Algorithms and Computation in Mathematics • Volume 15

Editors

Arjeh M. Cohen  Henri Cohen
David Eisenbud  Bernd Sturmfels
Classification Algorithms for Codes and Designs

With 61 Figures and 30 Tables

Springer
Authors

Petteri Kaski
Department of Computer Science and Engineering
Helsinki University of Technology
P. O. Box 5400
2015 HUT, Helsinki
Finland
e-mail: petteri.kaski@hut.fi

Patric R.J. Östergård
Department of Electrical and Communications, Engineering
Helsinki University of Technology
P.O. Box 3000
2015 HUT, Helsinki
Finland
e-mail: patric.ostergard@hut.fi

Library of Congress Control Number: 2005935445

Mathematics Subject Classification (2000): 05-02, 05Bxx, 05Cxx, 05E20, 51Exx, 68-02, 68Rxx, 94-02, 94Bxx

ISSN 1431-1550

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is
concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting,
reproduction on microfilm or in any other way, and storage in data banks. Duplication of this publication
or parts thereof is permitted only under the provisions of the German Copyright Law of September 9,
1965, in its current version, and permission for use must always be obtained from Springer. Violations are
liable for prosecution under the German Copyright Law.

Springer is a part of Springer Science+Business Media
springeronline.com
© Springer-Verlag Berlin Heidelberg 2006
Printed in The Netherlands

The use of general descriptive names, registered names, trademarks, etc. in this publication does not imply,
even in the absence of a specific statement, that such names are exempt from the relevant protective laws
and regulations and therefore free for general use.

Typesetting: by the authors and TechBooks using a Springer LaTeX macro package
Cover design: design & production GmbH, Heidelberg
Printed on acid-free paper SPIN: 11427841 46/TechBooks 5 4 3 2 1 0
## Contents

1 Introduction ................................................. 1

2 Graphs, Designs, and Codes ............................... 7  
   2.1 Graphs ................................................... 7  
   2.2 Designs ................................................... 13  
      2.2.1 Incidence Structures ................................. 14  
      2.2.2 t-Designs ........................................... 16  
      2.2.3 Balanced Incomplete Block Designs ................. 18  
      2.2.4 Steiner Triple Systems .............................. 20  
      2.2.5 Some Other Families of Designs .................... 21  
      2.2.6 Resolutions of Designs .............................. 22  
   2.3 Codes .................................................. 26  
      2.3.1 Preliminaries ......................................... 26  
      2.3.2 Equidistant Codes ................................... 29  
      2.3.3 Linear Codes ......................................... 33  
      2.3.4 Equivalence of Codes ................................ 35  
   2.4 More Combinatorial Objects ......................... 37  
      2.4.1 Orthogonal Arrays ................................... 37  
      2.4.2 Latin Squares ....................................... 39  
      2.4.3 Hadamard Matrices .................................. 43  

3 Representations and Isomorphism .......................... 47  
   3.1 Finite Groups and Group Actions ...................... 48  
      3.1.1 Permutation Groups .................................. 50  
      3.1.2 Group Actions ....................................... 53  
      3.1.3 Isomorphism and the Orbit-Stabilizer Theorem .... 55  
      3.1.4 Semidirect and Wreath Products .................... 59  
   3.2 Categories and Equivalence ............................ 64  
      3.2.1 Automorphisms and the Automorphism Group ....... 67  
      3.2.2 Functors ............................................. 70  
      3.2.3 Reconstructibility and Equivalence of Categories .. 76
VI Contents

3.3 Isomorphism Computations ........................................... 81
  3.3.1 Lexicographic Order ........................................... 82
  3.3.2 Representing Objects as Colored Graphs .................... 83
  3.3.3 Invariants and Certificates ................................... 88
  3.3.4 Subobject Invariants .......................................... 91
  3.3.5 Compounding and Iterative Refinement ....................... 95
  3.3.6 Isomorphism Problems and Tools .............................. 101

4 Isomorph-Free Exhaustive Generation ................................. 105
  4.1 Exhaustive Generation ............................................ 105
    4.1.1 Searching and Search Trees ................................. 106
    4.1.2 Backtrack Search ........................................... 109
    4.1.3 Estimating Resource Requirements ......................... 112
  4.2 Techniques for Isomorph Rejection .............................. 114
    4.2.1 Recorded Objects ........................................... 117
    4.2.2 Orderly Generation ......................................... 120
    4.2.3 Canonical Augmentation ..................................... 124
    4.2.4 Homomorphisms of Group Actions and Localization ....... 133

5 Auxiliary Algorithms .................................................... 145
  5.1 Clique Algorithms ................................................ 146
  5.2 Exact Cover Algorithms .......................................... 149
  5.3 Set Cover Algorithms ............................................. 152
  5.4 Diophantine Linear Systems of Equations ....................... 155
  5.5 Permutation Group Algorithms ................................... 159
  5.6 Isomorphism Algorithms .......................................... 164
  5.7 Distributing Computer Search .................................... 171

6 Classification of Designs .............................................. 175
  6.1 Balanced Incomplete Block Designs ............................... 175
    6.1.1 Classification Point by Point .............................. 175
    6.1.2 Testing Canonicity of Incidence Matrices ................ 182
    6.1.3 Classification Block by Block .............................. 187
    6.1.4 Isomorph Rejection for Designs Extending a Seed ......... 193
    6.1.5 Tailored Approaches ......................................... 195
    6.1.6 Results ...................................................... 197
  6.2 t-Designs .................................................................. 203
    6.2.1 Classification Point by Point .............................. 203
    6.2.2 Classification Block by Block .............................. 205
    6.2.3 Results ...................................................... 207
  6.3 Resolutions of Designs ............................................. 208
    6.3.1 Classification via the Underlying Design ................. 209
    6.3.2 Direct Classification ......................................... 210
    6.3.3 Results ...................................................... 211
  6.4 Designs with Additional Properties .............................. 215
7  Classification of Codes ........................................... 219
    7.1 Error-Correcting Codes ....................................... 219
        7.1.1 Classification via Subcodes ............................ 220
        7.1.2 Classification Codeword by Codeword .................. 223
        7.1.3 Constant Weight Codes .................................. 228
        7.1.4 Results .................................................. 229
    7.2 Covering Codes .............................................. 234
        7.2.1 Some Basic Approaches .................................. 235
        7.2.2 Stepwise Refinement of Hamming Spaces ............... 237
        7.2.3 Further Improvements ................................... 239
        7.2.4 Isomorph Rejection ...................................... 240
        7.2.5 Constant Weight Covering Codes ....................... 241
        7.2.6 Results .................................................. 242
    7.3 Linear Codes ................................................ 246
        7.3.1 Equivalence of Linear Codes .............................. 246
        7.3.2 Constructing Linear Codes via Subcodes ............... 247
        7.3.3 Isomorph Rejection using Words of Given Weights .... 249
        7.3.4 Isomorph Rejection in Projective Geometries .......... 250
        7.3.5 Implementation Issues ................................... 253
        7.3.6 Results .................................................. 255

8  Classification of Related Structures .............................. 259
    8.1 Triple Systems .............................................. 259
        8.1.1 One-Factorizations of Complete Graphs ................. 260
        8.1.2 Group Divisible Designs with Block Size 3 .......... 263
        8.1.3 Latin Squares .......................................... 264
    8.2 Hadamard Matrices .......................................... 265
    8.3 Orthogonal Arrays .......................................... 268

9  Prescribing Automorphism Groups ................................. 273
    9.1 Preliminaries ............................................... 274
    9.2 Designs ..................................................... 277
        9.2.1 The Kramer–Mesner Method ............................... 278
        9.2.2 Tactical Decompositions ................................ 281
        9.2.3 Example: STSs with Nontrivial Groups .................. 284
        9.2.4 Some Results ............................................ 289
    9.3 Codes ....................................................... 291
        9.3.1 Covering Codes .......................................... 291
        9.3.2 Error-Correcting Codes .................................. 292
        9.3.3 The Matrix Method ...................................... 293
        9.3.4 Linear Codes ............................................ 294
    9.4 Other Objects ................................................ 295
VIII Contents

10 Validity of Computational Results ...................................... 297
  10.1 Errors and Remedies ............................................... 298
  10.2 Double Counting Using the Orbit-Stabilizer Theorem .............. 299
  10.3 Double Counting by Identifying Subobjects ...................... 301
  10.4 Some Final Observations ......................................... 305

11 Computational Complexity ............................................. 307
  11.1 Preliminaries ..................................................... 307
  11.2 Completion Problems ............................................. 314
  11.3 Isomorphism Problems ............................................ 323
  11.4 Classification Problems ......................................... 330

12 Nonexistence of Projective Planes of Order 10 ..................... 339
  12.1 Projective Planes of Order 10 .................................... 339
  12.2 Codes of Designs .................................................. 341
  12.3 The Main Search .................................................. 345
    12.3.1 There are No Codewords of Weight 12 ...................... 345
    12.3.2 There are No Codewords of Weight 15 ...................... 348
    12.3.3 There are No Codewords of Weight 16 ...................... 351
    12.3.4 There are No Codewords of Weight 19 ...................... 355

References ................................................................. 365

Problem Index ............................................................ 399

Index ................................................................. 401
Preface

A new starting-point and a new method are requisite, to insure a complete [classification of the Steiner triple systems of order 15]. This method was furnished, and its tedious and difficult execution undertaken, by Mr. Cole.

F. N. Cole, L. D. Cummings, and H. S. White (1917) [129]

The history of classifying combinatorial objects is as old as the history of the objects themselves. In the mid-19th century, Kirkman, Steiner, and others became the fathers of modern combinatorics, and their work – on various objects, including (what became later known as) Steiner triple systems – led to several classification results. Almost a century earlier, in 1782, Euler [180] published some results on classifying small Latin squares, but for the first few steps in this direction one should actually go at least as far back as ancient Greece and the proof that there are exactly five Platonic solids.

One of the most remarkable achievements in the early, pre-computer era is the classification of the Steiner triple systems of order 15, quoted above. An onerous task that, today, no sensible person would attempt by hand calculation. Because, with the exception of occasional parameters for which combinatorial arguments are effective (often to prove nonexistence or uniqueness), classification in general is about algorithms and computation.

The approach of using computers to obtain mathematical results used to be controversial – and still is, in some circles – but has grown into an invaluable tool in many areas of contemporary mathematics. Notably, in the recent decades we have, for example, seen computer-aided solutions of two challenging problems: the four-color theorem and the nonexistence of a projective plane of order 10. As a matter of fact, the latter result is surveyed in Chap. 12.

Looking back at the history again, the reader may contemplate whether the tedious calculations by Mr. Cole should be trusted more than a computer search. The results in [129] are correct; a verification of this result published in 1955 was among the first scientific computations when computers became
available to researchers [247]. In the past there are many examples of manual calculations that have later been corrected in a computer search. Note, however, that the results on Steiner triple systems of order 15 in a paper published by Fisher [189] in 1940, occasionally claimed to be erroneous (incomplete, with one design missing), were never meant to form a complete classification, a fact that a careful reading of the paper reveals. Certainly, correctness of computational results is of central importance and will be discussed separately in the main text.

Mr. Cole and his co-authors found (all) 80 Steiner triple systems of order 15. But when are two objects different? This question is here answered through a proper definition (and motivation) of the concepts of equivalence and isomorphism for the objects considered.

Among the vast number of combinatorial objects, there is an idea behind the choice of classifying codes and designs and objects closely related to these. Namely, they can all be viewed as some sort of incidence structures, whereas, on the other hand, graphs, which are not considered from a classification perspective, are adjacency structures. The same applies to algebraic objects such as groups.

In studying this book, Chaps. 2, 3, and 4 are essential. Chapter 2 treats the foundations of the combinatorial objects considered; Chap. 3 the concepts of isomorphism, representations, and symmetry; and Chap. 4 presents the generic algorithms for classifying combinatorial objects. Chapter 5 contains several algorithms for solving subproblems in the classification of the objects discussed later. This chapter may be omitted unless one wants to implement these particular algorithms (but it may also be studied separately by anyone who wants to get an insight into contemporary algorithms for several important hard problems). Chapters 6 to 8 contain specific results for, respectively, designs, codes, and related structures. There is some dependency between these chapters, but hardly any overlapping. Constructions of objects with prescribed automorphism groups are studied in Chap. 9, validity of computational results is addressed in Chap. 10, and complexity issues are considered in Chap. 11. Finally, the celebrated nonexistence proof for projective planes of order 10 is surveyed in Chap. 12.

This book serves several different audiences. We have attempted to completely cover all important classification results in the areas of the book, and hope that researchers will find it an invaluable reference showing the state of the art. In fact, some of the results presented were obtained during the very process of writing this text. Most notably, these include a classification of the Steiner triple systems of order 19, the next open case after order 15, and a classification of the Steiner quadruple systems of order 16.

Due to its multidisciplinary nature, the book can be used as course material for graduate courses in computer science and discrete mathematics, as well as in coding theory (and selectively even for undergraduate courses). Elementary background knowledge in group theory will make the book more accessible, but it has been our intention to make the book as self-contained as possible.
Anyone wanting to implement classification algorithms (for any conceivable objects) will here find a basis for such work. Further research in the area is encouraged by a number of open research problems. Many of these problems are descriptions of new research ideas rather than specific problems that no one has managed to solve so far (although there are problems of this type, too). Whenever classification results are tabulated in the text, there is a smallest open case. Such instances are not explicitly stated as open problems, with a few exceptions where the instances are of greater general interest.

We also hope that the supplementary DVD with its comprehensive lists of combinatorial objects will prove a useful source for those whose main interest is in studying and using the classified objects for various purposes.

Last but not least, the following colleagues contributed to this project via valuable discussions and remarks: Gunnar Brinkmann, Harri Haanpää, Tommi Junttila, Gerzson Kéri, Clement Lam, Ilkka Niemelä, Pekka Orponen, Alexander Rosa, and Gordon Royle. Needless to say, we are more than grateful for this assistance. The suggestions of the anonymous reviewers were also of great help in preparing the final version of this book. The work was supported in part by the Helsinki Graduate School in Computer Science and Engineering (HeCSE), the Foundation of Technology (Tekniikan Edistämissäätiö), the Nokia Foundation, and by the Academy of Finland under Grants No. 44517, No. 100500, No. 107493, and No. 202315.

Espoo, Petteri Kaski
September 2005

Patric Östergård