Copyright is owned by the Author of the thesis. Permission is given for a copy to be downloaded by an individual for the purpose of research and private study only. The thesis may not be reproduced elsewhere without the permission of the Author.
On Fast and Space-Efficient Database Normalization

A dissertation presented in partial fulfilment
of the requirements for the degree of

Doctor of Philosophy
in
Information Systems

at Massey University, Palmerston North, New Zealand

Henning Koehler

2007
Abstract

A common approach in designing relational databases is to start with a relation schema, which is then decomposed into multiple subschemas. A good choice of subschemas can often be determined using integrity constraints defined on the schema.

Two central questions arise in this context. The first issue is what decompositions should be called “good”, i.e., what normal form should be used. The second issue is how to find a decomposition into the desired form.

These question have been the subject of intensive research since relational databases came to life. A large number of normal forms have been proposed, and methods for their computation given. However, some of the most popular proposals still have problems:

- algorithms for finding decompositions are inefficient
- dependency preserving decompositions do not always exist
- decompositions need not be optimal w.r.t. redundancy/space/update anomalies

We will address these issues in this work by

- designing efficient algorithms for finding dependency preserving decompositions
- proposing a new normal form which minimizes overall storage space

This new normal form is then characterized syntactically, and shown to extend existing normal forms.
Acknowledgement

I would like to thank my supervisor Sven Hartmann for his constant support, ranging from fruitful discussions and extensive proofreading to help with administrative hurdles and moral support.

Furthermore, my thanks go to my co-supervisor Klaus-Dieter Schewe, and to my colleagues Sebastian Link and Markus Kirchberg, who helped and supported me in various forms.

I dedicate this thesis to my parents, Klaus and Waltraud Koehler, and to my partner Jane Zhao.
## Contents

1 **Introduction**  
  1.1 Relational Databases and Dependencies .......................... 3  
  1.2 Normal Forms .................................................. 5  
  1.3 Contributions and Outline ...................................... 7

2 **Linear Resolution and Faithful BCNF Decomposition**  
  2.1 Linear Resolution .................................................. 11  
    2.1.1 The Basic Algorithm ........................................ 13  
    2.1.2 Improvements and Complexity Analysis ....................... 17  
    2.1.3 Polynomial Cases ........................................... 20  
    2.1.4 Updating the Atomic Closure ................................ 22  
    2.1.5 Other Applications .......................................... 24  
    2.1.6 Related work ................................................ 25  
  2.2 Faithful BCNF Decomposition .................................... 26  
    2.2.1 The Basic Algorithm ........................................ 26  
    2.2.2 Improvements and Complexity Analysis ....................... 29  
    2.2.3 Partial Determination Cycles ................................ 31  
    2.2.4 Related Work ................................................ 36  
  2.3 Complex-Valued Databases ...................................... 36  
    2.3.1 Introduction ................................................ 36  
    2.3.2 Representation Basis ....................................... 40  
    2.3.3 Linear Resolution .......................................... 42  
    2.3.4 Faithful BCNF-Decomposition ................................. 46  
    2.3.5 Lossless Decomposition ...................................... 48  
    2.3.6 Testing for BCNF ............................................. 54

3 **Canonical Covers** ........................................... 55  
  3.1 Hypergraph Decomposition ..................................... 57  
    3.1.1 Autonomous Sets .......................................... 57  
    3.1.2 Superedges and Partial Superedges .......................... 62  
    3.1.3 Computing Autonomous Sets ................................ 63  
  3.2 Computing all Canonical Covers ................................ 65  
    3.2.1 Partial Covers ............................................. 65  
    3.2.2 Relative Covers ............................................ 67  
    3.2.3 Implication Dependencies ................................... 69  
    3.2.4 The Algorithm .............................................. 72  
    3.2.5 Improvements and Complexity Analysis ....................... 74