Analyzing Conceptual Content of International Informatics Curricula for Secondary Education

Erik Barendsen[1(✉)] and Tim Steenvoorden[2]

[1] Radboud University and Open University, Nijmegen, The Netherlands
e.barendsen@cs.ru.nl
[2] Radboud University, Nijmegen, The Netherlands
t.steenvoorden@cs.ru.nl

Abstract. Various countries are in the process of curriculum innovation with respect to informatics, which makes it interesting to conduct a systematic international comparison. As a first step, we focus on the analysis of conceptual content of curriculum specifications, that is, formal descriptions and guidelines. As a case study, we apply our method to analyze five curriculum specifications, including the former (2007) and new (2016) Dutch informatics curriculum for upper secondary education. The results indicate interesting similarities and differences with respect to emphasis of specific conceptual areas such as algorithms, software engineering and social aspects. The method appears fruitful to determine, for example, the position of the new Dutch curriculum relative to the former curriculum and to the three other recent international specifications.

Keywords: Curriculum · Concepts · Content analysis

1 Introduction

In the past few years, several organizations and individuals in Europe and the United States have expressed concerns about the state of informatics education (Académie des Sciences 2013; Furber 2012; Gander et al. 2013; Kaczmarczyk and Dopplick 2014; KNAW 2012; Samaey et al. 2014).

Although the underlying motivations vary, the common outcome of the above reports is that our society is becoming more and more digitized and therefore a broad group of people (especially children) need to learn about ICT as well as the skillful and responsible use of digital tools. Moreover, interested young people should get the opportunity to receive further education in informatics.

Various countries are in the process of curriculum innovation or have recently completed such a reform. The developments have been documented in formal curriculum documents and in guidelines.

In *England*, for example, a new subject Computing has been introduced for all students (British Department for Education 2013). The organization Computing at School developed guidelines for the new subject (Computing at School Working Group 2012). The *US* teacher organization CSTA published standards

A. Brodnik and F. Tort (Eds.): ISSEP 2016, LNCS 9973, pp. 14–27, 2016.
DOI: 10.1007/978-3-319-46747-4_2

for K–12 computer science (CSTA 2011). In *France*, an informatics curriculum has been introduced for grades 9–12 (Ministère de l'Éducation Nationale 2012). The current informatics curriculum in *The Netherlands* for grades 10–12 dates from 1998 and has not changed since then, except for a minor reformulation in 2007 (Grgurina and Tolboom 2008). In March 2016, a new curriculum proposal, commissioned by the Ministry of Education, was completed (Barendsen and Tolboom 2016).

The variety of developments make it interesting to conduct a systematic content analysis of curricula and guidelines, in order to support international comparison and curriculum development. However, it is not easy to compare the above curriculum documents, as the composition, length, and formulation of the specifications vary a lot.

In order to support the development of a new curriculum in the Netherlands, we were interested in the conceptual content (i.e., topics and ideas belonging to the informatics subject matter) of existing curricula and guidelines.

The so-called *Darmstadt Model* is a more general framework for classifying implementations of informatics education in various countries (Hubwieser et al. 2011; Hubwieser 2013). Our analysis relates to the categories *knowledge* and *intentions* within the dimension *educational relevant areas* of the Darmstadt Model. In the process of developing the framework, Hubwieser et al. (2011) perform a global categorization of the learning objectives in four countries using the ACM classification scheme and the CSTA strands as categories.

In our study, we aimed for a detailed and in-depth analysis of concepts, regardless of the skills or attitudes in which they appear (cf. Barendsen et al. 2015).

An alternative type of conceptual content analysis, based on a survey among local experts, was part of a comparison of teaching practices in Germany and the UK (Dagiene et al. 2013).

2 Aim of the Study

Part of the research was carried out during the construction of the new Dutch informatics curriculum, aiming at positioning the ideas of the curriculum committee in international perspective (Steenvoorden 2015). The starting point of the curriculum development was an international workshop in September 2014 at the *Lorentz Center* at Leiden University in the Netherlands. The curricula and documents discussed in the workshop constituted the first sample for our analysis: the former Dutch curriculum (Schmidt 2007), the French informatics curriculum (Ministère de l'Éducation Nationale 2012), the CAS guidelines (Computing at School Working Group 2012), and the CSTA standards (CSTA 2011).

The other part of the research was conducted after completion of the new curriculum (Barendsen and Tolboom 2016), to determine similarities and differences between the new curriculum and the other curricula investigated thus far.

The Dutch informatics subject only spans upper secondary education (grades 10–12). The French curriculum is intended for a similar range (grades 9–12). CAS and CSTA constructed guidelines for grades K–12. For a proper comparison, we decided to analyze the latter documents as a whole instead of their respective 10–12 segments, since it is reasonable to expect that some basic concepts (comparable to those found in the Dutch and French 10–12 curricula) appear in the K–9 part of CAS and CSTA documents.

Our research question was: *How can the conceptual content of the new Dutch curriculum, the former Dutch curriculum, the French curriculum, and the* CAS *and* CSTA *guidelines be characterized?*

3 Method

We used a variant of the method developed in Barendsen, Fisser, Krüger, and Tolboom (2014) and Steenvoorden (2015), also applied by Barendsen et al. (2015).

Our starting point was a classification of informatics subjects in terms of knowledge categories, based on the 'knowledge areas' of the Computing Curricula (2013). These knowledge areas were developed for higher education, but can be applied fruitfully in our case, since they are complete, that is, certainly cover the secondary education topics. Moreover, the knowledge area descriptions contain detailed specifications, which adds to the reliability of the analysis. The knowledge areas have been clustered into a conveniently small number of categories while maintaining sufficient detail to distinguish variations in content, see Table 1.

We applied an open coding procedure (Cohen, Manion and Morrison 2013) to the documents to extract literal concepts from the curriculum texts. In a second (more axial, cf. Cohen et al. (2013)) coding phase, similar codes were merged into one, slightly more abstract, code. Then the resulting codes were grouped into the general knowledge categories mentioned earlier.

The authors coded samples of the documents (10 %) together, while discussing and documenting the code descriptions. Then the remaining texts were coded by the second author. About half of these were reviewed by the first author. Coding differences were discussed and whenever necessary, the category descriptions were refined to reflect the consensus reached in the discussions.

For the analysis, the resulting codes were first used to get a global overview of occurrences of codes in each category. We regard the distribution of occurrences over the categories as an indication of the relative importance of the categories. Then we conducted a more qualitative, in-depth content analysis with respect to selected categories, using the (relative) frequencies and codes as pointers to relevant text segments.

Table 1. Knowledge categories

Knowledge category	Included ACM/IEEE knowledge areas
Algorithms	Algorithms and complexity (AL)
	Parallel and distributed computing (PD)
	Algorithms and design (SDF/AL)
	Remark: concepts about data structures are covered by *Data*
Architecture	Architecture and organization (AR)
	Operating systems (OS)
	System fundamentals (SF)
Modeling	Computational science (CN)
	Graphics and visualisation (GV)
Data	Information management (IM)
	Fundamental data structures (SDF/IM)
Engineering	Software engineering (SE)
	Development methods (SDF/SE)
	Remarks: also contains ideas on collaboration; concepts without an engineering component are covered by programming
Intelligence	Intelligent systems (IS)
Mathematics	Discrete structures (DS)
Networking	Networking and communication (NC)
Programming	Programming languages (PL)
	Platform based development (PBD)
	Fundamental programming concepts (SDF/PL)
Security	Information assurance and security (IAS)
	Remark: concepts about privacy are covered by society
Society	Social issues and professional practice (SP)
Usability	Human-computer interaction (HCI)

4 Results

We present our results in two ways. Firstly, in Table 2 we list the categories for each curriculum, sorted according to (absolute) number of concept occurrences. Secondly, we show the (relative) distribution of concepts across the categories for every document in Fig. 1. The new Dutch curriculum consists of a core curriculum and a number of elective themes. Below, we distinguish between the core curriculum and the curriculum as a whole (including the elective themes).

The total number of concept occurrences (i.e., coded quotations) is given at the bottom of each list in Table 2. The reason that France and the Netherlands

Table 2. Lists of knowledge categories for each curriculum document, sorted from most to least occurring concepts. The number of concept occurrences in each category is displayed between parentheses. The total number of concept occurrences in the document is given at the end of each list.

CSTA

1. Algorithms (44)
2. Engineering (40)
3. Architecture (37)
4. Society (30)
5. Networking (27)
6. Programming (25)
7. Data (23)
8. Security (13)
9. Modeling (12)
10. Intelligence (11)
11. Mathematics (8)
12. Usability (2)
13. Rest (0)

(Total: 272)

CAS

1. Algorithms (44)
2. Networking (40)
3. Architecture (38)
4. Data (33)
5. Programming (19)
6. Engineering (17)
7. Mathematics (5)
8. Security (4)
9. Society (2)
10. Intelligence (1)
11. Modeling (0)
    Rest (0)
    Usability (0)

(Total: 203)

France

1. Data (28)
2. Programming (15)
3. Architecture (14)
   Networking (14)
4. Algorithms (13)
5. Mathematics (8)
6. Society (5)
7. Engineering (4)
   Modeling (4)
8. Intelligence (2)
9. Rest (1)
10. Security (0)
    Usability (0)

(Total: 108)

Netherlands 2007

1. Architecture (13)
2. Data (12)
3. Engineering (10)
4. Networking (4)
   Rest (4)
5. Programming (3)
6. Usability (3)
7. Modeling (2)
8. Security (1)
9. Algorithms (0)
   Intelligence (0)
   Mathematics (0)
   Society (0)

(Total: 52)

Netherlands 2016 (core)

1. Programming (18)
2. Engineering (17)
3. Data (11)
4. Society (10)
5. Architecture (9)
6. Security (7)
7. Algorithms (6)
8. Usability (3)
9. Networking (2)
10. Intelligence (0)
    Mathematics (0)
    Modeling (0)
    Rest (0)

(Total: 83)

Netherlands 2016 (complete)

1. Programming (22)
2. Architecture (19)
   Society (19)
3. Data (18)
   Engineering (18)
   Usability (18)
4. Security (16)
5. Algorithms (14)
6. Networking (11)
7. Modeling (7)
8. Mathematics (4)
9. Intelligence (3)
10. Rest (0)

(Total: 169)

have less coded concepts, is that the learning goals are formulated in a relatively compact way and concepts often are mentioned only once. The CAS and the CSTA documents formulate their guidelines in a more spiral-like way, first formulating learning goals for lower grades and after that for higher grades.

Figure 1 provides a global overview of the five documents and how they compare on the twelve respective knowledge categories and a rest category.

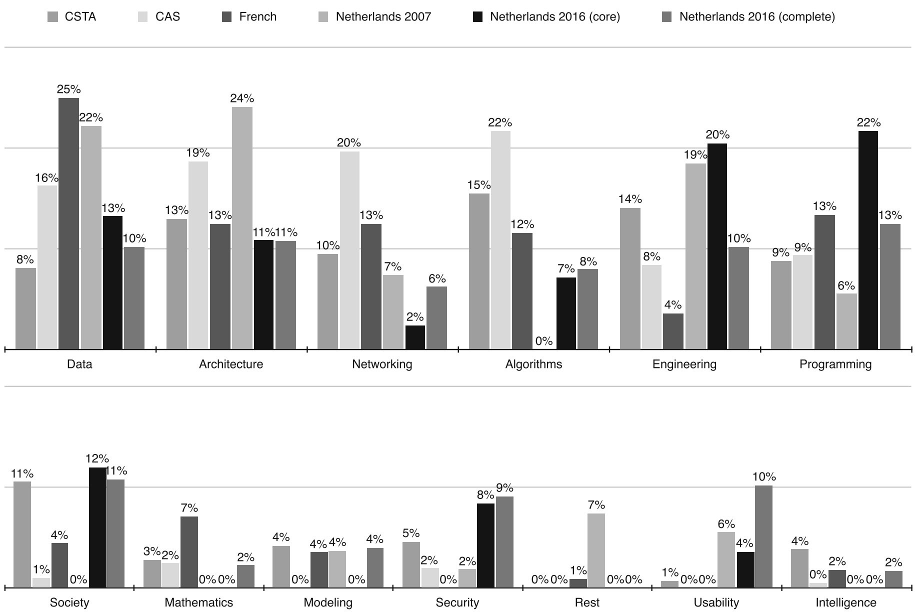

Fig. 1. Relative distribution of concept occurrences across the knowledge categories. The percentages show the fraction of the concept occurrences to the respective categories. For example, 25 % of the concept occurrences in the CSTA guidelines concerns data, while 7 % is about modeling. Categories are sorted by average occurrence.

The frequencies show that *data*, *architecture*, *networking*, *algorithms* and *engineering* cover the biggest parts of the studied specifications.

In this paper we will highlight some interesting differences. Firstly, we note the focus on *data* in the French curriculum and the gap until the next category, *programming*, as we can see in Table 2. Next, the CAS guidelines have the highest score on *algorithms*. Algorithmic concepts appear frequently in several curricula and guidelines. The old Dutch curriculum does not mention any concepts from this category, however. Another interesting observation with respect to the top five categories is the variation in scores within the *engineering* category. For this category, the French curriculum has lower scores than the other documents. Furthermore the high percentages with respect to *society* in the new Dutch curriculum and the CSTA guidelines are remarkable. Finally, the high score of the old Dutch curriculum in the rest category is exceptional.

Below, we will analyze the above observations in more depth. We illustrate our findings with characteristic quotations from the curriculum. In the case of the Dutch and French curricula, we have translated the original texts into English.

4.1 Data

The code frequencies suggest that the French curriculum has the highest emphasis on data (25 %), with *programming* appearing next in the ranking (13 %). This difference of 12 % (13 concepts) may be explained by the structure of the curriculum. Almost all (19 of 28) of the coded concepts in the Data category appear in the section 'Representation of Information'. This is the biggest section in the curriculum description, containing more than a third of the total learning objectives (8 of 21). Of the remaining concepts, 7 appear in the section on 'Languages and Programming'.

In the section on 'Representation of Information', the French curriculum includes objectives about *document formats* and *directory structure*, which furthermore appear only in the CSTA standards.

> *"Formats: Digital data is arranged to facilitate storage and processing. The structuring of digital data respects either de facto standards or norms.*
> *Skills: Identify some document formats, images and sound data. Choose an appropriate format compared to a use or need, quality or limitations."* (France)

The curriculum also mentions explicitly that students should learn about the representation of characters, text, numbers, floating points and images.

> *"Digitalization: The computer handles only numeric values. A digitalization step of physical world objects is essential.*
> *Skills: Encode a number, a character through a standard code, a text in the form of a list of numeric values. Encode an image or sound as an array of numeric values. [...]"* (France)

The CAS and CSTA curricula only mention *information representation* in general terms.

> *"Analyze the representation and trade-offs among various forms of digital information."* (CSTA, p. 18)

The old Dutch curriculum does not contain any objectives regarding *information representation*. The new Dutch curriculum however, specifies the ability to use *standard representations.*

> *"The candidate is able to use standard representations of numerical data and media, and is able to relate these to each other."* (Netherlands 2016)

In the section on 'Languages and Programming', the French curriculum explicitly states which *data types* students should master.

> *"Data types: Integer; floating point; boolean; natural number; array; string.*
> *Skills: Choosing a data type based on a problem to solve."* (France)

In contrast, the new Dutch curriculum refrains from explicitly mentioning specific data types. The same holds for the CSTA guidelines.

> *"The candidate is able to represent data in a suitable data structure, keeping the purpose in mind; the candidate is able to compare the elegance, efficiency and implementability of various representations."* (Netherlands 2016)

When going down to the bit level, the CSTA prescribes the following objective.

> *"Demonstrate how 0s and 1s can be used to represent information.* (CSTA, p. 13)

The new Dutch curriculum describes this implicitly as a *physical layer.*

> *"The candidate is able to explain the structure and functioning of digital artefacts through architectural elements, i.e., in terms of the physical, logical and application layer levels, and in terms of the components in these layers together with their interaction."* (Netherlands 2016)

The high score on data by the old Dutch curriculum can be attributed to the learning objectives on *information systems*, *databases*, *relational schemas* and *query languages.*

> *"The candidate can name the elements of a relational schema and describe the significance of each element, and can convert information needs into a command formulated in a query language for a relational database. He can describe the features and aspects of database management systems, and name and use them for specific systems [...]"* (Netherlands 2007, p. 3)

All these concepts are absent from the other four curricula. In the new Dutch curriculum, these concepts are treated in an elective theme on 'Databases'.

4.2 Algorithms

In this category, the documents differ in the amount of detail in which the learning objectives are described. We observed the CAS guidelines contains almost three times as many different concepts on algorithms as the French curriculum. The CAS guidelines, for example, explicitly states the notions of *sequence*, *selection* and *repetition.*

> *"- The idea of a program as a sequence of statements written in a programming language. - One or more mechanisms for selecting which statement sequence will be executed, based upon the value of some data item. - One or more mechanisms for repeating the execution of a sequence of statements, and using the value of some data item to control the number of times the sequence is repeated."* (CAS, p. 14)

The CSTA guidelines go even further and, instead of *repetition* in general, explicitly specify *iteration* and *recursion.*

> *"Explain how sequence, selection, iteration, and recursion are building blocks of algorithms."* (CSTA, p. 18)

Remarkably, the CSTA guidelines are the only curriculum specification in our sample that includes *recursion*. Likewise, CAS and the CSTA highlight the underlying notion of *instruction*, whereas France and the Netherlands do not.

> *"A computer program is a sequence of instructions written to perform a specified task with a computer."* (CAS, p. 14)

The new Dutch curriculum mentiones *instruction* only in the context of assembly languages.

> *"The candidate is able to write a simple program in a machine language, based on the description of an instruction set."* (Netherlands 2016)

We highlight some other concepts occurring in one single document.

Firstly, the explicit inclusion of *concurrency*, *parallelism* and *thread* in the CSTA guidelines is interesting. It is the only document to include these concepts.

> *"Describe the process of parallelization as it relates to problem solving."* (CSTA, p. 16)

> *"Demonstrate concurrency by separating processes into threads and dividing data into parallel streams."* (CSTA, p. 21)

Next, although *searching* and *sorting* appear in the French curriculum and the CAS and CSTA guidelines, the French curriculum is the only one of the three mentioning specific algorithms. It explicitly mentions *merge sort*, *breadth first search* and *depth first search.*

> *"Advanced algorithms: Merge sort; search for a path in a graph by a depth first search (DFS); finding a shortest path through a wide path (BFS). Skills: Understand and explain (orally or in writing) an algorithm. Questioning the effectiveness of an algorithm"* (France)

Finally, the French and new Dutch curriculum are the only ones including *state machines.*

> *"... describe a single event system with a finite state machine."* (France)

> *"The candidate is able to use finite automata for the characterization of certain algorithms."* (Netherlands 2016)

The old Dutch curriculum does not contain any concepts in the algorithm category. The new curriculum states objectives on the usage of *standard algorithms* and the *correctness* and *efficiency* of algorithms. It also provides an elective theme on 'Algorithms, Computability and Logic'.

4.3 Engineering

In the French curriculum, the category engineering has lower presence (4 %) compared to the other specifications. It does, however, contain pointers to *testing* and *verification.*

> *"Fixing a program: Test; instrumentation; error situations or bugs. Skills: Testing a developed program.* Optional*: using a development tool."* (France)

Testing and verification can be found in all other curricula, except for the old Dutch curriculum. The high score of the old Dutch curriculum in this category can be explained by the section on project management and related concepts like *specification*, *requirement*, *client* and *prototype.*

> *"The candidate can asses progress of a simple system development process, test prototypes, verify whether the final product meets the specifications of the client and evaluate whether the system meets the requirements."* (Netherlands 2007, p. 3)

Although the curriculum of the CSTA does not explicitly state concepts like *specification* and *requirement*, it does mention the *software development process* and *software life cycle* and the creation of problem statements in general.

> *"Describe a software development process used to solve software problems (e.g., design, coding, testing, verification)."* (CSTA, p. 18)

Furthermore, the CSTA standards have a strong focus on collaboration during software development. This is not surprising when we take the structure of the document into account. One of the five strands the document is built on is 'Collaboration' and a substantial part of the curriculum is dedicated to this strand. Concepts related to *teamwork* and *collaboration* are *peers*, *experts*, *pair programming*, *project teams*, *feedback*, *communication*, *feedback* and *socialization.* The CSTA document also mentions multiple productivity tools, development tools and collaboration tools explicitly.

> *"Use productivity technology tools (e.g., word processing, spreadsheet, presentation software) for individual and collaborative writing, communication, and publishing activities. Use collaborative tools to communicate with project team members (e.g., discussion threads, wikis, blogs, version control, etc.)."* (CSTA, p. 13)

The new Dutch curriculum contains a section dedicated to 'Informatics-specific skills', containing, amongst others, an objective on cooperation with(in) an inter- and intradisciplinary teams.

> *"The candidate is able to structurally cooperate in a team during the design and development of digital artefacts, and is able to cooperate with people from an application field."* (Netherlands 2016)

The inclusion of collaboration and tools make the CSTA guidelines and the new Dutch curriculum stand apart from the other specifications.

4.4 Society

A major difference between the CSTA guidelines and the new Dutch curriculum on one hand and the other specifications on the other, is the focus on computer science and society (both 11 %). One of the five strands of the CSTA guidelines is 'Society'. Therefore the subject covers a substantial part of the curriculum. The CAS curriculum contains a reference to *privacy*, whereas the French curriculum mentions *personal information* and *ownership.*

> *"Persistence of information: Data, including personal, may be stored for long periods without control by the persons concerned.*
> *Skills: Awareness of the persistence of information on digital networks. Understand the general principles to behave responsibly in relation to the rights of individuals in digital platforms."* (France)

The CSTA extends further with the inclusion of various other concepts ranging from *career perspectives*, via *software licenses* to *software piracy* and *legal behavior.*

> *"Exhibit legal and ethical behaviors when using information and technology and discuss the consequences of misuse."* (CSTA, p. 17)

The new Dutch curriculum contains objectives on 'Computer Science as a Perspective' and 'Ethical Conduct', both in computer science specific skills section. The curriculum includes a domain on 'Interaction', containing 'Social Aspects' and 'Privacy'. Furthermore, an elective theme on 'Social and Individual Influence of Informatics' contains concepts on social as well as legal influences of computer science on society.

> *"-The candidate is able to explain and predict the positive and negative effects of informatics and the networking society on the lives of individuals and on society. - The candidate is able to analyze legal aspects of the application of informatics in society. - The candidate is able to investigate the effects of technical, legal and social measures for privacy-related issues. - The candidate is able to reason about the influence of informatics on cultural expressions."* (Netherlands 2016)

4.5 Rest Category

The old Dutch curriculum contains a high number of concept occurrences in the rest category (8 %). This can be explained by the fact that the curriculum includes subjects on management and organization structures which are not mentioned in the ACM-IEEE body of knowledge. The old Dutch curriculum explicitly states students should know about project management and business structures, for example in the following learning objective taken from the domain 'Basic Concepts and Techniques'.

"The candidate knows the overall organizational structure of companies. He knows the characteristics of a project and can indicate why, during major changes in a information system of a company, one often chooses to use a project." (Netherlands 2007, p. 2)

No other document, including the revised Dutch curriculum, contains these kind of 'contextual' objectives.

5 Conclusion and Discussion

Our analysis suggests that there are similarities between recent curriculum specifications with respect to a number of knowledge categories such as algorithms and data.

We also found differences with respect to emphasis. The French curriculum specification appears to have a stronger emphasis on data. In the new Dutch curriculum and the CSTA standards, concepts around engineering receive substantial attention. The new Dutch curriculum and the CSTA guidelines appear to stress societal aspects more than the other documents in our sample. Software engineering and social and ethical topics cover together a quarter of the concepts occurrences in the CSTA standards. We found the most specific descriptions with respect to algorithms in the CAS guidelines and in the French curriculum. The CSTA standards are more outspoken on collaboration in the context of engineering.

Compared to the old curriculum, the new Dutch core curriculum appears to have smaller emphasis on hardware (architecture, networking) in favor of software and engineering content.

Our method appeared very useful to compare curriculum documents that differ considerably with respect to description style and level of detail. Moreover, the combination of a quantitative and qualitative investigation turned out to be valuable.

Our findings are consistent with those found for K–9 guidelines (Barendsen et al. 2015), but some are more outspoken, for example with respect to algorithms.

The categorization in our study has more categories than the ACM classification used by Hubwieser et al. (2011) and appears to give a more balanced distribution of conceptual content.

This paper contains a selection of observations; the full analysis was instrumental during the construction of the new Dutch curriculum. Our small-scale study spanned a limited number of documents. We plan to apply our method to a larger collection of curriculum documents and guidelines.

The stepwise procedure in this small-scale pilot study made it possible to code an entire document in a reasonable amount of time. The authors quickly reached consensus concerning coding differences. Throughout the process, the intercoder agreement appeared high. We plan to analyze reliability of the method in a more quantitative way.

It will be interesting to investigate whether conceptual differences between curricula can be related to other characteristics, such as the global curriculum intention in the sense of the 'goals' described by Biesta (2015).

References

Académie des Sciences: L'enseignement de l'informatique en France: Il est urgent de ne plus attendre. http://www.academie-sciences.fr/pdf/rapport/rads_0513.pdf. Accessed Apr 2016

ACM/IEEE-CS Joint Task Force on Computing Curricula: Computer science curricula 2013 (Technical report). ACM Press and IEEE Computer Society Press, December 2013

Barendsen, E., Fisser, P., Krüger, J., Tolboom, J.: Herziening van het Nederlandse informaticacurriculum havo-vwo. Paper presented at ORD 2014, Groningen (2014)

Barendsen, E., Mannila, L., Demo, B., Grgurina, N., Izu, C., Mirolo, C., Stupurienė, G.: Concepts in K–9 computer science education. In: Proceedings of the 2015 ITiCSE on Working Group Reports, pp. 85–116. ACM (2015)

Barendsen, E., Tolboom, J.: Advies examenprogramma informatica vwo-havo: inhouden invoering. SLO, Enschede (2016)

Biesta, G.J.J.: Good Education in an Age of Measurement: Ethics, Politics, Democracy. Routledge, Abingdon (2015)

British Department for Education: Computing programmes of study: key stages 1 and 2. National curriculum in England (2013). http://www.computingatschool.org.uk

Cohen, L., Manion, L., Morrison, K.: Research Methods in Education. Routledge, London, New York (2013)

Computing at School Working Group: Computer science: a curriculum for schools (2012). http://www.computingatschool.org.uk/data/uploads/ComputingCurric.pdf. Accessed Sep 2013

CSTA: K-12 computer science standards. ACM (2011). http://csta.acm.org/Curriculum/sub/K12Standards.html

Dagiene, V., Jevsikova, T., Schulte, C., Sentance, S., Thota, N., et al.: A comparison of current trends within computer science teaching in school in Germany and the UK. In: Informatics in Schools: Proceedings of the 6th International Conference ISSEP 2013—selected papers, pp. 63–75 (2013)

Furber, S.: Shut down or restart? The way forward for computing in UK schools. The Royal Society, London (2012)

Gander, W., Petit, A., Berry, G., Demo, B., Vahrenhold, J., McGettrick, A., Meyer, B.: Informatics education: Europe cannot afford to miss the boat. (Report of the Joint Informatics Europe & ACM Europe Working Group on Informatics Education) (2013). http://www.informatics-europe.org/images/documents/informatics-education-europe-report.pdf. Accessed Aug 2013

Grgurina, N., Tolboom, J.: The first decade of informatics in Dutch high schools. Inf. Educ. **7**(1), 55–74 (2008)

Hubwieser, P.: The Darmstadt model: a first step towards a research framework for computer science education in schools. In: International Conference on Informatics in Schools: Situation, Evolution, and Perspectives, pp. 1–14 (2013)

Hubwieser, P., Armoni, M., Brinda, T., Dagienė, V., Diethelm, I., Giannakos, M.N., Schubert, S.: Computer science/informatics in secondary education. In: Proceedings of the 16th Annual Conference Reports on Innovation and Technology in Computer Science Education-Working Group Reports, pp. 19–38 (2011)

Kaczmarczyk, L., Dopplick, R.: Preparing students for computing workforce needs in the US. ACM SIGCSE Bull. **46**(2), 8 (2014)

KNAW: Digitale geletterdheid in het voortgezet onderwijs: vaardigheden en attitudes voor de 21ste eeuw. Koninklijke Nederlandse Akademie van Wetenschappen, Amsterdam (2012)

Ministère de l'Éducation Nationale: Enseignement de spécialité d'informatique et sciences du numérique de la série scientifique – classe terminale (2012). http://www.education.gouv.fr/pid25535/bulletin_officiel.html?cid_bo=57572

Samaey, G., Van Remortel, J., Bersini, H., Bruynseraede, Y., Dekelver, J., Laender, F.D., Wyffels, F.: Informaticawetenschappen in het leerplichtonderwijs. Koninklijke Vlaamse Academie van België voor Wetenschappen en Kunsten, Brussel (2014)

Schmidt, V.: Handreiking schoolexamen informatica havo/vwo. SLO, Enschede (2007)

Steenvoorden, T.: Characterizing fundamental ideas in international computer science curricula (unpublished Master's thesis). Radboud University, The Netherlands (2015)

http://www.springer.com/978-3-319-46746-7

Informatics in Schools: Improvement of Informatics Knowledge and Perception
9th International Conference on Informatics in Schools: Situation, Evolution, and Perspectives, ISSEP 2016, Münster, Germany, October 13-15, 2016, Proceedings
Brodnik, A.; Tort, F. (Eds.)
2016, XII, 205 p. 29 illus., Softcover
ISBN: 978-3-319-46746-7