
50. Ilkka Jormanainen. *Supporting teachers in unpredictable robotics learning environments*. PhD thesis, University of Eastern Finland, 2013.
51. Ilkka Jormanainen and Markku Tukiainen. Attractive educational robotics motivates younger students to learn programming and computational thinking. In *Proceedings of the Eighth International Conference Technological Ecosystem for Enhancing Multiculturality Conference*, 2020.
52. Erkki Kaila. Utilizing educational technology in computer science and programming courses: theory and practice. *TUCS Dissertations 230. Turku Centre for Computer Science*, 2018.
53. Ville Karavirta. *Facilitating algorithm visualization creation and adoption in education*. PhD thesis, Helsinki University of Technology, 2009.
54. Ville Karavirta, Petri Ihantola, and Teemu Koskinen. Service-oriented approach to improve interoperability of e-learning systems. In *2013 IEEE 13th International Conference on Advanced Learning Technologies*, pages 341–345. IEEE, 2013.

55. Ville Karavirta, Ari Korhonen, Lauri Malmi, and Kimmo Stålnacke. Matrixpro - A tool for ex tempore demonstration of data structures and algorithms. In *Proceedings of the Third Program Visualization Workshop, University of Warwick, UK*, pages 27–33, 2004.
56. Ville Karavirta and Clifford A Shaffer. Creating engaging online learning material with the JSAV javascript algorithm visualization library. *IEEE Transactions on Learning Technologies*, 9(2):171–183, 2015.
57. Jyri Kemppainen. *Appropriating IT service management education in a Tanzanian university: Global and local perspectives*. PhD thesis, University of Eastern Finland, 2014.
58. Teemu Kerola and Harri Laine. SQL-trainer. *Kolin Kolistelut–Koli Calling Proceedings of the First Annual Finnish/Baltic Sea Conference on Computer Science Education*, 2001.
59. Teemu Kerola and Harri Laine. Creation of self tests and exam questions as a learning method. *Kolin Kolistelut—Koli Calling. Proceedings of the Fourth Finnish/Baltic Sea Conference on Computer Science Education*, 2004.
60. Päivi Kinnunen. *Challenges of teaching and studying programming at a university of technology-Viewpoints of students, teachers and the university*. PhD thesis, Helsinki University of Technology, 2009.
61. Päivi Kinnunen and Lauri Malmi. Do students work efficiently in a group? - Problem-based learning groups in basic programming course. In *Kolin Kolistelut - Koli Calling Proceedings of the Fourth Finnish/Baltic Sea Conference of Computer Science Education*, pages 57–66. Citeseer, 2004.
62. Antti Knutas, Timo Hynninen, and Maija Hujala. To get good student ratings should you only teach programming courses? Investigation and implications of student evaluations of teaching in a software engineering context. In *2021 IEEE/ACM 43rd International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET)*, pages 253–260. IEEE, 2021.
63. Antti Knutas, Jouni Ikonen, and Jari Porras. Communication patterns in collaborative software engineering courses: A case for computer-supported collaboration. In *Proceedings of the 13th Koli Calling International Conference on Computing Education Research*, pages 169–177, 2013.
64. Antti Knutas, Ahmed Seffah, Lene Sorensen, Andrey Sozykin, Fawaz Al-Zaghoul, and Alain Abran. Crossing the borders and the cultural gaps for educating PhDs in software engineering. In *2017 IEEE 30th Conference on Software Engineering Education and Training (CSEE&T)*, pages 256–265. IEEE, 2017.
65. Antti Knutas, Rob Van Roy, Timo Hynninen, Marco Granato, Jussi Kasurinen, and Jouni Ikonen. A process for designing algorithm-based personalized gamification. *Multimedia Tools and Applications*, 78(10):13593–13612, 2019.
66. Sandhya Kode. *Enhancing Information Technology education in Indian context: a design story*. PhD thesis, University of Eastern Finland, 2019.
67. Marja Kopponen. *CAI in CS*. PhD thesis, University of Joensuu, 1997.
68. Ari Korhonen. *Visual algorithm simulation*. PhD thesis, Helsinki University of Technology, 2003.
69. Ari Korhonen and Lauri Malmi. Algorithm simulation with automatic assessment. In *Proceedings of the 5th annual SIGCSE/SIGCUE ITiCSE Conference on Innovation and Technology in Computer Science Education*, pages 160–163, 2000.
70. Jami Kousa, Petri Ihantola, Arto Hellas, and Matti Luukkainen. Teaching container-based devops practices. In *International Conference on Web Engineering*, pages 494–502. Springer, 2020.
71. Jaakko Kurhila. *Considering individual differences in computer-supported special and elementary education*. PhD thesis, University of Helsinki, 2003.
72. Jaakko Kurhila and Erkki Sutinen. Sharing an open learning space by individualizing agents. *Journal of Interactive Learning Research*, 10(3):287, 1999.
73. Jaakko Kurhila and Erkki Sutinen. From intelligent tutoring systems to intelligent learning materials. In *EdMedia+ Innovate Learning*, pages 546–551. Association for the Advancement of Computing in Education (AACE), 2000.

74. Jaakko Kurhila and Arto Vihavainen. Management, structures and tools to scale up personal advising in large programming courses. In *Proceedings of the 2011 Conference on Information Technology Education*, pages 3–8, 2011.
75. Jaakko Kurhila and Arto Vihavainen. A purposeful MOOC to alleviate insufficient CS education in Finnish schools. *ACM Transactions on Computing Education (TOCE)*, 15(2):1–18, 2015.
76. Einari Kurvinen. Effects of regular use of scalable, technology enhanced solution for primary mathematics education. *TUCS Dissertations 260. Turku Centre for Computer Science*, 2020.
77. Mikko-Jussi Laakso. Promoting programming learning. Engagement, automatic assessment with immediate feedback in visualizations. *TUCS Dissertations 131. Turku Centre for Computer Science*, 2010.
78. Mikko-Jussi Laakso, Erkki Kaila, and Teemu Rajala. ViLLE – collaborative education tool: Designing and utilizing an exercise-based learning environment. *Education and Information Technologies*, 23:1655—1676, 2018.
79. Essi Lahtinen, Kirsti Ala-Mutka, and Hannu-Matti Järvinen. A study of the difficulties of novice programmers. *ACM SIGCSE Bulletin*, 37(3):14–18, 2005.
80. SP Lahtinen, T Lamminjoki, E Sutinen, J Tarhio, and AP Tuovinen. Towards automated animation of algorithms. In *Proceedings of Fourth International Conference in Central Europe on Computer Graphics and Visualization*, volume 96, pages 150–161, 1996.
81. Tommi Lahtonen and Ville Isomöttönen. Parsi: A tool for automatic assessment of office documents and basic IT skills. In *Proceedings of the 12th Koli Calling International Conference on Computing Education Research*, pages 174–180, New York, NY, 2012. ACM.
82. Antti-Jussi Lakanen. On the impact of computer science outreach events on K-12 students. In *Jyväskylä studies in computing*, volume 236. University of Jyväskylä, 2016.
83. Vesa Lappalainen, Jonne Itkonen, Ville Isomöttönen, and Sami Kollanus. Comtest: A tool to impart TDD and unit testing to introductory level programming. In *ITiCSE '10: Proceedings of the Fifteenth Annual Conference on Innovation and Technology in Computer Science Education*, pages 63–67, New York, NY, 2010. ACM.
84. Jari Laru, Kati Mäkitalo, Matti Tedre, Teemu Valtonen, and Henriikka Vartiainen. Ohjelmoinnista digitaalisen ajatteluun – kuinka edistää ohjelmoinnin ja tietotekniikan opetusta esi- ja alkuopetuksessa. In *Esi- ja alkuopetuksen käsikirja*, pages 243–268. PS-Kustannus, 2020.
85. Juho Leinonen. *Keystroke data in programming courses*. PhD thesis, University of Helsinki, 2019.
86. Juho Leinonen, Petri Ihantola, and Arto Hellas. Preventing keystroke based identification in open data sets. In *Proceedings of the Fourth (2017) ACM Conference on Learning@Scale*, pages 101–109, 2017.
87. Juho Leinonen, Petri Ihantola, Antti Leinonen, Henrik Nygren, Jaakko Kurhila, Matti Luukkainen, and Arto Hellas. Admitting students through an open online course in programming: A multi-year analysis of study success. In *Proceedings of the 2019 ACM Conference on International Computing Education Research*, pages 279–287, 2019.
88. Juho Leinonen, Krista Longi, Arto Klami, and Arto Vihavainen. Automatic inference of programming performance and experience from typing patterns. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education*, pages 132–137, 2016.
89. Krista Longi, Juho Leinonen, Henrik Nygren, Joni Salmi, Arto Klami, and Arto Vihavainen. Identification of programmers from typing patterns. In *Proceedings of the 15th Koli Calling Conference on Computing Education Research*, pages 60–67, 2015.
90. Jan Lönnberg. *Understanding and debugging concurrent programs through visualisation*. PhD thesis, Aalto University, 2012.
91. Jan Lönnberg, Mordechai Ben-Ari, and Lauri Malmi. Java replay for dependence-based debugging. In *Proceedings of the Workshop on Parallel and Distributed Systems: Testing, Analysis, and Debugging*, pages 15–25, 2011.

92. Sonsoles López-Pernas, Mohammed Saqr, and Mikko Apiola. Scientometrics: A concise introduction and a detailed methodology for the mapping of the scientific field of computing education. In Mikko Apiola, S López-Pernas, and Mohammed Saqr, editors, *Past, Present and Future of Computing Education Research*, pages xx–yy. Springer, 2023.
93. Sonsoles Lòpez-Pernas, Muhammed Saqr, and Olga Vberg. Putting it all together: Combining learning analytics methods and data sources to understand students' approaches to learning programming. *Sustainability*, 13(9), 2021.
94. Matti Luukkainen, Arto Vihavainen, and Thomas Vikberg. A software craftsman's approach to data structures. In *Proceedings of the 43rd ACM Technical Symposium on Computer Science Education*, pages 439–444, 2012.
95. Hanna Mäenpää, Samu Varjonen, Arto Hellas, Sasu Tarkoma, and Tomi Männistö. Assessing IOT projects in university education - A framework for problem-based learning. In *2017 IEEE/ACM 39th International Conference on Software Engineering: Software Engineering Education and Training Track (ICSE-SEET)*, pages 37–46. IEEE, 2017.
96. Lauri Malmi, Ville Karavirta, Ari Korhonen, Jussi Nikander, Otto Seppälä, and Panu Silvasti. Visual algorithm simulation exercise system with automatic assessment: TRAKLA2. *Informatics in education*, 3(2):267–288, 2004.
97. Linda Mannila. Teaching mathematics and programming - new approaches with empirical evaluation. *TUCS Dissertations 124. Turku Centre for Computer Science*, 2009.
98. Linda Mannila, Mia Peltomäki, and Tapio Salakoski. What about a simple language? Analyzing the difficulties in learning to program. *Computer Science Education*, 16(3):211–228, 2006.
99. Andèrs Moreno. *Re-designing program animation*. PhD thesis, University of Eastern Finland, 2014.
100. Salsen Mrong, Ilkka Jormanainen, and Tapani Toivonen. Visualization tool for teaching and learning artificial neural networks. In *Proceedings of the 9th Technological Ecosystems for Enhancing Multiculturality Conference*, 2021.
101. Nkundwe Moses Mwasaga, Mikko Apiola, Jarkko Suhonen, and Mike Joy. Integrating high performance computing into a Tanzanian IT engineering curriculum. In *Proceedings of the 21st ICE/IEEE International Technology Management Conference*, 2015.
102. Niko Myller. *Collaborative software visualization for learning: theory and applications*. PhD thesis, University of Joensuu, 2009.
103. Matti Nelimarkka and Arto Hellas. Social help-seeking strategies in a programming MOOC. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*, pages 116–121, 2018.
104. Seppo Nevalainen and Jorma Sajaniemi. Comparison of three eye tracking devices in psychology of programming research. In *Proceedings of the 16th Annual Workshop of the Psychology of Programming Interest Group*, 2004.
105. Pia Niemelä. *From Legos and Logos to Lambda: A hypothetical learning trajectory for computational thinking*. PhD thesis, Tampere University of Technology, 2018.
106. Pia Niemelä, Aulikki Hyrskykari, Timo Poranen, Heikki Hyyrö, and Juhani Linna. Flipped learning with peer reviews in the introductory CS course. In *Assessment, Testing, and Measurement Strategies in Global Higher Education*, pages 35–58. IGI Global, 2020.
107. Pia Niemelä, Tiina Partanen, Linda Mannila, Timo Poranen, and Hannu-Matti Järvinen. Code ABC MOOC for math teachers. In P. Escudeiro, G. Costagliola, S. Zvacek, J. Uhomoibhi, and B. McLaren, editors, *Proceedings of International Conference on Computers Supported Education. CSEDU 2017. Communications in Computer and Information Science*, pages 66–96. Springer, 2017.
108. Jussi Nikander. *Interaction and visualization methods in teaching spatial algorithms and analyzing spatial data*. PhD thesis, Aalto University, 2012.
109. Jussi Nikander, Juha Helminen, and Ari Korhonen. Algorithm visualization system for teaching spatial data algorithms. *Journal of Information Technology Education*, 9, 2010.

110. Uolevi Nikula, Jorma Sajaniemi, Matti Tedre, and Stuart Wray. Python and roles of variables in introductory programming: Experiences from three educational institutions. *Journal of Information Technology Education*, 6:199–214, 2007.
111. Jyrki Nummenmaa, Erkki Mäkinen, and Isto Aho (eds.). IOI'2001 competition. Technical report A-2001-7, University of Tampere, Department of Computer and Information Sciences, 2001.
112. Aletta Nylén and Ville Isomöttönen. Exploring the critical incident technique to encourage reflection during project-based learning. In *Proceedings of the 17th Koli Calling International Conference on Computing Education Research*, pages 88–97, New York, NY, 2017. ACM.
113. Solomon Sunday Oyelere. *Design and development of a mobile learning system for computer science education in Nigerian higher education context*. PhD thesis, University of Eastern Finland, 2018.
114. Jukka Paakki. *Opista tieteeksi - Suomen tietojenkäsittelytieteiden historia*. Tietojenkäsittelytieteen Seura ry, 2014.
115. Tiina Partanen, Pia Niemelä, and Timo Poranen. Racket programming material for Finnish elementary math education. In *Proceedings of Constructionism 2018*, pages 415–425, 2018.
116. Kukka-Maaria Polso, Heta Tuominen, Arto Hellas, and Petri Ihantola. Achievement goal orientation profiles and performance in a programming MOOC. In *Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education*, pages 411–417, 2020.
117. Timo Poranen, Valentina Dagienė, Åsmund Eldhuset, Heikki Hyyrö, Kubica Marcin, Antti Laaksonen, Mārtiņš Opmanis, Wolfgang Pohl, Jūratė Skūpienė, Pär Söderhjelm, and Ahto Truu. Baltic olympiads in informatics: Challenges for training together. *Olympiads in Informatics*, 3:112–131, 2009.
118. Jari Porras, Antti Knutas, Jouni Ikonen, Ari Happonen, Jayden Khakurel, and Antti Herala. Code camps and hackathons in education - literature review and lessons learned. In *Proceedings of the Annual Hawaii International Conference on System Sciences*, 2019.
119. Teemu Rajala, Mikko-Jussi Laakso, Erkki Kaila, and Tapio Salakoski. Ville: A language-independent program visualization tool. *Proceedings of the Seventh Baltic Sea Conference on Computing Education Research, Australian Computer Society*, 88:151–159, 2007.
120. Jorma Sajaniemi and Marja Kuittinen. Three-level teaching material for computer-aided lecturing. *Computers & Education*, 32:269–284, 1999.
121. Jorma Sajaniemi and Marja Kuittinen. An experiment using roles of variables in teaching introductory programming. *Computer Science Education*, 15(1):59–82, 2005.
122. Jorma Sajaniemi, Marja Kuittinen, and Taina Tikansalo. A study of the development of students' visualization of program state during an elementery object-oriented programming course. *Journal on Educational Resources in Computing*, 7(3):1–31, 2005.
123. Ismaila Temitayo Sanusi, Solomon Sunday Oyelere, and Joseph Olamide Omidora. Exploring teachers' preconceptions of teaching machine learning in high school: A preliminary insight from Africa. *Computers and Education Open*, 3, 2021.
124. Otto Seppälä. *Advances in assessment of programming skills*. PhD thesis, Aalto University, 2012.
125. Teemu Sirkiä. *Creating, tailoring, and distributing program animations-Supporting the production process of interactive learning content*. PhD thesis, Aalto University, 2017.
126. Teemu Sirkiä. Jsvee & Kelmu: Creating and tailoring program animations for computing education. *Journal of Software: Evolution and Process*, 30(2):e1924, 2018.
127. Teemu Sirkiä and Lassi Haaranen. Improving online learning activity interoperability with ACOS server. *Software: Practice and Experience*, 47(11):1657–1676, 2017.
128. Juha Sorva. *Visual program simulation in introductory programming education*. PhD thesis, Aalto University, 2012.
129. Juha Sorva and Teemu Sirkiä. UUhistle: A software tool for visual program simulation. In *Proceedings of the 10th Koli Calling International Conference on Computing Education Research*, pages 49–54, 2010.

130. Calkin Suero Montero. Facilitating computational thinking through digital fabrication. In *Proceedings of the 18th Koli Calling International Conference on Computing Education Research*, 2018.
131. Calkin Suero Montero and Ilkka Jormanainen. Theater meets robot – toward inclusive STEAM education. In *Educational Robotics in the Makers Era (Edurobotics 2016), Advances in Intelligent Systems and Computing*, pages 34–40. Springer, 2017.
132. Jarkko Suhonen. *A formative development method for digital learning environments in sparse learning communities*. PhD thesis, University of Joensuu, 2005.
133. Jarkko Suhonen and Erkki Sutinen. Learning computer science over the web: the ViSCoS odyssey. In *Cases on Global E-learning Practices: Successes and Pitfalls*, pages 176–188. IGI Global, 2007.
134. Jarkko Suhonen and Erkki Sutinen. The four pillar model - analysing the sustainability of online doctoral programmes. *TechTrends*, 58:81–88, 2014.
135. Ahmad Taherkhani. *Automatic algorithm recognition based on programming schemas and beacons - A supervised machine learning classification approach*. PhD thesis, Aalto University, 2013.
136. Toni Taipalus. Persistent errors in query formulation. In *JYU dissertations*, volume 283. University of Jyväskylä, 2020.
137. Toni Taipalus, Hilkka Grahn, and Hadi Ghanbari. Error messages in relational database management systems: A comparison of effectiveness, usefulness, and user confidence. *Journal of Systems and Software*, 181:111034, 2021.
138. Matti Tedre, Henriikka Vartiainen, Juho Kahila, Tapani Toivonen, and Valtonen Teemu. Machine learning introduces new perspectives to data agency in K-12 computing education. In *Proceedings of the IEEE Frontiers in Education Conference*. IEEE, 2020.
139. Matti Tedre, Henriikka Vartiainen, Juho Kahila, Tapani Toivonen, Teemu Valtonen, Ilkka Jormanainen, and Arnold Pears. Teaching machine learning in K-12 classroom: Pedagogical and technological trajectories for artificial intelligence education. *IEEE Access*, 9:110558–110572, 2021.
140. Ville Tirronen and Ville Isomöttönen. On the design of effective learning materials for supporting self-directed learning of programming. In *Proceedings of the 12th Koli Calling International Conference on Computing Education Research*, pages 74–82, New York, NY, 2012. ACM.
141. Ville Tirronen, Vesa Lappalainen, Ville Isomöttönen, Antti-Jussi Lakanen, Toni Taipalus, Paavo Nieminen, and Anthony Ogbechie. Incorporating teacher-student dialogue into digital course material: Usage patterns and first experiences. In *2020 IEEE Frontiers in Education Conference (FIE)*, pages 1–5. IEEE, 2020.
142. Tapani Toivonen, Ilkka Jormanainen, Calkin Suero Montero, and Andrea Alessandrini. Innovative maker movement platform for K-12 education as a smart learning environment. In *Proceeding of 2018 International Conference on Smart Learning Environments, Challenges and Solutions in Smart Learning, Lecture Notes in Educational Technology*, pages 61–66, 2018.
143. UNESCO. UNESCO prize awarded to a collaborative learning platform ViLLE from Finland. https://en.unesco.org/news/unesco-prize-awarded-collaborative-learning-platform-ville-finland. Accessed: 2022-05-04.
144. Teemu Valtonen, Matti Tedre, Kati Mäkitalo, and Henriikka Vartiainen. Media literacy education in the age of machine learning. *Journal of Media Literacy Education*, 11(2):20–36, 2019.
145. Ashok Kumar Veerasamy. Predictive models as early warning systems for student academic performance in introductory programming. *TUCS Dissertations 259. Turku Centre for Computer Science*, 2020.
146. Mikko Vesisenaho. *Developing university-level introductory ICT education in Tanzania: a contextualized approach*. PhD thesis, University of Joensuu, 2007.
147. Mikko Vesisenaho, Jyri Kemppainen, Carolina Islas Sedano, Matti Tedre, and Erkki Sutinen. How to contextualize ICT in higher education: A case study in Tanzania. *African Journal of Information & Communication Technology*, 2(2):88–109, 2006.

148. Arto Vihavainen, Matti Paksula, and Matti Luukkainen. Extreme apprenticeship method in teaching programming for beginners. In *Proceedings of the 42nd ACM Technical Symposium on Computer Science Education*, pages 93–98, 2011.
149. Arto Vihavainen, Thomas Vikberg, Matti Luukkainen, and Martin Pärtel. Scaffolding students' learning using test my code. In *Proceedings of the 18th ACM Conference on Innovation and Technology in Computer Science Education*, pages 117–122, 2013.
150. Thomas Vikberg, Arto Vihavainen, Matti Luukkainen, and Jaakko Kurhila. Early start in software coaching. In *International Conference on Agile Software Development*, pages 16–30. Springer, 2013.
151. Marjo Virnes. *Four seasons of educational robotics: Substansive theory on the encounters between educational robotics and children in the dimension of access and ownership*. PhD thesis, University of Eastern Finland, 2014.