NUI MAYNOOTH

Ollscoil na hÉireann Má Nuad

Department of Computer Science

Technical Report: NUIM-CS-TR-2010-01

Discrete Structures Teaching: A Systematic Literature Review

Thomas Whelan, Susan Bergin and James F. Power

Department of Computer Science
National University of Ireland
Maynooth, Co. Kildare, Ireland.

Date: September 2010

Abstract

The ACM curriculum defines Discrete Structures as foundational material for computer science; material that the ability to work with is necessary for many other areas of computer science. However a significant amount of uncertainty remains on how exactly to teach the subject to computer science undergraduates in a tangible way. This technical report presents a systematic literature review of the literature relevant to the teaching of discrete structures and evaluates the findings of the process. A categorisation of the results is described followed by an analysis of each category.

Discrete Structures Teaching: A Systematic Literature Review

Thomas Whelan, Susan Bergin and James F. Power

Technical Report NUIM-CS-TR-2010-01,
Department of Computer Science, NUI Maynooth.

1 Introduction

This technical report describes a system literature review of the teaching of discrete structures based on the guidelines outlined by Kitchenham [KC07] for performing systematic literature reviews in software engineering. Our evaluation contains cues from other sources of analysis, such as the meta-analysis presented by Valentine [Val04]. Rather than containing the entire systematic literature review process as described by Kitchenham, this report mainly documents the more in depth parts of the process including definition of search space, identification of relevant literature, selection of studies, data extraction and quality assessment. Upon approaching the topic of discrete structures teaching we find that there is a sizeable amount of research in the area spanning over multiple decades. In order to get a good bearing on the current state of the area, a systematic literature review can be performed to provide us with a good foundation.

2 Search Space - Title based filtering

Our initial search space for sources will be the ACM Digital Library. When displaying results in the ACM Digital Library we have two options; "Sort by" and "Form". During the search the "Sort by" field was left at its default setting; "relevance", while "Form" was set to "condensed form", which contains 50 papers per page with publication date, journal/conference and title. (As opposed to the default "expanded form" which contains 20 papers per page, publication date, journal/conference, title and partial abstract).

Beginning with the search terms "Discrete structures" and "Discrete mathematics" (referred to our "base terms" henceforth) we assume that the set of results returned by these queries will be a superset of the set of results returned by a base search term also including "teaching", "education" or "learning".

2.1 Search Term Cut-off Points

The following criteria was used for selecting papers;

- Papers not including any of the terms "Discrete Math", "Discrete Structures" or "Discrete Mathematics" in their title will be filtered / ignored.
- Each page of condensed form search results contains 50 papers, as such, the search will be stopped upon reaching a page of results containing two or less accepted papers.

Papers not meeting the criteria will not be shown in results. It should be noted that the majority of results for our base terms are related to education, which allows us to more leniently accept papers based on title alone, giving us a better a chance of finding relevant papers that only describe their applicability in the abstract.

On the availability of sources, a paper is considered "Available" if there is a direct or indirect link to download a full version of the paper on the site on which on the result was found.

2.2 Base Term Results - Title Based

2.2.1 "Discrete Structures"

Page Reach	Papers Inspected	Total Accepted	Total Available
2	100	18	14

Sources:
[Ber76, DRT75, Pra76, Nef10, WW06, TM74, Mau06, Wax75, Fle85, EJ73, Mar84, YGM73, Hei93, GEEH09, DV04, LL06, Fla07, Mar89]

2.2.2 "Discrete Mathematics"

Page Reach	Papers Inspected	Total Accepted	Total Available
4	200	55	52

Sources:
[Mar00, Noh07, Fle93, AHH^{+}06, Pag03, CH05, Kro07, Set09, McG02, Buc05, ZJ10, Mar89, Bro93, GEEH09, Sid86, MH08, Sta09, Ber97, Mar05, Rau08, HM06, EHM00, SS05, Gol04, Wai92, SW93, HE09, War95, Oli04, BMW04, Mar06, MH04, Hen90, Bri93, HR05, Lev90, Kob98, Kos00, Mic92, LL06, DV04, Nef10, HD07, Sut05, Pio06, GH05, WW06, Mau06, BSHV00, JJK03, HDH06, Hen07, Ber76, DRT75, Fle85]

2.2.3 "Discrete Mathematics" and "Discrete Structures" common results

Sources:
[Ber76, GEEH09, Mau06, Nef10, Mar89, LL06, WW06, DV04, DRT75, Fle85]

Upon inspection of titles, it can be seen that almost all of the papers fetched are related to the teaching of discrete structures/mathematics. It is also evident that "Discrete Mathematics" is a more popular name for the subject.

2.3 Inclusion / Exclusion of "Teaching", "Education" or "Learning"

As assumed, the inclusion of these terms did not reveal many extra sources, but there were a small amount of new papers discovered. "Total new" denotes number of papers which have no parent term, i.e. were not previously found when using either base term.

2.3.1 "Discrete Structures Teaching"

Page Reach	Papers Inspected	Total	Total New
3	150	14	2

Parent Term:	Sources:
"Discrete Structures"	[YGM73, TM74, EJ73]
"Discrete Mathematics"	[SS05, Lev90, Pag03, McG02]
Both	[DV04, Nef10, GEEH09, LL06, BMW04]
None	[SWR05, MAR07]

2.3.2 "Discrete Structures Education"

Page Reach	Papers Inspected	Total	Total New
3	150	11	1

Parent Term:	Sources:
"Discrete Structures"	None
"Discrete Mathematics"	[Pag03, McG02, BMW04, Mar05, AHH$^+$06]
Both	[DV04, Nef10, WW06, LL06, Mau06]
None	[SWR05]

2.3.3 "Discrete Structures Learning"

Page Reach	Papers Inspected	Total	Total New
1	50	2	0

Parent Term:	Sources:
"Discrete Structures"	None
"Discrete Mathematics"	[SS05, HE09]
Both	None
None	None

It should be noted that many papers are common between the inclusion of "Teaching", "Education" and "Learning".

2.3.4 "Discrete Mathematics Teaching"

Page Reach	Papers Inspected	Total	Total New
3	150	30	1

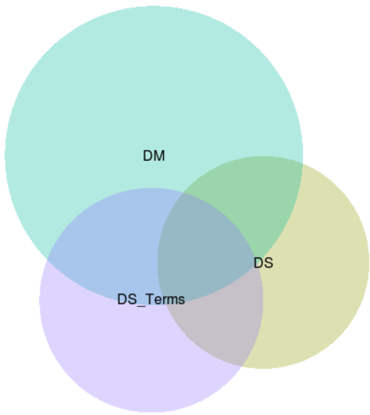

Figure 1: Proportional Diagram of Overlapping Results for Discrete Structures Terms
DS: "Discrete Structures", DM: "Discrete Mathematics", DS_Terms: "Discrete Structures" + "teaching", "education" and "learning".

Parent Term:	**Sources:**
"Discrete Structures"	None
"Discrete Mathematics"	[ZJ10, CH05, Sta09, Noh07, HR05, Set09, Lev90, War95, EHM00, JJK03, GH05, BMW04, McG02, HDH06, Pio06, Buc05, HD07, SS05, Gol04, Hen07, Bro93, Pag03, Sut05, AHH+06, Mar00]
Both	[GEEH09, LL06, DV04, Nef10]
None	[MAR07]

2.3.5 "Discrete Mathematics Education"

Page Reach	**Papers Inspected**	**Total**	**Total New**
5	250	24	1

Parent Term:	**Sources:**
"Discrete Structures"	None
"Discrete Mathematics"	[Pag03, HM06, MH04, AHH+06, Rau08, Gol04, SS05, Mar06, Mar00, McG02, Mar05, CH05, HR05, BMW04, Pio06, Sut05, GH05, Buc05]
Both	[DV04, Nef10, WW06, LL06, GEEH09]
None	[SWR05]

2.3.6 "Discrete Mathematics Learning"

Page Reach	**Papers Inspected**	**Total**	**Total New**
4	200	21	1

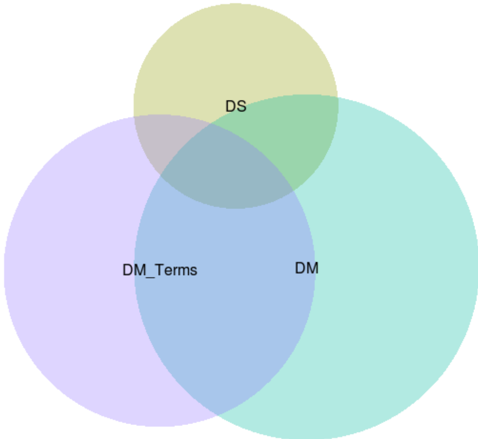

Figure 2: Proportional Diagram of Overlapping Results for Discrete Mathematics Terms
DS: "Discrete Structures", DM: "Discrete Mathematics", DM_Terms: "Discrete Mathematics" + "teaching", "education" and "learning".

Parent Term:	**Sources:**
"Discrete Structures"	None
"Discrete Mathematics"	[HE09, McG02, SS05, CH05, HD07, Pio06, GH05, Kro07, Mar06, Oli04, JJK03, Sut05, HDH06, Set09, AHH^{+}06, Mar00, Buc05, Pag03]
Both	[LL06, WW06]
None	[MAR07]

As can be seen, the effect of adding on "teaching", "education" or "learning" to the base search times has a negligible effect on results, turning up only 2 papers that were missed in the initial search.

2.3.7 Other Sources

The ACM Digital Library has an option to search "The Guide" which is a collection of bibliographic citations and abstracts of works published by both ACM and other publishers. It contains over 1.2 million citations from over 3,000 publishers, enabling a larger search space. The following are the results of extending our search into "The Guide" using the same criteria as before. However, results not in the ACM Digital Library do not provide direct links to the text and as such a small amount are unavailable. In addition to this, many books are returned in a search.

The following new papers were found in "The Guide" and not previously in the ACM Digital Library:

Sources:
[Li08, Xiu09]

Another port of call when it comes to searching for anything is a general web search engine - Google specifically. However, it must be taken into consideration that many results returned when using such a search engine will not be relevant and / or academically published. In order to remedy this, more precise search terms must be used in order to find websites with [links to] more relevant publications. As such, the base terms were only ever queried along with either "teaching", "education" or "learning". All queries were done as individual words and also as complete phrases. Due to the ambiguity of results returned, clarity in the nature of results returned appeared to decrease around the 5th page of results. Many papers returned were already discovered in the ACM searches, while a significant number of results were actually lectures and course work from different courses in different institutions. Another number of results were in article format but unpublished and / or unrelated.

2.3.8 Final Paper List - According to Title based filtering

This list is contained at the end of this document under the "Title Filtered Sources" section.

3 Further Filtering - Abstract Based

3.1 Motivation

Papers whose abstracts don't mention key words such as "course", "teaching", "education" or "learning" will be filtered out. Unless the abstract gives us reason to believe the full paper pertains to the teaching of discrete structures - be it a new course, new approach, analysis of curricula or otherwise - it will be excluded.

3.2 Results - Abstract Based

3.2.1 Final Paper List - According to Abstract based filtering

Sources:
[Mar00, SS05, SW93, BMW04, Mar06, HR05, Lev90, Sut05, Pio06, Mau06, HDH06, Hen07, Noh07, Fle93, AHH$^+$06, DV04, Li08, Xiu09, SWR05, JJK03, MAR07, Hei93, Wax75, Mar84, TM74, Ber76, Pra76, Ber97, WW06, BSHV00, Nef10, LL06, Bri93, Hen90, War95, Wai92, Pag03, Sid86, GEEH09, Bro93, Buc05, Mar89, ZJ10, CH05, Kro07, Set09, McG02]

The main reason for rejection was the nature of the paper (e.g. book review, poster and discussion sessions, overly technical papers, documentation) rather than the content.

4 Analysis

4.1 Precision and Recall

In order to measure the exactness and completeness of our search for sources we can calculate the *precision* and *recall* associated with each search term. *Precision* is defined as the number of relevant documents retrieved by a search divided by the total number of documents retrieved by that search, while *recall* is defined as the number of relevant documents retrieved by a search divided by the total number of existing relevant documents (which should have been retrieved). We will calculate the precision and recall values for each search term on both sets of papers; Titled filtered and Abstract filtered.

In the following tables our searching terms are abbreviated to their associated acronyms, e.g. "Discrete Structures" is DS while "Discrete Mathematics Education" is DM+E.

4.1.1 Precision and Recall for Title Filtered Papers

Total Relevant: 67

Search Term	Retrieved	Relevant	Precision	Recall
DS	100	18	0.18	0.269
DM	200	55	0.275	0.821
DS+T	150	14	0.0934	0.209
DM+T	150	30	0.2	0.448
DS+E	150	11	0.0734	0.164
DM+E	250	24	0.096	0.358
DS+L	50	2	0.04	0.0299
DM+L	200	21	0.105	0.313

4.1.2 Precision and Recall for Abstract Filtered Papers

Total Relevant: 47

Search Term	Retrieved	Relevant	Precision	Recall
DS	100	13	0.13	0.277
DM	200	38	0.19	0.787
DS+T	150	12	0.08	0.255
DM+T	150	25	0.1667	0.532
DS+E	150	10	0.0667	0.213
DM+E	250	18	0.072	0.383
DS+L	50	1	0.02	0.0213
DM+L	200	17	0.085	0.362

This analysis gives us a clear indication that the strongest search term for finding relevant and useful sources was definitely "Discrete Mathmatics", for both sets of papers.

4.2 Chronological Differences

Observing figure 3 we see that the term "Discrete Structures" is clearly not used as much as "Discrete Mathematics" when it comes to discussing the topic. However "Discrete Structures" was seemingly more prevalent in the early years.

4.3 Publication Differences

The following table is a breakdown of the title filtered papers according to the search term used and the place of publishing. Papers which are shared between different search terms are weighted accordingly in order to give a total of 67.

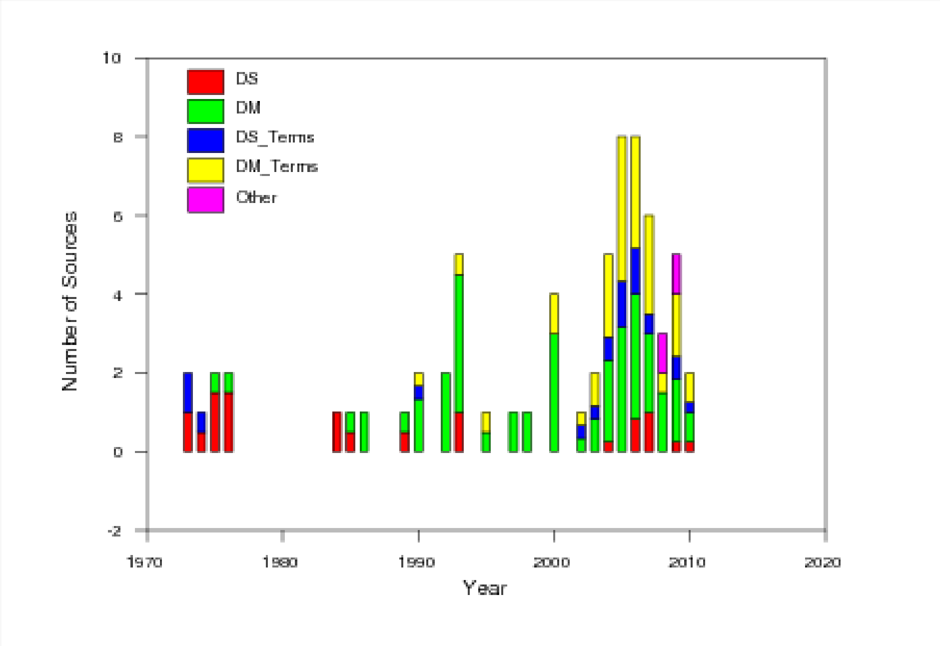

Figure 3: Citations and Search Terms Versus Time
DS: "Discrete Structures", DM: "Discrete Mathematics", D_Terms: "Discrete Mathematics / Structures" + "teaching", "education" and "learning", Other: As discussed in section 2.3.7

Place	**Discrete Structures**	**Discrete Mathematics**	**Other**
ACM-SE	0	1	0
CCSC	0	2	0
CSC	0.5	1.5	0
FIE	0	1	0
ICFP	0	1	0
ITiCSE	0	4	0
JCSC	1	11	0
JERIC	0	4	0
SIGACT News	0	3	0
SIGCSE	7.5	14.5	2
SIGCSE Bulletin	3	5	0
SIGITE	0	1	0
SIGPLAN	0	1	0
SODA	1	0	0
ETCS	0	0	1
ICYCS	0	0	1

As can be seen, there is a correlation between where papers are published between the search terms, but "Discrete Structures" results are too sparse to properly analyse this.

5 Categorisation

Having acquired our set of sources the next step is categorisation. Each paper will be evaluated under each of the following headings; "Distribution of Discrete Structures Course Content", "What, Why and When Students Should Learn Discrete Structures", "Data Structures and Discrete Structures Crossover", "Programming in Discrete Structures" and finally "Novel Teaching Methods". It should be noted that a small set of papers did not fall under any category and are contained within section 5.3.

5.1 Categorical Break Down

Category	Sources	Total
Distribution of Course Content	[Ber76, TM74, BMW04, Fle93, Nef10, Bro93, BSHV00]	7
What, Why and When Students Should Learn	[Ber76, BMW04, Pio06, HDH06, Mau06, SW93, Mar06, Mar00, Mar89, GEEH09, Pra76, Bro93, Wax75, War95, AHH$^+$06, LL06, BSHV00]	17
Data Structures and Discrete Structures Crossover	[DV04, Set09, Mar84, Ber97, MAR07, Pag03, Ber76, Sid86, Wai92, LL06, WW06, BSHV00]	12
Programming in Discrete Structures	[Set09, CH05, Lev90, HR05, Mar84, Ber97, Hei93, MAR07, Xiu09, Noh07, Hen90, Bri93, Wai92, McG02, SWR05, WW06]	16
Novel Teaching Methods	[CH05, SW93, Sut05, Ber97, HR05, Kro07, Buc05, Nef10, ZJ10, Li08, Xiu09, Noh07, Hen90, Bri93, AHH$^+$06, Wai92, SWR05, WW06]	18

5.2 Categorical Discussion

The following is a discussion of each category citing multiple examples from each set of papers belonging to the different categories.

5.2.1 Distribution of Course Content

The concept of dissolving discrete structures and moving its individual components into the courses they belong to ties into our other category of "Data Structures and Discrete Structures Crossover". For example, moving logic into software verification or formal methods while moving all study of trees and graphs into data structures. This idea appears across a large time span, from Berztiss, Tremblay and Manohar [Ber76, TM74] in the 70's to Fleury and Neff [Fle93, Nef10] in the 90's and 2000's. It is highlighted quite frequently that one of the main ideas of teaching students discrete structures is to prepare them for more theoretical content they meet in courses later on [Mar00, Mar89], which brings up the question that if computer science majors are finding the content too difficult or aimless so early on in their studies, why not push it back to "where it belongs"? With Tremblay and Manohar [TM74] for example, counting techniques, permutations and probability are omitted from their discrete structures course (although part of the curriculum) as they are covered in other courses. A more novel approach is taken specifically by Neff [Nef10]. Parts of the discrete structures course are outsourced and the subject is then taught with an approach of rather than "Here's the tools, you'll need them sometime", it says "Here's a computer science problem we need to solve, what tools do we need to solve it?".

5.2.2 What, Why and When Students Should Learn

These questions are three that are discussed quite frequently across many papers. It is acknowledged that discrete structures is just a basis for the more specialised computer science topics which students don't encounter for quite some time [Ber76]. This is one of the main reasons why these questions are asked so frequently. Timing and duration of the subject is a significant concern [Mar00], if the course is later on for students does that justify increased complexity? Marion [Mar89] draws us back to our previous point about dismantling the subject, but instead segments the course rather than completely dissolving it. This would seem to allow an easier introductory course for discrete structures and a subsequently more in-depth course later on when students are more experienced. But the appreciation of the subject is always a problem early on [GEEH09], and the idea of "Why are we teaching something that students [believe they] may never use?" does rear its head. If discrete structures is to serve as a basis for the wide variety of more in-depth computer science modules that students meet in later years, it should justify its existence by demonstrating its necessity and application.

5.2.3 Data Structures and Discrete Structures

In line with the category of "Programming in Discrete Structures", another noticeable feature of many of the papers reviewed is an apparent crossover between topics covered in discrete structures and data structures (CS2). From reading the curriculum for discrete structures [ACM08] we immediately notice that the most obvious crossovers been discrete structures and data structures are graphs and trees, such a crossover (as well as some others) are hinted at by Decker and Ventura [DV04]. Granted the context in which these are meant to be studied are different between the two subjects but nevertheless this does give us some evidence that discrete structures may not be as coherent as we may like. Looking at some of the papers discussed previously in relation to programming in discrete structures courses [Set09, Mar84, Ber97, MAR07], the concept of coding up actual algorithms and running them is very similar to the way in which data structures are studied. Showing a student a structure (such as a linked list) and then asking them to implement one is a similar process to giving a student the definition of list and concatenation of a list and then implementing this in a functional language, as with Page [Pag03]. There is a substantial amount of coupling between data structures and discrete structures when evaluating performance and course effectiveness [Ber76, Sid86, Pag03, DV04], and as a result it is evident that this relationship should be investigated further.

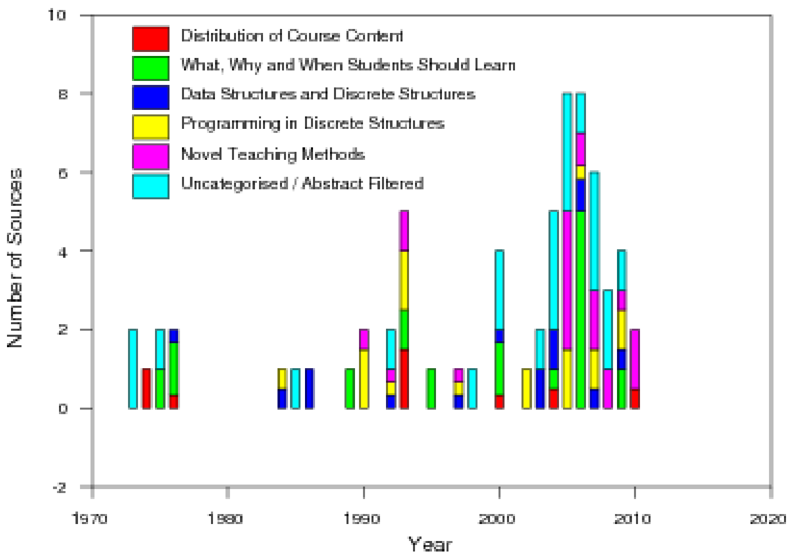

Figure 4: Citations and Categorisations Versus Time

5.2.4 Programming in Discrete Structures

A number of papers describe curricula that involve programming or coding up of a discrete structures related problem. This approach, by Setzer [Set09] for example, can demonstrate some real world examples and aid students in seeing the applicability of what they're being taught and motivate them to excel at the course work. With work by Cigas and Hsin [CH05], we see the introduction of a *specification* for problems; one of the primary applications of the logic learned in discrete structures. The idea of determining an algorithm based on the specification and then trying it out in an applet appears to be something that would be quite beneficial for a student trying to understand the motivations behind the course. Again with Martin [Mar84] we see students being given the chance to actually implement an algorithm of their choice which they learned in their discrete structures course, enforcing their understanding of the material. Looking at work by Berry [Ber97], we see a more advanced application of coding involving the use of the multi-paradigm language Scheme. As a language which contains elements of functional programming there is already a clear link between this approach and the *functions* aspect of discrete structures. Similarly with Hein [Hei93], we see the use of FP, ML and Prolog to teach students discrete structures concepts. But with some of these approaches we do drift towards more formal and complex abstractions, which do tend to confuse less experienced students. McMaster et al. [MAR07] do draw the conclusion that teaching discrete structures with programming does make it easier for students (at least for those students who already know how to program). Implementing a theory learned in discrete structures as a program does require a full understanding of it, rather than the regurgitation of a rote learned mathematical proof. However, it is important not to turn discrete structures into another programming course as the goal of the subject is not to improve knowledge of a programming language [ACM08]. The main goal of discrete structures cannot be overlooked forever by covering it up with an easy to understand language as sometime or another in future subjects students will have to face cold hard algebra with new notation; there is no avoiding it.

5.2.5 Novel Teaching Methods

Finally, in response to the uncertainties of educating students on discrete structures a number of novel teaching methods are described for us. For example, with Berry, Cigas and Hsin [CH05, Ber97] visual tools are often a lot less scary when introducing a student to something as [potentially] complex as discrete structures. Exploitation of a student's familiarity with programming is also an interesting way to tackle teaching problems; as students may be a lot more comfortable in a software engineering setting than in a pure mathematical one (in which discrete structures is sometimes presented). The approach taken by Krone [Kro07] is also quite interesting as often times new notation can catch people out when attempting to learn a new concept. A simple reinforcement of exactly what a symbol means can uncloud a problem for a student, allowing them to focus on understanding the theory rather than the theory's representation. The collaborative approach discussed by Buchele [Buc05] does take a more traditional pen and paper approach but the increased student engagement can certainly be seen as a beneficial. Collaborative homework also enables mingling of those who understand certain concepts and those who don't, in a sense making the education of a concept somewhat "viral". The problem-directed approach of Neff [Nef10] is certainly very different from many of the other courses discussed and does do a lot to tackle the appreciation and motivation problems with discrete structures. Finally, the computer science-friendly method by Zeng and Jiang [ZJ10] builds on the foundation of staying in a student's comfort zone when introducing a new idea. This appears to ease students into the formality of certain areas of discrete structures.

5.3 Evaluation of Course Success

Given that a very large number of papers discuss some kind of discrete structures course it is important to analyse the ways in which these courses are evaluated. Notably, it appears that statistics in some cases can be quite sparse when it comes to discrete structures. With Martin [Mar84] for example, a set of 20 students may not tell us as much as a set of 100 students would when it comes to certain questions. Going by statistics alone, in taking radical approaches such as scattering discrete structures course content to later on modules we effectively move into unknown territory. For this reason, it is necessary to collect a significant amount of statistics for a wide variety of approaches taken to teaching the subject. Having said that, there does exist some more sizeable data sets such as those discussed by Decker and Ventura [DV04], Page [Pag03] and Sidbury [Sid86], each analysing the final grades for students. As well as these grade based statistics there also exists some student survey type course evaluations as presented by Martin [Mar84] and Buchele [Buc05] which provide more anecdotal type feedback on courses. This anecdotal type evaluation is a lot stronger in some papers than in others, sometimes with just a sentence or two given for student thoughts on a course [Ber76, Kro07, MAR07, TM74]. There appears to be quite some variety in course evaluation techniques and as a result it is difficult to know what to expect in terms of course outcome analysis.

Following on from this, we can breakdown the set of papers that contain course evaluation into three evaluation sub-categories: "Grade Based Statistics", "Survey Based Statistics", "Anecdotal Feedback";

Category	**Sources**	**Total**
Grade Based Statistics	[Pag03, Pio06, Sid86, DV04, Xiu09, JJK03, McG02]	7
Survey Based Statistics	[Mar84, Wai92, Sut05, Mar06, SS05, SW93, Buc05]	7
Anecdotal Feedback	[Ber76, Hen07, HR05, Kro07, MAR07, TM74, Li08, Bro93, Hen90, Bri93, SWR05]	11

6 Concluding Remarks

In this report we presented a systematic literature review of the teaching of discrete structures. An analysis of the results discovered in the review was described followed by the categorisation of each source paper into any number of six named categories. The process has brought into light a number of different aspects of the topic of discrete structures teaching.

7 Epilogue

The Educational Resources Information Centre (ERIC) collection is the world's largest digital library of multi-disciplinary education literature. Although the ACM library is far more likely to house papers related to this work, we have recently extended the review to include the ERIC collection. However, given that the topic of discrete structures is not only taught to computer science students but also to other undergraduate mathematics students and high-school students, it was envisaged that many of the papers found in the collection on teaching discrete structures would relate to a different student body and thus not be relevant here. The search terms used and the papers found are outlined in 7.0.1. In addition to the standard search terms, given the multi-disciplinary nature of the ERIC collection, the terms "Discrete Structures and Computer Science" and "Discrete Mathematics and Computer Science" were included. An analysis of the papers found in ERIC revealed only four new unique papers related to this study; confirming our hypothesis that ERIC was an unlikely repository for publications in this area.

Three of the papers are suitable for inclusion in the novel teaching methods category. As with Buchele [Buc05], both Hagelgans [Hag98] and Bravaco and Simonson [BS04], use collaborative learning approaches to improve student learning of discrete mathematics. In the former case, second year computer science and mathematics students worked in collaborative groups using the DERIVE programming language. The author found that students had a positive perception of the class and had improved problem-solving skills. The latter study relates also to the relationship between discrete mathematics and data structures. The authors describe the use of a learning community that utilised seminars to act as a link between data structures and discrete mathematics. The students used this medium to investigate how mathematics is used to formalise computer science, and how computer science can be used as a tool to explore mathematics. Students provided positive written evaluations for this approach. The third paper in this category describes an approach to teaching discrete mathematics in context by using examples from topics within computer science such as artificial intelligence and software agents [Oxl10]. The author provides numerous examples of the application of discrete mathematics to computer science topics, but no evaluation of this approach is given.

The final paper, by Galpin and Sanders [GS07], outlines a study on the perceptions of first year computer science students on a variety of topics, including the relationship between computer science and mathematics. Interestingly, they found that fewer students thought that there was strong relationship between the two disciplines at the end of first year than at the start of first year. The authors explain that the majority of students taking computer science in first year also take multiple continuous mathematics courses and propose that the lack of connection between the disciplines is more likely caused by these additional continuous courses rather than the discrete mathematics course in the CS program. This is worth consideration with regards to when students should study discrete mathematics, as not introducing students to the topic in first year could result in students perceiving mathematics not to be important in CS.

7.0.1 Review of the ERIC collection

Search Term	Papers Found	Total Accepted	Total Available
DS	68	2	2
DS and Teaching	14	0	0
DS and Learning	26	1	1
DS and Education	38	1	1
DS and Computer Science	4	1	1
DM	357	5	5
DM and Teaching	109	0	0
DM and Learning	124	1	1
DM and Education	285	2	2
DM and Computer Science	32	3	3

References

[ACM08] ACM/IEEE Interim Review Task Force, *Computer science curriculum 2008: An interim revision of CS 2001*, Report from the interim review task force, Association for Computing Machinery and the IEEE Computer Society, December 2008.

[KC07] Barbara Kitchenham and Stuart Charters, *Guidelines for performing Systematic Literature Reviews in Software Engineering*, Tech. Report EBSE 2007-001, Keele University and Durham University Joint Report, 2007.

[Val04] David W. Valentine, *CS educational research: a meta-analysis of SIGCSE technical symposium proceedings*, 35th SIGCSE Technical Symposium on Computer Science Education (Norfolk, Virginia, USA), 2004, pp. 255–259.

Title Filtered Sources

[AHH+06] Vicki L. Almstrum, Peter B. Henderson, Valerie Harvey, Cinda Heeren, William Marion, Charles Riedesel, Leen-Kiat Soh, and Allison Elliott Tew, *Concept inventories in computer science for the topic discrete mathematics*, Working group reports on ITiCSE on innovation and technology in computer science education (Bologna, Italy), 2006, pp. 132–145.

[Ber76] A. T. Berztiss, *The why and how of discrete structures*, 6th SIGCSE Technical Symposium on Computer Science Education, July 1976, pp. 22–25.

[Ber97] Jonathan Berry, *Improving discrete mathematics and algorithms curricula with LINK*, 2nd Conference on Integrating Technology into Computer Science Education, June 1997, pp. 14–20.

[BMW04] Doug Baldwin, Bill Marion, and Henry Walker, *Status report on the SIGCSE committee on the implementation of a discrete mathematics course*, 35th SIGCSE Technical Symposium on Computer Science Education (Norfolk, Virginia, USA), 2004, pp. 98–99.

[Bri93] Susan Bridges, *Graphics assignments in discrete mathematics*, 24th SIGCSE Technical Symposium on Computer Science Education (Indianapolis, Indiana), 1993, pp. 83–86.

[Bro93] David T. Brown, *Discrete mathematics II*, SIGCSE Bulletin **25** (1993), no. 4, 13–17.

[BS04] Ralph Bravaco and Shai Simonson, *Mathematics and Computer Science: Exploring a Symbiotic Relationship*, Mathematics and Computer Education **38** (2004), no. 3, 307–317.

[BSHV00] Doug Baldwin, Carl H. Smith, Peter B. Hendersen, and Venu Vadisigi, *CS1 and CS2 (panel session): foundations of computer science and discrete mathematics*, 31st SIGCSE Technical Symposium on Computer Science Education (Austin, Texas), 2000, pp. 397–398.

[Buc05] Suzanne Fox Buchele, *Increased student participation in a discrete mathematics course*, Journal of Computing Sciences in Colleges **20** (2005), no. 4, 68–76.

[CH05] John Cigas and Wen Jung Hsin, *Teaching proofs and algorithms in discrete mathematics with online visual logic puzzles*, J. Educ. Resour. Comput. **5** (2005), no. 2.

[DRT75] Margaret E. Dexter, Margaret L. Rhoden, and Jerry Sue Townsend, *A discrete structures course for a small college*, 5th SIGCSE Technical Symposium on Computer Science Education, 1975, pp. 61–64.

[DV04] Adrienne Decker and Phil Ventura, *We claim this class for computer science: a non-mathematician's discrete structures course*, 35th SIGCSE Technical Symposium on Computer Science Education, March 2004, pp. 442–446.

[EHM00] Susanna Epp, Peter Henderson, and Bill Marion, *Benefits of an early introduction to discrete mathematics (panel discussion)*, Proceedings of the seventh annual consortium on Computing in small colleges midwestern conference (Valparaiso, Indiana), Consortium for Computing Sciences in Colleges, 2000, pp. 122–123.

[EJ73] G. L. Engel and N. D. Jones, *Discrete structures in the undergraduate computer science curriculum*, 3rd SIGCSE Technical Symposium on Computer Science Education, 1973, pp. 56–59.

[Fla07] Philippe Flajolet, *Analytic combinatorics: a calculus of discrete structures*, 18th Annual ACM-SIAM Symposium on Discrete Algorithms (New Orleans, Louisiana), 2007, pp. 137–148.

[Fle85] Ann Fleury, *The discrete structures course (abstract only): making its purpose visible*, 13th Annual Conference on Computer Science (New Orleans, Louisiana), 1985, p. 431.

[Fle93] Ann E. Fleury, *Evaluating discrete mathematics exercises*, 24th SIGCSE Technical Symposium on Computer Science Education, April 1993, pp. 73–77.

[GEEH09] David Gries, Michael Eckmann, Ali Erkan, and James Heliotis, *Discrete mathematics/structures: How do we deal with the late appreciation problem?*, Journal of Computing Sciences in Colleges **24** (2009), no. 6, 110–112.

[GH05] Timothy S. Gegg-Harrison, *Constructing contracts: Making discrete mathematics relevant to beginning programmers*, J. Educ. Resour. Comput. **5** (2005), no. 2, 3.

[Gol04] Evan Golub, *Handwritten slides on a tablet PC in a discrete mathematics course*, 35th SIGCSE Technical Symposium on Computer Science Education (Norfolk, Virginia), 2004, pp. 51–55.

[GS07] Vashti C. Galpin and Ian D. Sanders, *Perceptions of Computer Science at a South African University*, Computers and Education **49** (2007), no. 4, 1330–1356.

[Hag98] Nancy Hagelgans, *Learning Discrete Mathematics with Derive*, Proceedings of the Third International DERIVE/TI-92 Conference (Gettysburg, PA, USA), 1998.

[HD07] Kevin L. Huggins and Rachelle DeCoste, *Reflections on teaching discrete math for the first time*, SIGCSE Bulletin **39** (2007), no. 2, 28–31.

[HDH06] Peter B. Henderson, Rachelle DeCoste, and Kevin L. Huggins, *Preparing to teach discrete math for the first time*, Working group reports on ITiCSE on Innovation and technology in computer science education (Bologna, Italy), 2006, pp. 20–21.

[HE09] Timothy Highley and Anne E. Edlin, *Discrete mathematics assessment using learning objectives based on Bloom's taxonomy*, 39th IEEE International Conference on Frontiers in Education (San Antonio, Texas), 2009, pp. 341–346.

[Hei93] James L. Hein, *A declarative laboratory approach for discrete structures, logic, and computability*, SIGCSE Bulletin **25** (1993), no. 3, 19–25.

[Hen90] Peter B. Henderson, *Discrete mathematics as a precursor to programming*, 21st SIGCSE Technical Symposium on Computer Science Education (Washington D.C.), 1990, pp. 17–21.

[Hen07] ______, *Reflections on teaching discrete math for the first time*, SIGCSE Bulletin **39** (2007), no. 2, 24–24.

[HM06] Peter B. Henderson and William Marion, *Interactive activities for learning discrete mathematics concepts: tutorial presentation*, Journal of Computing Sciences in Colleges **22** (2006), no. 1, 65–66.

[HR05] Valerie J. Harvey and Susan H. Rodger, *Editorial for the special issue on software support for teaching discrete mathematics*, J. Educ. Resour. Comput. **5** (2005), no. 2, 1.

[JJK03] Lisa Jamba-Joyner and William F. Klostermeyer, *Predictors for success in a discrete math course*, SIGCSE Bulletin **35** (2003), no. 2, 66–69.

[Kob98] Neal Koblitz, *Book review: Discrete Mathematics in the Schools*, SIGACT News **29** (1998), no. 1, 8–12.

[Kos00] Martha J. Kosa, *Distributed algorithms in the discrete mathematics course (poster session)*, 5th Annual SIGCSE/SIGCUE ITiCSE Conference on Innovation and Technology in Computer Science Education (Helsinki, Finland), 2000, pp. 189–190.

[Kro07] Joan Krone, *Meeting the challenges of discrete mathematics for computer science: the power of symbols*, Journal of Computing Sciences in Colleges **23** (2007), no. 1, 31–37.

[Lev90] Gary Marc Levin, *ISETL: a language for teaching discrete mathematics (abstract)*, ACM Annual Conference on Cooperation (Washington, D.C.), 1990, p. 455.

[Li08] Yi Li, *A new approach to teaching logic in discrete mathematics*, 9th International Conference for Young Computer Scientists, November 2008, pp. 2432–2437.

[LL06] Mark D. LeBlanc and Rochelle Leibowitz, *Discrete partnership: a case for a full year of discrete math*, 37th SIGCSE Technical Symposium on Computer Science Education (Houston, Texas, USA), 2006, pp. 313–317.

[Mar84] Kenneth E. Martin, *The role of discrete structures and operations research in a computer science curriculum*, SIGCSE Bulletin **16** (1984), no. 4, 4–6.

[Mar89] William Marion, *Discrete mathematics for computer science majors - where are we? How do we proceed?*, 20th SIGCSE Technical Symposium on Computer Science Education, February 1989, pp. 273–277.

[Mar00] Bill Marion, *Discrete mathematics: support of and preparation for the study of computer science*, Seventh annual CCSC Midwestern Conference on Small Colleges, December 2000, pp. 190–199.

[Mar05] Bill Marion, *Status report on the SIGCSE committee on the implementation of a discrete mathematics course*, 36th SIGCSE Technical Symposium on Computer Science Education (St. Louis, Missouri), 2005, pp. 194–195.

[Mar06] ______, *Final oral report of the SIGCSE committee on the implementation of a discrete mathematics course*, 37th SIGCSE Technical Symposium on Computer Science Education (Houston, Texas), 2006, pp. 268–269.

[MAR07] Kirby McMaster, Nicole Anderson, and Brian Rague, *Discrete math with programming: better together*, 38th SIGCSE Technical Symposium on Computer Science Education, March 2007, pp. 100–104.

[Mau06] Ward Douglas Maurer, *An alternative to group theory in the discrete structures course*, Journal of Computing Sciences in Colleges **21** (2006), no. 5, 110–115.

[McG02] James W. McGuffee, *The discrete mathematics enhancement project*, Journal of Computing Sciences in Colleges **17** (2002), no. 5, 162–166.

[MH04] William Marion and Peter B. Henderson, *Nifty examples in discrete mathematics: tutorial presentation*, Journal of Computing Sciences in Colleges **20** (2004), no. 1, 149–149.

[MH08] ______, *Nifty examples in discrete mathematics*, Journal of Computing Sciences in Colleges **23** (2008), no. 3, 58–58.

[Mic92] James Bret Michael, *Book review of Discrete Mathematics: An Introduction for Software Engineers*, SIGPLAN Not. **27** (1992), no. 4, 10.

[Nef10] Norman Neff, *Problem-directed discrete structures course*, 41st SIGCSE Technical Symposium on Computer Science Education, March 2010, pp. 148–151.

[Noh07] Daniel Nohl, *Using an automated reasoning program as a CS application in discrete mathematics*, Journal of Computing Sciences in Colleges **22** (2007), no. 4, 7–13.

[Oli04] Carlos A. S. Oliveira, *Review of "Selected Papers in Discrete Mathematics"*, SIGACT News **35** (2004), no. 4, 11–14.

[Oxl10] Alan Oxley, *Discrete Mathematics and its Applications*, Teaching Mathematics and Its Applications: An International Journal of the IMA **29** (2010), no. 3, 155–163.

[Pag03] Rex L. Page, *Software is discrete mathematics*, 8th International Conference on Functional Programming, August 2003, pp. 79–86.

[Pio06] B. T. Pioro, *Introductory computer programming: gender, major, discrete mathematics, and calculus*, Journal of Computing Sciences in Colleges **21** (2006), no. 5, 123–129.

[Pra76] Ronald E. Prather, *Another look at the discrete structures course*, 6th SIGCSE Technical Symposium on Computer Science Education, February 1976, pp. 247–252.

[Rau08] James V. Rauff, *Review of "A Beginner's Guide to Discrete Mathematics"*, SIGACT News **39** (2008), no. 1, 16–18.

[Set09] Ben Setzer, *A lab course for discrete mathematics*, 47th Annual Southeast Regional Conference, March 2009.

[Sid86] James R. Sidbury, *A statistical analysis of the effect of discrete mathematics on the performance of computer science majors in beginning computing classes*, 17th SIGCSE Technical Symposium on Computer Science Education, February 1986, pp. 134–137.

[SS05] Amber Settle and Chad Settle, *Graduate student satisfaction with an online discrete mathematics course*, Journal of Computing Sciences in Colleges **21** (2005), no. 1, 79–87.

[Sta09] Timothy Daryl Stanley, *Using digital logic simulation as a teaching aid in discrete mathematics, hardware and operating systems, networking, computer organization and computer architecture: a workshop outline*, 10th ACM Conference on SIG-Information Technology Education (Fairfax, Virginia), 2009, pp. 1–2.

[Sut05] Klaus Sutner, *CDM: Teaching discrete mathematics to computer science majors*, J. Educ. Resour. Comput. **5** (2005), no. 2, 4.

[SW93] Dale A. Schoenefeld and Roger L. Wainwright, *Integration of discrete mathematics topics into the secondary mathematics curriculum using Mathematica: a summer institute for high school teachers*, 24th SIGCSE Technical Symposium on Computer Science Education (Indianapolis, Indiana), 1993, pp. 78–82.

[SWR05] Daniel E. Stevenson, Michael R. Wick, and Steven J. Ratering, *Steganography and cartography: interesting assignments that reinforce machine representation, bit manipulation, and discrete structures concepts*, 36th SIGCSE Technical Symposium on Computer Science Education (St. Louis, Missouri), 2005, pp. 277–281.

[TM74] J. P. Tremblay and R. Manohar, *A first course in discrete structures with applications to computer science*, 4th SIGCSE Technical Symposium on Computer Science Education, February 1974, pp. 155–160.

[Wai92] Roger L. Wainwright, *Introducing functional programming in discrete mathematics*, 23rd SIGCSE Technical Symposium on Computer Science Education (Kansas City, Missouri), 1992, pp. 147–152.

[War95] J. Stanley Warford, *An experience teaching formal methods in discrete mathematics*, SIGCSE Bulletin **27** (1995), no. 3, 60–64.

[Wax75] Jerry Waxman, *Reflections on B3, discrete structures*, SIGCSE Bulletin **7** (1975), no. 2, 51–54.

[WW06] Michael R. Wick and Paul J. Wagner, *Using market basket analysis to integrate and motivate topics in discrete structures*, 37th SIGCSE Technical Symposium on Computer Science Education (Houston, Texas, USA), 2006, pp. 323–327.

[Xiu09] Wu Xiuguo, *Discrete mathematics teaching reformation: Adding experiments*, International Workshop on Education Technology and Computer Science, vol. 2, 2009, pp. 566–569.

[YGM73] Raymond T. Yeh, Donald I. Good, and David R. Musser, *New directions in teaching the fundamentals of computer science - discrete structures and computational analysis*, 3rd SIGCSE Technical Symposium on Computer Science Education, 1973, pp. 60–67.

[ZJ10] Hongbiao Zeng and Keyu Jiang, *Teaching mathematical proofs to CS major students in the class of discrete mathematics*, Journal of Computing Sciences in Colleges **25** (2010), no. 5, 326–332.