The Domain Theory in Computer Science

Edited by: Jovan Pehcevski

AP | ARCLER PRESS

The Domain Theory in Computer Science

The Domain Theory in Computer Science

Edited by:

Jovan Pehcevski

ARCLER
PRESS

www.arclerpress.com

The Domain Theory in Computer Science

Jovan Pehcevski

Arcler Press
224 Shoreacres Road
Burlington, ON L7L 2H2
Canada
www.arclerpress.com
Email: orders@arclereducation.com

e-book Edition 2023

ISBN: 978-1-77469-677-4 (e-book)

This book contains information obtained from highly regarded resources. Reprinted material sources are indicated. Copyright for individual articles remains with the authors as indicated and published under Creative Commons License. A Wide variety of references are listed. Reasonable efforts have been made to publish reliable data and views articulated in the chapters are those of the individual contributors, and not necessarily those of the editors or publishers. Editors or publishers are not responsible for the accuracy of the information in the published chapters or consequences of their use. The publisher assumes no responsibility for any damage or grievance to the persons or property arising out of the use of any materials, instructions, methods or thoughts in the book. The editors and the publisher have attempted to trace the copyright holders of all material reproduced in this publication and apologize to copyright holders if permission has not been obtained. If any copyright holder has not been acknowledged, please write to us so we may rectify.

Notice: Registered trademark of products or corporate names are used only for explanation and identification without intent of infringement.

ISBN: 978-1-77469-440-4 (Hardcover)

DECLARATION

Some content or chapters in this book are open access copyright free published research work, which is published under Creative Commons License and are indicated with the citation. We are thankful to the publishers and authors of the content and chapters as without them this book wouldn’t have been possible.

ABOUT THE EDITOR

Jovan currently works as a presales Technology Consultant at Dell Technologies. He is a result-oriented technology leader with demonstrated subject matter expertise in planning, architecting and managing ICT solutions to reflect business objectives and achieve operational excellence. Jovan has broad deep technical knowledge in the fields of data center and big data technologies, combined with consultative selling approach and exceptional client-facing presentation skills. Before joining Dell Technologies in 2017, Jovan spent nearly a decade as a researcher, university professor and IT business consultant. In these capacities, he served as a trusted advisor to a multitude of customers in financial services, health care, retail, and academic sectors. He holds a PhD in Computer Science from RMIT University in Australia and worked as a post-doctoral visiting scientist at the renowned INRIA research institute in France. He is a proud father of two, an aspiring tennis player, and an avid Science Fiction/Fantasy book reader.”

TABLE OF CONTENTS

Section 1: Partial Orders and Groups

Section 2: Power Domains and Metrics Concepts

Section 3: Recursive Functions and Data Types (Binary Trees)

Section 4: Algebraicity and Boolean Algebras

LIST OF CONTRIBUTORS

Xiaoji Liu
School of Mathematics and Physics, Guangxi University for Nationalities, Nanning 530006, China

Fang Gui
School of Mathematics and Physics, Guangxi University for Nationalities, Nanning 530006, China

Yanisa Chaiya
Department of Mathematics, Chiang Mai University, Chiang Mai 50200, Thailand

Preeyanuch Honyam
Department of Mathematics, Chiang Mai University, Chiang Mai 50200, Thailand

Jintana Sanwong
Department of Mathematics, Chiang Mai University, Chiang Mai 50200, Thailand

Hacı Aktaş
Department of Mathematics, Faculty of Science, Erciyes University, 38039 Kayseri, Turkey

Şerif Özlü
Kilis 7 Aralık University, 79000 Kilis, Turkey

M. R. Darafsheh
Center for Theoretical Physics and Mathematics, AEOI, Iran
Department of Mathematics and Computer Sciences, Faculty of Sciences, University of Tehran, Iran

G. R. Rezaeezadeh
Department of Mathematics, Faculty of Basic Sciences, University of Tarbiat Moddares, Iran

Yayan Yuan
College of Mathematics and Information Science, Henan Normal University, Xinxiang, Henan 453007, China

Jibo Li
College of Mathematics and Information Science, Henan Normal University, Xinxiang, Henan 453007, China

Akbar Dehghan Nezhad
Faculty of Mathematics, Yazd University, Yazd, Iran

Zohreh Aral
Faculty of Mathematics, Yazd University, Yazd, Iran

Shasha Huang
College of Mathematics and Information, North China University of Water Resources and Electric Power, Zhengzhou, Henan 450045, China

Jing Hao
College of Mathematics and Information, Henan University of Economics and Law, Zhengzhou, Henan 450000, China

Dang Luo
College of Mathematics and Information, North China University of Water Resources and Electric Power, Zhengzhou, Henan 450045, China

Ray-Ming Chen
School of Mathematics and Statistics, Baise University, 21, Zhongshan No. 2 Road, Guangxi Province, China

Hua Li
Department of Information Engineering, Hangzhou Polytechnic College, Hangzhou, China.

Joonghoon Shin
Department of Forest Engineering, Resources and Management, Oregon State University, Peavy Hall, Corvallis, OR, USA.

Hailemariam Temesgen
Department of Forest Engineering, Resources and Management, Oregon State University, Peavy Hall, Corvallis, OR, USA.

Abdul Rauf Nizami
Division of Science and Technology, University of Education, Lahore, Pakistan

Mobeen Munir
Division of Science and Technology, University of Education, Lahore, Pakistan

Umer Saleem
Division of Science and Technology, University of Education, Lahore, Pakistan

Ansa Ramzan
Division of Science and Technology, University of Education, Lahore, Pakistan

Zhenyu Liu
College of Computer Science and Engineering, Northeastern University, Shenyang 110819, China
Department of Computer Science and Technology, Dalian Neusoft University of Information, Dalian, Liaoning, China

Tao Wen
College of Computer Science and Engineering, Northeastern University, Shenyang 110819, China
Department of Computer Science and Technology, Dalian Neusoft University of Information, Dalian, Liaoning, China

Wei Sun
Department of Computer Science and Technology, Dalian Neusoft University of Information, Dalian, Liaoning, China

Qilong Zhang
College of Computer Science and Engineering, Northeastern University, Shenyang 110819, China

B. G. Konopelchenko
Dipartimento di Fisica, Universita del Salento and INFN, Sezione di Lecce, 73100 Lecce, Italy

George Szeto
Department of Mathematics, Bradley University, USA

Lianyong Xue
Department of Mathematics, Bradley University, USA

Vladimir Leontiev
Moscow State University, Moscow, Russia.

Garib Movsisyan
BIT Group, Moscow, Russia.

Zhirayr Margaryan
Yerevan State University, Yerevan, Armenia.

Fadi A. Aloul
Department of Computer Science and Engineering, American University of Sharjah, United Arab Emirates

Mohamed El-Tarhuni
Department of Electrical Engineering, American University of Sharjah, United Arab Emirates

LIST OF ABBREVIATIONS

AWGN	Additive white Gaussian noise
ABA	Area-based approach
BA	Basal area
BER	Bit error rate
SAT	Boolean Satisfiability
CHM	Canopy Height Model
CS	Central system
CNF	Conjunctive normal form
DLL	Davis-Logemann-Loveland
DNN	Deep neural network
DDAs	Deformation driving algebras
DBH	Diameter at breast height
FDT	Fisher’s decision tree
ITD	Individual tree detection
ILP	Integer linear programming
LiDAR	Light detection and ranging
LP	Logic programming
PB	Pseudo-Boolean
RMSE	Root mean squared error
SNRc	Signal-to-noise ratio per chip
SADT	Simulated Annealing Decision Tree
SPH	Stems per hectare
OWL	Web Ontology Language

PREFACE

Domain theory is a branch of mathematics that studies special types of partially ordered sets, commonly known as domains. Consequently, domain theory can be considered as a branch of order theory. The field has major applications in computing, where it is used to determine denotational semantics, especially for functional programming languages. The domain theory very generally formalizes intuitive ideas about convergence, and convergence is closely related to topology.

The domain theory as a field of mathematics deals with information and computation. It treats the idea of states that contain partial information, which are ordered by the quantity of information they contain. As a result of that, the developed mathematical theory has a large number of applications in many computer science topics, particularly in the semantics of programming languages. In this book, we will cover both the mathematical theory and the applications of domain theory. The main themes include: the topics of approximation and continuity defined in the domain theory and their important connections with topology; the bases for computation with infinite objects; the development of a rich theory of fix-points, as a foundation for recursive definitions; the development of a rich set of data type constructions, and recursive definitions of domains themselves; and power domains, to support ideas of non-deterministic and probabilistic computation.

This edition covers different topics from domain theory in computer science, including: partial orders and groups, power domains and metrics, recursive data types (binary trees) and algebraicity and Boolean algebras.

Section 1 focuses on partial orders and groups, describing characterizations and properties of a new partial order, natural partial orders on transformation semigroups with fixed sets, cyclic soft groups and their applications on groups, factorization of groups involving symmetric and alternating groups.

Section 2 focuses on power domains and metrics, describing FS+ domains, topology of GB-metric spaces, incoherency problems in a combination of description logics and rules, metrics for multiset-theoretic subgraphs.

Section 3 focuses on recursive data types (binary trees), describing binary tree's recursion traversal algorithm and its improvement, the design of the minimum spanning tree algorithms, generating tree-lists by fusing individual tree detection and nearest neighbor imputation using airborne LIDAR data, a recursive approach to the Kauffman bracket, a novel multiway splits decision tree for multiple types of data.

Section 4 focuses on algebraicity and Boolean algebras, describing the deformation theory of structure constants for associative algebras, the Boolean algebra and central Galois algebras, on addition of sets in Boolean space, algebra and geometry of sets in Boolean space, multipath detection using Boolean satisfiability techniques.

SECTION 1: PARTIAL ORDERS AND GROUPS

Chapter

1

SOME CHARACTERIZATIONS AND PROPERTIES OF A NEW PARTIAL ORDER

Xiaoji Liu and Fang Gui

School of Mathematics and Physics, Guangxi University for Nationalities, Nanning 530006, China

ABSTRACT

On the basis of Löwner partial order and core partial order, we introduce a new partial order: LC partial order. By applying matrix decomposition, core inverse, core partial order, and Löwner partial order, we give some characteristics of LC partial order, study the relationship between LC partial order and Löwner partial order under constraint conditions, and illustrate its differences with some classical partial orders, such as minus, CL, and GL partial orders.

Citation: Xiaoji Liu and Fang Gui, "Some Characterizations and Properties of a New Partial Order", Journal of Mathematics, volume 2020, article ID 3215038, https://doi.org/10.1155/2020/3215038.

INTRODUCTION

A binary relation on a nonempty set is called partial order if it satisfies reflexivity, transitivity, and antisymmetry. In recent years, more and more mathematicians have turned their attention to matrix partial ordering: Hauke and Markiewicz [1] introduced generalized Löwner order by polar decomposition; Baksalary and Trenkler [2] studied the core partial order of complex matrices; and Ando [3] studied the square inequality and strong order relation on Hilbert space. In this paper, a new partial order is introduced on the complex matrix set by matrix decomposition and Löwner and core partial orders.

First, we use the following notations. The symbol $\mathbb{C}_{m,n}$ denotes the set of m × n matrices with complex entries. $\mathbb{C}_n^H$ and $\mathbb{C}_n^{\geq}$ denote the set of n × n Hermitian matrices and Hermitian nonnegative definite matrices, respectively. The symbols A^*, $\mathscr{R}(A)$, and rk(A) represent the conjugate transpose, range space, and rank of $A \in \mathbb{C}_{m,n}$, respectively. The symbol ρ(A) represents spectral radius of $A \in \mathbb{C}_{n,n}$.

The smallest positive integer k for which $\mathrm{rk}(A^{k+1}) = \mathrm{rk}(A^k)$ is called the index of $A \in \mathbb{C}_{n,n}$ and is denoted by Ind(A). When A is nonsingular, the index of A is 0. The symbol $\mathbb{C}_n^{CM}$ stands for a set of n × n matrices of index less than or equal to 1.

Definition 1 (see [4, 5]). Let $A \in \mathbb{C}_{m,n}$. If $X \in \mathbb{C}_{n,m}$ satisfies the following equations:

$$\begin{aligned}&(1) AXA = A,\\&(2) XAX = X,\\&(3) (AX)^* = AX,\\&(4) (XA)^* = XA,\end{aligned} \quad (1)$$

then X is said to be the Moore–Penrose inverse of matrix A, and X is unique. It is usually defined by $X = A^\dagger$.

Furthermore, we denote $P_A = AA^\dagger$.

Definition 2 (see [4, 5]). Let $A \in \mathbb{C}_n^{CM}$. If $X \in \mathbb{C}_{n,n}$ satisfies the following equations:

$$\begin{aligned}&(1)\, AXA = A,\\ &(2')\, AX^2 = X,\\ &(3)\, (AX)^* = AX,\end{aligned} \qquad (2)$$

then X is said to be the core inverse of matrix A, and X is unique. It is usually defined by X = $A^{\#}$.

Lemma 1 (see [2]). *Let* $A \in \mathbb{C}_{n,n}$ *with rk(A) = r, then A can be expressed as*

$$A = U\begin{bmatrix} \Sigma K & \Sigma L \\ 0 & 0 \end{bmatrix}U^*, \qquad (3)$$

where $U \in \mathbb{C}_{n,n}$ *is unitary,* $\Sigma = diag(\alpha_1, \alpha_2, \ldots, \alpha_r)$ *is the diagonal matrix of singular values of* $A, \alpha_1 \geq \alpha_2 \geq \cdots \geq \alpha_r > 0,$ $K \in \mathbb{C}_{r,r}$ *and* $L \in \mathbb{C}_{r,n-r}$ *satisfy* $KK^* + LL^* = I_r,$ *and* $\rho(K) \leq 1$.

Furthermore, when rk(A) = rk(A^2), K is nonsingular, and

$$A^{\textcircled{\#}} = U\begin{bmatrix} K^{-1}\Sigma^{-1} & 0 \\ 0 & 0 \end{bmatrix}U^*. \qquad (4)$$

We give the definitions of some classical partial orders such as minus, Löwner, sharp, core, and C-N partial orders [2, 6–8].

- $A \underset{-}{\leq} B, A, B \in \mathbb{C}_{m,n}, \mathrm{rk}(B) - \mathrm{rk}(A) = \mathrm{rk}(B - A)$
- $A \overset{L}{\leq} B, A, B \in \mathbb{C}_{n,n}, B - A \geq 0$
- $A \overset{\#}{\leq} B, A, B \in \mathbb{C}_n^{CM}, A^{\#}A = A^{\#}B$ and $AA^{\#} = BA^{\#}$
- $A \overset{\textcircled{\#}}{\leq} B, A, B \in \mathbb{C}_n^{CM}, A^{\textcircled{\#}}A = A^{\textcircled{\#}}B$ and $AA^{\textcircled{\#}} = BA^{\textcircled{\#}}$
- $A \leq^{\#,-} B\colon A, B \in \mathbb{C}_{n,n}, A_1 \overset{\#}{\leq} B_1$, and $A_2 \leq B_2$, in which A = A_1 + A_2 and B = B_1 + B_2 are the C-N decompositions of A and B, respectively

Matrix decomposition is an important tool to study the theory of matrix partial orders. It is used to discuss some characteristics and properties of matrix partial orders and then to establish some matrix partial orders. For example, C-N partial order and core partial order are based on C-N decomposition and core decomposition, respectively [2, 7, 9].

A particular concern is the generalized polar decomposition ([3], Chapter 6, Theorem 7). Let $A \in \mathbb{C}_{m,n}$. Then, A can be written as

$$A = G_A E_A = E_A H_A, \tag{5}$$

where $E_A \in \mathbb{C}_{m,n}$ is a partial isometry, i.e., $E_A^* = E_A^\dagger$, $G_A \in \mathbb{C}_{m,m}$, and $H_A \in \mathbb{C}_{n,n}$ are Hermitian nonnegative definite matrices. The matrices E_{Δ}, G_{Δ}, and H_A are uniquely determined by $\mathscr{R}(E_A) = \mathscr{R}(G_A)$ and $\mathscr{R}(E_A^*) = \mathscr{R}(H_A)$, in which $G_A = |A| = (AA^*)^{1/2}$, $H_A = |A^*| = (A^*A)^{1/2}$, and $E_A = G_A^\dagger A = AH_A^\dagger$.

Based on the generalized polar decomposition, Hauke and Markiewicz [1] introduced the GL partial order: let $A, B \in \mathbb{C}_{m,n}$, and $A = G_{\Delta}E_{\Delta}$ and $B = G_BE_B$ be their polar decompositions, where $\mathscr{R}(E_A) = \mathscr{R}(G_A)$ and $\mathscr{R}(E_B) = \mathscr{R}(G_B)$. Then,

$$\begin{aligned} A \overset{GL}{\leq} B &\Longleftrightarrow E_A \leq^* E_B \text{ and } G_A \overset{L}{\leq} G_B, \\ &\Longleftrightarrow E_A \overset{*}{\leq} E_B \text{ and } H_A \overset{L}{\leq} H_B. \end{aligned} \tag{6}$$

After that, Wang and Liu [10] made the polar-like decomposition: let $A \in \mathbb{C}_{m,n}$. Then, A can be written as

$$A = G_A^{\frac{1}{2}} E_A H_A^{\frac{1}{2}}, \tag{7}$$

where E_A, G_A, and H_A are given in ([3], Chapter 6, Theorem 7). On the basis of the polar-like decomposition, the WL partial order [10] is defined as

$$A \overset{WL}{\leq} B \Longleftrightarrow G_A^{\frac{1}{2}} \overset{L}{\leq} G_B^{\frac{1}{2}},\ E_A \overset{*}{\leq} E_B, \text{ and } H_A^{\frac{1}{2}} \overset{L}{\leq} H_B^{\frac{1}{2}}, \tag{8}$$

in which $A = G_A^{1/2} E_A H_A^{1/2}$ and $B = G_B^{1/2} E_B H_B^{1/2}$ are the polar-like decompositions of A and B, respectively. In [11], Wang and Liu introduced the CL partial order:

$$A \overset{CL}{\leq} B \Longleftrightarrow A^{\textcircled{\#}} A \overset{\textcircled{\#}}{\leq} B^{\textcircled{\#}} B \text{ and } A^2 A^{\textcircled{\#}} \leq^L B^2 B^{\textcircled{\#}}, \tag{9}$$

in which $A, B \in \mathbb{C}_n^{CM}$. It is also worthy to note that, under certain conditions, CL partial order is equivalent to GL and Löwner partial orders [1, 11]. In this paper, we consider matrices over complex fields. Based on the above research and inspired by generalized polar decomposition, we introduce a new partial order on the set of core matrices by using Löwner partial order and core partial order. It is dominated neither by minus partial order nor by Löwner partial order. Interestingly, under some conditions, LC partial order is equivalent to CL, GL, and Löwner partial orders.

MAIN RESULT

In this section, we introduce a new partial order on $\mathbb{C}_n^{CM}$, derive some of its characteristics, consider its relationship with Löwner partial order under some constraints, and illustrate its difference from other partial orders with examples.

Let $A \in \mathbb{C}_n^{CM}, P_A = AA^{\dagger},$ and E_A be as given in (5). Then,

$$A = F_A Q_A, \qquad (10)$$

where

$$F_A = U\begin{bmatrix} K & 0 \\ 0 & 0 \end{bmatrix}U^*,$$
$$Q_A = U\begin{bmatrix} K^{-1}\Sigma K & K^{-1}\Sigma L \\ 0 & 0 \end{bmatrix}U^*. \qquad (11)$$

We call (10) the P-2 expression of A. Furthermore, it is easy to check that

$$F_A = E_A^{\textcircled{\#}}\left(A^{\textcircled{\#}}\right)^{\textcircled{\#}},$$
$$Q_A = A^{\textcircled{\#}}E_A A, \qquad (12)$$

$$F_A = AQ_A^{\textcircled{\#}},$$
$$A = Q_A Q_A^{\textcircled{\#}} A, \qquad (13)$$

$F_A \in \mathbb{C}_{n,n}$ is a EP-matrix, $\rho(F_A) \le 1,$ and $Q_A \in \mathbb{C}_{n,n}$.

Let $A, B \in \mathbb{C}_n^{CM}$. Consider the binary operation:

$$A \overset{LC}{\le} B \Leftrightarrow F_A \overset{L}{\le} F_B \text{ and } Q_A \overset{\textcircled{\#}}{\le} Q_B, \qquad (14)$$

in which $A = F_AQ_A$ and $B = F_BQ_B$ are the P-2 expressions of A and B, respectively. In the following Theorem 1, we check that the binary operation is a partial order and call it the LC partial order.

Theorem 1. *The binary operation (14) is a partial order on* $\mathbb{C}_n^{CM}$.

Proof.

(1) Reflexive: let $A \in \mathbb{C}_n^{CM}$, and $A = F_AQ_A$ is the P-2 expression of A. We have

$$\begin{gathered} F_A \overset{L}{\leq} F_A, \\ Q_A \overset{\textcircled{\#}}{\leq} Q_A. \end{gathered} \qquad (15)$$

So, $A \leq^{LC} A$.

(2) Antisymmetric: let $A, B \in \mathbb{C}_n^{CM}$, $A = F_AQ_A$ and $B = F_BQ_B$ be their P-2 expressions. If $A \leq^{LC} B$ and $B \leq^{LC} A$, that is, if

$$\begin{cases} F_A \overset{L}{\leq} F_B \text{ and } Q_A \overset{\textcircled{\#}}{\leq} Q_B, \\ F_B \overset{L}{\leq} F_A \text{ and } Q_B \overset{\textcircled{\#}}{\leq} Q_A. \end{cases} \qquad (16)$$

Then, A = B.

(3) Transitive: suppose that $A \leq^{LC} B$ and $B \leq^{LC} C$, that is, suppose

$$\begin{cases} F_A \overset{L}{\leq} F_B \text{ and } F_B \overset{L}{\leq} F_C, \\ Q_A \overset{\textcircled{\#}}{\leq} Q_B \text{ and } Q_B \overset{\textcircled{\#}}{\leq} Q_C. \end{cases} \qquad (17)$$

From the transitivity of the Löwner partial order and core partial order, we have $F_A \overset{L}{\leq} F_C$ and $Q_A \overset{\textcircled{\#}}{\leq} Q_C$, that is, $A \overset{LC}{\leq} C$. By (1), (2), and (3), we know that the binary operation (14) is a partial order on $\mathbb{C}_n^{CM}$.

Next, we give the characteristics of the LC partial order.

Theorem 2. *Let* $A, B \in \mathbb{C}_n^{CM}$, $rk(B) \geq rk(A) \geq 1$, $A \leq^{LC} B$, *and* $A = F_AQ_A$, *and* $B = F_BQ_B$ *be the P-2 expressions of A and B, respectively. Then, there exists a unitary matrix U such that*

$$F_A = U\begin{bmatrix} A_1T_{A1}^{-1} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}U^*,$$
$$Q_A = U\begin{bmatrix} T_{A1} & T_{A2} & T_{A3} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}U^*,$$
$$F_B = U\begin{bmatrix} B_1T_{A1}^{-1} & \left(B_4T_{A1}^{-1}\right)^* & 0 \\ B_4T_{A1}^{-1} & -B_4T_{A1}^{-1}T_{A2}T_{A4}^{-1} + B_5T_{A4}^{-1} & 0 \\ 0 & 0 & 0 \end{bmatrix}U^*,$$
$$Q_B = U\begin{bmatrix} T_{A1} & T_{A2} & T_{A3} \\ 0 & T_{A4} & T_{A5} \\ 0 & 0 & 0 \end{bmatrix}U^*,$$
$$B = U\begin{bmatrix} B_1 & B_2 & B_3 \\ B_4 & B_5 & B_6 \\ 0 & 0 & 0 \end{bmatrix}U^*, \tag{18}$$

where $A_1, B_1 \in \mathbb{C}_{rk(A),rk(A)}$, *$T_{A1}$ and T_{A4} are both nonsingular, T_{A2}, T_{A3}, T_{A4}, and T_{A5} are arbitrary matrices with appropriate sizes,* $\rho(A_1T_{A1}^{-1}) \leq 1$,

$$\rho\left(\begin{bmatrix} B_1T_{A1}^{-1}, & \left(B_4T_{A1}^{-1}\right)^*, \\ B_4T_{A1}^{-1}, & -B_4T_{A1}^{-1}T_{A2}T_{A4}^{-1} + B_5T_{A4}^{-1}, \end{bmatrix}\right) \leq 1,$$
$$\begin{bmatrix} (B_1 - A_1)T_{A1}^{-1}, & \left(B_4T_{A1}^{-1}\right)^*, \\ B_4T_{A1}^{-1}, & -B_4T_{A1}^{-1}T_{A2}T_{A4}^{-1} + B_5T_{A4}^{-1}, \end{bmatrix} \geq 0. \tag{19}$$

Proof. Let $A \in \mathbb{C}_n^{CM}$. Applying (11), we get A = F_AQ_A, where

$$F_A = U\begin{bmatrix} K_A & 0 \\ 0 & 0 \end{bmatrix}U^*,$$
$$Q_A = U\begin{bmatrix} K_A^{-1}\Sigma_AK_A & K_A^{-1}\Sigma_AL_A \\ 0 & 0 \end{bmatrix}U^*. \tag{20}$$

Furthermore, we write

$$Q_A = U\begin{bmatrix} T_{A1} & T_{A2} & T_{A3} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}U^*,$$
$$F_A = U\begin{bmatrix} A_1T_{A1}^{-1} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}U^*, \tag{21}$$

where $T_{A1} = K_A^{-1}\Sigma_AK_A, A_1 = K_AT_{A1}$ and $T_{A2} \in \mathbb{C}_{\mathrm{rk}(A),\mathrm{rk}(B)-\mathrm{rk}(A)}$.

Let $B \in \mathbb{C}_n^{CM}, \mathrm{rk}(B) \geq \mathrm{rk}(A) \geq 1$, and B = F_BQ_B be the P-2 expression of B. Since A $\leq^{LC}$B, by applying (14), we have $Q_A \overset{\oplus}{\leq} Q_B$. Then, by applying ([2], Lemma 3), we get

$$Q_B = U\begin{bmatrix} T_{A1} & T_{A2} & T_{A3} \\ 0 & T_{A4} & T_{A5} \\ 0 & 0 & 0 \end{bmatrix}U^*, \tag{22}$$

where T_{A4} is nonsingular. It is easy to check that

$$Q_B^{\circledast} = U\begin{bmatrix} T_{A1}^{-1} & -T_{A1}^{-1}T_{A2}T_{A4}^{-1} & 0 \\ 0 & T_{A4}^{-1} & 0 \\ 0 & 0 & 0 \end{bmatrix}U^*. \tag{23}$$

We write

$$B = U\begin{bmatrix} B_1 & B_2 & B_3 \\ B_4 & B_5 & B_6 \\ B_7 & B_8 & B_9 \end{bmatrix}U^*. \tag{24}$$

Then, applying (13), we get $B_7 = 0$, $B_8 = 0$, $B_9 = 0$, and

$$\begin{aligned} F_B &= BQ_B^{\circledast} = U\left(U^*BU\begin{bmatrix} T_{A1}^{-1} & -T_{A1}^{-1}T_{A2}T_{A4}^{-1} & 0 \\ 0 & T_{A4}^{-1} & 0 \\ 0 & 0 & 0 \end{bmatrix}\right)U^* \\ &= U\begin{bmatrix} B_1T_{A1}^{-1} & -B_1T_{A1}^{-1}T_{A2}T_{A4}^{-1} + B_2T_{A4}^{-1} & 0 \\ B_4T_{A1}^{-1} & -B_4T_{A1}^{-1}T_{A2}T_{A4}^{-1} + B_5T_{A4}^{-1} & 0 \\ 0 & 0 & 0 \end{bmatrix}U^*. \end{aligned} \tag{25}$$

Since $F_A \overset{L}{\le} F_B$ and $F_B - F_A \ge 0$, then we get $B_4T_{A1}^{-1} = (-B_1T_{A1}^{-1}T_{A2}T_{A4}^{-1} + B_2T_{A4}^{-1})^*$, and

$$F_B - F_A = U\begin{bmatrix} (B_1 - A_1)T_{A1}^{-1} & \left(B_4T_{A1}^{-1}\right)^* & 0 \\ B_4T_{A1}^{-1} & -B_4T_{A1}^{-1}T_{A2}T_{A4}^{-1} + B_5T_{A4}^{-1} & 0 \\ 0 & 0 & 0 \end{bmatrix}U^* \ge 0, \tag{26}$$

that is,

$$\begin{bmatrix} (B_1 - A_1)T_{A1}^{-1} & \left(B_4T_{A1}^{-1}\right)^* \\ B_4T_{A1}^{-1} & -B_4T_{A1}^{-1}T_{A2}T_{A4}^{-1} + B_5T_{A4}^{-1} \end{bmatrix} \ge 0. \tag{27}$$

Since $\rho(K_A) \le 1$, $\rho(K_B) \le 1$, F_A and F_B are EP, and $F_A \overset{L}{\le} F_B$, we obtain $\rho(A_1T_{A1}^{-1}) \le 1$ and $\rho\left(\begin{bmatrix} B_1T_{A1}^{-1} & (B_4T_{A1}^{-1})^* \\ B_4T_{A1}^{-1} & -B_4T_{A1}^{-1}T_{A2}T_{A4}^{-1} + B_5T_{A4}^{-1} \end{bmatrix}\right) \le 1$. Therefore, we have (18).

Next, we use Examples 1 and 2 to explain the difference between LC partial order and minus (Löwner, GL, or CL) partial order.

Example 1. Let

$$A = U\begin{bmatrix} \frac{1}{2} & 0 & 0 & \frac{1}{2} \\ 0 & \frac{1}{3} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}U^*,$$

$$B = U\begin{bmatrix} \frac{3}{4} & 0 & 0 & \frac{3}{4} \\ 0 & \frac{2}{3} & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}U^*,$$

$$U = \begin{bmatrix} \frac{1}{2} & 0 & \frac{\sqrt{3}}{2} & 0 \\ 0 & \frac{1}{3} & 0 & \frac{2\sqrt{2}}{3} \\ \frac{\sqrt{3}}{2} & 0 & -\frac{1}{2} & 0 \\ 0 & \frac{2\sqrt{2}}{3} & 0 & -\frac{1}{3} \end{bmatrix}. \tag{28}$$

Then, rk(B) = 3, rk(A) = 2, and

$$Q_A = U\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}U^*,$$

$$Q_B = U\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}U^*,$$

$$Q_A^{\oplus} = U\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}U^*,$$

$$\begin{aligned}
Q_B^{\textcircled{\#}} &= U\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & \frac{1}{3} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}U^*, \\
F_A &= U\begin{bmatrix} \frac{1}{2} & 0 & 0 & 0 \\ 0 & \frac{1}{6} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}U^*, \\
F_B &= U\begin{bmatrix} \frac{3}{4} & 0 & 0 & 0 \\ 0 & \frac{1}{3} & 0 & 0 \\ 0 & 0 & \frac{1}{6} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}U^*, \\
B - A &= U\begin{bmatrix} \frac{1}{4} & 0 & 0 & \frac{1}{4} \\ 0 & \frac{1}{3} & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}U^*.
\end{aligned} \tag{29}$$

Since

$$F_B - F_A = U\begin{bmatrix} \frac{1}{4} & 0 & 0 & 0 \\ 0 & \frac{1}{6} & 0 & 0 \\ 0 & 0 & \frac{1}{6} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}U^* \geq 0 \Rightarrow F_A \overset{L}{\leq} F_B,$$

$$Q_A Q_A^{\textcircled{\#}} = U \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} U^* = Q_B Q_A^{\textcircled{\#}},$$

$$Q_A^{\textcircled{\#}} Q_A = U \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} U^* = Q_A^{\textcircled{\#}} Q_B, \tag{30}$$

then $Q_A \overset{\textcircled{\#}}{\leq} Q_B$. We have $A \overset{LC}{\leq} B$.

(1) Since rk(B − A) = 3 ≠ rk(B) − rk(A) = 1, A is not below B under the minus partial order

(2) Since B − A is not a positive semi-definite matrix, A is not below B under the Löwner partial order

Example 2. Let

$$A = U \begin{bmatrix} \frac{1}{5} & \frac{1}{10} & \frac{405}{908} & -\frac{366}{11279} \\ \frac{1}{15} & \frac{1}{15} & -\frac{425}{19646} & \frac{676}{1464} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} U^*,$$

$$B = U \begin{bmatrix} \frac{1}{3} & \frac{2}{15} & \frac{1700}{1223} & -\frac{1429}{1656} \\ \frac{1}{30} & \frac{1}{12} & \frac{199}{444} & -\frac{198}{3077} \\ 0 & 0 & \frac{1}{100} & \frac{1}{100} \\ 0 & 0 & 0 & 0 \end{bmatrix} U^*, \tag{31}$$

in which

$$U=\begin{bmatrix} \frac{1}{2} & 0 & \frac{\sqrt{3}}{2} & 0 \\ 0 & \frac{1}{3} & 0 & \frac{2\sqrt{2}}{3} \\ \frac{\sqrt{3}}{2} & 0 & -\frac{1}{2} & 0 \\ 0 & \frac{2\sqrt{2}}{3} & 0 & -\frac{1}{3} \end{bmatrix}. \tag{32}$$

Then, rk(B) = 3, rk(A) = 2, and

$$Q_A=U\begin{bmatrix} \frac{2}{3} & \frac{1}{6} & \frac{3269}{1398} & -\frac{977}{556} \\ -\frac{1}{3} & \frac{1}{6} & -\frac{1715}{701} & \frac{1723}{514} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}U^{*},$$

$$Q_B=U\begin{bmatrix} \frac{2}{3} & \frac{1}{6} & \frac{3269}{1398} & -\frac{977}{556} \\ -\frac{1}{3} & \frac{1}{6} & -\frac{1715}{701} & \frac{1723}{514} \\ 0 & 0 & \frac{1}{10} & \frac{1}{10} \\ 0 & 0 & 0 & 0 \end{bmatrix}U^{*},$$

$$Q_A^{\textcircled{\#}}=U\begin{bmatrix} 1 & -1 & 0 & 0 \\ 2 & 4 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}U^{*},$$

$$Q_B^{\oplus} = U \begin{bmatrix} 1 & -1 & \frac{1579}{33} & 0 \\ 2 & 4 & -\frac{13131}{257} & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} U^*,$$

$$F_A = U \begin{bmatrix} \frac{2}{5} & \frac{1}{5} & 0 & 0 \\ \frac{1}{5} & \frac{1}{5} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} U^*,$$

$$F_B = U \begin{bmatrix} \frac{3}{5} & \frac{1}{5} & 0 & 0 \\ \frac{1}{5} & \frac{3}{10} & 0 & 0 \\ 0 & 0 & \frac{1}{10} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} U^*.$$

(33)

Since

$$F_B - F_A = U \begin{bmatrix} \frac{1}{5} & 0 & 0 & 0 \\ 0 & \frac{1}{10} & 0 & 0 \\ 0 & 0 & \frac{1}{10} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} U^* \geq 0 \Longrightarrow F_A \overset{L}{\leq} F_B,$$

$$Q_A Q_A^{\textcircled{\#}} = U \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} U^* = Q_B Q_A^{\textcircled{\#}},$$

$$Q_A^{\textcircled{\#}} Q_A = U \begin{bmatrix} 1 & 0 & \frac{1579}{330} & -\frac{1916}{375} \\ 0 & 1 & -\frac{2243}{439} & \frac{1870}{189} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} U^* = Q_A^{\textcircled{\#}} Q_B, \qquad (34)$$

so $Q_A \overset{\textcircled{\#}}{\leq} Q_B$. We have A≤ LCB.

(1) But

$$A^{\textcircled{\#}} = U \begin{bmatrix} 10 & -15 & 0 & 0 \\ -10 & 30 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} U^*,$$

$$B^{\textcircled{\#}} = U \begin{bmatrix} \frac{25}{7} & -\frac{40}{7} & -\frac{25955}{108} & 0 \\ -\frac{10}{7} & \frac{100}{7} & -\frac{10601}{24} & 0 \\ 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} U^*,$$

$$A^2 A^{\textcircled{\#}} = U \begin{bmatrix} \frac{1}{5} & \frac{1}{10} & 0 & 0 \\ \frac{1}{15} & \frac{1}{15} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} U^*,$$

$$B^2B^{\oplus} = U\begin{bmatrix} \frac{1}{3} & \frac{2}{15} & \frac{613}{441} & 0 \\ \frac{1}{30} & \frac{1}{12} & \frac{610}{1361} & 0 \\ 0 & 0 & \frac{1}{100} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}U^*,$$

$$B^2B^{\oplus} - A^2A^{\oplus} = U\begin{bmatrix} \frac{2}{15} & \frac{1}{30} & \frac{613}{441} & 0 \\ -\frac{1}{30} & \frac{1}{60} & \frac{610}{1361} & 0 \\ 0 & 0 & \frac{1}{100} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}U^*. \tag{35}$$

$B^2B^{\oplus} - A^2A^{\oplus}$ is not a positive semidefinite matrix, so A is not below B under the CL partial order.

(2) Since

$$AA^* = U\begin{bmatrix} \frac{11206}{22413} & -\frac{115}{24122} & 0 & 0 \\ -\frac{115}{24122} & \frac{359}{761} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}U^*,$$

$$BB^* = U\begin{bmatrix} \frac{2022}{1237} & \frac{361}{987} & \frac{13}{40354} & 0 \\ \frac{361}{987} & \frac{474}{8516} & \frac{124}{9563} & 0 \\ \frac{13}{40354} & \frac{124}{9563} & \frac{135}{23954} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}U^*,$$

$$AB^{*} = U\begin{bmatrix} \frac{3905}{5364} & \frac{748}{3447} & \frac{33}{7979} & 0 \\ -\frac{215}{541} & -\frac{122}{3857} & \frac{61}{13860} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}U^{*},$$

$$(AA^{*})^{1/2}(BB^{*})^{1/2} = U\begin{bmatrix} \frac{557}{683} & \frac{1594}{8781} & \frac{5}{50378} & 0 \\ \frac{1749}{10616} & \frac{335}{2558} & \frac{173}{28289} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}U^{*},$$

$$AB^{*} \neq (AA^{*})^{1/2}(BB^{*})^{1/2}, \tag{36}$$

A is not below B under the GL partial order.

Theorem 3. *Let* $A, B \in \mathbb{H}_{\geq}(n),\ rk(A) = rk(B) \geq 1$. *Then,*

$$A \overset{LC}{\leq} B \Longleftrightarrow A \overset{L}{\leq} B. \tag{37}$$

Proof. When rk(A) = rk(B), we have

$$Q_A = Q_B = U\begin{bmatrix} T_{A1} & T_{A2} & T_{A3} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}U^{*}. \tag{38}$$

Then,

$$F_A = AQ_A^{\circledast} = U\begin{bmatrix} A_1T_{A1}^{-1} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}U^{*},$$

$$F_B = BQ_B^{\circledast} = U\begin{bmatrix} B_1T_{A1}^{-1} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}U^{*},$$

$$F_B - F_A = U\begin{bmatrix} (B_1 - A_1)T_{A1}^{-1} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}U^{*}. \tag{39}$$

Therefore, $A \overset{LC}{\leq} B \Longleftrightarrow F_A \overset{L}{\leq} F_B \Longleftrightarrow A \overset{L}{\leq} B$.

Corollary 1 (see [7]). *Let* $A, B \in \mathbb{H}_{\geq}(n)$. *Then,* $A \overset{CL}{\leq} B$ *if and only if* $A \overset{L}{\leq} B$.

In ([8], Theorem 8.5.15), Mitra has verified that when $A, B \in \mathbb{H}_{\geq}(n)$, there has $A \overset{CL}{\leq} B$ if and only if $A \overset{L}{\leq} B$. By applying Theorem 3, Corollary 1, and ([8], Theorem 8.5.15), we get Corollary 2.

Corollary 2. *Let* $A, B \in \mathbb{H}_{\geq}(n)$ *and rk(A) = rk(B) ≥* 1. *Then,*

$$A \overset{LC}{\leq} B \Leftrightarrow A \overset{CL}{\leq} B \Leftrightarrow A \overset{GL}{\leq} B. \tag{40}$$

An EP matrix is core invertible, and the core, Moore–Penrose, and group inverses of the matrix are identical. Next, we consider the case where both A and B are EP.

Theorem 4. *Let A, B be EP, rk(B) > rk(A) ≥* 1, $A \overset{LC}{\leq} B$, *$A = F_AQ_A$, and B = F_BQ_B be the P-2 expression of A and B. Then, there exists a unitary matrix U such that*

$$F_A = U\begin{bmatrix} I_{rk(A)} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}U^*,$$

$$Q_A = U\begin{bmatrix} T_{A1} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}U^*,$$

$$F_B = U\begin{bmatrix} B_1T_{A1}^{-1} & B_2T_{A4}^{-1} & 0 \\ (B_2T_{A4}^{-1})^* & B_5T_{A4}^{-1} & 0 \\ 0 & 0 & 0 \end{bmatrix}U^*,$$

$$Q_B = U\begin{bmatrix} T_{A1} & 0 & 0 \\ 0 & T_{A4} & 0 \\ 0 & 0 & 0 \end{bmatrix}U^*, \tag{41}$$

where $B_1 \in \mathbb{C}_{rk(A),rk(A)}$, $\rho\left(\begin{bmatrix} B_1T_{A1}^{-1} & B_2T_{A4}^{-1} \\ (B_2T_{A4}^{-1})^* & B_5T_{A4}^{-1} \end{bmatrix}\right) \leq 1$, T_{A1} *and T_{A4} are both nonsingular, and* $\begin{bmatrix} (B_1 - T_{A1})T_{A1}^{-1} & B_2T_{A4}^{-1} \\ (B_2T_{A4}^{-1})^* & B_5T_{A4}^{-1} \end{bmatrix} \geq 0.$

Proof. Let *A, B* be EP, rk(B) ≥ rk(A) ≥ 1, $A \overset{LC}{\leq} B$. There exists a unitary matrix U such that

$$A = U\begin{bmatrix} T_{A1} & 0 \\ 0 & 0 \end{bmatrix}U^* = F_A Q_A = U\begin{bmatrix} \Sigma K_A & \Sigma L_A \\ 0 & 0 \end{bmatrix}U^*$$
$$= U\begin{bmatrix} K_A & 0 \\ 0 & 0 \end{bmatrix}U^* U\begin{bmatrix} K_A^{-1}\Sigma K_A & K_A^{-1}\Sigma L_A \\ 0 & 0 \end{bmatrix}U^*. \quad (42)$$

Then, $L_A = 0$, $K_A = I_{rk}(A)$. Furthermore,

$$F_A = U\begin{bmatrix} I_{rk(A)} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}U^*,$$
$$Q_A = U\begin{bmatrix} \Sigma_A & 0 \\ 0 & 0 \end{bmatrix}U^* = U\begin{bmatrix} T_{A1} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}U^*. \quad (43)$$

Since B is EP,

$$B = U\begin{bmatrix} B_1 & B_2 & 0 \\ (B_2)^* & B_5 & 0 \\ 0 & 0 & 0 \end{bmatrix}U^* \quad (44)$$

and rk(B) ≥ rk(A) ≥ 1, we get

$$Q_B = U\begin{bmatrix} T_{A1} & 0 & 0 \\ 0 & T_{A4} & 0 \\ 0 & 0 & 0 \end{bmatrix}U^*,$$
$$Q_B^{\circledast} = U\begin{bmatrix} T_{A1}^{-1} & 0 & 0 \\ 0 & T_{A4}^{-1} & 0 \\ 0 & 0 & 0 \end{bmatrix}U^*. \quad (45)$$

Then,

$$F_B = BQ_B^{\circledast} = UU^*BQ_B^{\circledast} = U\left(U^*BU\begin{bmatrix} T_{A1}^{-1} & 0 & 0 \\ 0 & T_{A4}^{-1} & 0 \\ 0 & 0 & 0 \end{bmatrix}\right)U^*$$
$$= U\begin{bmatrix} B_1 & B_2 & 0 \\ (B_2)^* & B_5 & 0 \\ 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} T_{A1}^{-1} & 0 & 0 \\ 0 & T_{A4}^{-1} & 0 \\ 0 & 0 & 0 \end{bmatrix}U^*$$
$$= U\begin{bmatrix} B_1T_{A1}^{-1} & B_2T_{A4}^{-1} & 0 \\ (B_2)^*T_{A1}^{-1} & B_5T_{A4}^{-1} & 0 \\ 0 & 0 & 0 \end{bmatrix}U^*. \quad (46)$$

Since $\rho(K_A) \le 1$, $\rho(K_B) \le 1$, and F_A and F_B are both EP, we have $(B_2)^* T_{A1}^{-1} = (B_2T_{A4}^{-1})^*$, and $\rho\left(\begin{bmatrix} B_1T_{A1}^{-1} & B_2T_{A4}^{-1} \\ (B_2T_{A4}^{-1})^* & B_5T_{A4}^{-1} \end{bmatrix}\right) \le 1$. And because $F_A \overset{L}{\le} F_B$ and $F_B - F_A \ge 0$, then

$$F_B - F_A = U \begin{bmatrix} (B_1 - T_{A1})T_{A1}^{-1} & B_2 T_{A4}^{-1} & 0 \\ \left(B_2 T_{A4}^{-1}\right)^* & B_5 T_{A4}^{-1} & 0 \\ 0 & 0 & 0 \end{bmatrix} U^* \geq 0. \tag{47}$$

Therefore,

$$\begin{bmatrix} (B_1 - T_{A1})T_{A1}^{-1} & B_2 T_{A4}^{-1} \\ \left(B_2 T_{A4}^{-1}\right)^* & B_5 T_{A4}^{-1} \end{bmatrix} \geq 0. \tag{48}$$

So, we have (41).

It is noteworthy that $A \overset{LC}{\leq} B \nRightarrow A \overset{L}{\leq} B$, when A and B both are EP.

Example 3. Let

$$A = \begin{bmatrix} \frac{1}{3} & 0 & 0 \\ 0 & -\frac{1}{2} & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$B = \begin{bmatrix} \frac{1}{4} & 0 & 0 \\ 0 & -\frac{1}{4} & 0 \\ 0 & 0 & 0 \end{bmatrix}, \tag{49}$$

rk(B) = rk(A) = 2, and

$$Q_A = \begin{bmatrix} \frac{1}{3} & 0 & 0 \\ 0 & -\frac{1}{2} & 0 \\ 0 & 0 & 0 \end{bmatrix} = Q_B,$$

$$Q_A^{\oplus} = \begin{bmatrix} 3 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & 0 \end{bmatrix} = Q_B^{\oplus}. \tag{50}$$

We obtain $Q_B \overset{\oplus}{\leq} Q_A$. Since

$$F_A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$F_B = \begin{bmatrix} \frac{3}{4} & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$F_A - F_B = \begin{bmatrix} \frac{1}{4} & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 \end{bmatrix} \geq 0 \Longrightarrow F_B^L \leq F_A. \tag{51}$$

We have $B \overset{LC}{\leq} A$. But

$$A - B = \begin{bmatrix} \frac{1}{12} & 0 & 0 \\ 0 & -\frac{1}{4} & 0 \\ 0 & 0 & 0 \end{bmatrix} \ngeq 0. \tag{52}$$

B is not below A under the Löwner partial order. So, $B \overset{LC}{\leq} A \nRightarrow B \overset{L}{\leq} A$.

ACKNOWLEDGMENTS

The work was supported by Guangxi Natural Science Foundation (No. 2018GXNSFDA281023), High Level Innovation Teams and Distinguished Scholars in Guangxi Universities (No. GUIJIAOREN201642HAO), the National Natural Science Foundation of China (No. 11361009), the Special Fund for Bagui Scholars of Guangxi (No. 2016A17), and the Education Innovation Program for 2019 Graduate Students (No. gxun-chxzs2019026).

REFERENCES

1. J. Hauke and A. Markiewicz, “On partial orderings on the set of rectangular matrices,” *Linear Algebra and its Applications*, vol. 219, pp. 187–193, 1995.
2. O. M. Baksalary and G. Trenkler, “Core inverse of matrices,” *Linear and Multilinear Algebra*, vol. 58, no. 6, pp. 681–697, 2010.
3. T. Ando, “Square inequality and strong order relation,” *Advances in Operator Theory*, vol. 1, no. 1, pp. 1–7, 2016.
4. G. Wang, Y. Wei, and S. Qiao, *Generalized Inverse: Theory and Computations*, Springer, Berlin, Germany, 2018.
5. A. Ben-Israel and T. N. E. Greville, *Generalized Inverse: Theory and Applications*, Springer, Berlin, Germany, 2nd edition, 2003.
6. R. E. Hartwig, “How to partially orders regular elements,” *Math Japonica*, vol. 25, no. 1, pp. 1–13, 1980.
7. S. K. Mitra, P. Bhimasankaram, and S. B. Malik, *Matrix Partial Orders, Shorted Operators and Applications*, World Scientific, Toh Tuck Link, Singapore, 2010.
8. H. Wang and X. Liu, “Partial orders based on core-nilpotent decomposition,” *Linear Algebra and its Applications*, vol. 488, pp. 235–248, 2016.
9. H. Wang, “Core-EP decomposition and its applications,” *Linear Algebra and its Applications*, vol. 508, pp. 289–300, 2016.
10. H. Wang and X. Liu, “The polar-like decomposition and its applications,” *Filomat*, vol. 33, no. 12, pp. 3977–3983, 2019.
11. H. Wang and X. Liu, “A partial order on the set of complex matrices with index one,” *Linear and Multilinear Algebra*, vol. 66, no. 1, pp. 206–216, 2018.

Chapter 2

NATURAL PARTIAL ORDERS ON TRANSFORMATION SEMIGROUPS WITH FIXED SETS

Yanisa Chaiya, Preeyanuch Honyam, and Jintana Sanwong

Department of Mathematics, Chiang Mai University, Chiang Mai 50200, Thailand

ABSTRACT

Let X be a nonempty set. For a fixed subset Y of X, let $\mathrm{Fix}(X, Y)$ be the set of all self-maps on X which fix all elements in Y. Then $\mathrm{Fix}(X, Y)$ is a regular monoid under the composition of maps. In this paper, we characterize the natural partial order on $\mathrm{Fix}(X, Y)$ and this result extends the result due to Kowol and Mitsch. Further, we find elements which are compatible and describe minimal and maximal elements.

Citation: Yanisa Chaiya, Preeyanuch Honyam, and Jintana Sanwong, "Natural Partial Orders on Transformation Semigroups with Fixed Sets", International Journal of Mathematics and Mathematical Sciences, volume 2016, article ID 2759090, https://doi.org/10.1155/2016/2759090.

INTRODUCTION

For any semigroup S, the natural partial order on (S), the set of all idempotents on S, is defined by

$$e \le f \quad \text{iff } e = ef = fe.$$

(1)

In 1980, Hartwig [1] and Nambooripad [2] proved that if S is a regular semigroup, then the relation

$$a \le b \quad \text{iff } a = eb = bf \text{ for some } e, f \in E(S)$$

(∗)

is a partial order on S which extends the usual ordering of the set $E(S)$.

Later in 1986, the natural partial order on a regular semigroup was further extended to any semigroup S by Mitsch [3] as follows:

$$a \le b \quad \text{iff } a = xb = by,$$

$$xa = a \quad \text{for some } x, y \in S^1.$$

(2)

Let X be a set and (X) denote the semigroup of binary relations on the set X under the composition of relations. A *partial transformation semigroup* is the collection of functions from a subset of X into X with composition which is denoted by (X). Let (X) be the set of all transformations from X into itself and it is called the *full transformation semigroup* on X. Then (X) and (X) are subsemigroups of (X). It is well known that (X) and (X) are regular semigroups.

In 1986, Kowol and Mitsch [4] characterized the natural partial order on (X) in terms of images and kernels. They also proved that an element $\alpha \in (X)$ is maximal with respect to the natural order if and only if α is surjective or injective; α is minimal if and only if α is a constant map. Moreover, they described lower and upper bounds for two transformations and gave necessary and sufficient conditions for their existence.

Later in 2006, Namnak and Preechasilp [5] studied two natural partial orders on B(X) and characterized when two elements of $B(X)$ are related under these orders. They also described the minimality, maximality, left compatibility, and right compatibility of elements with respect to each order.

Let Y be a subset of X. Recently, Fernandes and Sanwong [6] defined

$$PT(X, Y) = \{\alpha \in P(X) : X\alpha \subseteq Y\},$$

(3)

where $X\alpha$ denotes the image of α. Moreover, they defined (X, Y) to be the set of all injective transformations in (X, Y). Hence (X, Y) and (X, Y) are subsemigroups of $P(X)$.

In [7], Sangkhanan and Sanwong described natural partial order ≤ on (X, Y) and (X, Y) in terms of domains, images, and kernels. They also compared ≤ with the subset order and characterized the meet and join of these two orders. Furthermore, they found elements of (X, Y) and (X, Y) which are compatible and determined the minimal and maximal elements.

Let Y be a fixed subset of X and

$$\mathrm{Fix}(X, Y) = \{\alpha \in T(X) : y\alpha = y \ \forall y \in Y\}. \tag{4}$$

In 2013, Honyam and Sanwong [8] proved that Fix(X, Y) is a regular semigroup and they also determined its Green's relations and ideals. Moreover, they proved that Fix(X, Y) is never isomorphic to $T(Z)$ for any set Z when $\emptyset \neq Y \subsetneq X,$ and every semigroup S is isomorphic to a subsemigroup of Fix(X', Y') for some appropriate sets X' and Y' with $Y' \subseteq X'$. Note that this also follows trivially from the fact that (X) embeds in Fix($X \cup Z$, Z) for any set Z with $X \cap Z = \emptyset$. Recently, the authors in [9] proved that there are only three types of maximal subsemigroups of Fix(X, Y) and these maximal subsemigroups coincide with the maximal regular subsemigroups when $X \backslash Y$ is a finite set with $|X \backslash Y| \geq 2$.They also gave necessary and sufficient conditions for Fix(X, Y) to be factorizable, unit-regular, and directly finite.

In this paper, we characterize the natural partial order on Fix(X, Y) and find elements which are compatible under this order in Section 3. In Section 4, we describe the minimal elements, the maximal elements, and the covering elements. Moreover, we find the number of upper covers of minimal elements and the number of lower covers of maximal elements.

PRELIMINARIES AND NOTATIONS

In [8], the authors proved that Fix(X, Y) is a regular subsemigroup of $T(X)$. Note that Fix(X, Y) contains 1_X, the identity map on X. If $Y = \emptyset$, then Fix(X, Y) = $T(X)$; and if $|X| = 1$ or $X = Y$, then Fix(X, Y) consists of one element, 1_X. So, throughout this paper we will consider the case $Y \subsetneq X$ and $|X| > 1$.

For any $\alpha \in (X)$, the symbol π_α denotes the partition of X induced by the map α, namely,

$$\pi_\alpha = \left\{ x\alpha^{-1} : x \in X\alpha \right\}. \tag{5}$$

For $\alpha, \beta \in T(X), \mathscr{A} \subseteq \pi_\alpha,$ and $\mathscr{B} \subseteq \pi_\beta,$ we say that $\mathscr{A}$ refines $\mathscr{B}$ if for each $A \in \mathscr{A}$ there exists $B \in \mathscr{B}$ such that .

Throughout this paper, unless otherwise stated, let $\{ : i \in I\}$.

For each $\alpha \in \mathrm{Fix}(X, Y)$, we have $y_i\alpha = y_i$ for all $i \in I$. So $Y = Y\alpha \subseteq X\alpha$. If $\alpha \in \mathrm{Fix}(X, Y)$, then we write

$$\alpha = \begin{pmatrix} A_i & B_j \\ y_i & b_j \end{pmatrix} \tag{6}$$

and take as understood that the subscripts i and j belong to the index sets I and J, respectively, such that $X\alpha = \{yi : i \in I\} \cup \{bj : j \in J\}$, $yi\alpha-1 = Ai$, and $bj\alpha-1 = Bj$.Thus $Ai \cap Y = \{yi\}$ for all $i \in I$, $Bj \subseteq X \setminus Y$ for all $j \in J$ and $\{bj : j \in J\} \subseteq X \setminus Y$. Here J can be an empty set.

An idempotent e in a semigroup S is said to be minimal if e has the property $f \in (S)$ and $f \leq e$ implies $f = e$.

In [8] the authors showed that

$$E_m = \left\{ \begin{pmatrix} A_i \\ y_i \end{pmatrix} : \{A_i : i \in I\} \text{ is a partition of } X \text{ with } y_i \in A_i \right\} \tag{7}$$

is the set of all minimal idempotents in $\mathrm{Fix}(X, Y)$ and it is an ideal of $\mathrm{Fix}(X, Y)$. We note that E_m is simply the set $\{\alpha \in \mathrm{Fix}(X, Y) : X\alpha = Y\}$ and α is an idempotent in $\mathrm{Fix}(X, Y)$ if and only if $x\alpha = x$ for all $x \in X\alpha \setminus Y$.

NATURAL PARTIAL ORDER ON FIX(X, Y)

Kowol and Mitsch [4] gave a characterization of the natural partial order on (X). Later in 1994, Higgins [10] showed that if T is a regular subsemigroup of a semigroup S, then the natural partial order on T is the restriction to T of the natural partial order on S. Here we describe the natural partial order on $\mathrm{Fix}(X, Y)$ which is a regular subsemigroup of $T(X)$ without making use of Higgins' result and when we take $Y = 0$, we recapture the result above by Kowol and Mitsch.

We note that if $\alpha, \beta \in \mathrm{Fix}(X, Y)$ and $\alpha = \beta\gamma$ for some $\gamma \in \mathrm{Fix}(X, Y)$, then π_β refines π_α.

Since $\mathrm{Fix}(X, Y)$ is regular, we use $(*)$ to study the natural partial order on this semigroup.

Theorem 1. *Let* $\alpha, \beta \in \mathrm{Fix}(X, Y)$. *Then* $\alpha \leq \beta$ *if and only if the following statements hold:*

- $X\alpha \subseteq X\beta$;
- π_β *refines* π_α;
- if $x\beta \in X\alpha$, *then* $x\beta = x\alpha$.

Proof. Suppose that $\alpha \leq \beta$. Then, by $(*)$, we have

$$\alpha = \lambda\beta = \beta\gamma \tag{8}$$

for some $\lambda, \gamma \in E(\mathrm{Fix}(X, Y))$. Thus $X\alpha = (X\lambda) \subseteq X\beta$. Since $\alpha = \beta\gamma$, we get that π_β refines π_α. Now, let $x\beta \in X\alpha$. Then $x\beta = x'\,\alpha$ for some $x' \in X$ and thus $x\beta = x'\,\alpha = x'\,\beta\gamma = (x'\,\beta)$. Hence $x\beta \in X\gamma$ and then $x\alpha = x\beta\gamma = x\beta$ since γ is an idempotent.

Conversely, assume that conditions (1)–(3) hold. By condition (1), we can write

$$\alpha = \begin{pmatrix} A_i & B_j \\ y_i & b_j \end{pmatrix},$$

$$\beta = \begin{pmatrix} A'_i & C_j & C_k \\ y_i & b_j & b_k \end{pmatrix}, \tag{9}$$

where $y_i \in A_i \cap A'_i, b_j, b_k \in X\setminus Y,$ and $B_j, C_j, C_k \subseteq X\setminus Y$. Since $y_i \in A_i \cap A'_i$ and π_β refines π_α, we obtain $A'_i \subseteq A_i$ for all $i \in I$. If $J = \emptyset$, then define $\lambda = \alpha$ and thus $\alpha = \lambda\beta$. If $J \neq \emptyset$, then, for each $j \in J$, let $c_j \in C_j$. So $c_j\beta = b_j \in X\alpha$. By condition (3), $c_j\alpha = c_j\beta = b_j$; that is, $c_j \in B_j$ and hence $C_j \subseteq B_j$. Define

$$\lambda = \begin{pmatrix} A_i & B_j \\ y_i & c_j \end{pmatrix}. \tag{10}$$

We get $\lambda \in (\mathrm{Fix}(X, Y))$ and $\alpha = \lambda\beta$.

If $K = \emptyset$, then $\alpha = \beta 1_X$. If $K \neq \emptyset$, then, for each $k \in K$, we choose $c_k \in C_k$.

Case 1. Consider $X\beta = X$. Then $X \setminus X\alpha = \{\ : k \in K\}$. We define $\gamma \in \mathrm{Fix}(X, Y)$ by

$$x\gamma = \begin{cases} x, & x \in X\alpha, \\ c_k\alpha, & x = b_k \in X \setminus X\alpha. \end{cases} \tag{11}$$

To prove that $\alpha = \beta\gamma$, let $x \in X$. If $x \in A'_i$ for some i or $x \in C_j$ for some j, then it is clear that $x\alpha = x\beta\gamma$. Now, if $x \in C_k$ for some k, then $x\beta = c_k\beta$ and thus $x\alpha = c_k\alpha$ since π_β refines π_α. So $(x\beta) = b_k\gamma = c_k\alpha = x\alpha$. Hence $\alpha = \beta\gamma$. It remains to show that γ is an idempotent. Let $x\gamma \in X\gamma \setminus X\alpha$. Then $x\gamma = c_k\alpha$ for some k. Thus $(x\gamma) = (c_k\alpha) = c_k\alpha = x\gamma$ since $c_k\alpha \in X\alpha$.

Case 2. Consider $X\beta \subsetneq X$. We choose $c_0 \in X \setminus X\beta$ and define $\gamma' \in \mathrm{Fix}(X, Y)$ by

$$x\gamma' = \begin{cases} x, & x \in X\alpha, \\ c_k\alpha, & x = b_k \in X\beta \setminus X\alpha, \\ c_0, & x \in X \setminus X\beta. \end{cases} \tag{12}$$

By the same prove as given in Case 1, we get $\alpha = \beta\gamma'$ and $(x\gamma')' = x\gamma'$ for all $x\gamma' \in X\beta$. If $x\gamma' = c_0$, then $(x\gamma')' = c_0\gamma' = c_0 = x\gamma'$. So γ' is an idempotent. Therefore, $\alpha \leq \beta$ by $(*)$.

Remark 2. If $Y = \emptyset$, then $\mathrm{Fix}(X, Y) = T(X)$, and we have the characterization of $\leq$ on $T(X)$ which first appeared in [4, Proposition 2.3].

As a direct consequence of Theorem 1, we get the following corollary.

Corollary 3. *Let $\alpha, \beta \in$ Fix(X, Y) with $\alpha \leq \beta$.* If *$X\alpha \setminus Y = X\beta \setminus Y$, then $\alpha = \beta$.*

Let S be a semigroup. An element $a \in S$ is said to be left (right) compatible with respect to the partial order $\leq$ if $ab \leq ac$ $(ba \leq ca)$ whenever $b \leq c$.

The following results describe all the left compatible and right compatible elements in $\mathrm{Fix}(X, Y)$ when $\emptyset \neq Y \subsetneq X$. We also write $\alpha < \beta$ instead of $\alpha \leq \beta$ and $\alpha \neq \beta$ for $\alpha, \beta \in \mathrm{Fix}(X, Y)$.

Theorem 4. *Assume that $\emptyset \neq Y \subsetneq X$ and let $\lambda \in$ Fix(X, Y). Then λ is left compatible if and only if λ is a minimal idempotent or λ is surjective.*

Proof. Suppose that λ is left compatible. Assume by contrary that λ is not a minimal idempotent and λ is not surjective. So there are $a \in X\lambda \setminus Y$ and $b \in X \setminus X\lambda$. Define

$$\alpha = \begin{pmatrix} y_i & X \setminus Y \\ y_i & a \end{pmatrix},$$

$$\beta = \begin{pmatrix} y_i & b & X \setminus (Y \cup \{b\}) \\ y_i & a & b \end{pmatrix}. \qquad (13)$$

Then α, $\beta \in \mathrm{Fix}(X, Y)$ with $\alpha < \beta$ and thus $\lambda\alpha \leq \lambda\beta$ since λ is left compatible. However, $X\lambda\alpha \subseteq X\lambda\beta /$ since $a \in X\lambda\alpha$ but $a \notin X\lambda\beta$, a contradiction.

Conversely, let $\alpha{\leq}\beta$. If λ is a minimal idempotent, then $\lambda\alpha = \lambda = \lambda\beta$. Now, assume that λ is surjective. So $X\lambda\alpha = X\alpha \subseteq X\beta = X\lambda\beta$. Let $A \in \pi_{\lambda\beta}$. So $A = (\lambda\beta)^{-1} = (x\beta^{-1})\lambda^{-1}$ for some $x \in X\lambda\beta$. Since $\alpha \leq \beta$, we have that π_β refines π_α and hence $x\beta^{-1} \subseteq x'\,\alpha^{-1}$ for some $x' \in X\alpha$. Since $x' \in X\alpha$, we get $x' = u\alpha$ for some $u \in X$ and $u = \mathrm{V}\lambda$ for some $\mathrm{V} \in X$ because λ is surjective. Hence $\mathrm{V}\lambda\alpha = u\alpha = x'$; that is, $x' \in X\lambda\alpha$. Further, $A = (x\beta^{-1})^{-1} \subseteq (x'\,\alpha^{-1})\lambda^{-1} = x'\,(\lambda\alpha)^{-1} \in \pi_{\lambda\alpha}$, thus $\pi_{\lambda\beta}$ refines $\pi_{\lambda\alpha}$. Let $a\lambda\beta \in X\lambda\alpha$. So $(a\lambda) \in X\alpha$ and then $a\lambda\beta = a\lambda\alpha$. By Theorem 1, we have $\lambda\alpha \leq \lambda\beta$ which implies that λ is left compatible.

Theorem 5. *The following statements hold.*

- *If* $|Y| = 1$, *then* $\lambda \in \mathrm{Fix}(X, Y)$ *is right compatible if and only if* λ *is a minimal idempotent or is injective.*
- *If* $|Y| = 2$, *then* $\lambda \in \mathrm{Fix}(X, Y)$ *is right compatible if and only if* λ *is injective.*

Proof. (1) Assume that $Y = \{y\}$ and λ is right compatible. Suppose in the contrary that λ is not a minimal idempotent and λ is not injective. So we can write

$$\lambda = \begin{pmatrix} A & B_j \\ y & b_j \end{pmatrix}, \qquad (14)$$

where $y \in A$ and $J \neq 0$. Since λ is not injective, two cases arise.

Case 1. Consider $|A| \geq 2$. Choose $a \in A \setminus \{y\}$ and $c \in B_{j_0}$ for some $j_0 \in J$. Let $X \setminus \{a, c\} = \{ : k \in K\}$ and define $\alpha \in \mathrm{Fix}(X, Y)$ by

$$\alpha = \begin{pmatrix} \{a, c\} & x_k \\ c & x_k \end{pmatrix}; \qquad (15)$$

we get $\alpha < 1_X$. Moreover, we have $(1_X\lambda) = a\lambda = y = (1_X\lambda)$, hence there is $B \in \pi_{1_X\lambda}$ such that $\{a, y\} \subseteq B$. However, $\{a, y\} \not\subseteq C$ for all $C \in \pi_{\alpha\lambda}$

since $a(\alpha\lambda) = c\lambda = b_{j_0} \neq y = y(\alpha\lambda)$. This means that $\pi_{1_X\lambda}$ does not refine $\pi_{\alpha\lambda}$. By Theorem 1, we get $\alpha\lambda \nleq 1_X\lambda$, a contradiction.

Case 2. Consider $|B_{j_0}| \geq 2$ for some $j_0 \in J$. Choose $a, b \in B_{j_0}$ such that $a \neq b$. Let $X \setminus \{a, b, y\} = \{ : k \in K\}$. Define $\alpha, \beta \in \mathrm{Fix}(X, Y)$ by

$$\alpha = \begin{pmatrix} \{y, a\} & b & x_k \\ y & a & x_k \end{pmatrix},$$

$$\beta = \begin{pmatrix} a & b & y & x_k \\ b & a & y & x_k \end{pmatrix}, \qquad (16)$$

we get $\alpha < \beta$. Since $(\beta\lambda) = b\lambda = a\lambda = (\beta\lambda)$, there is $B \in \pi_{\beta\lambda}$ such that $\{a, b\} \subseteq B$. However, $\{a, b\} \nsubseteq C$ for all $C \in \pi_{\alpha\lambda}$ since $a(\alpha\lambda) = y\lambda = y \neq b_{j_0} = a\lambda = b(\alpha\lambda)$. So $\pi_{\beta\lambda}$ does not refine $\pi_{\alpha\lambda}$. By Theorem 1, we get $\alpha\lambda \nleq \beta\lambda$, a contradiction.

Conversely, let $\alpha, \beta \in \mathrm{Fix}(X, Y)$ be such that $\alpha \leq \beta$. If λ is a minimal idempotent, then $\lambda = \binom{X}{y}$ and $\alpha\lambda = \lambda = \beta\lambda$; that is, λ is right compatible. Now, assume that λ is injective. Since $X\alpha \subseteq X\beta$, we get $X\alpha\lambda \subseteq X\beta\lambda$. Let $A \in \pi_{\beta\lambda}$. So $A = (\beta\lambda)^{-1} = (x\lambda^{-1})\beta^{-1}$ for some $x \in X\beta\lambda$ and hence $(x\lambda^{-1})\beta^{-1} \subseteq x'\alpha^{-1}$ for some $x' \in X\alpha$. So $x' = u\alpha$ for some $u \in X$. Since λ is injective, $\{x'\} = v\lambda^{-1}$ for some $v \in X\lambda$ and $u\alpha\lambda = x'\lambda = v$; that is, $v \in X\alpha\lambda$. Thus $x'\alpha^{-1} = (v\lambda^{-1})^{-1} = v(\alpha\lambda)^{-1} \in \pi_{\alpha\lambda}$ which implies that $\pi_{\beta\lambda}$ refines $\pi_{\alpha\lambda}$. Let $a\beta\lambda \in X\alpha\lambda$. So $a\beta\lambda = b\alpha\lambda$ for some $b \in X$. Since λ is injective, $a\beta = b\alpha$ and then $a\beta \in X\alpha$. Thus $a\beta = a\alpha$ since $\alpha \leq \beta$ and that $a\beta\lambda = a\alpha\lambda$. Therefore, $\alpha\lambda \leq \beta\lambda$, and we conclude that λ is right compatible.

(2) Suppose that λ is right compatible and λ is not injective. Write

$$\lambda = \begin{pmatrix} A_i & B_j \\ y_i & b_j \end{pmatrix}, \qquad (17)$$

where $y_i \in A_i$ and $|I| \geq 2$. Since λ is not injective, two cases arise.

Case 1. $|A_{i_0}| \geq 2$ for some $i_0 \in I$. Choose $a \in A_{i_0} \setminus \{y_{i_0}\}$ and $y_{i_1} \in Y \setminus \{y_{i_0}\}$. Let $X \setminus \{y_{i_1}, a\} = \{x_k : k \in K\}$ and define

$$\alpha = \begin{pmatrix} \{y_{i_1}, a\} & x_k \\ y_{i_1} & x_k \end{pmatrix}. \qquad (18)$$

Then $\alpha < 1_X$ and hence $\alpha\lambda \leq 1_X\lambda$. We can see that $\{y_{i_0}, a\} \subseteq A_{i_0} \in \pi_\lambda = \pi_{1_X\lambda}$, but $\{y_{i_0}, a\} \nsubseteq B$ for all $B \in \pi_{\alpha\lambda}$ since $a\alpha\lambda = y_{i_1}\lambda = y_{i_1} \neq y_{i_0} = y_{i_0}\alpha\lambda$. This means that $\pi_{1_X\lambda}$ does not refine $\pi_{\alpha\lambda}$, a contradiction.

Case 2. $|B_{j_0}| \geq 2$ for some $j_0 \in J$. This is virtually identical to Case 2 of (1) above.

MINIMAL AND MAXIMAL ELEMENTS

Let S be a semigroup together with the partial order $\leq$. S is said to be directed downward if every pair of elements has a lower bound. In other words, for any a and b in S, there exists c in S with $c \leq a$ and $c \leq b$. A directed upward semigroup is defined dually.

If $Y = \emptyset$, then Fix(X, Y) = T(X) and it has neither minimum nor maximum elements under the natural order (see [4]). So, in Lemmas 6 and 7 we assume that $\emptyset \neq Y \subsetneq X$.

Lemma 6. *Assume that* $\emptyset \neq Y \subsetneq X$. *Then the following statements are equivalent.*

(1) Fix(X, Y) *has a minimum element.*

(2) Fix(X, Y) *is directed downward.*

(3) $|Y| = 1$.

Proof. (1)⇒(2) This is clear.

(2)⇒(3) Assume that Fix(X, Y) is directed downward. Let $y_{i_1}, y_{i_2} \in Y$ and $J = I \setminus \{i_1, i_2\}$. Consider

$$\alpha = \begin{pmatrix} y_j & y_{i_2} & (X \setminus Y) \cup \{y_{i_1}\} \\ y_j & y_{i_2} & y_{i_1} \end{pmatrix},$$

$$\beta = \begin{pmatrix} y_j & y_{i_1} & (X \setminus Y) \cup \{y_{i_2}\} \\ y_j & y_{i_1} & y_{i_2} \end{pmatrix}. \qquad (19)$$

We have α, $\beta \in$ Fix(X, Y) and there is $\gamma \in$ Fix(X, Y) such that $\gamma \leq \alpha$ and $\gamma \leq \beta$. By Theorem 1, $\pi\alpha$ refines $\pi\gamma$ and $\pi\beta$ refines $\pi\gamma$. Then there is $A \in \pi\gamma$ such that $(X \setminus Y) \cup \{yi1\} \subseteq A$ and $(X \setminus Y) \cup \{yi2\} \subseteq A$.Thus $yi1$, $yi2 \in A$ and hence $yi1 = yi2$. Since $yi1$, $yi2$ are arbitrary elements in Y, we obtain that $|Y| = 1$.

(3)⇒(1) Assume that $Y = \{y\}$. It is easy to see that $\theta = \begin{pmatrix} X \\ y \end{pmatrix}$ is the minimum element in Fix(X, Y).

Lemma 7. *Assume that* $\emptyset \neq Y \subsetneq X$. *Then the following statements are equivalent.*

(1) Fix(X, Y) *has a maximum element.*

(2) Fix(X, Y) *is directed upward.*

(3) $|X \setminus Y| = 1$.

Proof. (1)⇒(2) This is clear.

(2)⇒(3) Assume that Fix(X, Y) is directed upward. Let $a, b \in X \setminus Y$ and $X \setminus \{a, b\} = \{ : k \in K\}$. Define

$$\alpha = \begin{pmatrix} a & b & x_k \\ b & a & x_k \end{pmatrix} \in \text{Fix}(X, Y). \qquad (20)$$

Then there is $\gamma \in$ Fix(X, Y) such that $\alpha \leq \gamma$ and $1_X \leq \gamma$. Since α and 1_X are bijective, γ is also bijective and thus $b_\gamma \in (X\alpha \setminus Y) \cap (X1X \setminus Y)$. So $a = b\alpha = b\gamma = b1_X = b$. Since a, b are arbitrary elements in $X \setminus Y$, we get $|X \setminus Y| = 1$.

(3)⇒(1) Assume that $|X \setminus Y| = 1$. It is easy to see that 1_X is the maximum element in Fix(X, Y).

We now describe minimal and maximal elements in Fix(X, Y) when $\emptyset \neq Y \subsetneq X$. If $|Y| = 1$, then Fix(X, Y) has a minimum element by Lemma 6 and it is minimal. In the same way, if $|X \setminus Y| = 1$, then Fix(X, Y) has a maximum element by Lemma 7 and it is maximal.

Theorem 8. *Assume that* $\emptyset \neq Y \subsetneq X$ *and let* $\alpha \in$ Fix(X, Y). *Then α is minimal if and only if α is a minimal idempotent.*

Proof. Assume that α is minimal but α is not a minimal idempotent. So we can write

$$\alpha = \begin{pmatrix} A_i & B_j \\ y_i & b_j \end{pmatrix}, \qquad (21)$$

where $J \neq \emptyset$. Choose $i_0 \in I$ and $j_0 \in J$. Let $I' = I \setminus \{i_0\}$, $J' = J \setminus \{j_0\}$ and define $\beta \in$ Fix(X, Y) by

$$\beta = \begin{pmatrix} A_{i_0} \cup B_{j_0} & A_{i'} & B_{j'} \\ y_{i_0} & y_{i'} & b_{j'} \end{pmatrix}. \qquad (22)$$

Hence $\beta < \alpha$, which contradicts the minimality of α.

Conversely, assume that α is a minimal idempotent and $\beta \leq \alpha$. Since $Y \subseteq X\beta \subseteq X\alpha = Y$, we get $X\beta = X\alpha$ and hence $X\beta \setminus Y \subseteq X\alpha \setminus Y$. By Corollary 3, we obtain $\beta = \alpha$.

Theorem 9. *Assume that* $\emptyset \neq Y \subsetneq X$ *and let* $\alpha \in$ Fix(X, Y). *Then α is maximal if and only if α is injective or α is surjective.*

Proof. Let α be maximal. Assume that α is not injective and surjective. So there are $a, b, c \in X$ such that $a\alpha = b\alpha$ with $a \neq b$ and $c \in X \setminus X\alpha$. Write

$$\alpha = \begin{pmatrix} A_i & B_j \\ y_i & b_j \end{pmatrix}. \qquad (23)$$

Case 1. $a, b \in A_{i_0}$ for some $i_0 \in I$. We may assume that $a \neq y_{i_0}$. Let I' = $I \setminus \{i_0\}$ and define

$$\beta = \begin{pmatrix} A_{i_0} \setminus \{a\} & a & A_{i'} & B_j \\ y_{i_0} & c & y_{i'} & b_j \end{pmatrix}. \qquad (24)$$

Then $\beta \in$ Fix(X, Y) and $\alpha < \beta$ which contradicts the maximality of α.

Case 2. $a, b \in B_{j_0}$ for some $j_0 \in J$. Then we let $J' = J \setminus \{j_0\}$ and define

$$\gamma = \begin{pmatrix} A_i & B_{j_0} \setminus \{a\} & a & B_{j'} \\ y_i & b_{j_0} & c & b_{j'} \end{pmatrix}. \qquad (25)$$

Then $\gamma \in$ Fix(X, Y) and $\alpha < \gamma$ which contradicts the maximality of α.

Conversely, assume that α is injective or α is surjective and $\alpha \leq \beta$ for some $\beta \in$ Fix(X, Y). Then $X\alpha \subseteq X\beta$ and $X\alpha \setminus Y \subseteq X\beta \setminus Y$. Consider the case where α is injective, by letting $z \in X\beta \setminus Y$. Then $z = x\beta$ for some $x \in X \setminus Y$ and $x\alpha \in X\alpha \setminus Y \subseteq X\beta \setminus Y$; that is, $x\alpha = x'\beta$ for some $x' \in X \setminus Y$. So $x'\beta \in X\alpha$ and $x'\beta = x'\alpha$ by Theorem 1. Since α is injective, we get $x = x'$ and thus $z = x\beta = x'\beta = x\alpha \in X\alpha \setminus Y$, whence $X\beta \setminus Y \subseteq X\alpha \setminus Y$. Hence, in this case, $X\alpha \setminus Y = X\beta \setminus Y$ and by Corollary 3 we obtain $\alpha = \beta$. In the case α is surjective, we get $X \setminus Y = X\alpha \setminus Y \subseteq X\beta \setminus Y \subseteq X \setminus Y$; that is, $X\alpha \setminus Y = X\beta \setminus Y$. Again by Corollary 3, we have that $\alpha = \beta$. Therefore, α is maximal.

Figure 1 shows the diagram of Fix(X, Y) when X = {1, 2, 3, 4} and Y = {1, 2}. The notation ($abcd$) for $\alpha \in$ Fix(X, Y) means that $1\alpha = a$, $2\alpha = b$, $3\alpha = c$, and $4\alpha = d$.

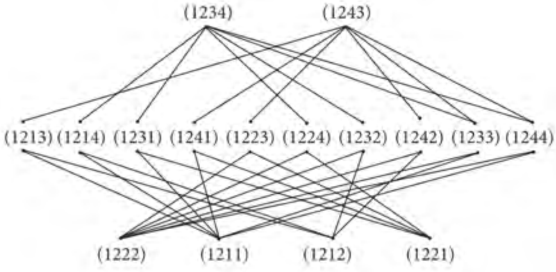


Figure 1.

An element $\beta \in \mathrm{Fix}(X, Y)$ is called an upper cover for $\alpha \in \mathrm{Fix}(X, Y)$ if $\alpha < \gamma$

Lemma 10. *Assume that* $\emptyset \neq Y \subsetneq X$ *and let* $\alpha \in \mathrm{Fix}(X, Y)$. *Then the following statements hold.*

- *If α is not minimal in* $\mathrm{Fix}(X, Y)$, *then there is some lower cover of α in* $\mathrm{Fix}(X, Y)$.
- *If α is not maximal in* $\mathrm{Fix}(X, Y)$, *then there is some upper cover of α in* $\mathrm{Fix}(X, Y)$.

Proof. (1) Let $\alpha \in \mathrm{Fix}(X, Y)$ be not minimal. By Theorem 8, α is not a minimal idempotent. So we can write

$$\alpha = \begin{pmatrix} A_i & B_j \\ y_i & b_j \end{pmatrix}, \qquad (26)$$

where $J \neq \emptyset$. Define β as in the proof of Theorem 8, we get $\beta < \alpha$. Suppose that there is $\lambda \in \mathrm{Fix}(X, Y)$ such that $\beta \leq \lambda \leq \alpha$. Then by Theorem 1, $X\beta \subseteq X\lambda \subseteq X\alpha$ and thus $X\beta \setminus Y \subseteq X\lambda \setminus Y \subseteq X\alpha \setminus Y$. Since $X\alpha \setminus Y = (X\beta \setminus Y) \cup \{b_{j_0}\}$ which implies $X\lambda \setminus Y = X\beta \setminus Y$ or $X\lambda \setminus Y = X\alpha \setminus Y$, thus $\lambda = \beta$ or $\lambda = \alpha$ by Corollary 3. Therefore, β is a lower cover of α.

(2) The proof is similar to (1), using β or γ from the proof of Theorem 9 as appropriate.

Now, we aim to find the number of upper covers of minimal elements and the number of lower covers of maximal elements when X is a finite set. The following lemma is needed in finding such numbers.

Lemma 11. *Assume that* $\emptyset \neq Y \subsetneq X$ *and let* $\alpha, \beta \in \mathrm{Fix}(X, Y)$ *with* $\alpha < \beta$. *Then β is an upper cover of α if and only if* $|X\beta \setminus X\alpha| = 1$.

Proof. Write

$$\alpha = \begin{pmatrix} A_i & B_j \\ y_i & b_j \end{pmatrix}. \qquad (27)$$

Since $\alpha < \beta$, we can write

$$\beta = \begin{pmatrix} A_i' & C_j & C_k \\ y_i & b_j & b_k \end{pmatrix}, \qquad (28)$$

where $y_i \in A_i' \subseteq A_i, C_j \subseteq B_j,$ and C_k is contained in either A_i for some i or B_j for some j. We get $|K| = |X\beta \setminus X\alpha|$.

Assume that β is an upper cover of α. If $|X\beta \setminus X\alpha| = 0$, then $X\beta = X\alpha$ which implies that $\beta = \alpha$, a contradiction. For the case $|X\beta \setminus X\alpha| > 1$, we choose $k_0 \in K$ and hence $C_{k_0} \subseteq A_{i_0}$ for some $i_0 \in I$ or $C_{k_0} \subseteq B_{j_0}$ for some $j_0 \in J$. Assume that $C_{k_0} \subseteq A_{i_0}$ (the other case being similar). Let $I' = I \setminus \{i_0\}$ and $K' = K \setminus \{k_0\}$. Define

$$\gamma = \begin{pmatrix} A'_{i_0} \cup C_{k_0} & A'_{i'} & C_j & C_{k'} \\ y_{i_0} & y_{i'} & b_j & b_{k'} \end{pmatrix} \in \mathrm{Fix}\,(X, Y). \tag{29}$$

Since $K' \neq \emptyset$, we get $\alpha < \gamma < \beta$, a contradiction. Therefore, $|X\beta \setminus X\alpha| = 1$.

The converse is proved in similar fashion to Lemma 10 (1).

Let X be a finite set with n elements and Y a nonempty proper subset of X with r elements. If $|Y| = 1$, then Fix(X, Y) has unique minimal element, say $\alpha = \binom{X}{y}$. By Lemma 11, each of upper covers of α is of the form $\begin{pmatrix} X\setminus B & B \\ y & b \end{pmatrix}$, where $\emptyset \neq B \subseteq X \setminus \{y\}$ and $b \in X \setminus Y$. Since there are $(2^{n-1} - 1)$ ways to choose B and $n-1$ choice of b, in this case there are in total $(2^{n-1} - 1)(n - 1)$ upper covers of α.

If $|X \setminus Y| = 1$, then Fix(X, Y) has unique maximal element, the identity map. Let $I = \{1, 2, \dots, n - 1\}$, $Y = \{y_i : i \in I\}$, and $X \setminus Y = \{b\}$. Then each of lower covers of 1_X is of the form $\begin{pmatrix} \{y_{i_0}, b\} & y_{i'} \\ y_{i_0} & y_{i'} \end{pmatrix}$, where $I' = I \setminus \{i_0\}$. Since i_0 can be chosen from I, there are in total $n - 1$ lower covers of 1_X.

Theorem 12. *Assume that* $\emptyset \neq Y \subsetneq X$ *and let* $\alpha \in$ Fix(X, Y). *Then the following statements hold.*

- *If* $\alpha = \binom{A_i}{y_i}$ *is minimal, then there are*

  $$\sum_{i=1}^{r} \left(2^{|A_i|-1} - 1\right)(n - r) \tag{30}$$

  upper covers of α.
- *If* α *is maximal, then there are* $(n - r)(n - 1)$ *lower covers of* α.

Proof. Since Y is a finite set with r elements, $Y = \{y_1, \dots, y_r\}$ and $I = \{1, \dots, r\}$.

- Let $\alpha = \binom{A_i}{y_i}$ be minimal in Fix(X, Y) and β an upper cover of α. Then $|X\beta \setminus X\alpha| = 1$ by Lemma 11; that is, $X\beta = Y \cup \{b\}$ for some $b \in X \setminus Y$. Since π_β must refine π_α, we can write

$$\beta = \begin{pmatrix} A_i' & B \\ y_i & b \end{pmatrix}, \tag{31}$$

where $A_i' \subseteq A_i$ and $\emptyset \neq B \subseteq A_{i_0} \setminus \{y_{i_0}\}$ for some $i_0 \in I$. We claim that $A_i' = A_i$ for all $i \in I \setminus \{i_0\}$. Assume by contrary that there is $i_1 \in I \setminus \{i_0\}$ such that $A_{i_1}' \subsetneq A_{i_1}$. Let $B_1 = A_{i_1} \setminus A_{i_1}'$. So $\emptyset \neq B_1 \cap A_i' \subseteq A_i$ for some $i \neq i_1$, but $B_1 \subseteq A_{i_1}$; that is $A_i \cap A_{i_1} \neq \emptyset$, a contradiction. So we can write

$$\beta = \begin{pmatrix} A_{i'} & A_{i_0} \setminus B & B \\ y_{i'} & y_{i_0} & b \end{pmatrix}, \tag{32}$$

where $I' = I \setminus \{i_0\}$. Since there are $2^{|A_{i_0}|-1} - 1$ ways to choose B and $n - r$ choices of b, in this case β can have $(2^{|A_{i_0}|-1} - 1)(n-r)$ forms, but i_0 can be chosen from $I = \{1, \ldots, r\}$, so that there are in total $\sum_{i=1}^{r}(2^{|A_i|-1} - 1)(n - r)$ upper covers of α.

- Assume that α is maximal. Then α is a bijection and we can write

$$\alpha = \begin{pmatrix} y_i & b_j \\ y_i & c_j \end{pmatrix}, \tag{33}$$

where $J = \{1, \ldots, n - r\}$ and $\{b_j : j \in J\} = X \setminus Y = \{c_i : j \in J\}$. Let β be a lower cover of α. Then $|X\alpha \setminus X\beta| = 1$; that is, $X\beta = X\alpha \setminus \{c_{j_0}\}$ for some $j_0 \in J$. Let $J' = J \setminus \{j_0\}$ and $b_{j'} \in \{b_j : j \in J'\}$. So $b_{j'}\alpha = c_{j'} \in X\beta \setminus Y$, then $b_{j'}\alpha = b_{j'}\beta$ since $\beta < \alpha$. Hence $x\alpha = x\beta$ for all $x \in X \setminus \{b_{j_0}\}$ and $b_{j_0}\beta = y_{i_0}$ for some $i_0 \in I$ or $b_{j_0}\beta = b_{j_1}\beta = c_{j_1}$ for some $j_1 \in J'$. Thus

$$\beta = \begin{pmatrix} \{y_{i_0}, b_{j_0}\} & y_{i'} & b_{j'} \\ y_{i_0} & y_{i'} & c_{j'} \end{pmatrix}, \tag{34}$$

where $I' = I \setminus \{i_0\}$, or

$$\beta = \begin{pmatrix} y_i & \{b_{j_1}, b_{j_0}\} & b_k \\ y_i & c_{j_1} & c_k \end{pmatrix}, \tag{35}$$

where $K = J \setminus \{j_0, j_1\}$. For the first form and the second form, the numbers of ways of placing b_{j_0} is r and $n - r - 1$, respectively. So the total number of ways of placing b_{j_0} is $n - 1$. But j_0 varies in the index set J; hence there are in total $(n - 1)(n - r)$ lower covers of α.

ACKNOWLEDGMENTS

This research was supported by Chiang Mai University.

REFERENCES

1. R. Hartwig, “How to partially order regular elements,” *Mathematica Japonica*, vol. 35, pp. 1–13, 1980.
2. K. S. Nambooripad, “The natural partial order on a regular semigroup,” *Proceedings of the Edinburgh Mathematical Society (Series 2)*, vol. 23, no. 3, pp. 249–260, 1980.
3. H. Mitsch, “A natural partial order for semigroups,” *Proceedings of the American Mathematical Society*, vol. 97, no. 3, pp. 384–388, 1986.
4. G. Kowol and H. Mitsch, “Naturally ordered transformation semigroups,” *Monatshefte f\”ur Mathematik*, vol. 102, no. 2, pp. 115–138, 1986.
5. C. Namnak and P. Preechasilp, “Natural partial orders on the semigroup of binary relations,” *Thai Journal of Mathematics*, vol. 4, no. 3, pp. 39–50, 2006.
6. V. H. Fernandes and J. Sanwong, “On the ranks of semigroups of transformations on a finite set with restricted range,” *Algebra Colloquium*, vol. 21, no. 3, pp. 497–510, 2014.
7. K. Sangkhanan and J. Sanwong, “Partial orders on semigroups of partial transformations with restricted range,” *Bulletin of the Australian Mathematical Society*, vol. 86, no. 1, pp. 100–118, 2012.
8. P. Honyam and J. Sanwong, “Semigroups of transformations with fixed sets,” *Quaestiones Mathematicae. Journal of the South African Mathematical Society*, vol. 36, no. 1, pp. 79–92, 2013.
9. Y. Chaiya, P. Honyam, and J. Sanwong, “Maximal subsemigroups and finiteness conditions on transformation semigroups with fixed sets,” *Turkish Journal of Mathematics*, In press.
10. P. M. Higgins, “The Mitsch order on a semigroup,” *Semigroup Forum*, vol. 49, no. 2, pp. 261–266, 1994.

Chapter

CYCLIC SOFT GROUPS AND THEIR APPLICATIONS ON GROUPS

3

Hacı Aktaş[1] and Şerif Özlü[2]

[1]Department of Mathematics, Faculty of Science, Erciyes University, 38039 Kayseri, Turkey
[2]Kilis 7 Aralık University, 79000 Kilis, Turkey

ABSTRACT

In crisp environment the notions of order of group and cyclic group are well known due to many applications. In this paper, we introduce order of the soft groups, power of the soft sets, power of the soft groups, and cyclic soft group on a group. We also investigate the relationship between cyclic soft groups and classical groups.

Citation: Hacı Aktaş and Şerif Özlü, "Cyclic Soft Groups and Their Applications on Groups", Emerging Trends in Soft Set Theory and Related Topics, volume 2014, article ID 437324, https://doi.org/10.1155/2014/437324.

INTRODUCTION

Most of our real life problems in economics, engineering, environment, social science, and medical science involve imprecise data that contain uncertainties. To solve these kinds of problems, it is quite difficult to successfully use classical methods. However, there are some well-known theories (probability, fuzzy sets [1], vague sets [2], rough sets [3], etc.) which can be considered as mathematical tools for dealing with uncertainties. All these theories have their inherent difficulties pointed out by Molodtsov [4].

The theory of soft sets, proposed by Moldtsov [4], is an extension of set theory for the study of intelligent systems characterized by insufficient and incomplete information. His pioneer paper has undergone tremendous growth and applications in the last few years. Maji et al. [5] give an application of soft set theory in a decision making problem by using the rough sets and they conducted a theoretical study on soft sets in a detailed way [6]. Chen et al. [7] proposed a reasonable definition of parameterizations reduction of soft sets and compared them with the concept of attributes reduction in rough set theory.

The algebraic structures of set theories which deal with uncertainties have been studied by some authors. Rosenfeld [8] proposed fuzzy groups to establish results for the algebraic structures of fuzzy sets. Rough groups are defined by Biswas and Nanda [9], and some others (i.e., Bonikowaski [10] and Iwinski [11]) studied algebraic properties of rough sets. Fuzzification of algebraic structures was studied by many authors [8, 12, 13].

Many papers on soft algebras have been published since Aktaş and Çağman [14] introduced the notion of a soft group in 2007. Recently, Jun et al. [15] studied soft ideals and idealistic soft BCK/BCI-algebras. Acar et al. [16] introduced initial concepts of soft rings. Aygünoğlu and Aygün [17] introduced the concept of fuzzy soft group and, in the meantime, they studied its properties and structural characteristics. Atagün and Sezgin [18] introduced and studied the concepts of soft subrings, soft ideal of a ring, and soft subfields of a field.

Our interest, in this paper, is to define order of the soft group and cyclic soft group by using definition of the soft group that was defined in [14]. We then find out the relationships between cyclic soft groups and classical groups. Finally, we conclude the study with suggestions for future work.

PRELIMINARIES

The following definitions and preliminaries are required in the sequel of our work and they are presented in brief.

Throughout this work, U is an initial universe set, E is a set of parameters, (U) is the power set of U, $A \subset E$, and G denotes a group with identity e.

Definition 1 (see [4]). A pair (F, A) is called a soft set over U, where F is a mapping given by

$$F : A \longrightarrow P(U). \tag{1}$$

In other words, a soft set over U is a parameterized family of subsets of the universe U.

Definition 2 (see [6]). For two soft sets (F, A) and (G, B) over U, (F, A) is called a soft subset of (G, B), if

- $A \subset B$ and
- $\forall \varepsilon \in A$; (ε) and (ε) are identical approximations.

It is denoted by $(F, A)\tilde{\subset}(G, B)$.

(F, A) is called a soft superset of (G, B) if (G, B) is a soft subset of (F, A). It is denoted by $(F, A)\tilde{\supset}(G, B)$.

Definition 3 (see [6]). If (F, A) and (G, B) are two soft sets, then (F, A) AND (G, B) is denoted $(F, A) \wedge (G, B)$. $(F, A) \wedge (G, B)$ is defined as $(H, A \times B)$, where $(\alpha, \beta) = (\alpha) \cap G(\beta)$, for all $(\alpha, \beta) \in A \times B$.

Definition 4 (see [6]). If (F, A) and (G, B) are two soft sets, then (F, A) OR (G, B), denoted by $(F, A) \vee (G, B)$, is defined by $(F, A) \vee (G, B) = (H, A \times B)$, where $(\alpha, \beta) = (\alpha) \cup G(\beta)$, $\forall(\alpha, \beta) \in A \times B$.

Definition 5 (see [6]). Union of two soft sets of (F, A) and (G, B) over U is the soft set (H, C), where $C = A \cup B$ and $\forall e \in C$,

$$H(e) = \begin{cases} F(e), & \text{if } e \in A - B, \\ G(e), & \text{if } e \in B - A, \\ F(e) \cup G(e), & \text{if } e \in A \cap B. \end{cases} \tag{2}$$

It is denoted by $(F, A)\tilde{\cup}(G, B) = (H, C)$.

Definition 6 (see [14]). Let (F, A) be a soft set over G. Then (F, A) is said to be a soft group over G if and only if (x) is subgroup of G for all $x \in A$.

Definition 7 (see [14]). One considers the following.

- (F, A) is said to be an identity soft group over G if $(x) = \{e\}$ for all $x \in A$, where e is the identity element of G.

- (F, A) is said to be an absolute soft group over G if $(x) = G$ for all $x \in A$.

Definition 8 (see [14]). Let (F, A) and (H, K) be two soft groups over G. Then (H, K) is a soft subgroup of (F, A), written as $(H, K)\tilde{<}(F, A)$, if

- $K \subset A$,
- (x) is a subgroup of (x) for all $x \in K$.

Definition 9 (see [14]). Let (F, A) and (H, B) be two soft groups over G and K, respectively, and let $f{:}G \to K$ and $g{:}A \to B$ be two functions. Then one says (f, g) is a soft homomorphism and (F, A) is soft homomorphic to (H, B), denoted by $(F, A) \sim (H, B)$, if the following conditions are satisfied:

- f is a homomorphism from G onto K,
- g is a mapping from A onto B,
- $(F(x)) = (g(x))$ for all $x \in A$.

In this definition, if f is an isomorphism from G to K and g is a one-to-one mapping from A onto B, then we say that (f, g) is a soft isomorphism and (F, A) is soft isomorph to (H, B) which is denoted by $(F, A) \simeq (H, B)$. The image of soft group (F, A) under soft homomorphism (f, g) will be denoted by $((F), (A))$.

Definition 10 (see [14]). Let (F, A) and (H, B) be two soft groups over G and K, respectively. The product of soft groups (F, A) and (H, B) is defined as $(F, A) \times (H, B) = (U, A \times B)$, where $(x, y) = (x) \times H(y)$, for all $(x, y) \in A \times B$.

THE ORDER OF SOFT GROUPS

Since the elements of a cyclic group are the powers of the element, properties of cyclic groups are closely related to the properties of the powers of an element. In this paper, we define order of the soft group by using definition of the soft group that was defined in [14]. We then investigate their properties.

Definition 11. A pair (F, A) is called surjective soft set over U, where F is a surjective mapping given by $: A \to P(U)$.

Throughout this study, G denotes a group and the soft set (F, A) will be a surjective soft set. The element (a) is used instead of the element $(a, (a))$ of (F, A).

Definition 12. Let (F, A) be a soft set over G and $(x) \in (F, A)$ for $x \in A$. Then $(x) = \{a^n : a \in F(x), n \in Z\}$ is called n- power of $F(x)$.

Example 13. Let (F, A) be a soft set over S_3, where $A = S_3$, and let

$$\begin{aligned}(F, A) = \ &\{F(e) = \{e\}, F(12) = \{e, (12)\}, \\ &F(13) = \{e, (13)\}, F(23) = \{e, (23)\}, \\ &F(123) = \{e, (123), (132)\}\}\end{aligned} \qquad (3)$$

be a soft set over group S_3. And the third power of (123) is $(123)^3 = \{e^3, (123)^3, (132)^3\} = \{e, e, e\} = \{e\}$.

Of course, when the group is additive, the nth power of (x) will be written by $n(x) = \{na : a \in F(x), n \in Z\}$.

Theorem 14. *Let (F, A) be a soft set over G and $(x), (y) \in (F, A)$ for $x, y \in A$. Then, for all $n \in Z$,*

- $(F(x) \cap F(y))^n \subseteq F(x)^n \cap F(y)^n$ *, for all $n \in Z$,*
- $(F(x) \cup F(y))^n = F(x)^n \cup F(y)^n$ *, for all $n \in Z$,*
- $(F(x) \times F(y))^n = (x) \times F(y)^n$ *, for all $n \in Z$.*

Proof. Let $a^n \in ((x) \cap (y))^n$, for $n \in Z$. From Definition 12 $a \in (x) \cap (y)$ and $a^n \in F(x)^n$ and $a^n \in F(y)^n$. This means that $a^n \in (x) \cap F(y)^n$. This completes the proof. Theorem 14(2) and Theorem 14(3) can be proved similarly by using Definition 12.

In general, the opposite of Theorem 14(1) is not true. We illustrate an example of this situation.

Example 15. Let $A = \{0, 1\}$ and let : $A \to P(Z)$ be a function such that $F(0) = \{2k : k \in Z\}$ and $F(1) = \{2k + 1 : k \in Z\}$. The intersection is $(0) \cap (1) = \emptyset$. So $((0) \cap (1))^2 = \emptyset$. On the other hand $(0)^2 \cap (1)^2 \neq \emptyset$. Consequently $((x) \cap (y))^n \neq F(x)^n \cap F(y)^n$.

Definition 16. Let(F, A) be a soft set over$G$and $F(x) \in (F, A)$. If there is a positive integer n such that $(x) = \{e\}$, then the least such positive integer n is called the order of $F(x)$. If no such n exists, then (x) has infinite order. The order of (x) is denoted by $|(x)|$.

If (F, A) is a soft group over G, then the order of $(x) \in (F, A)$ coincides with the order of (x), which is subgroup of G. Of course if there is any element x in A such that $(x) = \{e\}$, then the order of (x) is 1.

Example 17. In Example 13 the order of element (123) is 3.

Let (F, N) be a soft group over group of integer numbers Z, where F is a mapping from natural numbers N to (Z) such that $(n) = nZ$ for all $n \in N$. There is no any positive integer m such that $(n) = \{0\}$, so $F(n)$ has infinite order for all $n \in N - \{0\}$.

Theorem 18. *Let G be finite group and (F, A) a soft group over G. Then, the orders of elements of (F, A) are finite.*

Proof. It is straightforward.

Theorem 19. *Let (F, A) be a soft set over finite group G and $(x) \in (F, A)$ for $x \in A$. Then, the order of (x) is the least common multiple (LCM) of order of elements of (x).*

Proof. Let n be the order of (x). Then $(x) = \{e\}$. This means that $a^n = e$ for all $a \in (x)$. We know from classical group theory that $|a| \mid n$, namely, a, divides n for all $a \in (x)$. Thus, n is common multiple of elements of (x). Let m be another common multiple of elements of (x). Then, by reason of $a^m = e$ for all $a \in (x)$, $(x)^m = \{e\}$. However, since n is the least number that satisfies the condition $(x) = \{e\}$, hence $n|m$. This completes the proof.

Theorem 20. *Let G be finite group, (F, A) a soft group over G, and (x) and (y) the elements of (F, A). Then, for all $x, y \in A$, one has the following:*

- $|F(x) \cap F(y)| \leq GCD(|F(x)|, |F(y)|)$ *for* $x, y \in A$,
- $|F(x) \cup F(y)| = LCM(|F(x)|, |F(y)|)$ *for* $x, y \in A$,
- $|(x) \times F(y)| = |F(x)||F(y)|$ *for* $x, y \in A$.

Proof. We consider the following.

- $F(x) \cap F(y)$ is subgroup of $F(x)$ and $F(y)$, so $|F(x) \cap F(y)| \mid |F(x)|$ and $|F(x) \cap F(y)| \mid |F(y)|$. It follows $|(x) \cap F(y)| \leq GCD(|F(x)|, |F(y)|)$.
- Let $|(x) \cup F(y)| = k$, $|F(x)| = m$, and $|F(y)| = n$. From Theorem 14$((x) \cup (y))^k = F(x)^k \cup F(y)^k = \{e\}$. This follows $n \mid k$ and $m \mid k$. Thus k is a common multiple of m and n. Let t be another common multiple of m and n. Consider $((x) \cap (y))^t = F(x)^t \cup F(y)^t = \{e\} \cup \{e\} = \{e\}$. Since k is the least positive integer that satisfied the condition $((x) \cup (y))^k = \{e\}$, k divides t. Hence k is LCM order of (x) and (y). This completes the proof.
- Since (x) and (y) are subgroups of G, it is seen easily.

Definition 21. Let G be group and (F, A) soft set over G. The set

$$(F, A)^n = \{F(x)^n : x \in A, n \in Z\} \qquad (4)$$

is called *n*th power of soft set (F, A).

Example 22. Let (F, A) be a soft set over S_3 defined in Example 13. Then, the second power of (F, A) is that $(F, A)^2 = \{F(e)^2 = F(e)$, and $F(12)^2 = F(e)$, $F(13)^2 = F(e)$, $F(23)^2 = F(e)$, $F(123)^2 = F(132)\}$.

Theorem 23. *Let* (F, A) *and* (E, B) *be two soft sets over . Then,*

- $((F, A) \vee (E, B))^n = (F, A)^n \vee (E, B)^n$,
- *if* $A \subseteq B$ *and, for all* $a \in A$, $F(a)$ *and* $E(a)$ *are identical approximations, then* $((F, A) \wedge (E, B))^n \subseteq (F, A)^n \wedge (E, B)^n$.

Proof. We consider the following.

- Suppose that $(F, A) \vee (E, B) = (H, A \times B)$ and $(F, A) \vee (E, B) = (T, A \times B)$. We can write $((F, A) \vee (E, B)) = (H, A \times B)$. Using Definition 21 and Theorem 14 we have

$$\begin{aligned}(H, A \times B)^n &= \{H(a,b)^n : (a,b) \in A \times B\}\\ &= \{(F(a) \cup E(b))^n : (a,b) \in A \times B\}\\ &= \{F(a)^n \cup E(b)^n : (a,b) \in A \times B\}\\ &= \{T(a,b) : (a,b) \in A \times B\}\\ &= (F, A)^n \vee (E, B)^n.\end{aligned} \tag{5}$$

- Suppose that $(F, A) \wedge (E, B) = (H, A \times B)$ and $(F, A) \wedge (E, B) = (T, A \times B)$. Using the same arguments in (1), we have

$$\begin{aligned}(H, A \times B)^n &= \{H(a,b)^n : (a,b) \in A \times B\}\\ &= \{(F(a) \cap E(b))^n : (a,b) \in A \times B\}\\ &\subseteq \{F(a)^n \cap E(b)^n : (a,b) \in A \times B\}\\ &= \{T(a,b) : (a,b) \in A \times B\}\\ &= (F, A)^n \wedge (E, B)^n.\end{aligned} \tag{6}$$

If (F, A) and (E, B) are both soft groups, then, for all $a, b \in A$, $F(a)$ and $F(b)$ are all subgroups of G, and so $F(a)$ and $E(b)$ contain the identity element e of G. Thus, the set $(a) \cap (b)$ contains at least e; hence $(F(a) \cap E(b))^n \neq \emptyset$. It means that if we take inTheorem 23(2)(F, A) and (E, B) as soft groups, not soft sets, then the extra condition can be added in Theorem 23(2).

In classical groups, the order of a group is defined as the number of elements it contains. But in soft groups, it differs from classical groups.

Definition 24. Let (F, A) be soft group over G. If G is a finite group, then the least common multiple of orders of elements of (F, A) is called order of (F, A). If G is an infinite group, then the order of (F, A) is defined to be the number of elements of (F, A) and the order of soft group (F, A) is denoted by $|(F, A)|$.

Of course, if (F, A) is surjective, then the number of elements of (F, A) is the number of elements in A.

Example 25. In Example 13 the order of (F, A) is 6 and in Example 17 if we chose $N = N_4 = \{0, 1, 2, 3, 4\}$ and $(n) = nZ$, for $n \in N_4$, then the order of soft group (F, N_4) is 5.

We give the following results similar to Lagrange Theorem in group theory.

Theorem 26. *Let (F, A) be a soft group over a finite group G and $(x) \in (F, A)$. Then one has the following.*

- *The order of (x) divides the order of (F, A). In particular, $(x)^{|(F,A)|} = \{e\}$.*
- *The order of (F, A) divides the order of G.*

When G is a finite group, then the order (F, A) is a finite and the orders of elements of (F, A) divide the order of (F, A). However when G is an infinite group, then the order of (F, A) can be finite or infinite. Hence, Theorem 26 is not true for infinite groups.

Theorem 27. *Let G be a finite group and let (F, A) and (E, B) be two soft groups over G. Then $|(F, A) \wedge (E, B)| \leq |(F, A)|$ and $|(F, A) \wedge (E, B)| \leq |(E, B)|$.*

Proof. Suppose that $(F, A) \wedge (E, B) = (H, C)$, where $C = A \times B$. Using fundamental theorems and definitions in group theory and Definitions 24 and 3, we have

$$\begin{aligned} |(F, A) \wedge (E, B)| &= \mathrm{LCM}\left(H\left(a_i, b_i\right)\right), \text{for } \left(a_i, b_i\right) \in A \times B \\ &= \mathrm{LCM}\left(F\left(a_i\right) \cap E\left(b_i\right)\right) \\ &\leq \mathrm{LCM}\left(F\left(a_i\right)\right) = |(F, A)|. \end{aligned} \tag{7}$$

The other inequality is shown similarly.

CYCLIC SOFT GROUPS

The class of cyclic groups is an important class in group theory. In this section, we study soft groups which are generated by one element of (G). We define cyclic soft group and prove some of their properties which are analogous to the crisp case.

Definition 28. Let (F, A) be a soft group over G and X an element of (G). The set $\{(a, \langle x \rangle) : F(a) = \langle x \rangle, x \in X\}$ is called a soft subset of (F, A)

generated by the set X and denoted by $\langle X\rangle$. If $(F, A) = \langle X\rangle$, then the soft group (F, A) is called the cyclic soft group generated by X.

If (F, A) is a cyclic soft group over G, then we can write it in this form $(F, A) = \{(a) = \langle x\rangle : a \in A, x \in G\}$, where $\{x \in G\}$ is element of $P(G)$. That is to say, if all the elements of (F, A) are generated by any elements of X of (G), then (F, A) is a cyclic soft group over G.

If G is a cyclic group, then (F, A) is a soft cyclic group over G since all subgroups of cyclic group are cyclic but the reverse is not always true.

As an illustration, let us consider the following example.

Example 29. Let $G = S_3$ be the symmetric group and $A = \{e$, (12), (13), (23), (123)$\}$ the set of parameters. If we construct a soft set (F, A) over G such that $(x) = \{y \in\ : y = x^n , n \in Z\}$ for all $x \in A$, then one can easily show that (F, A) is a soft cyclic group over G; however G is not a cyclic group.

In the following theorem, we can give some properties of cyclic soft groups that has similar features of the classical cyclic groups.

Theorem 30. *One considers the following.*

- *If (F, A) is a finite cyclic soft group generated by X, then $|(F, A)| = LCM(|x_i|)$, where $x_i \in X$.*
- *If (F, A) is an infinite cyclic soft group generated by X, then $|(F, A)| = |X|$.*
- *If (F, A) is an identity soft group, then it is a cyclic soft group generated by $\{e\}$.*
- *Let (F, A) be an absolute soft group defined on G. Then, (F, A) is a cyclic soft group if and only if G is a cyclic group.*
- *Let (F, A) be a soft group on G. If the order of G is prime, then (F, A) is a cyclic soft group.*
- *A soft subgroup of a cyclic soft group is cyclic soft group.*

Proof. It is easily seen from Definitions 24, 7, and 8.

Theorem 31. *Let (f, g) be a soft homomorphism of the soft group (F, A) over G into the soft group (H, B) over K. If (F, A) is a cyclic soft group over G, then $((F), (A))$ is a cyclic soft group over K.*

Proof. First, we show that $((F), (A))$ is a soft group over K. Since f is a homomorphism from G to K, $((x)) = H(g(x))$ is a subgroup of K for all $g(x) \in g(A)$. Thus $((F), (A))$ is a soft group over K. Since (x) is cyclic subgroup for all x in A, image of (x) under f is cyclic; that is, $f(F(x)) = H(g(x))$ is cyclic subgroup of K for all $g(x) \in g(A)$. Consequently, $((F), (A))$ is a cyclic soft group over K.

Theorem 32. *Let (F, A) and (H, B) be two soft isomorphic soft groups over G and K, respectively. If (F, A) is a cyclic soft group, then so is (H, B).*

Proof. First of all note that since (F, A) is a soft isomorphic to (H, B), there is an f homomorphism from G to K such that $((x)) = H(g(x))$ for all $x \in A$, where g is a one-to-one mapping from A onto B. Since (x) is a cyclic subgroup for all x in A and f is a homomorphism, then $(g(x))$ is a cyclic subgroup of K. Thus, all elements of (H, B) are cyclic. This result completes the proof.

Theorem 33. *If (F, A) and (H, B) are two cyclic soft groups over G, then $(F, A) \wedge (H, B)$ is a cyclic soft group over G.*

Proof. Let $(F, A) \wedge (H, B) = (E, A \times B)$, where $(\alpha, \beta) = (\alpha) \cap H(\beta)$, $\forall(\alpha, \beta) \in A \times B$. Since $F(\alpha)$ and $H(\beta)$ are cyclic subgroups of G for all $\alpha \in A$ and $\beta \in B$ and $F(\alpha) \cap H(\beta)$ is a subgroup of both $F(\alpha)$ and $H(\beta)$, $F(\alpha) \cap H(\beta)$ is cyclic subgroup of G for all $(\alpha, \beta) \in A \times B$. Hence, $(H, A \times B)$ is cyclic soft group over G.

Theorem 34. *Let (F, A) and (H, B) be two cyclic soft groups over G and $A \cap B = \emptyset$. Then, $(F, A)\tilde{\cup}(H, B)$ is a cyclic soft group over G.*

Proof. It is trivial.

Theorem 35. *Let (F, A) and (H, B) be two cyclic soft groups of finite orders m and n over G and K, respectively. If m and n are relatively prime, then the product $(F, A) \times (H, B)$ is a cyclic soft group.*

Proof. Let$(m, n) = 1$. According to LagrangeTheorem, $|(x)|$ divides m and $|H(y)|$ divides n for all $x \in A$ and for all $y \in B$. Since $(m, n) = 1$, $|(x)|$ and $|H(y)|$ are relatively prime. So $(x) \times (y)$ is cyclic group for all $(x, y) \in A \times B$. This completes the proof.

CONCLUSION

In this paper, we have expanded the soft set theory. We have focused on order of the soft groups and investigated relationship between the order of soft groups and the order of classical groups. Additionally, we have studied the algebraic properties of cyclic soft groups with respect to a group structure. Our future work will focus on the relationships between cyclic soft groups and other algebraic structures such as rings and fields.

REFERENCES

1. L. A. Zadeh, “Fuzzy sets,” *Information and Computation*, vol. 8, pp. 338–353, 1965.
2. W. Gau and D. J. Buehrer, “Vague sets,” *IEEE Transactions on Systems, Man and Cybernetics*, vol. 23, no. 2, pp. 610–614, 1993.
3. Z. Pawlak, “Rough sets,” *International Journal of Computer and Information Sciences*, vol. 11, no. 5, pp. 341–356, 1982.
4. D. Molodtsov, “Soft set theory—first results,” *Computers & Mathematics with Applications*, vol. 37, no. 4-5, pp. 19–31, 1999.
5. P. K. Maji, A. R. Roy, and R. Biswas, “An application of soft sets in a decision making problem,” *Computers & Mathematics with Applications*, vol. 44, no. 8-9, pp. 1077–1083, 2002.
6. P. K. Maji, R. Biswas, and A. R. Roy, “Soft set theory,” *Computers & Mathematics with Applications*, vol. 45, no. 4-5, pp. 555–562, 2003.
7. D. Chen, E. C. C. Tsang, D. S. Yeung, and X. Wang, “The parameterization reduction of soft sets and its applications,” *Computers & Mathematics with Applications*, vol. 49, no. 5-6, pp. 757–763, 2005.
8. A. Rosenfeld, “Fuzzy groups,” *Journal of Mathematical Analysis and Applications*, vol. 35, pp. 512–517, 1971.
9. R. Biswas and S. Nanda, “Rough groups and rough subgroups,” *Bulletin of the Polish Academy of Sciences Mathematics*, vol. 42, no. 3, pp. 251–254, 1994.
10. Z. Bonikowski, “Algebraic structures of rough sets,” in *Rough Sets, Fuzzy Sets and Knowledge Discovery*, W. P. Ziarko, Ed., pp. 242–247, Springer, Berlin, Germany, 1994.
11. T. B. Iwinski, “Algebraic approach to rough sets,” *Bulletin of the Polish Academy of Sciences Mathematics*, vol. 35, no. 9-10, pp. 673–683, 1987.
12. S. Abdullah, M. Aslam, T. A. Khan, and M. Naeem, “A new type of fuzzy normal subgroups and fuzzy cosets,” *Journal of Intelligent & Fuzzy Systems*, vol. 25, no. 1, pp. 37–47, 2013.
13. S. Abdullah, M. Aslam, T. A. Khan, and M. Naeem, “A note on ordered semigroups characterized by their ((ϵ, ϵ Vq)),” *UPB Scientific Bulletin A*, vol. 75, pp. 41–44, 2013.

14. H. Aktaş and N. Çağman, “Soft sets and soft groups,” *Information Sciences*, vol. 177, no. 13, pp. 2726–2735, 2007.

15. Y. B. Jun and C. H. Park, “Applications of soft sets in ideal theory of BCK/BCI-algebras,” *Information Sciences*, vol. 178, no. 11, pp. 2466–2475, 2008.

16. U. Acar, F. Koyuncu, and B. Tanay, “Soft sets and soft rings,” *Computers & Mathematics with Applications*, vol. 59, no. 11, pp. 3458–3463, 2010.

17. A. Aygünoǧlu and H. Aygün, “Introduction to fuzzy soft groups,” *Computers and Mathematics with Applications*, vol. 58, no. 6, pp. 1279–1286, 2009.

18. A. O. Atagün and A. Sezgin, “Soft substructures of rings, fields and modules,” *Computers & Mathematics with Applications*, vol. 61, no. 3, pp. 592–601, 2011.

Chapter

FACTORIZATION OF GROUPS INVOLVING SYMMETRIC AND ALTERNATING GROUPS

4

M. R. Darafsheh[1,2] and G. R. Rezaeezadeh[3]

[1] Center for Theoretical Physics and Mathematics, AEOI, Iran
[2] Department of Mathematics and Computer Sciences, Faculty of Sciences, University of Tehran, Iran
[3] Department of Mathematics, Faculty of Basic Sciences, University of Tarbiat Moddares, Iran

ABSTRACT

We obtain the structure of finite groups of the form $G = AB$ where B is a group isomorphic to the symmetric group on n letters S_n, $n \geq 5$ and A is a group isomorphic to the alternating group on 6 letters.

Citation: M. R. Darafsheh and G. R. Rezaeezadeh, "Factorization of Groups involving Symmetric and Alternating Groups", International Journal of Mathematics and Mathematical Sciences, volume 27, article ID 461380, https://doi.org/10.1155/S0161171201010754.

INTRODUCTION

For a finite group G if there exist subgroups A and B of G such that $G = AB$, then G is called a factorizable group. Of course if neither of A nor B is contained in the other, then the factorization is called nontrivial. A knowledge of the factorizations of finite simple groups will help to investigate the general theory of factorizing finite groups. All possible factorizations of sporadic simple groups have been obtained in [4] and those of simple groups of Lie type of Lie rank 1 or 2 in [3].

A factorization $G = AB$ where both A and B are maximal subgroups of G is called a maximal factorization of G. In [9], all the maximal factorizations of all the finite simple groups and their automorphism groups have been determined completely.

In another direction some results have been obtained assuming $G = AB$ is a factorization of G with A and B simple subgroups of G. For example, in [8] finite groups $G = AB$ where both A and B are isomorphic to the simple group of order 60 are classified, and in [10] finite groups $G = AB$ where A is a non-abelian simple group and B A_5 are determined. In [5], $G = AB$ where A and B are simple groups of small order are considered.

In a series of papers, Walls considered groups which are a product of simple groups [13, 14]. In [15], groups which are product of a symmetric group and a group isomorphic to A_5 are classified. This result is interesting because in the factorization $G = AB$ one of the factors is not a simple group. Motivated by this result, in this paper we classify all groups G which are product of subgroups A and B such that $A \cong A_6$ and $B \cong S_n, n \geq 6$. In this paper, A_n and S_n are the alternating and symmetric groups on n letters, respectively, and all groups are assumed to be finite.

PRELIMINARY RESULTS

Now A_6 is a simple group of order 360 and it is easy to verify that the order of any proper subgroup of A_6 is one of the numbers 1, 2, 3, 4, 5, 6, 8, 9, 10, 12, 18, 24, 36, or 60. Therefore the size of sets on which A_6 acts transitively and faithfully is one of the numbers 360, 180, 120, 90, 72, 60, 45, 40, 36, 30, 20, 15, 10, or 6. Also since $A_6 \cong L_2(9)$, A_6 has a 2-transitive action on a set of 10 points and by consulting [10] one can see that if A_6 acts k-transitively, $k \geq 2$, on a set of cardinality m, then either $m = 10$, $k = 2$ or $m = 6$ and $k = 2,3$, or 4. Now by [14, Lemma 7] we have the following decomposition.

Lemma 2.1. *For n a positive integer, $S_{n+1} = A_6\, S_n$ and $A_{n+1} = A_6\, A_n$ if and only if n =* 5, 9, 14, 19, 29, 35, 39, 44, 59, 71, 89, 119, 179, *or* 359. *We can write $A_{10} = AB$ where $A \cong A_6$ and $B \cong A_8$. Further, we can write $S_{10} = AB$ and $A_{10} \times Z_2 = AB$ where $A \cong A_6$ and $B \cong S_8$.*

The only nontrivial decomposition $A_m = AB$, where $A \cong A_6$ and $B \cong A_n$, occurs if and only if $m = n + 1$, where n is one of the numbers mentioned in Lemma 2.1 or $m = 10$ and $n = 8$. To see this one can use [14, Theorem 9]. Because according to this theorem one of the groups in the decomposition $G = AB$, say A, must be a k-transitive permutation group and according to what we said earlier all the k-transitive permutation representations of $A \cong A_6$ are known.

For our work it is necessary to know if it is possible to decompose an alternating group as the product of A_6 and S_n.

Lemma 2.2. *It is not possible to decompose the alternating group A_m, $m \geq 7$, as the product of A_6 and a symmetric group S_n, $n > 1$, unless $m = 10$ and $n = 8$.*

Proof. According to [9, Theorem D], if A_m acts naturally on a set **Ω** of cardinality m, and $A_m = A_6\, S_n$, then there are two possibilities.

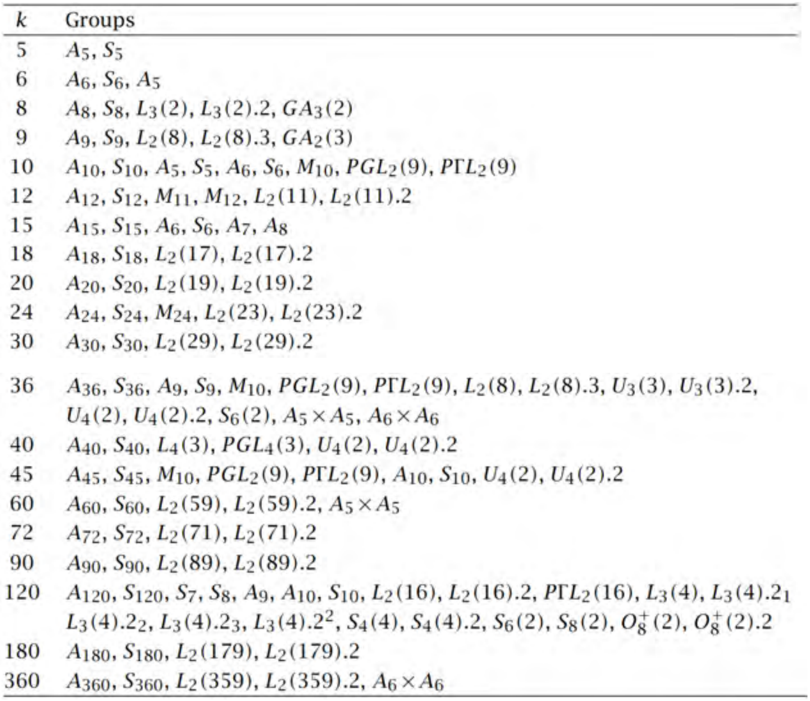
Case (i). $A_{m-k} \trianglelefteq S_n \leq S_{m-k} \times S_k$ for some k, $1 \leq k \leq 5$, and A_6 is k-homogeneous on Ω. If $k = 1$, then $A_{m-1} \trianglelefteq S_n \leq S_{m-1}$ and it is easy to deduce $n = m-1$. Therefore $A_{n+1} = A_6\, S_n$ and so, $S_n \leq A_{n+1}$ from which it follows that $n = 1$ which is not the case. If $k \geq 2$ then by [7] A_6 can only be k-transitive for k = 2,3, or 4. If $k = 2$, then m = 6 or 10. Since we have assumed that $m \geq 7$, therefore, if $m = 10$, then $A_{10} = A_6\, S_n$ and from $A_8 \trianglelefteq S_n \leq S_8 \times S_2$ we obtain $n \geq 8$ and the order consideration in $A_{10} = A_6\, S_n$ leads to $A_{10} = A_6\, S_8$. If k = 3 or 4, then $m = 6$ and again $A_6 = A_6\, S_n$, a contradiction. Since in [9, Theorem D] the role of S_n and A_6 may be interchanged, hence we may assume that $A_{m-k} \trianglelefteq A_6 \leq S_{m-k} \times S_k$ and S_n is k-homogeneous for some $1 \leq k \leq 5$. However, a contradiction is obtained in this case again.

Case (ii). m = 6,8, or 10. If $m = 6$ then $A_6 = A_6\, S_n$, a contradiction. If $m = 8$, then $A_8 = A_6\, S_n$ from which it follows that $n \geq 7$, but it is easy to see that A_8 has no subgroup isomorphic to S_7 . If $m = 10$, then $A_{10} = A_6\, S_n$ from which it follows that n = 7 or 8.

Now to rule out the case $n = 7$. We will use [16, Result 1.4]. Using the notation used in [16] the decomposition $A_{10} = A_6\, S_7$ is exact and we have $p = 7$ and $|\Delta| = k = 3$ and therefore A_6 must be 3-homogeneous which is impossible by [7] unless A_6 acts on Ω, $|\Omega| = 6$ in a natural way and this is also a contradiction. However, if $n = 8$, then $A_{10} = A_6\, S_8$ and this possibility holds because by Lemma 2.1 we have $A_{10} = A_6\, A_8$ and since A_{10} has a subgroup isomorphic to S_8, namely $A_8\langle(1\ 2)(9\ 10)\rangle$ we obtain $A_{10} = A_6S_8$.

In this paper we also use the following result which can be proved using the sub-group structure of $L_2\,(q)$ given in [6].

Table 1. Primitive groups of degree $k \geq 5$, $k \mid 360$.



k	Groups
5	A_5, S_5
6	A_6, S_6, A_5
8	$A_8, S_8, L_3(2), L_3(2).2, GA_3(2)$
9	$A_9, S_9, L_2(8), L_2(8).3, GA_2(3)$
10	$A_{10}, S_{10}, A_5, S_5, A_6, S_6, M_{10}, PGL_2(9), P\Gamma L_2(9)$
12	$A_{12}, S_{12}, M_{11}, M_{12}, L_2(11), L_2(11).2$
15	$A_{15}, S_{15}, A_6, S_6, A_7, A_8$
18	$A_{18}, S_{18}, L_2(17), L_2(17).2$
20	$A_{20}, S_{20}, L_2(19), L_2(19).2$
24	$A_{24}, S_{24}, M_{24}, L_2(23), L_2(23).2$
30	$A_{30}, S_{30}, L_2(29), L_2(29).2$
36	$A_{36}, S_{36}, A_9, S_9, M_{10}, PGL_2(9), P\Gamma L_2(9), L_2(8), L_2(8).3, U_3(3), U_3(3).2,$ $U_4(2), U_4(2).2, S_6(2), A_5 \times A_5, A_6 \times A_6$
40	$A_{40}, S_{40}, L_4(3), PGL_4(3), U_4(2), U_4(2).2$
45	$A_{45}, S_{45}, M_{10}, PGL_2(9), P\Gamma L_2(9), A_{10}, S_{10}, U_4(2), U_4(2).2$
60	$A_{60}, S_{60}, L_2(59), L_2(59).2, A_5 \times A_5$
72	$A_{72}, S_{72}, L_2(71), L_2(71).2$
90	$A_{90}, S_{90}, L_2(89), L_2(89).2$
120	$A_{120}, S_{120}, S_7, S_8, A_9, A_{10}, S_{10}, L_2(16), L_2(16).2, P\Gamma L_2(16), L_3(4), L_3(4).2_1$ $L_3(4).2_2, L_3(4).2_3, L_3(4).2^2, S_4(4), S_4(4).2, S_6(2), S_8(2), O_8^+(2), O_8^+(2).2$
180	$A_{180}, S_{180}, L_2(179), L_2(179).2$
360	$A_{360}, S_{360}, L_2(359), L_2(359).2, A_6 \times A_6$

Lemma 2.3. *It is not possible to decompose the group $L_2\,(q)$ as the product of A_6 and S_n, where $n > 4$.*

Proof. By [6, page 213] if $G = L_2\,(q)$, $q = p^f$, has a subgroup isomorphic to A_6 , then this subgroup must be of the form $L_2\,(p^m)$ where $m \mid f$. But $L_2(p^m) \cong A_6$ if and only if $p = 3$ and $m = 2$, hence $G = L_2(3^{2k}),\ k \geq 1$. But again by [6] a symmetric group S_n can be a subgroup of G if and only if $n \leq 4$, a contradiction.

[15, Lemma 3] is essential in this paper and so we will reproduce it here. We mention that it is not necessary to assume that B is a complete group and our rephrasing of the lemma is as follows.

Lemma 2.4. *Suppose $G = AB$ is such that A is a simple group and B has a unique proper normal subgroup N which is simple. Let $G \not\cong A \times B$ and let M be a minimal normal subgroup of G. Then one of the following holds:*

(i) $G = AB = M$ *is a simple group*
(ii) $G = MB,\ M = A \times N,\ N \cong A$
(iii) $G = MB,\ M \cong NA$ *is simple*
(iv) $M = A$ *or* $N,\ [G : AN] = [B : N],\ AN \cong A \times N$
(v) $M \cap X = 1,\ |M| \mid [X : A \cap B]$ *for* $X \in \{A, B\},\ |M| | A \cap B = |AM/M \cap BM/M|$.

Our work also depends on the primitive groups of certain degrees. Primitive groups of degree up to 20 were obtained in [11] and up to 1000 in [2]. In Table 2.1, the list of all primitive groups of degree $k \geq 5$ where k is a divisor of $|A_6| = 360$ is given. Notation for the names of groups in Table 2.1 is taken from [1].

MAIN RESULTS

In this section, using Lemma 2.4, we characterize finite groups $G = AB$ where $A \cong A_6$ and $B \cong S_n, n \geq 5$. But first we deal with the possibilities which arise as different cases in Lemma 2.4.

Lemma 3.1. *There is no simple group M such that $M = AB$ where $A \cong A_6$ and $B \cong S_n, n \geq 5$, unless $M \cong A_{10}$ and $n = 8$.*

Proof. We will assume that M is a simple group having subgroups $A \cong A_6$ and $B \cong S_n, n \geq 5$, such that $M = AB$ and derive a contradiction. If C is a maximal subgroup of G containing B, then $M = AC$ and $k = [M : C] = [A : A \cap C] \mid 360$. Therefore M is a primitive simple group of degree k, where k is a divisor of 360 and $k \geq 5$. By Lemmas 2.2 and 2.3, we know that M cannot be isomorphic to an alternating group or a linear group $L_2(g)$, unless $M \cong A_{10}$ for which the decomposition $A_{10} = A_6 S_8$ is possible by Lemma 2.2.

Therefore by Table 2.1 we have the following possibilities for $M : M_{11},\ M_{12},\ M_{24}, U_3(3),\ U_4(2),\ L_4(3),\ L_3(4),\ S_4(4),\ S_6(2),\ S_8(2),\ O_8^+(2)$. Since $5 \nmid |U_3(3)|$, therefore $M = U_3(3)$ is impossible. If $M = M_{11}$ or M_{12} then $11 \mid |M|$ and hence $n \geq 11$ which implies that $7 \mid |M|$ a contradiction. If $M = M_{24}$, then $23 \mid |M|$ and so $n \geq 23$ implying that $17 \mid |M|$, a contradiction. The same reasoning rules out $M = L_4(3)$, $S_4(4)$ and $S_8(2)$ considering $13 \mid |M|$ in the first case and $17 \mid |M|$ in the remaining two cases. If $M = U_4(2)$, then as $|U_4(2)| = 2^6 \cdot 3^4 \cdot 5 = |A_6 S_n|$ we must have $n = 6$ and therefore $U_4(2) = A_6 S_6$, but by [1] $U_4(2)$ has only one conjugacy class of subgroups isomorphic to S_6 and hence by [9, Proposition C, page 31] there is $g \in U_4$ (2) such that $U_4(2) = S_6^g S_6$ which by [12, page 26] is impossible. If $M = L_3$ *(4)* $= A_6 S_n$, then $n \geq 7$ but by [1] the group L_3 *(4)* has no subgroup isomorphic to S_7 . If $M = S_6$ *(2)* $= A_6 S_n$, then as $|S_6$ *(2)*$| = 2^9 \cdot 3^4 \cdot 5 \cdot 7$ we obtain $7 \leq n \leq 10$ and since by [1] the group S_6 *(2)* has no subgroup isomorphic to S_9 hence $n = 7$ or 8. But order consideration yields $n = 8$ and so S_6 *(2)* $= A_6$ S_8 . By [1], the group A_6 cannot be contained in a maximal subgroup of the form $2^5 : S_6$. Again by [1], the group S_6 *(2)* has only one conjugacy class of subgroups isomorphic to S_8 and so $S_6(2) = S_8^g S_8$ for some $g \in S_6$ *(2)* which is impossible by [12, page 26]. Finally, if $M = A_6 S_n = O_8^+(2)$, then by [1] $n \leq 8$ and order consideration gives a contradiction.

Lemma 3.2. *Let G be a group such that G = AB where* $A \cong A_6$ *and* $B \cong A_n,\ n \geq 5,$ *then either* $G \cong A \times B$ *or one of the following cases holds:*

- $G = A_{n+1}$, $n = 5,9,14,19,29,35,39,44,59,71,89,119,179,$ *or* 359
- $G = A_n$, $n \geq 6$, *or*
- $G = A_{10}$, $n = 8$.

Proof. First suppose that G is simple. By Lemma 2.1 the cases (i) and (iii) are possible and the case (ii) arise from the trivial factorization of A_n. Now assume that the simple group G has the desired decomposition $G = A_6$ A_n, $n \geq 5$ and let C be a maximal subgroup of G containing A_n. Therefore $G = A_6 C$ and $m = [G : C] = [A_6 : A_6 \cap C] \mid 360$.

Maximality of C in G implies that G is a simple primitive permutation group of degree m where m is a divisor of 360. Now by Table 2.1 we know that simple primitive permutation groups are alternating groups, sporadic simple groups and simple groups of Lie type with small orders. We consider the following cases:

- The group G is isomorphic to an alternating group. In this case by Lemma 2.1 and what follows after that we obtain all the cases (i), (ii), and (iii) of the lemma.
- The group $G \cong L_2$ (q) is a 2-dimensional linear group over the finite field *GF(q)*. In this case by [3] factorization L_2 *(q)* $= A_6 A_n$ is possible if and only if $n = 6$ and $q = 9$ which gives the trivial factorization.
- The group G is isomorphic to a sporadic simple group. In this case by Table 2.1 we have the following possibilities for $G = M_{11}$,M_{12} ,M_{24} . But by [4] the factorization $G = A_6 S_n$, $n \geq 5$, is not possible for G.

The group G is isomorphic to one of the following linear groups: $U_3(3)$, $U_4(2)$, $L_4(3)$, $L_3(4)$, $S_4(4)$, $S_6(2)$, $S_8(2)$, $O_8^+(2)$. Since $5 \nmid |U_3(3)|$ therefore $G \neq U_3$ (3). If $G = S_4$ (4) or S_8 (2), then since $17 \mid |G|$ we must have $n \geq 17$ and since $13 \nmid |G|$ we get a contradiction. If $G = L_4$ (3), then $13 \mid |G|$ and so $n \geq 13$ which is impossible because $11 \nmid |G|$. If $G = U_4$ $(2) = A_6$ A_n, then order consideration yields $n = 6$. But by [1] the maximal subgroup of U_4 (2) containing one of the A_6 subgroups is conjugate to an S_6 subgroup which is maximal in G. Therefore U_4 $(2) = A_6$ S_6 which is impossible by the proof of Lemma 3.1. If $G = L_3$ $(4) = A_6$ A_n, then by [1] $n = 6$, a contradiction because $7 \mid |G|$. If $G = S_6$ $(2) = A_6$ A_n then since $3^4 \mid |G|$ and S_6 (2) has no subgroup isomorphic to A_9 we must have $n = 8$. But $G = A_6$ A_8 and A_8 is contained in a maximal subgroup of S_6 (2) isomorphic to S_8 and so $G = A_6$ S_8 which is impossible by Lemma 3.1. Finally if $G = O_8{}^+(2) = A_6$ A_n, then by [1] $n \leq 9$ and order consideration gives a contradiction.

Now suppose that G is not isomorphic to $A\times B$ and let $1 \neq M$ be a minimal normal subgroup of G. By [14, Lemma 1] M is elementary abelian, M $A = M \cap B = 1$, and $|M|$ divides 360 the order of A_6 . Thus M is an elementary abelian subgroup of order 2, 2^2 , 2^3 , 3, 3^2 , or 5. By induction, as $G/M = (AM/M)(BM/M)$ with $AM/M \cong A$ and $BM/M \cong B$, that G/M is simple. Hence, either $C_G^{(M)} = M$ or $M \leq Z(G)$. Now $C_G^{(M)} = M$ implies that $A_6 \leq \mathrm{Aut}(M)$, contrary to the possibilities for M. Now $M = Z(G)$ and G/M is an alternating group. It follows that G is a covering group of an alternating group, contrary to [14, Theorem 10].

Theorem 3.3. *Let G be a group such that $G = AB$, $A \cong A_6$ and $B \cong S_n$, $n \geq 5$. Then one of the following cases occurs:*

- $G \cong A_6 \times S_n$
- $G \cong A_{10} \cong A_6 S_8$, $n = 8$
- $G \cong (A_6 \times A_6)\langle \tau \rangle$, τ *an automorphism of order* 2 *and* $A_6 \times A_6$ *is the minimal normal subgroup of* G, $n = 6$
- $G \cong S_{n+1}$, $n = 5,9,14,19,29,35,39,44,59,71,89,119,179,359$
- $G \cong S_n$, $n \geq 6$
- $G \cong A_{10} \times Z_2$, $n = 8$
- $G \cong (A_6 \times A_n)\langle \tau \rangle$, $n \geq 5$, *where* τ *acts as an automorphism of order* 2 *on both factors.*

Proof. Our proof is based on the results of Lemma 2.4 and here we use the same notation used in this lemma. Therefore, let M be a minimal normal subgroup of G and note that $N \cong A_n$. If $G \ncong A \times B$, then one of the following possibilities occurs:

- $M = G = AB$ is a simple group. In this case by Lemma 3.1 we have $M \cong A_{10}$ and $n = 8$ and case (b) occurs.
- $G = MB$, $M \cong A_6 \times \mathrm{N}$, $N \cong A_6$.
- In this case $n = 6$ and $G \cong A_6\, S_6$, S_6 acts on A_6 by conjugation and $A_6 \times \mathrm{A}_6$ is the minimal normal subgroup of G and this is case (c) in the theorem.
- $G = MB$, $M \cong A_6\, A_n$ is simple. In this case by Lemma 3.2 three cases occur. If $M = A_{n+1}$, n = 5,9,14,19,29,35,39,44,59,71,89,119,179, or 359, then the same reasoning used in the proof of [15, Theorem 4] yields case (d). If $M = A_n$, then $G = S_n$, $n \geq 6$ and this is the case (e). If $M = A_{10}$ and $n = 8$, then a simple argument forces $G \cong S_{10}$ or $A_{10} \times Z_2$. If $G \cong S_{10}$ we have case (e) again. If we consider the alternating group A_{10} on the set $\{1,2,...,10\}$. Then since A_6 has a 2-transitive action on 10 letters we obtain $A_{10} = A_6 A_8$ where A_8 is the pointwise stabilizer of the set {9,10}. Now the set stabilizer of {9,10} is isomorphic to S_8 and is a subgroup of A_{10} containing this A_8. Therefore $A_{10}\langle(9\ 10)\rangle = A_6A_8\langle(9\ 10)\rangle$ implying $A_{10} \times \mathrm{Z}_2 \cong A_6\, S_8$ which is the case (f).
- $M = A_6$ or $A_n, [G : A_6A_n] = 2$, $A_6A_n \cong A_6 \times A_n$. In this case $G \cong (A_6 \times A_n) \cdot \langle\tau\rangle$ where τ acts as an outer automorphism of order 2 on both factors and this is the case (g).
- $M \cap A = 1$, $M \cap B = 1$ and $|M|$ divides $|A_6|$.

Since M is isomorphic to a direct product of simple groups either M is isomorphic to A_6 , A_5 or M is elementary abelian of order 2, 2^2 , 2^3 , 3, 3^2 , or 5. If $M \cong A_6$, then as $MS_n \leq G$ and $M \cap B = 1$, $G = MB \cong A_6\, S_n$ with A_6 as a minimal normal subgroup. This is the case (4) treated above. Consider $C_G^{(M)}$. Suppose that $A \cap C_G^{(M)} = 1$. Then A is isomorphic to a subgroup of Aut(M). Considering the possibilities for M, this is impossible. Thus, $A \leq C_G^{(M)}$ and by the modular law $C_G^{(M)} = A(B \cap C_G^{(M)})$. Now since $B \cap C_G^{(M)}$ is a normal subgroup of B, we must have either $B \cap C_G^{(M)} = 1, B,$ or $B \cap C_G^{(M)} \cong A_n$ A_n. If $B \cap C_G^{(M)} = 1,$ then as before B is isomorphic to a subgroup of

Aut*(M)*, contrary to the possibilities for *M* unless $M A_5$ and $n = 5$. Now *AM* has index 2 in *G*, so $AM = C_G^{(M)} \times M$ is a normal subgroup of *G*. This is case (4), above. If $B \cap C_G^{(M)} \cong A_n$, then $C_G^{(M)}$ is as in Lemma 3.2. However, none of these groups has a nontrivial center, a contradiction. Thus, we must have $B \leq C_G^{(M)}$ and $M \leq Z(G)$ and hence, *M* has prime order. By induction, $G/M = (AM/M)(BM/M)$ must be in the list, but $(AM/M) \cap (BM/M) \neq 1$ so only the parts (b), (d), (e), and (f) are possible. If part (e) holds, then we would have $G = BM = B \times M$ contrary to the fact that *A* has no subgroup of prime index. If part (b) or (d) holds, then *G* is the covering group of the symmetric group. Now we can see that *BM/M* must contain an involution which is the product of 2-cycles. It is known, see [10], that such an involution must lift to an element of order 4 in *G*, contrary to the fact that $M \cap B = 1$. (Note that *BM/M* lifts to *BM* in *G*, see the argument in [14].) Now suppose that $G/M = A_{10} \times Z_2$ and $n = 8$. Thus *G* has a normal subgroup of order $2|M|$ which arguing as above must be the center of *G*. It follows that *G* is a covering group of A_{10}. But as the Schur multiplier of A_{10} has order 2 this is impossible. This completes the proof.

ACKNOWLEDGEMENTS

We would like to thank Professor Gary L. Walls for his interest and guidance in preparation of this manuscript. We also wish to thank the referee for his comments concerning improvements in proving Lemma 3.2 and Theorem 3.3.

REFERENCES

1. J. H. Conway, R. T. Curtis, S. P. Norton, R. A. Parker, and R. A. Wilson, Atlas of Finite Groups. Maximal Subgroups and Ordinary Characters for Simple Groups, Oxford University Press, Eynsham, 1985. MR 88g:20025. Zbl 568.20001.
2. J. D. Dixon and B. Mortimer, The primitive permutation groups of degree less than 1000, Math. Proc. Cambridge Philos. Soc. 103 (1988), no. 2, 213–238. MR 89b:20014. Zbl 646.20003.
3. T. R. Gentchev, Factorizations of the groups of Lie type of Lie rank 1 or 2, Arch. Math. (Basel) 47 (1986), no. 6, 493–499. MR 87k:20033. Zbl 589.20006.
4. ______, Factorizations of the sporadic simple groups, Arch. Math. (Basel) 47 (1986), no. 2, 97–102. MR 88f:20031. Zbl 591.20022.
5. H. Hanes, K. Olson, and W. R. Scott, Products of simple groups, J. Algebra 36 (1975), no. 2, 167–184. MR 55#10558. Zbl 311.20007.
6. B. Huppert, Endliche Gruppen I, Die Grundlehren der Mathematischen Wissenschaften, vol. 134, Springer-Verlag, Berlin, 1967 (German). MR 37#302. Zbl 217.07201.
7. W. M. Kantor, k-homogeneous groups, Math. Z. 124 (1972), 261–265. MR 46#5422. Zbl 232.20003.
8. O. H. Kegel and H. Lüneburg, Über die kleine reidemeisterbedingung II, Arch. Math. 14 (1963), 7–10 (German). MR 26#4253. Zbl 108.16302.
9. M. W. Liebeck, C. E. Praeger, and J. Saxl, The maximal factorizations of the finite simple groups and their automorphism groups, Mem. Amer. Math. Soc. 86 (1990), no. 432, iv–151. MR 90k:20048. Zbl 703.20021.
10. W. R. Scott, Products of A_5 and a finite simple group, J. Algebra 37 (1975), no. 1, 165–171. MR 52#3321. Zbl 317.20012.
11. C. C. Sims, Computational methods in the study of permutation groups, Computational Problems in Abstract Algebra (Proc. Conf., Oxford, 1967), Pergamon, Oxford, 1970, pp. 169–183. MR 41#1856. Zbl 215.10002.
12. M. Suzuki, Group Theory I, Grundlehren der Mathematischen Wissenschaften, vol. 247, Springer-Verlag, Berlin, 1982, translated from the Japanese by the author. MR 82k:20001c. Zbl 472.20001.

13. G. L. Walls, Groups which are products of finite simple groups, Arch. Math. (Basel) 50 (1988), no. 1, 1–4. MR 88k:20049. Zbl 611.20017.

14. ______, Nonsimple groups which are the product of simple groups, Arch. Math. (Basel) 53 (1989), no. 3, 209–216. MR 90k:20049. Zbl 672.20011.

15. ______, Products of simple groups and symmetric groups, Arch. Math. (Basel) 58 (1992), no. 4, 313–321. MR 93a:20032. Zbl 764.20015.

16. J. Wiegold and A. G. Williamson, The factorisation of the alternating and symmetric groups, Math. Z. 175 (1980), no. 2, 171–179. MR 82a:20008. Zbl 439.20003.

Section 2: Power Domains and Metrics Concepts

Chapter

ON FS+-DOMAINS

5

Yayan Yuan and Jibo Li

[1]College of Mathematics and Information Science, Henan Normal University, Xinxiang, Henan 453007, China

ABSTRACT

We introduce a new construction—FS_+-domain—and prove that the category with FS_+-domains as objects and Scott continuous functions as morphisms is a Cartesian closed category. We obtain that the Plotkin powerdomain (L) over an FS-domain L is an FS_+-domain.

Citation: Yayan Yuan and Jibo Li, "On FS+ Domains", Nonlinear Analysis: Algorithm, Convergence, and Applications 2014, volume 2014, article ID 850298, https://doi.org/10.1155/2014/850298.

INTRODUCTION

Powerdomains are very important structures in Domain theory, which play an important role in modeling the semantics of nondeterministic programming languages. Three classical powerdomains are the Hoare or lower powerdomain [1], the Smyth or upper powerdomain [2], and the Plotkin or convex powerdomain [3].They are all free dcpo-algebras over (continuous) dcpos with special binary operators satisfying some equations and inequalities (see [4–12]).

In [13], Huth et al. concluded that the Hoare powerdomain (L) over a pointed domain L is a distributive $FS_{\vee}$-lattice. In [14], Meng and Kou obtained that the Smyth powerdomain (L) of a Lawson compact domain L is an $FS_{\wedge}$-domain. Then we have a problem whether the Plotkin powerdomain can be characterized by some special FS- domain. In this paper, we will introduce a new domain construction called the FS_{+}-domain which is a +-semilattice and there exists a directed family of finitely separated Scott continuous and +-semilattice homomorphisms which can approximate id_L, where the operation + is Scott continuous which satisfied the commutative, associative, and idempotency laws. And the category with FS_{+}-domains as objects and Scott continuous functions as morphisms is a Cartesian closed category. We will show that the Plotkin powerdomain (L) over an FS-domain L is an FS_{+}-domain, where the Plotkin powerdomain is the free dcpo-semilattice over a continuous dcpo.

Next, we collect some basic notions needed in this paper. The reader can also consult [4, 5, 15, 16]. A poset L is called a directed complete poset (a dcpo, for short) if any nonempty directed subset of L has a sup in L. For x, $y \in L$, x is way below y (denoted by $x \ll y$) if and only if, for all directed subsets $D \subseteq L$ for which sup D exists, the relation $y \leq$ sup D implies the existence of a $d \in D$ with $x \leq d$. A dcpo L is called a continuous domain if, for all $x \in L$, $x = \bigvee^{\uparrow} \twoheaddownarrow x$; that is, the set $\twoheaddownarrow x = \{a \in \ : a \ll x\}$ is directed and $x = \bigvee\{a \in L : a \ll x\}$. For a subset A of L, let $\uparrow A = \{x \in \ : \exists a \in A, a \leq x\}$, $\downarrow A = \{x \in L : \exists a \in A, x \leq a\}$. We use $\uparrow a$ (resp., $\downarrow a$) instead of $\uparrow \{a\}$ (resp., $\downarrow \{a\}$) when $A = \{a\}$. A is called an upper (resp., a lower) set if $A = \uparrow A$ (resp., $A = \downarrow A$). If $(L, \leq)$ is a dcpo, we define the Scott topology, denoted by (L), which has as its topology of closed sets all directed complete lower subsets, that is, lower sets closed under directed sups. A function f from a dcpo L into a dcpo

P is continuous with respect to the Scott topologies if f preserves suprema of directed subsets.

Recall the definition of FS-domain: a dcpo L is called an FS-domain if id_L is approximated directly by a family of finitely separated Scott continuous functions. A Scott continuous function $: L \to L$ is called finitely separated if there exists a finite set M_f such that, for each $x \in L$, there exists $m \in M_f$ such that $f(x) \leq m \leq x$.

FS_+-DOMAINS

Categories of FS_+-Domains

For dcpos L and P, the function space $[L \to P]$ of all Scott continuous functions from L to P with the pointwise order is a dcpo. Then for dcpo +-semilattices D and E, we conclude that the function space $[D \to {}_+E]$ of all the Scott continuous and +-semilattice homomorphisms from D to E with the pointwise order is a dcpo +-semilattice from the following theorem, where the operation + satisfies the commutative, associative, and idempotency laws.

Theorem 1

Let D and E be dcpo +-semilattices; then $[D \to {}_+E]$ is a dcpo +-semilattice.

Proof. For any directed family $\{f_j \in [D \to {}_+E] : j \in J\}$ and $x \in D$, set $f(x) = \bigvee_{j\in J} f_j(x)$. It is obvious that f is Scott continuous. Then

$$\begin{aligned} f(x+y) &= \bigvee_{j\in J} f_j(x+y) = \bigvee_{j\in J}\left(f_j(x)+f_j(y)\right) \\ &= \left(\bigvee_{j\in J} f_j(x)\right)+\left(\bigvee_{j\in J} f_j(y)\right) = f(x)+f(y). \end{aligned} \tag{1}$$

So f is also a Scott continuous and +-semilattice homomorphism. Hence $[D \to {}_+E]$ is a dcpo.

For any $x \in D$, f, $g \in [D \to {}_+E]$, we define $(f+g)(x) = f(x) + g(x)$. For a directed set $\{x_k \in \ : k \in K\}$, we have

$$\begin{aligned}(f+g)\left(\bigvee_{k\in K}(x_k)\right) &= f\left(\bigvee_{k\in K}(x_k)\right)+g\left(\bigvee_{k\in K}(x_k)\right)\\ &= \bigvee_{k\in K} f(x_k)+\bigvee_{k\in K} g(x_k)\\ &= \bigvee_{k\in K}\bigvee_{k'\in K}\left[f(x_k)+g(x_{k'})\right]\\ &= \bigvee_{k\in K}\left[f(x_k)+g(x_k)\right]\\ &= \bigvee_{k\in K}\left[(f+g)(x_k)\right].\end{aligned} \tag{2}$$

Then $f + g$ is Scott continuous.

For a pair of points x, y in D

$$\begin{aligned}(f+g)(x+y) &= f(x+y)+g(x+y)\\ &= (f(x)+f(y))+(g(x)+g(y))\\ &= (f(x)+g(x))+(f(y)+g(y))\\ &= (f+g)(x)+(f+g)(y).\end{aligned} \tag{3}$$

That is, $f + g$ is a +-semilattice homomorphism. So $[D \longrightarrow {}_{+}E]$ is a +-semilattice.

Finally, by the Scott continuity of the operation +, we obtain the following conclusion. For the sup of the directed set $\{f_j \in [D \longrightarrow {}_{+}E] : j \in J\}$ and $g \in [D \longrightarrow {}_{+}E]$, if $x \in D$, then

$$\begin{aligned}\left[g+\left(\bigvee_{j\in J} f_j\right)\right](x) &= g(x)+\left(\bigvee_{j\in J} f_j(x)\right)\\ &= \bigvee_{j\in J}\left[g(x)+f_j(x)\right]\\ &= \bigvee_{j\in J}\left[(g+f_j)(x)\right]\\ &= \left[\bigvee_{j\in J}(g+f_j)\right](x).\end{aligned} \tag{4}$$

So $+ : [D \to {}_{+}E] \times [D \to {}_{+}E] \to [D \to {}_{+}E]$ is Scott continuous.

We have obtained that $[D \to {}_{+}E]$ is a dcpo +-semilattice.

With respect to these special Scott continuous functions, we will introduce some new order structures.

Definition 2

A dcpo L is called an FS_{+}-domain if it is a +- semilattice and there exists a directed family of finitely separated Scott continuous and +-semilattice homomorphisms which can approximate id_{L}

For example, an $FS_{\wedge}$-domain is a continuous dcpo ∧- semilattice where id is approximated by a directed family of finitely separated Scott continuous functions preserving finite infs.

We know that an FS_{+}-domain is an FS-domain.

Theorem 3.

Let D and E be FS_{+}-domains; then $[D \to {}_{+}E]$ and $[D \to E]$ are FS_{+}-domains.

Proof. Suppose that $\mathscr{D}$ and $\mathscr{E}$ are approximate identities for D and E, respectively. Then we claim that the family

$$\mathscr{D} \otimes \mathscr{E} = \{\delta \otimes \epsilon : \delta \in \mathscr{D}, \epsilon \in \mathscr{E}\}, \tag{5}$$

defined by

$$f \longmapsto \epsilon^2 f \delta^2, \tag{6}$$

for $f \in [D \to {}_{+}E]$ is an approximate identity for $[D \to {}_{+}E]$ where $\delta \otimes \epsilon$ is finitely separated. The proof is similar to the case of FS-domains.

It suffices to show that $\delta \otimes \epsilon \in [D \to {}_{+}E] \to {}_{+}[D \to {}_{+}E]$. Firstly, it is obvious that $\delta \otimes \epsilon$ is Scott continuous. Secondly, for a pair of points $f, g \in [D \to {}_{+}E]$, we have for any $x \in \mathrm{D}$

$$\begin{aligned}[(\delta \otimes \epsilon)(f + g)](x) &= \left[\epsilon^2 (f + g) \delta^2\right](x) \\ &= \epsilon^2 \left[f\delta^2(x) + g\delta^2(x)\right]\end{aligned}$$

$$= \epsilon\left[\epsilon\left(f\delta^2(x) + g\delta^2(x)\right)\right]$$

$$= \epsilon^2 f\delta^2(x) + \epsilon^2 g\delta^2(x)$$

$$= \left[\epsilon^2 f\delta^2 + \epsilon^2 g\delta^2\right](x)$$

$$= [(\delta \otimes \epsilon)(f) + (\delta \otimes \epsilon)(g)](x). \tag{7}$$

So we conclude that $\delta \otimes \epsilon$ is a +-semilattice homomorphism. Then $[D \to {}_{+}E]$ is an FS_{+}-domain. Similarly, $[D \to E]$ is also an FS_{+}-domain.

Theorem 4

The category with FS_{+}-domains as objects and Scott continuous functions as morphisms is a Cartesian closed category.

Note that the category with FS_{+}-domains as objects and Scott continuous and +-semilattice homomorphisms as morphisms is not a Cartesian closed category generally, because the evaluation maps do not preserve the finite +- operation.

Classify the Powerdomains

Definition 5 (see [5]). Let L be a dcpo-algebra equipped with a Scott continuous binary operation + that satisfies the following equations: for any $a, b, c, \in L$

- $a + a = a$ (idempotency law);
- $a + b = b + a$ (commutative law);
- $a + (b + c) = (a + b) + c$ (associative law).

Then the dcpo-algebra is a commutative idempotent semigroup, called a dcpo-semilattice. The free dcpo-semilattice over a dcpo L is called the convex or Plotkin powerdomain of L and it is denoted by (L).

If the binary operation + satisfies the inequality $a + b \le a$, then we obtain the upper or Smyth powerdomain, and it is denoted by $P(L)$, where $a + b = a \wedge b$.

Similarly, if the binary operation + satisfies $a + b \ge a$, then it is called the lower or Hoare powerdomain, denoted by $P(L)$, where $a + b = a \vee b$.

Proposition 6 (see [5]). For subsets C and D of a preordered set $(L, \le)$ one has

- $C =_H \downarrow C$;
- $C \leq_H D$ iff $\downarrow C \subseteq \downarrow D$;
- $C \ll_H D$ iff there exists a finite subset $F \subseteq L$ such that $C \subseteq \downarrow F \subseteq \twoheaddownarrow D$;
- $C =_S \uparrow C$;
- $C \leq_S D$ iff $\uparrow D \subseteq \uparrow C$;
- $C \ll_S D$ iff $D \subseteq \mathrm{int}_\sigma(\uparrow C)$ iff $D \subseteq \twoheaduparrow C$;
- $C =_P \downarrow C \cap \uparrow C = \sup\{\downarrow F \cap \uparrow F : F \prec C, F \subseteq_{fin} L\}$, where $F \prec C$ iff $F \subseteq \twoheaddownarrow C$ and $C \subseteq \twoheaduparrow F$;
- $C \leq_P D$ iff $\downarrow C \subseteq \downarrow D$ and $\uparrow D \subseteq \uparrow C$;
- $C \ll_P D$ iff $C \ll_H D$ and $C \ll_S D$.

Next, we draw the conclusion that some special FS- domain categories concerning the operation + can be used to classify the powerdomains.

Theorem 7

If L is an FS-domain, then the convex powerdomain (L) is an FS_+-domain.

Proof. Suppose that L is an FS-domain; then L is a Lawson compact domain. Thus, (L) is also a domain. Assume that $\mathscr{F} = \{f_i : L \to L\}_{i \in I}$ is the approximate identity for L, where $\mathscr{F}$ is a family of finitely separated Scott continuous functions; that is, for any f_i, there exists a finite set $M_i \subseteq L$ such that, for any $x \in L$, there exists some $m \in M_i$ such that $f(x) \leq m \leq x$. We claim that $\{(f_i):P(L) \to P^P(L)\}_{i \in I}$ is the approximate identity for $P^P(L)$. It suffices to consider four steps as follows.

- $P(f_i) \leq P(id)$. For $A \in (L)$, define $P(f_i)(A) = P(f_i(A)) = \downarrow f_i(A) \cap \uparrow f_i(A)$. By Proposition 6, $\downarrow (A) \cap \uparrow f_i(A) \in P^P(L)$. For any $x \in A$, let $(A) = \{m \in M_i : \exists x \in A, f_i(x) \leq m \leq x\}$; then $f_i(x) \leq x$ implies $\downarrow f_i(A) \subseteq \downarrow M_i(A) \subseteq \downarrow A$ and $\uparrow A \subseteq \uparrow M_i(A) \subseteq \uparrow f_i(A)$. Hence $(f_i)(A) = P(f_i(A)) \leq_P A$.
- $\sup\{P(f_i) : i \in I\} = P(id)$. For any $A \in (L)$, it is obvious that $\sup\{P(f_i)(A) : i \in I\} \leq A$. Suppose $A \nleq \sup\{P(f_i)(A) : i \in I\}$. There is $B \in (L)$ such that $B \ll_P A$ and $B \nleq \sup\{P(f_i)(A) : i \in I\}$. By $B \ll_P A$ and $A = \sup\{\downarrow F \cap \uparrow F : F \prec A, F \subseteq_{fin} L\}$, there is some finite set $F \prec A$ such that $B \leq \downarrow F \cap \uparrow F$. But for any finite set $F \prec A$, we have $F = \sup\{f_i(F) : i \in I\}$, where $F \prec A$ iff $F \subseteq \twoheaddownarrow A$ and $A \subseteq \twoheaduparrow F$. Then $\downarrow F \cap \uparrow F = \sup\{\downarrow f_i(F) \cap \uparrow f_i(F) : i \in I\}$

$\leq \sup\{\downarrow f_i(A) \cap \uparrow f_i(A) : i \in I\}$. This is a contradiction. Then we conclude that $\sup\{P(f_i) : i \in I\} = P(id)$.

- (f_i) is Scott continuous and finitely separated. For a directed family $\mathscr{D}$ in $P^P(L)$, we have

$$\begin{aligned} P(f_i)(\sup \mathscr{D}) &= P(f_i(\sup \mathscr{D})) \\ &= P(\sup\{f_i(D) : D \in \mathscr{D}\}) \\ &= \sup\{P(f_i(D)) : D \in \mathscr{D}\}. \end{aligned} \tag{8}$$

Then (f_i) is Scott continuous.

For any $A \in (L)$, $M_i(A)$ is a finite set. By $\downarrow (A) \subseteq \downarrow M_i(A) \subseteq \downarrow A$ and $\uparrow A \subseteq \uparrow M_i(A) \subseteq \uparrow f_i(A)$, it follows that $\downarrow M_i(A) \cap \uparrow M_i(A) \in P^P(L)$. Let $\mathscr{M}_i = \{\downarrow (A) \cap \uparrow M_i(A) : A \in P^P(L)\}$. Since $(A) \subseteq M_i$ and M_i is finite, it follows that $\mathscr{M}_i$ is a finite family of $P^P(L)$. And we have that, for any A, there exists $\downarrow M_i(A) \cap \uparrow M_i(A) \in \mathscr{M}_i$ such that $P(f_i)(A) \leq_P \downarrow M_i(A) \cap \uparrow M_i(A) \leq_P A$; that is, $P(f_i)$ is finitely separated.

- (f_i) is a +-semilattice homomorphism. For $A, B \in P^P(L)$, since $P^P(L)$ is a +-semilattice, $A + B = \downarrow (A \cup B) \cap \uparrow (A \cup B) \in P^P(L)$:

$$\begin{aligned} P(f_i)(A+B) &= P(f_i)[\downarrow (A \cup B) \cap \uparrow (A \cup B)] \\ &= \downarrow f_i[\downarrow (A \cup B) \cap \uparrow (A \cup B)] \\ &\quad \cap \uparrow f_i[\downarrow (A \cup B) \cap \uparrow (A \cup B)] \\ &= \downarrow f_i(A \cup B) \cap \uparrow f_i(A \cup B) \\ &= \downarrow [(\downarrow f_i(A) \cap \uparrow f_i(A)) \\ &\qquad \cup (\downarrow f_i(B) \cap \uparrow f_i(B))] \\ &\quad \cap \uparrow [(\downarrow f_i(A) \cap \uparrow f_i(A)) \\ &\qquad \cup (\downarrow f_i(B) \cap \uparrow f_i(B))] \\ &= P(f_i)(A) + P(f_i)(B). \end{aligned} \tag{9}$$

Then we conclude that $\{(f_i) : i \in I\}$ is the approximate identity for $P^P(L)$. Thus the convex powerdomain $P^P(L)$ is an FS_+-domain.

Combined with the work of Huth et al. [13] and Meng and Kou [14], we conclude the following theorem.

Theorem 8

Let L be a domain. Then the following statements hold:

- if L is Lawson compact, then the Smyth powerdomain $P^S(L)$ is an $FS_\wedge$-domain (in [14]);
- if L has a least point, then the Hoare powerdomain $P^H(L)$ is a distributive $FS_\vee$-lattice (in [13]);
- if L is an FS-domain, then the Plotkin powerdomain $\mathrm{P}^P(L)$ is an FS_+-domain.

ACKNOWLEDGMENTS

This work is supported by the Foundation of the Education Department of Henan Province (13A110552), the Foundation of the Science and Technology Department of Henan Province (142300410165), and the Foundation of Henan Normal University (2013PL03).

REFERENCES

1. M. B. Smyth, “Power domains and predicate transformers: a topological view,” in *Automata, Languages and Programming*, vol. 154 of *Lecture Notes in Computer Science*, pp. 662–675, Springer, Berlin, Germany, 1983.
2. M. B. Smyth, “Powerdomains,” *Journal of Computer and Systems Sciences*, vol. 16, pp. 23–36, 1978.
3. G. D. Plotkin, “A powerdomain construction,” *SIAM Journal on Computing*, vol. 5, no. 3, pp. 452–487, 1976.
4. S. Abramsky and A. Jung, “Domain theory,” in *Handbook of Logic in Computer Science*, vol. 3, Oxford University Press, New York, NY, USA, 1994.
5. G. Gierz, K. H. Hofmann, K. Keimel, J. D. Lawson, M. Mislove, and D. S. Scott, *Continuous lattices and domains*, vol. 93 of *Encyclopedia of Mathematics and its Applications*, Cambridge University Press, New York, NY, USA, 2003.
6. R. Heckmann and K. Keimel, “Quasicontinuous domains and the Smyth powerdomain,” *Electronic Notes in Theoretical Computer Science*, vol. 298, pp. 215–232, 2013.
7. R. Heckmann, *Power domain constructions [Ph.D. thesis]*, Universität des Saarlandes, Saarbrücken, Germany, 1990.
8. R. Heckmann, “An upper power domain construction in terms of strongly compact sets,” in *Mathematical Foundations of Programming Semantics*, vol. 598 of *Lecture Notes in Computer Science.*, pp. 272–293, Springer, Berlin, Germany, 1992.
9. R. Heckmann, “Stable power domains,” *Theoretical Computer Science*, vol. 136, no. 1, pp. 21–56, 1994.
10. R. Heckmann, “Characterising FS domains by means of power domains,” *Theoretical Computer Science*, vol. 264, no. 2, pp. 195–203, 2001.
11. J. H. Liang and H. Kou, “Convex power domain and Vietoris space,” *Computers & Mathematics with Applications*, vol. 47, no. 4-5, pp. 541–548, 2004.
12. M. Mislove, “On the Smyth power domain,” in *Mathematical Foundations of Programming Language Semantics*, vol. 298 of *Lecture Notes in Computer Science*, pp. 161–172, Springer, Berlin, Germany, 1988.

13. M. Huth, A. Jung, and K. Keimel, “Linear types, approximation, and topology,” in *Proceedings of the IEEE 9th Annual Symposium on Logic in Computer Science*, pp. 110–114, July 1994.

14. H. Meng and H. Kou, “Function spaces of semilattice homomorphisms and FS∧-domains,” *Chinese Annals of Mathematics A*, vol. 32, no. 1, pp. 107–114, 2011.

15. A. Jung, *Cartesian Closed Categories of Domains*, vol. 66 of *CWI Tracts*, Stichting Mathematisch Centrum Centrum voor Wiskunde en Informatica, Amsterdam, The Netherlands, 1989.

16. J. D. Lawson, “The versatile continuous order,” in *Mathematical Foundations of Programming Language Semantics*, vol. 298 of *Lecture Notes in Computer Science*, pp. 134–160, Springer, Berlin, Germany, 1988.

Chapter

THE TOPOLOGY OF GB-METRIC SPACES

6

Akbar Dehghan Nezhad and Zohreh Aral

Faculty of Mathematics, Yazd University, Yazd, Iran

ABSTRACT

We define the concepts of GB-metric in sets over σ-complete Boolean algebra and obtain some applications of them on the theory of topology. We also study some related properties of them.

INTRODUCTION

Numerous studies have been made concerning geometries and topologies induced in sets by general distance functions. A formulation of the notion "generalized metric space (or G-metric space)" has been given [1]. In

Citation: Akbar Dehghan Nezhad and Zohreh Aral, "The Topology of GB-Metric Spaces", International Scolarly Research Notes, volume 2011, article ID 523453, https://doi.org/10.5402/2011/523453.

this paper, we begin the elaboration of the topology induced in sets over σ-complete Boolean algebra.

In this paper, B shall always denote a σ-complete Boolean algebra. In B, we denote the operations of join, meet, and complement by $a \vee b$, $a \wedge b$ and a', respectively.

THE GB-METRIC SPACES

In 1952, a new structure of metric spaces, so called B-metric space was introduced by Ellis and Sprinkle [2], on the set X to Boolean algebra (for details see [3, 4]).

Definition 2.1 (see [2]). A B-metric space is a set X with a map $d : X \times X \rightarrow B$ (B is σ-Boolean algebra) with the properties

- $d(\xi, \eta) = 0$ if and only if $\xi = \eta$,
- $d(\eta, \xi) = d(\xi, \eta)$, (symmetry), and
- $(\xi, \zeta) \leq (\xi, \eta) \vee d(\eta, \zeta)$, for all ξ, η, ζ belong to X.

In [1], the present author has introduced a new structure of metric spaces which is a generalized idea of the ordinary metric space. The term generalized metric space is used in [5, 6].

Definition 2.2 (see [7]). Let X be a nonempty set and $G : X \times X \times X \rightarrow [0, \infty)$ be a function satisfying the following properties:

(G_1) $(x, y, z) = 0$ if $x = y = z$,

(G_2) $0 < (x, x, y)$; for all $x, y \in X$, with $x \neq y$,

(G_3) $(x, x, y) \leq (x, y, z)$, for all $x, y, z \in X$ with $z \neq y$,

(G_4) $\mathrm{G}(x, y, z) = G(x, z, y) = G(y, z, x) = \cdots$, (symmetry in all three variables), and

(G_5) $(x, y, z) \leq (x, a, a) + G(a, y, z)$, for all $x, y, z, a \in X$, (rectangle inequality).

Topology of GB-Metric Spaces

In this section we define generalized B-metric or GB-metric space and introduce some basic notions and results that are used in sequel.

Definition 2.3. Let X be a non empty set and $\mathrm{G_B} : \mathrm{X} \times \mathrm{X} \times \mathrm{X} \rightarrow \mathrm{B}$, be a function satisfying the following properties

(GB_1) $G_B(x,y,z) = 0$ if $x = y = z$,

(GB_2) $0 < G_B(x,x,y)$; for all $x, y \in X$, with $x \neq y$,

(GB_3) $G_B(x,x,y) \leq G_B(x,y,z)$, for all $x, y, z \in X$ with $z \neq y$,

(GB_4) $G_B(x,y,z) = G_B(x,z,y) = G_B(y,z,x) = \cdots$, (symmetry in all three variables), and

(GB_5) $G_B(x,y,z) \leq G_B(x,a,a) \vee G_B(a,y,z)$, forall $x, y, z, a \in X$, (rectangle inequality).

Then, the function G_B is called a generalized B-metric, or, more specifically, a G_B-metric on X, and the pair (X, G_B) is called a GB-metric space.

Example 2.4. Put $\mathbb{W} = \mathbb{N} \cup \{0\}$. Define a map $G_B : \mathbb{W} \times \mathbb{W} \times \mathbb{W} \to \{0, 1\}$

$$G_B(m,n,k) = \begin{cases} 0 & m = n = k, \\ 1 & \text{else.} \end{cases} \tag{2.1}$$

The map G_B is a GB-metric on $\mathbb{W}$

Remark 2.5. We can show that a GB-metric space is a generalized B-metric space over X.

Proposition 2.6. Every GB-metric space (X, G_B) will define a B-metric (X, d_{G_B}) by $d_{G_B}(x,y) = G_B(x,y,y)$.

Proof. Conditions 1 and 2 of B-metric are clearly and 3 follows from (GB_5)

Proposition 2.7. Let G_B be GB-metric on a ring X. For an element a ∈ X, the following maps are GB-metrics on the ring X:

ISRN Mathematical Analysis

(1) $G_B^*(x,y,z) = G_B(a-x, a-y, a-z)$ *and*

(2) $G_B^*(x,y,z) = G_B(x,y,z) \vee G_B(a-x, a-y, a-z)$.

Proposition 2.8. Let (X, d) be a B-metric space. Define a function, $G_B : X \times X \times X \to B$ by $G_B(x, y, z) = d(x, y) \vee d(y, z) \vee d(z, x)$ for x, y, z ∈ X. The map G_B is a GB-metric on X, and consequently (X, G_B) is a GB-metric space

Proof. Conditions (GB_1), (GB_2), and (GB_4) are clear. We show that (GB_3) and (GB_5) are valid too.

$$\begin{aligned} G_B(x,x,y) &= d(x,x) \vee d(x,y) \vee d(x,y) < d(x,z) \vee d(z,y) < d(x,z) \vee d(z,y) \vee d(x,y) \\ &= G_B(x,y,z). \end{aligned} \tag{2.2}$$

And for (GB_5), we have

$$\begin{aligned} G_B(x,y,z) &= d(x,y) \vee d(y,z) \vee d(z,x) < d(x,a) \vee d(y,a) \vee d(z,a) \vee d(y,z) \\ &= d(x,a) \vee G_B(a,y,z) \\ &= G_B(a,a,x) \vee G_B(a,y,z). \end{aligned} \tag{2.3}$$

Proposition 2.9. Let (X, G_B) be a GB-metric space, then for all x, y, z, a ∈ X, it follows that:

(1) $G_B(x,y,z) \leq G_B(x,x,y) \vee G_B(x,x,z)$,

(2) $G_B(x,y,y) = G_B(x,x,y)$,

(3) $G_B(x,y,z) \leq G_B(x,a,z) \vee G_B(a,y,z)$, *and*

(4) $G_B(x,y,z) \leq G_B(x,a,a) \vee G_B(y,a,a) \vee G_B(z,a,a)$.

Proposition 2.10. Let (X, G_B) be a GB-metric space, then the following are equivalent:

(1) $G_B(x,y,y) \leq G_B(x,y,a)$ *for all* $x,y,z,a \in X$,

(2) $G_B(x,y,z) \leq G_B(x,y,a) \vee G_B(z,y,b)$ *for all* $x,y,z,a \in X$.

Proof. Use 2 of Proposition 2.9.

Definition 2.11. Let (X, G_B) be a GB-metric. For all x_0 ∈ X, r ∈ B \ {0}, the GB-ball with center x_0 and radius r is $B_{GB}(x_0, r) = \{y \in X \mid G_B(x_0, y, y) < r\}$. For any GB-ball, we can define B_{GB} $(x_0, r) = \{y \in X \mid G_B(x_0, y, y) \leq r\}$.

Proposition 2.12. Let (X, G_B) be a G-metric space. Then for all x_0 ∈ X and r ∈ B \ {0}, we have

(1) *If* $G_B(x_0,x,y) < r$, *then* $x,y \in B_{GB}(x_0,r)$ *and*

(2) *if* $y \in B_{GB}(x_0,r)$, *then there exists one element* $\delta \in B \setminus \{0\}$ *such that,* $B_{G_B}(y,\delta) \subseteq B_{GB}(x_0,r)$.

Proof. (1) follows directly $G_B(x, x, y) \leq G_B(x, y, z)$. Put $\delta = r - G_B(x, y, y)$ and use (GB_5), then deduce (2).

It follows from 2 of the above proposition that the family of all GB-balls

$$\beta = \{B_{GB}(x,r) \mid x \in X, r \in B \setminus \{0\}\} \tag{2.4}$$

is the base of topology τ_{GB} on X, we call the GB-metric topology.

Definition 2.13. Let (X, G_B) be a GB-metric space. The sequence $\{x_n\} \subseteq$ X is GB-convergent to x if it converges to x in the GB-metric topology, τ_{GB}.

Remark 2.14. Another topological notions as GB-Hausdorff, GB-compact, GB-normal, and ... define similarly as usual.

Definition 2.15. Let (X, G_B) and (X', G'_B) be GB-metric spaces, a function f : X → X' is GB-continuous at a point x_0 ∈ X if $f^{-1(}B_{GB'})$ (f($x_{0)}$, r)) ∈ τ(G) for all r ∈ B \ {0}. We say f is GB-continuous if it is GB-continuous at all points of X, that is, continuous as a function from X with the τ_{GB}-topology to X' with the τ'_{GB}-topology.

Theorem 2.16. Let (X, G_B) be a GB-metric space and f be a self-map of X into itself. Suppose f is GB-continuous at $x_0 \in X$, if there is a point $x \in X$ such that the sequence of iterates $\{f^n(x)\}$ converges to x_0, then $f(x_0) = x_0$.

Proof. From (GB_5), *we derive* $G_B(f(x_0), x_0, x_0) \leq G_B(f(x_0), f^n(x), f^n(x)) \vee G_B(f^n(x), x_0, x_0) = G_B(f(x_0), f(f^{n-1}(x)), f(f^{n-1}(x))) \vee G_B(f^n(x), x_0, x_0)$ *as* $n \to \infty$ *we deduce,* $G_B(f(x_0), f(f^{n-1}(x)), f(f^{n-1}(x))) \vee G_B(f^n(x), x_0, x_0) \to 0$, *so the above equation will be* $G_B(f(x_0), x_0, x_0) \leq 0$, *therefore,* $f(x_0) = x_0$.

Theorem 2.17. *Let* (X, G_B) *be a* GB*-metric space and* $f : X \to X$ *be* G*-continuous. If for* $x \in X$, *the sequence* $\{f(x)\}$ *has a convergent subsequence* $\{f^{n_i}(x)\}$ GB*-converges to* p, *and* $G_B(f^{n_i}(x), f^{n_i+1}(x), f^{n_i+1}(x)) \to 0,$, *then* f *has a fixpoint.*

Proof. Since $\{f^{n_i}(x)\} \to p,$ then by GB-continuity of f we have $f(f^{n_i}(x)) = f^{n_i+1}(x) \to f(p),$ and we have

$$G_B(p, f(p), f(p)) \leq G_B(p, f^{n_i}(x), f^{n_i}(x)) \vee G_B\left(f^{n_i}(x), f^{n_i+1}(x), f^{n_i+1}(x)\right) \\ \vee G_B\left(f^{n_i+1}(x), f(p), f(p)\right). \tag{2.5}$$

As $i \to \infty$ we get $G(p, f(p), f(p)) \leq 0$, which implies that $f(p) = p$.

Theorem 2.18. *A* GB*-metric space* X *is* GB*-Hausdorff.*

Proof. Consider two elements x, y belong to X that $x \neq y$, and two subsets $A = \{x \in X \mid G(x, x, z) < G_B(x, x, y)\}$ and $B = \{x \in X \mid G_B(x, x, y) < G_B(x, x, z)\}$. These are two disjoint neighbourhoods of x and y, respectively. It is sufficient to prove that the subsets A and B are open. Suppose $t \in A$. Put $G(t, t, z) = r_1$, $G_B(t, t, y) = r_2$ and $\delta = r_2 - r_1$. If $s \in B_G(t, \delta)$, then $G_B(s, s, z) < G_B(t, t, z) \vee G_B(s, s, t) < r_2 = G_B(t, t, y) < G_B(t, t, s) \vee G_B(s, s, y) = G_B(s, s, y)$, therefore, $s \in A$ and $B_{GB}(t, \delta) \subset A$. This proves X is Hausdorff.

Theorem 2.19. *A* GB*-metric space* X *is* GB*-normal.*

Proof. Let $G(x, x, \text{D} \wedge) = \{G_B(x, x, y) \mid y \in D\}$. Consider two closed disjoint subsets E and F of X. Put $A = \{x \in X \mid (x, x, F) < G(x, x, E)\}$ and $B = \{x \in XG_B(x, x, E) < G_B(x, x, F)\}$. It is clear that A, B are disjoint open subsets of X, $E \subset A$, and $F \subset B$.

Corollary 2.20. *A* GB*-metric space* X *is* GB*-regular.*

Monoid Invariant GB-Metric

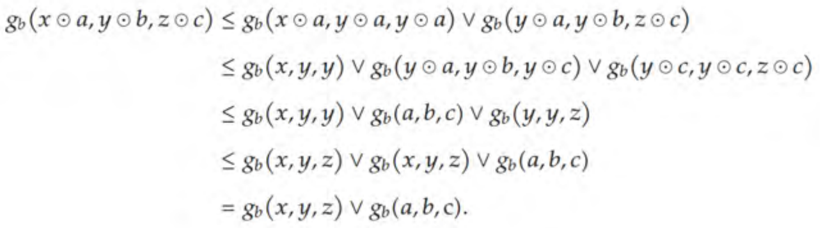
Definition 2.21. A GB-metric g_b on a $(X, \odot)$ is m-invariant (or monoid invariant) when $G_B(x \odot a, y \odot a, z \odot a) \leq G_B(x, y, z)$ and $G_B(a \odot x, a \odot y, a \odot z) \leq G_B(x, y, z)$ are valid for any $x, y, z, a \in X$.

Proposition 2.22. *Let $(X, \odot)$ be a monoid. If G_B is an m-invariant G-metric on X, then*

(1) *$G(x \odot a, y \odot b, z \odot c) \leq G_B(x, y, z) \vee G_B(a, b, c)$ and*

(2) *$G_B(x_1 \odot x_2 \odot x_3 \odot \cdots \odot x_n, y_1 \odot y_2 \odot y_3 \odot \cdots \odot y_n, z_1 \odot z_2 \odot z_3 \odot \cdots \odot z_n) \leq G_B(x_1, y_1, z_1) \vee G_B(x_2, y_2, z_2) \vee G_B(x_3, y_3, z_3) \vee \cdots \vee G_B(x_n, y_n, z_n)$.*

Proof. (1) We have



$$\begin{aligned} g_b(x \odot a, y \odot b, z \odot c) &\leq g_b(x \odot a, y \odot a, y \odot a) \vee g_b(y \odot a, y \odot b, z \odot c) \\ &\leq g_b(x, y, y) \vee g_b(y \odot a, y \odot b, y \odot c) \vee g_b(y \odot c, y \odot c, z \odot c) \\ &\leq g_b(x, y, y) \vee g_b(a, b, c) \vee g_b(y, y, z) \\ &\leq g_b(x, y, z) \vee g_b(x, y, z) \vee g_b(a, b, c) \\ &= g_b(x, y, z) \vee g_b(a, b, c). \end{aligned} \tag{2.6}$$

(2) It concludes by induction.

Put $(X, \odot)$ is a monoid and G_B be a GB-metric on X. We call G_B is left effective on X if $G(a \odot x, a \odot y, a \odot z) = G_B(x, y, z)$ for all $x, y, z \in X$ then $a = e$. And we say G_B is left free GB-metric on X if for a triple $(x, y, z) \in X \times X \times X$, $G(a \odot x, a \odot y, a \odot z) = G_B(x, y, z)$ then $a = e$. The right effective (free) is defined samely, and we call G_B is effective (free) if it is left and right effective (free) on X. Immediately we deduce, if G_B is free GB-metric on $(X, \odot)$ then it is effective GB-metric.

Proposition 2.23. *Put $f : (X, \odot) \to (X', \otimes)$ is a monoid homomorphism and GB' is a GB-metric on X' the following results are valid:*

- *$G_B : X \times X \times X \to B$, $(x, y, z) \mapsto G'(f(x), f(y), f(z))$ is a G-metric on $(X, \odot)$.*
- *If G'_B is m-invariant, then G_B is m-invariant.*
- *The GB-metric G_B on X is free (or effective) if G'_B is free (or effective) on X'.*

Let E and F be subsets of X. The set E is congruent to F, written $E \approx F$, provided there exist a map $f : E \to F$ that for any $x, y, z \in E$, $G(x, y, z) = G_B(f(x), f(y), f(z))$ and that F is congruent to E by the map $f^{-1} : F \to E$. The map ($or\ f^{-1}$) is called a congruence between E and F. Clearly every single set $\{x\}$, $x \in X$ and each map f, $\{x\}$ congruent to $\{(x)\}$.

Definition 2.24. A motion of X is a congruence of X with itself.

Proposition 2.25. *Let* (X, G_B) *be a* GB*-metric space. The set* $\mathcal{M}(X) = \{f : X \to X \mid f$ *is a group.*

Proof. The identity map belongs to $\mathcal{M}(X)$ so $\mathcal{M}(X)$ is not empty. Group action is composition of maps, so the identity map is the identity element. If $g \in \mathcal{M}(X)$, then $G_B(g^{-1}(x), g^{-1}(y), g^{-1}(z)) = G_B(g(g^{-1}(x)), g(g^{-1}(y)), g(g^{-1}(z))) = (x, y, z)$, so $g^{-1} \in \mathcal{M}(x)$.

Theorem 2.26. *Let* (X, G_B) *be a* GB*-metric space and* $f : X \to X$ *is a motion of* X*, with the properties: if* $\{f^n\} \to x$*, then* $G_B(f^{n_i+1}(x), f^{n_i+1}(x), f^n(x)) \to 0$*, then the set of fixpoints of* f *(or* (f)*) is equal to the set of fixpoints of* f^n*(or*$F(f^n)$*) for all* $n \in \mathbb{N}$.

Proof. Clearly $(f) \subseteq F(f^n)$. Suppose $x \in (f^n)$, so the sequence $\{f^{nk}(x)\}_{k=1}^{\infty} \to x$, because $f^{kn}(x) = x$, for all $k \in \mathbb{N}$. Now by Theorem 2.16, we conclude $(x) = x$.

Proposition 2.27. *Let* $(X, *)$ *be a monoid and* G_B *be a* GB*-metric on* X.

- *If for all* $a \in X$, L_a *and* R_a *be motions of* X*, then* G_B *is m-invariant.*
- L_a *(or* R_a*)* $(a \neq e)$ *cannot be a motion of* X*, if* G_B *is effective.*

Example 2.28. The GB-metric defined in Example 2.4 is invariant on the monoid $(\mathbb{W}, +)$, but it is not effective. Clearly for all $k \in \mathbb{N}$, $f_k : \mathbb{W} \to \mathbb{W}$, $n \mapsto n + k$ is a motion.

REFERENCES

1. Z. Mustafa and B. Sims, “A new approach to generalized metric spaces,” *Journal of Nonlinear and Convex Analysis*, vol. 7, no. 2, pp. 289–297, 2006.
2. D. Ellis and H. D. Sprinkle, “Topology of *B*-metric spaces,” *Compositio Mathematica*, vol. 12, pp. 250–262, 1956.
3. A. Avilés, “Extensions of Boolean isometries,” *Discrete Mathematics*, vol. 297, no. 1–3, pp. 1–12, 2005.
4. D. Ellis, “Autometrized Boolean algebras I: fundamental distance-theoretic properties of B,” *The American Mathematical Society*, no. 25, pp. 1–12, 1949.
5. R. Chugh, T. Kadian, A. Rani, and B. E. Rhoades, “Property *P* in *G*-metric spaces,” *Fixed Point Theory and Applications*, vol. 2010, Article ID 401684, 12 pages, 2010.
6. Z. Mustafa and W. Sims, “Fix point theorem on uncomplete *G*-metric spaces,” *Journal of Mathematics and Statistics*, vol. 4, pp. 196–201, 2006.
7. S. Romaguera and M. Schellekens, “Partial metric monoids and semivaluation spaces,” *Topology and Its Applications*, vol. 153, no. 5-6, pp. 948–962, 2005.

Chapter

7

INCOHERENCY PROBLEMS IN A COMBINATION OF DESCRIPTION LOGICS AND RULES

Shasha Huang,[1] Jing Hao,[2] and Dang Luo[1]

[1]College of Mathematics and Information, North China University of Water Resources and Electric Power, Zhengzhou, Henan 450045, China
[2]College of Mathematics and Information, Henan University of Economics and Law, Zhengzhou, Henan 450000, China

ABSTRACT

A paraconsistent semantics has been presented for hybrid MKNF knowledge bases—a combination method for description logics and rules. However, it is invalid when incoherency occurs in the knowledge base. In this paper, we introduce a semi-S_5 semantics for hybrid MKNF knowledge bases on the basis of nine-valued lattice, such that it is paraconsistent for incoherent

Citation: Shasha Huang, Jing Hao, and Dang Luo, “Incoherency Problems in a Combination of Description Logics and Rules”, Journal of Applied Mathematics, volume 2014, article ID 604753, https://doi.org/10.1155/2014/604753.

knowledge base. It is shown that a semi-S_5 model can be computed via a fixpoint operator and is in fact a paraconsistent MKNF model when the knowledge base is incoherent. Moreover, we apply six-valued lattice to hybrid MKNF knowledge bases and present a suspicious semantics to distinguish different trust level information. At last, we investigate the relationship between suspicious semantics and paraconsistent semantics.

INTRODUCTION AND MOTIVATION

The Semantic Web [1] extends the current World Wide Web by standards and techniques that help machines to understand the meaning of data on the web to enable more powerful intelligent system applications. The essence of the Semantic Web is to describe data on the web by metadata that conveys the meaning of the data and that is expressed by means of ontologies.

Web Ontology Language (OWL) [1] is based on the Description Logic $\mathcal{SROIQ}(\mathcal{D})$ [2] and has been recommended by the World Wide Web Consortium for representing ontologies. However, as monotonic logic, description logics (DLs for short) are not as expressive as needed for modeling some real world problems. Consequently, how to improve OWL has become a very important branch of research in the Semantic Web field, and one of the hot topics is how to better combine DL and rules in the sense of logic programming (LP), which is complementary to modeling in DL with respect to expressivity, have become a mature reasoning mechanism in the past thirty years.

Several integration methods have been proposed. As a bridge between monotonic reasoning and nonmonotonic reasoning, hybrid MKNF knowledge bases have favourable properties of decidability, flexibility, faithfulness, and tightness. However, due to nonmonotonicity of rules, hybrid MKNF knowledge bases may be incoherent; that is, they do not have an MKNF model due to cyclic dependencies of a modal atom from default negation of the atom in the rule part. Standard reasoning systems will break down in this case. Nevertheless, one might want to derive useful information from incoherent hybrid MKNF knowledge bases. This is similar to paraconsistency, where nontrivial consequences shall be derivable from an inconsistent theory. For distinguishing the former reasoning with the later, we use term paracoherent reasoning to denote reasoning with incoherent knowledge bases. Both types of reasoning for rules have been studied, for example, Sakama and Inoue [3] and Eiter et al. [4]. For hybrid MKNF knowledge bases, Huang et al. [5] presented paraconsistent semantics for it, where only inconsistency can be handled. In this paper, we study the incoherency problem in hybrid MKNF knowledge bases and present a paracoherent reasoning system such that nontrivial conclusions can be drawn from incoherent knowledge bases.

The remainder of the paper is organized as follows. In Section 2 we have a quick look over hybrid MKNF knowledge bases. In Section 3, we present paracoherent semantics for hybrid MKNF knowledge bases on the basis of nine-valued lattice. In Section 4, we give suspicious MKNF models for such knowledge bases to distinguish different trust level information. In Section 5, we discuss the related work. We conclude and discuss the future work in Section 6.

HYBRID MKNF KNOWLEDGE BASES

At first, the logic of MKNF is a variant of first-order modal logic with two modal operators: K and not. We present the syntax of MKNF formulae taken from [6]. Let Σ be a signature that consists of constants and function symbols and first-order predicates, including the binary equality predicate ≈. A first-order atom $(t_1,...,)$ is an MKNF formula, where P is a first-order predicate and t_i are first-order terms. Other MKNF formulae are built over Σ by using standard connectives in first-order logic and two modal operators as follows: true, $\neg\varphi$, $\varphi_1 \wedge \varphi_2$, $\exists : \varphi$, Kφ, notφ. Moreover, the symbols ∨, ⊃, ∀, and ≡ represent the usual boolean combination of previously introduced connectors. Formulae of the form Kφ (notφ) are called *modal* K-*atoms* (not-*atoms*). Modal K-*atoms* and not-*atoms* are called *modal* atoms. An MKNF formula φ is called *closed* if it contains no free variables and called *ground* if it is without any variables. An MKNF formula φ is called *modally closed* if it is closed and all modal operators are applied to closed subformulae. $[t/x]$ is the formula obtained from φ by substituting the term t for the variable x. Moreover, the equality predicate ≈ in Σ is interpreted as an equivalence relation on Δ, which is called a *universe* and contains an infinite supply of constants, besides the constants occurring in the formulae.

As shown in [6], hybrid MKNF knowledge bases consist of a finite number of MKNF rules and a decidable description logic knowledge base $\mathcal{O}$, which satisfies the following conditions: (i) each knowledge base $\mathcal{O} \in \mathcal{DL}$ can be translated to a formula $\pi(\mathcal{O})$ of function-free first-order logic with equality (see [2] for standard translation for description logic axioms), (ii) it supports ABox assertions of the form $P(t_1, \ldots, t_l)$, where is a predicate and each t_i a constant of $\mathcal{DL}$, and (iii) satisfiability checking and instance checking (i.e., checking entailments of the form $\mathcal{O} \models P(t_1, \ldots, t_l)$) are decidable.

Definition 1. Let $\mathcal{O}$ be a DL knowledge base. A first-order function-free atom (t_1,...,) over Σ such that P is ≈ or it occurs in $\mathcal{O}$ is called a DL atom; all other atoms are called non-DL atoms. An MKNF rule r has the following form where H_i, A_i, B_i, are first-order function-free atoms:

$$\mathbf{K}H_1 \vee \ldots \vee \mathbf{K}H_n \\ \longleftarrow \mathbf{K}A_{n+1} \wedge \ldots \wedge \mathbf{K}A_m \wedge \mathbf{not}B_{m+1} \wedge \ldots \wedge \mathbf{not}B_k. \tag{1}$$

The sets {KH_i}, {KA_i}, and {notB_i} are called the rule head, the positive body, and the negative body, respectively. An MKNF rule r is nondisjunctive if n = 1; r is positive if m = k; r is a fact if m = k = 0. A program $\mathcal{P}$ is a finite set of MKNF rules. A hybrid MKNF knowledge base $\mathcal{K}$ is a pair $(\mathcal{O}, \mathcal{P})$.

To ensure that the MKNF logic is decidable, DL safety is introduced as a restriction to MKNF rules.

Definition 2. An MKNF rule is DL safe if every variable in r occurs in at least one non-DL atom KB occurring in the body of r. A hybrid MKNF knowledge base $\mathcal{K}$ is DL safe if all its rules are DL safe.

In the rest of this paper, without explicitly stating it, we only consider hybrid MKNF knowledge bases which are DL safe.

Definition 3. Given a hybrid MKNF knowledge base $\mathcal{K} = (\mathcal{O}, \mathcal{P})$, the ground instantiation of $\mathcal{K}$ is the knowledge base $\mathcal{K}_G = (\mathcal{O}, \mathcal{P}_G)$, where $\mathcal{P}_G$ is obtained from $\mathcal{P}$ by replacing each rule r of $\mathcal{P}$ with a set of rules substituting each variable in r with constants from $\mathcal{K}$ in all possible ways.

Grounding the knowledge base $\mathcal{K}$ ensures that rules in $\mathcal{P}$ apply only to objects that occur in $\mathcal{K}$. And it has been proved by Motik and Rosati [6] that the MKNF models of $\mathcal{K}$ and $\mathcal{K}_G$ coincide.

Hybrid MKNF knowledge bases provide a paradigm for representing data sources on the web by rules and description logics simultaneously. Local closed world reasoning in the knowledge bases bridges the rules and DLs, accordingly overcomes the expressive limitation of rules and DLs, and enhance the expressivity.

PARACOHERENT SEMANTICS FOR HYBRID MKNF KNOWLEDGE BASE

Huang et al. [5] presented a four-valued paraconsistent semantics for hybrid MKNF knowledge bases, which can handle inconsistent information in the knowledge base. However, there is a kind of knowledge base which has no four-valued paraconsistent MKNF model but still contains useful information, for instance, the following example.

Example 4. Let $\mathcal{K}_G = (\mathcal{O}, \mathcal{P}_G)$ be a ground knowledge base, where $\mathcal{O} = \{p\}$ and $\mathcal{P}_G = \{\mathbf{K}a \leftarrow \mathbf{not}a\}$ (p, a are literals).

From [5], we know that $\mathcal{K}_G$ has no paraconsistent MKNF model. Generally, MKNF rule of the form Ka ← not$a$ will lead to incoherency, which is a kind of inconsistency but cannot be handled by four-valued paraconsistent semantics. Therefore, it is desirable to provide a framework for incoherent knowledge bases. In this section, we will present a nine-valued semantics which is paraconsistent for incoherent knowledge bases.

Firstly, we introduce the nine-valued lattice $\mathcal{N}_9$. Besides the four basic values t, f, ⊤, and ⊥, which constitute four-valued lattice $\mathcal{FOUR}$ and, respectively, represent true, false, contradictory (both true and false), and unknown (neither true nor false), $\mathcal{N}_9$ contains five extra truth values bt, bf, b⊤, tcb, and fcb, which denote believed true, believed false, believed contradictory, true with contradictory belief, and false with contradictory belief, respectively. These values constitute a lattice of nine-valued logic $\mathcal{N}_9$ [3] (as shown in Figure 1) such that ⊥≤ bf ≤ x ≤ xcb ≤ ⊤ and bx ≤ b⊤ ≤ xcb (x ∈ {t, f}).

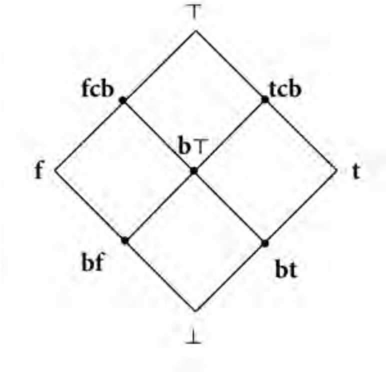


Figure 1. Nine-valued Lattice $\mathcal{N}_9$.

Let $\mathscr{F}$ be a first-order theory and $\mathscr{B}(\mathscr{F})$ is the Herbrand base of $\mathscr{F}$. Let $\mathscr{F}^k = \mathscr{B}(\mathscr{F}) \cup \{\mathsf{K}A \mid$ A is a literal in $\mathscr{B}(\mathscr{F})\}$, and let I be a subset of $\mathscr{F}^k$. Then a *nine-valued interpretation* under the logic $\mathscr{N}_9$ is defined as a function $I\colon \mathscr{B}(\mathscr{F}) \to \mathscr{N}_9$ such that, for each literal $A \in \mathscr{B}(\mathscr{F})$,

$$\begin{aligned}(A)^I = \mathrm{lub}\,\{x \mid {} & x = \mathbf{t} \text{ if } A \in I,\\ & x = \mathbf{f} \text{ if } \neg A \in I,\\ & x = \mathbf{bt} \text{ if } \mathsf{K}A \in I,\\ & x = \mathbf{bf} \text{ if } \mathsf{K}\neg A \in I,\\ & x = \bot \ \text{ otherwise}\},\end{aligned} \qquad (2)$$

where the term lub denotes *least upper bound*.

Every formula in $\mathscr{F}$ is assigned a value in $\mathscr{N}_9$. The intuitive meaning of new introduced operator K is "*belief*." For instance, $\mathsf{K}A \in I$ means we believe that literal A belongs to interpretation I, which coincides with the truth value bt. By the order structure of nine-valued lattice, $(A)^I = \mathbf{b}\top$ iff both $\mathsf{K}A \in I$ and $\mathsf{K}\neg A \in I$; $(A)^I = \mathbf{fcb}$ iff both $\mathsf{K}A \in I$ and $\neg A \in I$; $(A)^I = \mathbf{tcb}$ iff both $A \in I$ and $\mathsf{K}\neg A \in I$. Furthermore, $(A)^I = \mathbf{t}$ iff $(\neg A)^I = \mathbf{f}$, $(A)^I = \top$ iff $(\neg A)^I = \top$, $(A)^I = \bot$ iff $(\neg A)^I = \bot$, $(A)^I = \mathbf{bt}$ iff $(\neg A)^I = \mathbf{bf}$, $(A)^I = \mathbf{tbc}$ iff $(\neg A)^I = \mathbf{fbc}$, and $(A)^I = \mathbf{b}\top$ iff $(\neg A)^I = \mathbf{b}\top$.

Under this logic, satisfaction of literals and default negation is defined as follows: $I \models_9 A$ iff $\mathbf{t} \le (A)^I$, $I \models_9 \neg A$ iff $\mathbf{f} \le (A)^I$, $I \models_9 \mathbf{not}A$ iff $(A)^I \le \mathbf{f}$, and $I \models_9 \mathbf{not}\neg A$ iff $(A)^I \le \mathbf{t}$. Satisfaction of other connectors is defined as usual.

Now we come to the nine-valued semantics for hybrid MKNF knowledge bases.

For distinguishing the two hybrid MKNF knowledge bases with stable model semantics and nine-valued semantics, we call the latter $\mathscr{N}_9$-MKNF knowledge bases.

We use the syntax of para-MKNF knowledge base presented by Huang et al. [5] as the syntax of $\mathcal{N}_9$-MKNF knowledge bases, which is similar to the classical knowledge base as presented in Section 2. The only difference between them is that MKNF rules are restricted to literals in para-MKNF knowledge bases. In our paper, we use MKNF rules defined as follows:

$$\begin{aligned} &\mathbf{K}H_1 \vee \ldots \vee \mathbf{K}H_n \\ &\qquad \longleftarrow \mathbf{K}A_{n+1} \wedge \ldots \wedge \mathbf{K}A_m \wedge \mathbf{not}B_{m+1} \wedge \ldots \wedge \mathbf{not}B_k, \end{aligned} \tag{3}$$

where H_i, A_i, and B_i are first-order function-free literals.

Semantically, we first introduce $\mathcal{N}_9$-MKNF structure $(\mathcal{I}, \mathcal{M}, \mathcal{N})$.

Definition 5. An $\mathcal{N}_9$-MKNF structure $(\mathcal{I}, \mathcal{M}, \mathcal{N})$ consists of a nine-valued interpretation $\mathcal{I}$ and two nonempty sets of nine-valued interpretation interpretations $\mathcal{M}$ and $\mathcal{N}$. A nonempty set of nine-valued interpretations $\mathcal{M}$ is called a $\mathcal{N}_9$-MKNF interpretation.

Definition 6. Let $(\mathcal{I}, \mathcal{M}, \mathcal{N})$ be a $\mathcal{N}_9$-MKNF structure. $\mathcal{N}_9$ satisfaction of closed MKNF formulae is defined inductively as follows:

$$\begin{aligned}
&(\mathcal{I}, \mathcal{M}, \mathcal{N}) \vDash_9 P(t_1, \ldots, t_l) \quad \text{iff } P^{\mathcal{I}}(t_1, \ldots, t_l) \geq \mathbf{t}, \\
&(\mathcal{I}, \mathcal{M}, \mathcal{N}) \vDash_9 \neg\varphi \quad \text{iff } (\mathcal{I}, \mathcal{M}, \mathcal{N})(\varphi) \geq \mathbf{f}, \\
&(\mathcal{I}, \mathcal{M}, \mathcal{N}) \vDash_9 \varphi_1 \wedge \varphi_2 \\
&\quad \text{iff } (\mathcal{I}, \mathcal{M}, \mathcal{N}) \vDash_9 \varphi_i,\ i = 1, 2, \\
&(\mathcal{I}, \mathcal{M}, \mathcal{N}) \vDash_9 \exists x : \varphi \quad \text{iff } (\mathcal{I}, \mathcal{M}, \mathcal{N}) \vDash_9 \varphi[ax] \\
&\qquad \text{for some } \alpha \in \Delta, \\
&(\mathcal{I}, \mathcal{M}, \mathcal{N}) \vDash_9 \varphi_1 \supset \varphi_2 \quad \text{iff } (\mathcal{I}, \mathcal{M}, \mathcal{N}) \nvDash_9 \varphi_1 \\
&\qquad \text{or } (\mathcal{I}, \mathcal{M}, \mathcal{N}) \vDash_9 \varphi_2, \\
&(\mathcal{I}, \mathcal{M}, \mathcal{N}) \vDash_9 \mathbf{K}\varphi \quad \text{iff } (\mathcal{J}, \mathcal{M}, \mathcal{N}) \vDash_9 \varphi\ \forall \mathcal{J} \in \mathcal{M}, \\
&(\mathcal{I}, \mathcal{M}, \mathcal{N}) \vDash_9 \mathbf{not}\varphi \quad \text{iff } (\mathcal{J}, \mathcal{M}, \mathcal{N})(\varphi) \leq \mathbf{f} \\
&\qquad \text{for some } \mathcal{J} \in \mathcal{N}.
\end{aligned} \tag{4}$$

A $\mathcal{N}_9$-MKNF interpretation $\mathcal{M}$ is a *semi-S_5 model* of a given closed MKNF formula φ, written as $\mathcal{M} \vDash_9 \varphi$ if and only if $(\mathcal{I}, \mathcal{M}, \mathcal{M}) \vDash_9 \varphi$ for each $\mathcal{I} \in \mathcal{M}$.

How to obtain models of a knowledge base is a basic problem in the reasoning process. The next work is around this topic.

Let $\mathcal{K}_G = (\mathcal{O}, \mathcal{P}_G)$ be a ground $\mathcal{N}_9$-MKNF knowledge base. The set of K-atoms of $\mathcal{K}_G$, written as $\mathsf{KA}(\mathcal{K}_G)$, is the smallest set that contains (1) all ground K-atoms occurring in $\mathcal{P}_G$ and (3) a modal atom Kξ for each ground modal atom notξ occurring in $\mathcal{P}_G$. Let $\mathsf{KA}^k(\mathcal{K}_G) = \mathsf{KA}(\mathcal{K}_G) \cup \{\mathbf{KK}A \mid$ KA is an element in $\mathsf{KA}(\mathcal{K}_G)\}$. Furthermore, $\mathsf{HA}(\mathcal{K}_G)$ is the subset of $\mathsf{KA}(\mathcal{K}_G)$ that contains all K-atoms occurring in the head of some rule in $\mathcal{P}_G$. $\mathsf{HA}^k(\mathcal{K}_G)$ is a subset of $\mathsf{KA}^k(\mathcal{K}_G)$.

We now recall the fixpoint operator of positive paraconsistent MKNF knowledge base, which will be used to search for the semi-S_5 models of $\mathcal{N}_9$ -MKNF knowledge bases.

Definition 7 (see [5]). Let $\mathcal{K}_G = (\mathcal{O}, \mathcal{P})$ be a ground positive para-MKNF knowledge base and $\mathbb{S} \in 2^{2^{\mathsf{HA}(\mathcal{K}_G)}}$. A mapping $\mathfrak{T}_{\mathcal{K}_G} : 2^{2^{\mathsf{HA}(\mathcal{K}_G)}} \to 2^{2^{\mathsf{HA}(\mathcal{K}_G)}}$ is defined as

$$\mathfrak{T}_{\mathcal{K}_G}(\mathbb{S}) = \bigcup_{\mathsf{S} \in \mathbb{S}} T_{\mathcal{K}_G}(\mathsf{S}), \tag{5}$$

where the mapping $T_{\mathcal{K}_G} : 2^{\mathsf{HA}(\mathcal{K}_G)} \to 2^{2^{\mathsf{HA}(\mathcal{K}_G)}}$ is defined as follows.

- If $\mathsf{OB}_{\mathcal{O},\mathsf{S}} \models_4 A_i, n+1 \leq i \leq m$ for some ground integrity constraint $\leftarrow \mathbf{K}A_n + 1 \wedge \ldots \wedge \mathbf{K}A_m$ in $\mathcal{P}_G$, then $T_{\mathcal{K}_G}(\mathsf{S}) = \emptyset$.
- Otherwise, $T_{\mathcal{K}_G}(\mathsf{S}) = \{\mathsf{Q}_t \subseteq \mathsf{HA}(\mathcal{K}_G) \mid \mathsf{Q}_t = \mathsf{S} \cup \mathsf{R}_t \cup \mathsf{H}\}$, where $\mathsf{R}_t = \{\mathbf{K}H_t \mid$ for each ground MKNF rule $C_j \in \mathcal{P}_G : \mathsf{OB}_{\mathcal{O},\mathsf{S}} \models_4 A_i, n+1 \leq i \leq m\}$ and $\mathsf{H} = \{\mathbf{K}\xi \in \mathsf{HA}(\mathcal{K}_G) \mid \mathsf{OB}_{\mathcal{O},\mathsf{S}} \models_4 \xi\}$.

Then we can use the following fixpoint procedure to compute paraconsistent MKNF models of positive para-MKNF knowledge bases:

$$\begin{aligned} \mathfrak{T}_{\mathcal{K}_G} \uparrow 0 &= \emptyset, \\ \mathfrak{T}_{\mathcal{K}_G} \uparrow n+1 &= \mathfrak{T}_{\mathcal{K}_G}\left(\mathfrak{T}_{\mathcal{K}_G} \uparrow n\right), \\ \mathfrak{T}_{\mathcal{K}_G} \uparrow \omega &= \bigcup_{\alpha<\omega}\bigcap_{\alpha\leq n<\omega} \mathfrak{T}_{\mathcal{K}_G} \uparrow n, \end{aligned} \tag{6}$$

where n is a successor ordinal and ω is a limit ordinal.

For general para-MKNF knowledge bases, a transformation was presented.

Definition 8. Let $\mathcal{K}_G = (\mathcal{O}, \mathcal{P}_G)$ be a ground para-MKNF knowledge base. Then its transformation is defined as $\mathcal{K}_G^*$ obtained by replacing each general rule in $\mathcal{P}_G$ with the following positive MKNF rule

$$\begin{aligned} &\mathbf{K}\mu_1 \vee \ldots \vee \mathbf{K}\mu_n \vee \mathbf{K}\mathbf{K}B_{m+1} \vee \ldots \vee \mathbf{K}\mathbf{K}B_k \\ &\qquad \longleftarrow \mathbf{K}A_{n+1} \wedge \ldots \wedge \mathbf{K}A_m, \end{aligned} \tag{7}$$

$$\mathbf{K}H_i \longleftarrow \mathbf{K}\mu_i \quad \text{for } 1 \leq i \leq n, \tag{8}$$

$$\longleftarrow \mathbf{K}\mu_i \wedge \mathbf{K}B_j \quad \text{for } 1 \leq i \leq n,\ m+1 \leq j \leq k, \tag{9}$$

$$\mathbf{K}\mu_i \longleftarrow \mathbf{K}H_i \wedge \mathbf{K}\mu_j \quad \text{for } 1 \leq i,\ j \leq n. \tag{10}$$

Let $\gamma(\mathfrak{T}_{\mathcal{K}_G} \uparrow \omega) = \{\mathsf{S} \mid \mathsf{S} \in \mathfrak{T}_{\mathcal{K}_G} \uparrow \omega,$ and $\mathsf{S} \in \mathfrak{T}_{\mathcal{K}_G}(\{\mathsf{S}\})\}$ and min$(\mathbb{S}) = \{\mathsf{S} \mid$ there exists no $\mathsf{Q} \in \mathbb{S}$ such that $\mathsf{Q} \subset \mathsf{S}\}$. Given a set $\mathbb{S}^*$ that is a subset of $2^{\mathsf{HA}(\mathcal{K}_G^*)}$, $\mathbb{S}^*$ is *canonical* if $\mathbf{K}\mathbf{K}a \in \mathbb{S}^*$ implies $\mathbf{K}a \in \mathbb{S}^*$. $\Phi(\mathbb{S}^*) = \{\mathsf{S}^* \cap \mathsf{KA}(\mathcal{K}_G) \mid \mathsf{S}^* \in \mathbb{S}^*$ and $\mathbb{S}^*$ is *canonical*$\}$.

Theorem 9. *Let* $\mathcal{K}_G = (\mathcal{O}, \mathcal{P}_G)$ *be a ground para-MKNF knowledge base, and then each paraconsistent MKNF model of* $\mathcal{K}_G$ *equals* $\mathcal{M} = \{\mathcal{I} \mid \mathcal{I} \models_4 \mathsf{OB}_{\mathcal{O},P_h}\}$, *where* P_h *is an element of the set* $\mathbb{Q} = \Phi(\min(\gamma(\mathfrak{T}_{\mathcal{K}_G^*} \uparrow \omega)))$.

Note that the transformation of general MKNF rules is a little different from the one in [5]. However, this does not affect the result of Theorem 9. In fact, both two transformations have the same essence, transforming default negation in the rule body to a literal in rule head, and the unique difference is between “KKB_k” and “KB_k” in the transformed MKNF rules. But from the canonical condition, KKB_k implies KB_k, which does not change the original proof of [5,Theorem 4]. Therefore, Theorem 9 still holds if replacing Ka in (3) with KKa. In a previous work, we have mentioned that Ka is interpreted by “belief true,” corresponding to the truth value “bt.” As we have defined, a nine-valued interpretation can be represented by special Herbrand interpretation equipped with new elements of form Kε based on classical Herbrand interpretation, in which ε is a literal.

Given a set $\mathbb{S}^*$ that is a subset of $2^{\mathsf{HA}(\mathscr{K}_G^*)}$, $\mathbb{S}^*$ is *maximally canonical* if there is no subset $\mathbb{S}_1^*$ of $2^{\mathsf{HA}(\mathscr{K}_G^*)}$, such that $\{\mathbf{KK}a \mid \mathbf{KK}a \in \mathbb{S}_1^*$ and $\mathbf{K}a \notin \mathbb{S}_1^*\} \subset \{\mathbf{KK}a \mid \mathbf{KK}a \in \mathbb{S}^*$ and $\mathbf{K}a \notin \mathbb{S}^*\}$:

$$\Phi_{\mathrm{mc}}(\mathbb{S}^*) = \{S^* \cap \mathsf{KA}(\mathscr{K}_G) \mid S^* \in \mathbb{S}^* \text{ and } \mathbb{S}^* \text{ is canonical}\}. \tag{11}$$

With maximally canonical condition, semi-S_5 models of a hybrid MKNF knowledge base can be computed by the fixpoint operator presented in Definition 7.

Theorem 10. *Let* $\mathscr{K}_G = (\mathscr{O}, \mathscr{P}_G)$ *be a* $\mathscr{N}_9$*-MKNF knowledge base, if* P_h *is an element of the set* $\mathbb{Q} = \Phi_{mc}(\min(\gamma(\mathfrak{T}_{\mathscr{K}_G^*} \uparrow \omega)))$ *and* $\mathscr{M} = \{\mathscr{I} \mid \mathscr{I} \models_9 \mathsf{OB}_{\mathscr{O},P_h}\}$*, then* $\mathscr{M}$ *is a semi-*S_5 *model of* $\mathscr{K}_G$*.*

Proof. Given a maximally canonical element $P_h^* \in \min(\gamma(\mathfrak{T}_{\mathscr{K}_G^*} \uparrow \omega))$ and $\mathscr{M}^* = \{\mathscr{I} \mid \mathscr{I} \models_9 \mathsf{OB}_{\mathscr{O},P_h^*}\}$. $\mathscr{M}^*$ is a paraconsistent S_5 model of $\mathscr{K}_G^*$ by [5, Lemma 4]. For each transformed MKNF rules (3) and (7), if $\mathscr{M}^* \models_9 \mathbf{K}A_i$, for each $n + 1 \le i \le m$, then either $\mathscr{M}^* \models_9 \mathbf{K}H_j$, for some $1 \le k \le n$, or $\mathscr{M}^* \models_9 \mathbf{KK}B_t$, for some $m + 1 \le t \le k$. Case 1: $\mathscr{M}^* \models_9 \mathbf{K}H_j$, for some $1 \le k \le n$, then the corresponding MKNF rule of form (1) is satisfied. Case 2: $\mathscr{M}^* \models_9 \mathbf{KK}B_t$, for some $m + 1 \le t \le k$. Then for each nine-valued interpretation $I \in \mathscr{M}^*$, $I \models_9 \mathrm{K}B_t$, which means $B_t^{\mathscr{I}} \ge \mathbf{bt}$ and then $\mathscr{M} \not\models_9 \mathbf{not}B_t$. In either case, corresponding MKNF rule of form (1) is satisfied. Let $P_h = P_h^* \cap \mathsf{KA}(\mathscr{K}_G)$ and $\mathscr{M} = \{\mathscr{I} \mid \mathscr{I} \models_9 \mathsf{OB}_{\mathscr{O},P_h}\}$, and then $\mathscr{M}$ is a semi-S_5 model of $\mathscr{K}_G$.

Corollary 11. *If $\mathcal{K}_G$ is a coherent knowledge base, then its semi-S_5 model coincides with paraconsistent MKNF model.*

Proof. When $\min(\gamma(\mathfrak{T}_{\mathcal{K}^*_G} \uparrow \omega))$ contains canonical element, it is also maximally canonical. Therefore, the result holds.

Theorem 12. *Let $\mathcal{K}_G = (\mathcal{O}, \mathcal{P}_G)$ be a ground hybrid MKNF knowledge base, if $\mathcal{K}_G$ has an S_5 model, it has a semi-S_5 model.*

Proof. If $\mathcal{K}_G$ has an S_5 model, then it is easy to construct a MKNF interpretation that satisfies $\mathcal{K}^*_G$. Then $\min(\gamma(\mathfrak{T}_{\mathcal{K}^*_G} \uparrow \omega))$ contains maximally canonical elements. Thus $\Phi_{\mathrm{mc}}(\min(\gamma(\mathfrak{T}_{\mathcal{K}^*_G} \uparrow \omega)))$ is not empty. Theorem holds.

Example 13. Consider the incoherent knowledge base $\mathcal{K}_G$ from Example 4. By Definition 8, $\mathcal{P}_G$ is transformed to $\mathcal{P}^*_G$:

$$\begin{aligned} \mathbf{K}\mu \vee \mathbf{K}\mathbf{K}a &\longleftarrow \mathbf{K}a \\ &\longleftarrow \mathbf{K}\mu \\ &\longleftarrow \mathbf{K}\mu \wedge \mathbf{K}a. \end{aligned} \tag{12}$$

We compute the fixpoint by applying the procedure presented in Section 3 to the knowledge base $\mathcal{K}^*_G = (\mathcal{O},\mathcal{P}^*_G)$. By evaluating $\mathfrak{T}_{\mathcal{K}_G} \uparrow n+1 = \mathfrak{T}_{\mathcal{K}_G}\mathfrak{T}_{\mathcal{K}_G} \uparrow n$ recursively, $\min(\mathfrak{T}_{\mathcal{K}^*_G} \uparrow n) = \{\{\mathbf{K}\mathbf{K}a, \mathbf{K}p\}\}$. Also, it can be easily verified that $\mathcal{M} = \{\mathcal{I} \mid \mathcal{I} \models_9 \{\mathrm{K}a, p\}\}$.

SUSPICIOUS MKNF MODELS

As the beginning of this section, we give a motivation example as follows.

Example 14. Let $\mathcal{K}_G = (\mathcal{O}, \mathcal{P}_G)$ be a ground knowledge base, where $\mathcal{O} = \{p\}, \mathcal{P}_G = \{\mathbf{K}a \leftarrow \mathbf{K}\neg p, \mathbf{K}\neg p \leftarrow, \mathbf{K}c \leftarrow\}$ (p, a, c are literals).

In the above paraconsistent hybrid MKNF knowledge base $\mathcal{K}_G$, both c and a are the consequence of it. However, it is not difficult to find that a is derived by inconsistent information, while c is not. Apparently a is less credible than *c*.Therefore, it is necessary to distinguish information derived by inconsistencies from others.

In order to distinguish two kinds of information, we introduce six-valued lattice, which is used by Sakama and Inoue [3] to present suspicious stable models for a program. As shown in Figure 2, there are two new introduced values sf and sf in six-valued lattice VI, which, respectively, stand for *suspiciously false* and *suspiciously true*. These newly introduced values together with $\mathscr{FOUR}$ constitute six-valued lattice such that $\bot \leq \text{sx} \leq \text{x} \leq \top$ $(\text{x} \in \{\text{t}, \text{f}\})$.

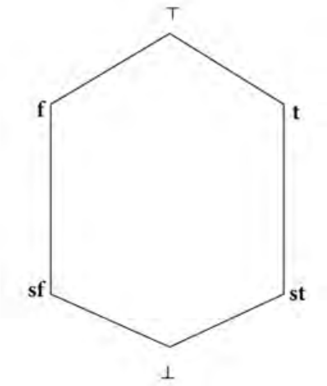


Figure 2. Six-valued Lattice VI.

Let $\mathscr{F}$ be a first-order theory, $\mathscr{F}^s = \mathscr{B}(\mathscr{F}) \cup \{A^s \mid A$ is a literal in $\mathscr{B}(\mathscr{F})\}$, and *I* be a subset of $\mathscr{F}^s$. Then a six-valued interpretation *I* under the logic VI is defined as a function *I*: $\mathscr{B}(\mathscr{F}) \rightarrow \mathbf{VI}$ such that, for each literal $A \in \mathscr{B}(\mathscr{F})$,

$$\begin{aligned}(A)^I = \text{lub}\,\{x \mid x = \mathbf{t} \text{ if } A \in I,& \\ x = \mathbf{f} \text{ if } \neg A \in I,& \\ x = \mathbf{st} \text{ if } A^s \in I,& \\ x = \mathbf{sf} \text{ if } \neg A^s \in I,& \\ x = \bot \ \text{ otherwise}\}.&\end{aligned} \quad (13)$$

Note that (A) = st if and only if $(\neg A)$ = sf. Under the logic VI, satisfaction of literals and default negation is defined as follows: $I \vDash_6 A$ if and only if st $\leq (A)^I$, $I \vDash_6 \neg A$ if and only if sf $\leq (A)^I$, $I \vDash_6 \text{not}A$ if and only if $(A)^I \leq$ f, and *I*

$\models_6$ not$\neg A$ if and only if $(A)^I \leq$ t. Satisfaction of other connectors is defined as usual.

Given a hybrid MKNF knowledge base, its suspicious S_5 models are defined by suspicious MKNF structure $(\mathcal{I}, \mathcal{M}, \mathcal{N})$, which is defined as usual.

Definition 15. A suspicious MKNF structure $(\mathcal{I}, \mathcal{M}, \mathcal{N})$ consists of a six-valued interpretation $\mathcal{I}$ and two nonempty sets of six-valued interpretation interpretations $\mathcal{M}$ and $\mathcal{N}$. A nonempty set of six-valued interpretations $\mathcal{M}$ is called a suspicious MKNF interpretation.

Definition 16. Let $(\mathcal{I}, \mathcal{M}, \mathcal{N})$ be a suspicious MKNF structure. Six-valued satisfaction of closed MKNF formulae is defined inductively as follows:

$$
\begin{aligned}
&(\mathcal{I},\mathcal{M},\mathcal{N}) \models_6 P(t_1,\ldots,t_l) \quad \text{iff } P^{\mathcal{I}}(t_1,\ldots,t_l) \geq \mathbf{st},\\
&(\mathcal{I},\mathcal{M},\mathcal{N}) \models_6 \neg\varphi \quad \text{iff } (\mathcal{I},\mathcal{M},\mathcal{N})(\varphi) \geq \mathbf{sf},\\
&(\mathcal{I},\mathcal{M},\mathcal{N}) \models_6 \varphi_1 \wedge \varphi_2 \quad \text{iff } (\mathcal{I},\mathcal{M},\mathcal{N}) \models_6 \varphi_i,\ i = 1, 2,\\
&(\mathcal{I},\mathcal{M},\mathcal{N}) \models_6 \exists x : \varphi \quad \text{iff } (\mathcal{I},\mathcal{M},\mathcal{N}) \models_6 \varphi\,[ax]\\
&\qquad \text{for some } \alpha \in \Delta,\\
&(\mathcal{I},\mathcal{M},\mathcal{N}) \models_6 \varphi_1 \supset \varphi_2 \quad \text{iff } (\mathcal{I},\mathcal{M},\mathcal{N}) \not\models_6 \varphi_1\\
&\qquad \text{or } (\mathcal{I},\mathcal{M},\mathcal{N}) \models_6 \varphi_2,\\
&(\mathcal{I},\mathcal{M},\mathcal{N}) \models_6 \mathbf{K}\varphi \quad \text{iff } (\mathcal{J},\mathcal{M},\mathcal{N}) \models_6 \varphi\ \forall \mathcal{J} \in \mathcal{M},\\
&(\mathcal{I},\mathcal{M},\mathcal{N}) \models_6 \mathbf{not}\varphi \quad \text{iff } (\mathcal{J},\mathcal{M},\mathcal{N})(\varphi) \leq \mathbf{f}\\
&\qquad \text{for some } \mathcal{J} \in \mathcal{N}.
\end{aligned}
\tag{14}
$$

A suspicious MKNF interpretation $\mathcal{M}$ is a *suspi- cious S_5 model* of a given closed MKNF formula φ, written as $\mathcal{M} \models_6 \varphi$ if and only if $(\mathcal{I}, \mathcal{M}, \mathcal{M}) \models_6 \varphi$ for each $\mathcal{I} \in \mathcal{M}$.

To compute the suspicious S_5 models, we introduce a new fixpoint operator $\mathfrak{T}^s_{\mathcal{K}_G}$, which is little different from the operator $\mathfrak{T}_{\mathcal{K}_G}$ on the definition of $T_{\mathcal{K}_G}(\mathrm{S})$. We replace $T_{\mathcal{K}_G}(\mathrm{S})$ with $T^s_{\mathcal{K}_G}(\mathrm{S})$, which is defined as follows.

- If $\mathrm{OB}_{\mathcal{O},\mathrm{S}} \models_6 A_i, n+1 \leq i \leq m$ for some ground integrity constraint $\leftarrow \mathbf{K}A_1 \wedge \ldots \wedge \mathbf{K}A_m$ in $\mathcal{P}_G$, then $T^s_{\mathcal{K}_G}(\mathrm{S}) = \emptyset$.
- Otherwise, $T^s_{\mathcal{K}_G}(\mathrm{S}) = \{\mathrm{S} \cup \mathrm{R}_t \cup \mathrm{H} \mid$ for each ground MKNF rule $C_j \in \mathcal{P}_G \colon \mathrm{OB}_{\mathcal{O},\mathrm{S}} \models_6 A_i, n+1 \leq i \leq m, \mathrm{R}_t = \cup_{C_j}\{\mathbf{K}H'_t\}\ (1 \leq t \leq n),$ where $H'_t = H_t,$ if $\mathrm{OB}_{\mathcal{O},\mathrm{S}} \models_6 A_i,$ and $\mathrm{OB}_{\mathcal{O},\mathrm{S}} \not\models_6 \neg A_i$ for each $n+1 \leq i \leq m; H'_t = H^s_t,$ otherwise. $\mathrm{H} = \{\mathbf{K}\xi \in \mathrm{HA}(\mathcal{K}_G) \mid \mathrm{OB}_{\mathcal{O},\mathrm{S}} \models_6 \xi\}\}$.

Note that the only difference between operators $T_{\mathcal{K}_G}$ and $T^s_{\mathcal{K}_G}$ is replacing KH_i with $\mathbf{K}H^s_i$ when KH_i is derived by inconsistent information. However, this will not affect the final results, since $\mathcal{M} \models_6 \mathbf{K}H^s_i$ implies $\mathcal{M} \models_6 \mathbf{K}H_i$ for any suspicious MKNF interpretation $\mathcal{M}$. The superscript "*s*" in $\mathbf{K}H^s_i$ is just like a label of suspicious information.

Given a hybrid MKNF knowledge base $\mathcal{K}_G$ and its transformation $\mathcal{K}^*_G$ as shown in Definition 8, let P_h be an element of the set $\mathbb{Q} = \Phi(\min(\gamma(\mathfrak{T}^s_{\mathcal{K}^*_G} \uparrow \omega)))$ and $\mathcal{M} = \{\mathcal{I} \mid \mathcal{I} \models_6 \mathrm{OB}_{\mathcal{O},P_h}\}$. We call $\mathcal{M}$ the *suspicious MKNF model* of $\mathcal{K}_G$.

Theorem 17. *Let $\mathcal{K}_G$ be a hybrid MKNF knowledge base. If $\mathcal{M}$ is a suspicious MKNF model of $\mathcal{K}_G$, then it is a suspicious S_5 model of $\mathcal{K}_G$.*

Proof. Let $\mathcal{M}'$ be the corresponding MKNF interpretation, in which each literal H^s_i is replaced by H_i. It is easy to see that $\mathcal{M}'$ is a paraconsistent MKNF model of $\mathcal{K}_G$. Moreover, for each literal $A, \mathcal{M}' \models_4 \mathbf{K}A$ if and only if $\mathcal{M} \models_6 \mathbf{K}A,$ and $\mathcal{M}' \models_4 \mathbf{not}A$ if and only if $\mathcal{M} \models_6 \mathbf{not}A$. Thus $\mathcal{M}$ satisfies each MKNF rule in $\mathcal{K}_G$. Hence the result follows.

RELATED WORKS

Huang et al. [5] presented a four-valued paraconsistent semantics for hybrid MKNF knowledge bases, which resolved the inconsistency problem but was invalid to incoherency.

Michael Fink [7] proposed paraconsistent hybrid theory for handling paraconsistent and paracoherent information in a combination of DL and rules, which is based on here-and-there logic.

Sakama and Inoue [3] proposed a paraconsistent stable semantics for extended disjunctive programs. Moreover, they introduced suspicious stable models to distinguish facts affected by inconsistent information from others in a program. At last, in order to handle incoherency occurring in a program, they employed nine-valued lattice and presented semistable models, which is also used in [8] to cope with instability and also is the inspiration of our work on incoherency handling in hybrid MKNF knowledge bases.

CONCLUSION

In this paper we presented a semi-S_5 semantics for hybrid MKNF knowledge bases which is paraconsistent for incoherent knowledge bases. We showed that a semi-S_5 model can be computed via a fixpoint operator and is in fact a paraconsistent MKNF model when the knowledge base is incoherent. Furthermore, we applied six-valued lattice to hybrid MKNF knowledge bases and present a suspicious semantics to distinguish different trust level information.

Our future work can be directed towards several paths. First of all, a well-founded semantics of hybrid MKNF knowledge bases has better complexity properties than paraconsistent semantics, and paraconsistent approach could be carried over to this paradigm. Moreover, in the real world, there are some other problems, such as probabilistic uncertainty, that cannot be coped with by classical reasoners. Then it is necessary to extend probabilistic semantics to hybrid MKNF knowledge bases.

REFERENCES

1. P. Hitzler, M. Krötzsch, and S. Rudolph, *Foundations of Semantic Web Technologies*, Chapman & Hall/CRC, 2009.
2. F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. F. Patel- Schneider, Eds., *The Description Logic Handbook: Theory, Implementation, and Applications*, Cambridge University Press, Cambridge, UK, 2003.
3. C. Sakama and K. Inoue, "Paraconsistent stable semantics for extended disjunctive programs," *Journal of Logic and Computation*, vol. 5, no. 3, pp. 265–285, 1995.
4. T. Eiter, M. Fink, and J. Moura, "Paracoherent answer set programming," in *Proceedings of the 12th International Conference on Principles of Knowledge Representation and Reasoning (KR '10)*, pp. 486–496, May 2010.
5. S. Huang, Q. Li, and P. Hitzler, "Paraconsistent semantics for hybrid MKNF knowledge bases," in *Proceedings of the 5th International Conference on Web Reasoning and Rule Systems*, vol. 6902 of *Lecture Notes in Computer Science*, pp. 93–107, Springer, 2011.
6. B. Motik and R. Rosati, "A faithful integration of description logic with logic programming," in *Proceedings of the 20th International Joint Conference on Artificial Intellegence (IJCAI '07)*, pp. 477–482, AAAI Press, Hyderabad, India, Janaray 2007.
7. Michael Fink: Paraconsistent Hybrid Theories. KR , 2012.
8. T. Eiter, M. Fink, and J. Moura, "Paracoherent answer set programming," in *Proceedings of the 12th International Conference on Principles of Knowledge Representation and Reasoning (KR '10)*, pp. 486–496, May 2010.

Chapter

METRICS FOR MULTISET-THEORETIC SUBGRAPHS

8

Ray-Ming Chen

School of Mathematics and Statistics, Baise University, 21, Zhongshan No. 2 Road, Guangxi Province, China

ABSTRACT

We show how to define a plethora of metrics for graphs—either full graphs or subgraphs. The method mainly utilizes the minimal matching between any two multisets of positive real numbers by comparing the multiple edges with respect to their corresponding vertices. In the end of this article, we also demonstrate how to implement these defined metrics with the help of adjacency matrices. These metrics are easy to be manipulated in real applications and could be amended according to different situations. By our metrics, one should be able to compare the distances between graphs, trees, and networks, in particular those with fuzzy properties.

Citation: Ray-Ming Chen, “Metrics for Multiset-Theoretic Subgraphs”, Journal of Function Spaces, volume 2019, article ID 7630242, https://doi.org/10.1155/2019/7630242.

INTRODUCTION

In the real world, we face a lot of uncertain mathematical objects. Among them, some are easier to be formalized via graphs or tree structures or networks. Henceforth, the distances between such structures are vital as they provide deeper information between two different structures. In the article [1], we have shown how to define metrics for graphs with single-edged vertices. Single-edge graphs are graphs with at most one edge between any two vertices. Such graphical structures normally are deterministic. The basic idea is to, based on minimal matching concepts, define matched parts and mismatched parts of incoming or outgoing edges. However, due to the complexity of real applications, in particular some fuzzy or indecisive mathematical objects, the metrics we had defined in that paper are insufficient to cover the needs. Therefore, we put forward some novel metrics which could accommodate such complexity in this article. We would consider graphs with multiple edges between any two vertices. This mechanism could then be applied in modelling some indecisive objects. Our research is also partially motivated by some articles regarding fuzzy mathematical objects [2–5]. Furthermore, if one is interested in other variants of metrics for graphs, he could consult either [6] or [7].

MULTISETS

Let us introduce some defnitions and operations of multisets. Let $\mathbb{R}^+$ denote the set of all positive real numbers. Let $\mathbb{N}_0$ denote the set of all the natural numbers including 0. Let Γ denote the set of all the functions $\mathbb{R}^+ \longrightarrow \mathbb{N}_0$. Let D_f be the domain of a function f. Let $D_f^* = \{r \in \mathbb{R}^+ : f(r) \neq 0\}$be the nonzero domain of f. Defne $\Gamma^< = \{f \in \Gamma : |D_f^*| < \infty\}$. In this article, we name each element in $\Gamma^<$ a multiset. Let $f, g \in \Gamma^<$ be arbitrary multisets. We use the notation $f \leq g$ (i.e., f is a multisubset of g) to denote that for all $x \in \mathbb{R}^+$, $f(x) \leq g(x)$.

Definition 1 (empty multiset). We call the zero function in $\Gamma^<$ the empty multiset.

Definition 2 (equality). $f = g$ iff $f \leq g$ and $g \leq f$.

Definition 3 (intersection $\wedge$). The intersection of multiset f and g, denoted by the function $f \wedge g : \mathbb{R}^+ \longrightarrow \mathbb{N}_0$, is defined by $(f \wedge g)(\alpha) := \min\{f(\alpha), g(\alpha)\}$ for all $\alpha \in \mathbb{R}^+$.

Definition 4 (union $\vee$*).* The union of multiset f and g, denoted by the function $f \vee g : \mathbb{R}^+ \longrightarrow \mathbb{N}_0$, is defined to $(f \vee g)(\alpha) := \max\{f(\alpha), g(\alpha)\}$ for all $\alpha \in \mathbb{R}^+$.

Definition 5 (difference $\ominus$*).* Exclusion of multiset g from f, denoted by the function $f \ominus g : \mathbb{R}^+ \longrightarrow \mathbb{N}_0$, is defined by $(f \ominus g)(\alpha) := f(\alpha) - (f \wedge g)(\alpha)$ for all $\alpha \in \mathbb{R}^+$.

Note that each multiset f in $\Gamma^<$ could be uniquely represented by either a set of descending form as follows:

$$f^- = \left(\alpha_1^{f(\alpha_1)}, \alpha_2^{f(\alpha_2)}, \ldots, \alpha_n^{f(\alpha_n)}\right), \tag{1}$$

or in short $f^- = \alpha_1^{f(\alpha_1)}\alpha_2^{f(\alpha_2)}\cdots\alpha_n^{f(\alpha_n)}$, or by a set of ascending form as follows:

$$f^+ = \left(\alpha_n^{f(\alpha_n)}\alpha_{n-1}^{f(\alpha_{n-1})}, \ldots, \alpha_2^{f(\alpha_2)}\alpha_1^{f(\alpha_1)}\right), \tag{2}$$

or in short $f^+ = \alpha_n^{f(\alpha_n)}\alpha_{n-1}^{f(\alpha_{n-1})}\cdots\alpha_1^{f(\alpha_1)}$, where $\alpha_1 > \alpha_2 > \alpha_3 \ldots > \alpha_n > 0$ and $\alpha_1, \alpha_2, \ldots, \alpha_n \in D_f^*$ and $f(\alpha_v) > 0$ for all $1 \le v \le n$. Define the cardinality of a multiset f by $|f| = \sum_{t=1}^{n} f(\alpha_t)$.

Definition 6 (descending order). Define the p-th element in f by function OD as follows:

$$OD(p, f) := \begin{cases} \alpha_1 & \text{if } 1 \le p \le f(\alpha_1); \\ \alpha_j & \text{if } \sum_{l=1}^{j-1} f(\alpha_l) < p \le \sum_{l=1}^{j} f(\alpha_l) \text{ and } \left|D_f^*\right| \ge j \ge 2; \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

Definition 7 (ascending order). Define the p-th element in f by function OA as follows:

$$OA(p,f) := \begin{cases} \alpha_n & \text{if } 1 \leq p \leq f(\alpha_n); \\ \alpha_{n-j} & \text{if } \sum_{l=0}^{j-1} f(\alpha_{n-l}) < p \leq \sum_{l=0}^{j} f(\alpha_{n-l}) \text{ and } \left|D_f^*\right| \geq j \geq 1; \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

METRICS

Let V be a set of vertices and E be a set of directed edges. Let $\mathscr{MP}^{<}(\mathbb{R}^{+})$ denote the finite multipower set of $\mathbb{R}^{+}$. In this section, we show how to define metrics for labelled graphs and unlabelled graphs. The distance is mainly defined based on weights of the corresponding edges between the graphs.

Definition 8. We call $G = (V, E, W : E \longrightarrow \mathscr{MP}^{<}(\mathbb{R}^{+}))$ a multiset-theoretic graph if and only if

(1) For each $v \in V[(v,v) \in E]$,

(2) For all $a, b \in V$, every element in $W(a,b)$ is non-negative and $W(a,b) = \{0\}$ iff $a = b$,

where W is a multiset-valued weight function. Let SG denote the set of all the multiset-theoretic graphs. The main purpose of this article is to define some metrics for SG which is divided into two categories: labelled vertices and unlabelled vertices.

Definition 9. Let $\|W(a,b)\|$ denote the sum of all the elements in the multiset $W(a,b)$.

Metrics for Labelled Graphs and Subgraphs

In this section, we show how to define the distance between any two graphs (whose vertices are all named) that could be graphs with either compatible or incompatible vertices. In this subsection, we assume all the vertices are labelled. To begin with, we show how to define metrics for $\mathscr{MP}^{<}(\mathbb{R}^{+})$. These metrics will serve as the foundations for further construction of metrics. By the representations of multisets in descending and ascending forms as shown in (1) and (2), we have the following definitions. Based on the minimal matchings between any two multisets of positive real numbers [8], we derive the following metrics.

Definition 10 (descending metric). Define

$$dd(f,g) := \sum_{k=1}^{\max\{|f|,|g|\}} \left|OD(k,f) - OD(k,g)\right|. \tag{5}$$

Definition 11 (ascending metric). Define

$$da(f,g) := \sum_{k=1}^{\max\{|f|,|g|\}} \left|OA(k,f) - OA(k,g)\right|. \tag{6}$$

Definition 12 (halved metric). Define

$$dh(f,g) := \alpha \cdot dd(f,g) + \beta \cdot da(f,g), \tag{7}$$

where $\alpha, \beta \geq 0$ and $\alpha + \beta = 1$.

In this article, we assume $\alpha = \beta = 1/2$. Indeed α and β could be decided by some mechanisms. We omit this part.

Example 13. Suppose $f^- = 8^3 6^4 3^5 2^3$ and $g^- = 9^4 4^3 3^3 2^4 1^6$. Then $|f| = 15$ and $|g| = 20$. Hence $dd(f,g) = 3 \cdot |9-8| + 1 \cdot |9-6| + 3 \cdot |4-6| + 3 \cdot |3-3| + 2 \cdot |2-3| + 2 \cdot |2-2| + 1 \cdot |1-2| + 5 \cdot |1-0| = 20$. Suppose $f^+ = 2^3 3^5 6^4 8^3$ and $g^+ = 1^6 2^4 3^3 4^3 9^4$. Then $da(f,g) = 78$ and $dh(f,g)) = (1/2)(20 + 78) = 49$ In the following, instead of explicitly showing the representative forms of a multiset, the distances of the corresponding form are understood from their contexts; for example, $dd(\{8,3,2,8,9\},\{2,2,3,1,5,2\})$ is deemed as $dd(9^1 8^2 3^1 2^1, 5^1 3^1 2^3 1^1)$, while $da(\{8,3,2,8,9\},\{2,2,3,1,5,2\})$ is deemed as $da(2^1 3^1 8^2 9^1, 1^1 2^3 3^1 5^1)$

Lemma 14. dd, da *and dh are all metrics on* $\Gamma^<$.

Proof. They follow from the absolute triangle property. Let $f, g, h \in \Gamma^<$ be arbitrary. One could show the triangle property via the relation of cardinalities of f, g, and h. For a full proof, one could also consult the results in [8].

Now we need to develop much more complicated metrics via dd, da, and dh metrics. For each vertex, there are two approaches to define its adjacent vertices: inbound or outbound. Hence we have the following definitions. Define outbound set

$$Out(a,E) := \{b \in V : (a,b) \in E\} \tag{8}$$

and inbound set

$$In(b,E) := \{a \in V : (a,b) \in E\}. \tag{9}$$

Definition 15. Define $\|Out(a,E)\| = \sum_{b\in Out(a,E)} \|W(a,b)\|$ and $\|In(a,E)\| = \sum_{c\in In(a,E)} \|W(c,a)\|$.

Example 16. Assume $E = \{(a,a),(c,a),(a,d),(a,k),(b,a),(a,b),(u,v)\}$, $W(a,a) = \{0\}$, $W(c,a) = \{1,3,6\}$, $W(a,d) = \{2,4,2,4,2\}$, $W(a,k) = \{3,5,3,1\}$, $W(b,a) = \{1,2,3,1\}$, $W(a,b) = \{1,6\}$, $W(u,v) = \{2,6,8,8\}$. Then $Out(a,E) = \{a,d,k,b\}$ and $In(a,E) = \{c,b\}$; thus $\|Out(a,E)\| = \|W(a,a)\| + \|W(a,d)\| + \|W(a,k)\| + \|W(a,b)\| = 0 + 14 + 12 + 7 = 33$. Moreover, $\|In(a,E)\| = \|W(c,a)\| + \|W(b,a)\| = 10 + 7 = 17$.

Let $G_1 = (V_1,E_1,W_1), G_2 = (V_2,E_2,W_2) \in SG$ be arbitrary. We have the following abbreviations:

(i) $Out_{12}(a) = Out(a,E_1) - Out(a,E_2)$;

(ii) $Out_{21}(a) = Out(a,E_2) - Out(a,E_1)$;

(iii) $In_{12}(a) = In(a,E_1) - In(a,E_2)$;

(iv) $In_{21}(a) = In(a,E_2) - In(a,E_1)$.

In this article, we put forward two equivalent categories of distance functions for subgraphs as follows.

Definition 17 (outbound measure: descending).

$$\begin{aligned} d^d_{out}(G_1,G_2) := {} & \sum_{v\in V_1-V_2} \|Out(v,E_1)\| \\ & + \sum_{v\in V_2-V_1} \|Out(v,E_2)\| \\ & + \sum_{a\in V_1\cap V_2} \left[\sum_{c\in Out_{12}(a)} \|W(a,c)\| \right. \\ & + \sum_{c\in Out_{21}(a)} \|W(c,a)\| \\ & \left. + dd(Out(a,E_1), Out(a,E_2)) \right]. \end{aligned} \tag{10}$$

Definition 18 (inbound measure: descending).

$$d_{in}^{d}(G_1, G_2) := \sum_{v \in V_1 - V_2} \|In(v, E_1)\| + \sum_{v \in V_2 - V_1} \|In(v, E_2)\| + \sum_{a \in V_1 \cap V_2} \left[\sum_{c \in In_{12}(a)} \|W(a, c)\| + \sum_{c \in In_{21}(a)} \|W(c, a)\| + dd\left(In(a, E_1), In(a, E_2)\right) \right]. \tag{11}$$

Definition 19 (outbound measure: ascending).

$$d_{out}^{a}(G_1, G_2) := \sum_{v \in V_1 - V_2} \|Out(v, E_1)\| + \sum_{v \in V_2 - V_1} \|Out(v, E_2)\| + \sum_{a \in V_1 \cap V_2} \left[\sum_{c \in Out_{12}(a)} \|W(a, c)\| + \sum_{c \in Out_{21}(a)} \|W(c, a)\| + da\left(Out(a, E_1), Out(a, E_2)\right) \right]. \tag{12}$$

Definition 20 (inbound measure: ascending).

$$d_{in}^{a}(G_1, G_2) := \sum_{v \in V_1 - V_2} \|In(v, E_1)\| + \sum_{v \in V_2 - V_1} \|In(v, E_2)\| + \sum_{a \in V_1 \cap V_2} \left[\sum_{c \in In_{12}(a)} \|W(a, c)\| + \sum_{c \in In_{21}(a)} \|W(c, a)\| + da\left(In(a, E_1), In(a, E_2)\right) \right]. \tag{13}$$

Definition 21 (outbound measure: halved).

$$\begin{aligned}
d^h_{out}\left(G_1, G_2\right) := & \sum_{v \in V_1 - V_2} \left\| Out\left(v, E_1\right)\right\| \\
& + \sum_{v \in V_2 - V_1} \left\| Out\left(v, E_2\right)\right\| \\
& + \sum_{a \in V_1 \cap V_2} \left[\sum_{c \in Out_{12}(a)} \| W\left(a, c\right) \| \right. \\
& + \sum_{c \in Out_{21}(a)} \| W\left(c, a\right) \| \\
& \left. + dh\left(Out\left(a, E_1\right), Out\left(a, E_2\right)\right) \right].
\end{aligned} \tag{14}$$

Definition 22 (inbound measure: halved).

$$\begin{aligned}
d^h_{in}\left(G_1, G_2\right) := & \sum_{v \in V_1 - V_2} \left\| In\left(v, E_1\right)\right\| \\
& + \sum_{v \in V_2 - V_1} \left\| In\left(v, E_2\right)\right\| + \sum_{a \in V_1 \cap V_2} \left[\sum_{c \in In_{12}(a)} \| W\left(a, c\right) \| \right. \\
& \left. + \sum_{c \in In_{21}(a)} \| W\left(c, a\right) \| + dh\left(In\left(a, E_1\right), In\left(a, E_2\right)\right) \right].
\end{aligned} \tag{15}$$

Theorem 23. *(1)* $d^d_{out}, d^d_{in}, d^a_{out}, d^a_{in}, d^h_{out}, d^h_{in}$ *are all metrics;* (2) $d^d_{out} = d^d_{in}, d^a_{out} = d^a_{in}, d^h_{out} = d^h_{in}$

Proof. They all follow from the definitions, in particular the property of set difference and the facts that dd, da, and dh are all metrics. For a full proof, one could also consult the results in [1, 8].

Since $d^d_{out} = d^d_{in}, d^a_{out} = d^a_{in}, d^h_{out} = d^h_{in}$, we use d^d, d^a , and d^h to represent them, respectively, in the following.

Metrics for Unlabelled Graphs and Subgraphs

In this section, we show how to define the distance between any two graphs (whose vertices are all unnamed) that could be either compatible or incompatible graphs. Since all the vertices are unnamed, the distance could not be defined as the one in the labelled cases. Hence all the possibilities of the interactions between G_1 and G_2 (whose vertices are all unnamed) should be taken into consideration. However, we could fix one graph, say G_1, and permute G_2. Then one computes all the possible distances and chooses the optimal permutation of G_2, in the sense of minimal distance. Let $\rho_G(k)$ denote the k-th graph whose vertices are identical to G with k-permutation of the names for the vertices. ρ indeed is treated as a naming system. There are $|V_2|!$ ways of assigning the names to the unnamed set V_2. Based on metrics d^d, d^a, d^h, we define the unlabelled distances as follows.

Definition 24. $d^{d*}(G_1, G_2) := \min\{d^d(G_1, \rho_{G_2}(k)) : 1 \le k \le |V_2|!\}$.

Definition 25. $d^{a*}(G_1, G_2) := \min\{d^a(G_1, \rho_{G_2}(k)) : 1 \le k \le |V_2|!\}$.

Definition 26. $d^{h*}(G_1, G_2) := \min\{d^h(G_1, \rho_{G_2}(k)) : 1 \le k \le |V_2|!\}$.

Theorem 27. d^{d*}, d^{a*}, *and* d^{h*} *, are all metrics.*

Proof. They follow from the definitions. For a full proof, one could also consult the results in [1, 8].

COMPUTATIONS AND IMPLEMENTATIONS

In this section, we show how to implement all the above-mentioned metrics via adjacency matrices. Let V=$\{v_1, v_2, v_3, v_4, v_5\}$. Suppose $\mathbb{G}(V)$ is the set of all the graphs whose vertices are comprised of part or all of V. Let $G_1, G_2, G_3, G_4 \in \mathbb{G}(V)$ defined as follows.

$G_1 = (V_1, E_1, W_1), G_2 = (V_2, E_2, W_2), G_3 = (V_3, E_3, W_3),$
, and $G_4 = (V_4, E_4, W_4)$, where $V_1 = V_2 = V, V_3 = \{v_1, v_3, v_4\}$, and $V_4 = \{v_2, v_3, v_4\}$; $E_k = (e^k_{ij})$; and W_1, W_2, W_3, W_4 are defined as follows:

$$W_1 = \begin{bmatrix} \{0\} & \{2,2,5\} & \{7,1,9,1\} & \{4,5,4,5,9\} & \{1,2,1\} \\ \{2,4,1\} & \{0\} & \{1,2,1\} & \{5,5\} & \{6,2,8,9\} \\ \{2,1,2\} & \{5,2,7,6\} & \{0\} & \{7,4,9\} & \{2,5,5\} \\ \{6,7\} & \{6,8,1\} & \{8,9,5\} & \{0\} & \{2,4,2\} \\ \{5,5\} & \{6,6,1\} & \{1,9,1\} & \{1,5,5\} & \{0\} \end{bmatrix}$$

$$W_2 = \begin{bmatrix} \{0\} & \{1,2,3,6\} & \{7,7,9,2\} & \{4,5,5\} & \{6,2,5\} \\ \{4,4,5\} & \{0\} & \{6,7,2\} & \{1,6,5\} & \{\{6,7,7\}\} \\ \{1,1\} & \{6,6,6\} & \{0\} & \{6,4,8,8\} & \{1,2,2\} \\ \{6,1,1\} & \{2,2\} & \{1,5,5\} & \{0\} & \{8,4,4,8\} \\ \{1,2,5\} & \{8\} & \{8,5,2,2\} & \{5,5\} & \{0\} \end{bmatrix}$$

$$W_3 = \begin{bmatrix} \{0\} & \{4,1,6\} & \{1,9,9,4\} \\ \{4,5\} & \{0\} & \{6,6,9,1\} \\ \{1,4,4,5\}\ \} & \{8,1,6\} & \{0\} \end{bmatrix},$$

$$W_4 = \begin{bmatrix} \{0\} & \{8,6,6\} & \{6,9,1\} \\ \{4,8,5\} & \{0\} & \{6,6,6,7\} \\ \{4,8,2\} & \{7,7\} & \{0\} \end{bmatrix}. \tag{16}$$

Computations for Labelled Full Graphs

We call G_1 and G_2 full graphs. Based on the definitions regarding their components of distances, we have the following computations (which correspond to the original definitions, in matrix form): $[W_1, W_2]^d_{in} = [W_1, W_2]^d_{out} = \begin{bmatrix} 0 & 3 & 7 & 13 & 9 \\ 6 & 0 & 11 & 2 & 5 \\ 3 & 4 & 0 & 8 & 7 \\ 7 & 11 & 11 & 0 & 16 \\ 4 & 9 & 8 & 1 & 0 \end{bmatrix}$ and $d^d(G_1, G_2) = \|[W_1, W_2]\|^d_{in} = \|[W_1, W_2]\|^d_{out} = 145$.

$[W_1, W_2]^a_{in} = [W_1, W_2]^a_{out} = \begin{bmatrix} 0 & 9 & 7 & 15 & 9 \\ 6 & 0 & 11 & 10 & 15 \\ 3 & 12 & 0 & 10 & 7 \\ 17 & 13 & 11 & 0 & 16 \\ 12 & 19 & 14 & 9 & 0 \end{bmatrix}$ and $d^a(G_1, G_2) = \|[W_1, W_2]\|^a_{in} = \|[W_1, W_2]\|^a_{out} = 225$. Furthermore, $d^h(G_1, G_2) = 1/2 \cdot [145 + 225] = 185$.

.

Computations for Labelled Subgraphs

We call G_3 and G_4 subgraphs. To begin with, we show how to compute the distance between a full graph and a subgraph. Their interactive components of distances are illustrated in the following matrix form:

$[W_1, W_3]^d_{in} = [W_1, W_3]^d_{out} = \begin{bmatrix} 0 & 9 & 7 & 12 & 4 \\ 7 & 0 & 4 & 10 & 25 \\ 6 & 20 & 0 & 4 & 12 \\ 9 & 15 & 7 & 0 & 8 \\ 10 & 13 & 11 & 11 & 0 \end{bmatrix}$ and $d^d(G_1, G_3) = \|[W_1, W_3]\|^d_{in} = \|[W_1, W_3]\|^d_{out} = 204$.

$[W_1, W_3]^a_{in} = [W_1, W_3]^a_{out} = \begin{bmatrix} 0 & 9 & 13 & 20 & 4 \\ 7 & 0 & 4 & 10 & 25 \\ 8 & 20 & 0 & 16 & 12 \\ 17 & 15 & 7 & 0 & 8 \\ 10 & 13 & 11 & 11 & 0 \end{bmatrix}$ and $d^a(G_1, G_3) = \|[W_1, W_3]^a_{in}\| = \|[W_1, W_3]^a_{out}\| = 240$. Furthermore, $d^h(G_1, G_3) = 1/2 \cdot [204 + 240] = 222$. .

Now we show how to implement the computation of the distance between any two subgraphs via a matrix form and its norm as follows:

$[W_3, W_4]^d_{in} = [W_3, W_4]^d_{out} = \begin{bmatrix} 0 & 31 & 39 \\ 26 & 0 & 7 \\ 28 & 3 & 0 \end{bmatrix}$ and $d^d(G_3, G_4) = \|[W_3, W_4]^d_{in}\| = \|[W_3, W_4]^d_{out}\| = 134$; $[W_3, W_4]^a_{in} = [W_3, W_4]^a_{out} = \begin{bmatrix} 0 & 31 & 39 \\ 26 & 0 & 7 \\ 28 & 15 & 0 \end{bmatrix}$ and $d^a(G_3, G_4) = \|[W_3, W_4]^a_{in}\| = \|[W_3, W_4]^a_{out}\| = 146$. Furthermore, $d^h(G_3, G_4) = 1/2 \cdot [134 + 146] = 140$. .

Computations for Unlabelled Full Graphs

Suppose the vertices in V_1, V_2, V_3, V_4 are all unnamed. To compute the distance between unnamed G_1 and G_2, according to the definition, we need to pick the smallest distances between G_1 and $\rho_{G_2}(k)$ for $1 \leq k \leq |V_2|!$. To implement this, we fix the adjacency matrix of G_1 and permutes the adjacency matrix of G_2 and compute all the respective distances and then choose the least one and its resulting permutation. Through computation, we have the following result:

By setting $\rho_{G_2}(k)$ as follows: $\rho_{G_2}(v_1) = v_1, \rho_{G_2}(v_2) = v_4, \rho_{G_2}(v_3) = v_2, \rho_{G_2}(v_4) = v_3, \rho_{G_2}(v_5) = v_5$ for some unique k, one has $d^{d*}(G_1, G_2) = d^d(G_1, \rho_{G_2}(k)) = 119$.

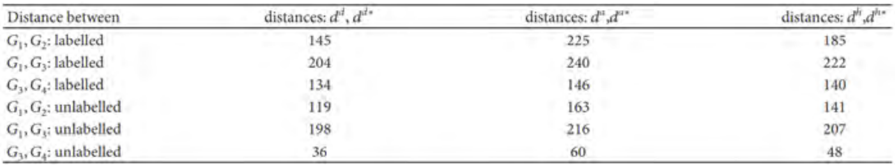
Similarly, by setting $\rho_{G_2}(k)$ as follows: $\rho_{G_2}(v_1) = v_5, \rho_{G_2}(v_2) = v_3, \rho_{G_2}(v_3) = v_1, \rho_{G_2}(v_4) = v_2, \rho_{G_2}(v_5) = v_4$ for some unique k, one has $d^{a*}(G_1, G_2) = d^a(G_1, \rho_{G_2}(k)) = 163$. Furthermore, $d^{h*} = 1/2 \times [119 + 163] = 141$.

Computations for Unlabelled Subgraphs

As for the $d^d(G_1, G_3)$, after our computations, the optimal corresponding subgraph in G$_1$ is the truncated one with vertex lying in $\{v_1, v_3, v_4\}$; i.e., d^{d*} is $\begin{bmatrix} \{0\} & \{7,1,9,1\} & \{4,5,4,5,9\} \\ \{2,1,2\} & \{0\} & \{7,4,9\} \\ \{6,7\} & \{8,9,5\} & \{0\} \end{bmatrix}$ the optimal subgraph with respect to . Then $d^{d*}(G_1, G_3) = 198$. Similarly, the optimal subgraph with respect to d^{a*} is $\begin{bmatrix} \{0\} & \{5,5\} & \{6,2,8,9\} \\ \{6,8,1\} & \{0\} & \{2,4,2\} \\ \{6,6,1\} & \{1,5,5\} & \{0\} \end{bmatrix}$. Then $d^{d*}(G_1, G_3) = 216$. Furthermore, $d^{h*} = 1/2 \times [198 + 216] = 207$. By setting $\rho_{G_4}(k)$ as follows: $\rho_{G_2}(v_1) = v_3, \rho_{G_2}(v_2) = v_1, \rho_{G_2}(v_3) = v_2$ for some unique k, one has $d^{d*}(G_3, G_4) = d^d(G_3, \rho_{G_4}(k)) = 36$. By setting $\rho_{G_4}(k)$ as follows: $\rho_{G_2}(v_1) = v_1, \rho_{G_2}(v_2) = v_2, \rho_{G_2}(v_3) = v_3$ for some unique k, one has $d^{a*}(G_3, G_4) = d^a(G_3, \rho_{G_4}(k)) = 60$.

To sum up all the results regarding different metrics and graphs, we have Table 1.

Table 1. Implementations of all the metrics.



Distance between	distances: d^d, d^{d*}	distances: d^a,d^{a*}	distances: d^h,d^{h*}
G_1, G_2: labelled	145	225	185
G_1, G_3: labelled	204	240	222
G_3, G_4: labelled	134	146	140
G_1, G_2: unlabelled	119	163	141
G_1, G_3: unlabelled	198	216	207
G_3, G_4: unlabelled	36	60	48

There are several observations worth mentioning:

(1) $d^{d*} \geq d^{h*} \geq d^{d*}$;

(2) the distance between a full graph and a subgraph is higher than either the distances for full graphs or the subgraphs;

(3) the distance between unlabelled graphs is less than or equal to the one between labelled ones.

All these results agree with our theoretical definitions and derivations.

REAL WORLD APPLICATION

In this section, we demonstrate how to make a decision via our derived metrics when facing some uncertain situation in the real. Suppose country

B's strategic deployment of air planes depends on country A's attack force. The degree of attack force ranges from 0 to 100, in which 0 indicates that there is no loss in the combat while 100 indicates the opponent's airbase is completely wiped out. Suppose A and B both have fives airbases in other countries C1, C2, C3, C4, and C5. Suppose A has 6 types of air planes A1, A2, A3, A4, A5, and A6. The number of each type of planes installed across different countries is listed in Table 2. Their respective attack forces are 30, 45, 55, 76, 88, and 97. Suppose B's observation of A's planes and air flight of his planes from one base to other bases is recorded in Tables 3 and 4. The potential flights from one base to other bases for A are listed in Table 2. This table could be directly converted into an adjacency matrix (named PAF), in which "none" is replaced by 0 and contents in each C_{ij} is rewritten in the forms of sets. Suppose B has 7 types of air planes: B1, B2, B3, B4, B5, B6, and B7. Their respective attack forces are 28, 41, 46, 52, 61, 70, 86. Suppose B owns $e_1 = 6, e_2 = 4, e_3 = 10, e_4 = 7, e_5 = 10, e_6 = 22$, and $e_7 = 11$ planes for each corresponding type. Now the problem for B is how he should send his air planes to counterbalance his opponent. Based on the metrics in this article, we could make a decision toward such uncertain situation. Let a_{ij}^k, b_{ij}^k denote the numbers of air flight of A's and B's air planes k from airbase C_i to C_j. Define

$$\mathbb{B} = \left\{ \tilde{B} : \tilde{B} = \left(\tilde{B}_{i,j}\right)_{i,j=1}^{5},\ \tilde{B}_{ij} = \left\{b_{i,j}^{k}\right\}_{k=1}^{7},\ \sum_{i,j=1}^{5} b_{i,j}^{k} = e_k,\ 1 \le k \le 7 \right\} \tag{17}$$

The optimal decision for B to counterbalance A is

$$\operatorname{argmin}\left\{d^{v}\left(PAF, \tilde{B}\right) : \tilde{B} \in \mathbb{B}\right\}, \tag{18}$$

where $v \in \{d, a, h\}$. Their individual solutions could be obtained via integer programming. Here we omit the final execution. If there is inconsistency between the choices of v, one could resort to subjective judgment or assigning weights between d^d, d^a and d^h to reach a final decision.

Table 2. Country A's installed air force.

	C1	C2	C3	C4	C5	subtotal
A1	2	0	1	1	3	7
A2	1	4	2	2	2	11
A3	2	3	3	2	0	10
A4	2	1	0	4	1	8
A5	1	1	0	2	0	4
A6	5	1	1	0	3	10
subtotal	13	10	7	11	9	50

Table 3. Country A's potential air flight.

	C1	C2	C3	C4	C5
C1	0	A1,A3	A2, A4,A6	A3,A6	A1,A5,A6
C2	None	0	A2,A3,A5,A6	A2	A3,A5
C3	A2,A3	A1,A3,A6	0	A2	None
C4	A1,A2,A3,A4	A2	A4	0	A3,A5
C5	A1,A6	None	A2,A4,A6	A6	0

Table 4. Country A's potential attack force.

	C1	C2	C3	C4	C5
C1	0	30,55	45, 76,76,97	55,97,97,97	30,88,97
C2	None	0	45,55,55,76,97	45,45,45	55,88
C3	45,55,55	30,55,97	0	45	None
C4	30,45,55,76	45	76,76,76	0	55,88,88
C5	30,30,30,97	None	45,45,76,97	97	0

CONCLUSION

In this article, we have shown how to define distances between graphs over either a set of labelled or unlabelled vertices via a plethora of metrics for graphs. We also give computational approaches to implement the computation of these metrics via the operations on adjacency matrices. This implementation gives an efficient and fast computation of the distance between any two such graphs. We also demonstrate how to apply these metrics in uncertain decision-making. Indeed, these metrics could be further applied in measuring the distance between real networks or tree-like structures.

ACKNOWLEDGMENTS

The work is supported by the Natural Science Foundation of Fujian Province of China (Grant no. 2017J01566).

REFERENCES

1. R. Chen, "Metrics for Single-Edged Graphs over a Fixed Set of Vertices," *Mathematical and Computational Applications*, vol. 23, no. 4, p. 66, 2018.
2. Z. Pawlak, *Rough Sets: Theoretical Aspects of Reasoning about Data*, Kluwer Academic, Dordrecht, Netherlands, 1991.
3. C. Xu, "Improvement of the distance between intuitionistic fuzzy sets and its applications," *Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology*, vol. 33, no. 3, pp. 1563–1575, 2017.
4. L. A. Zadeh, "Fuzzy sets," *Information and Computation*, vol. 8, pp. 338–353, 1965.
5. X. C. Liu, "Entropy, distance measure and similarity measure of fuzzy sets and their relations," *Fuzzy Sets and Systems*, vol. 52, no. 3, pp. 305–318, 1992.
6. M. Sarwar and M. Akram, "An algorithm for computing certain metrics in intuitionistic fuzzy graphs," *Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology*, vol. 30, no. 4, pp. 2405–2416, 2016.
7. M. Akram and N. Waseem, "Certain metrics in m-polar fuzzy graphs," *New Mathematics and Natural Computation*, vol. 12, no. 2, pp. 135–155, 2016.
8. R. Chen, "A Metric for Finite Power Multisets of Positive Real Numbers Based on Minimal Matching," *Axioms*, vol. 7, no. 4, p. 94, 2018.

Section 3: Recursive Functions and Data Types (Binary Trees)

Chapter

9

BINARY TREE'S RECURSION TRAVERSAL ALGORITHM AND ITS IMPROVEMENT

Hua Li

Department of Information Engineering, Hangzhou Polytechnic College, Hangzhou, China.

ABSTRACT

Binary tree is a very important data structure in computer science. Some major properties are discussed. Both recursive and non-recursive traversal methods of binary tree are discussed in detail. Some improvements in programming are proposed.

Keywords: Binary Tree, Traversal, Stack

Citation: Li, H. (2016), "Binary Tree's Recursion Traversal Algorithm and Its Improvement". Journal of Computer and Communications, 4, 42-47. doi: 10.4236/jcc.2016.47006.

INTRODUCTION

Binary tree is a very important data structure in which each node has at most two children, which are referred to as the left child and the right child. In computing, binary trees are seldom used solely for their structure. Much more typical is to define a labeling function on the nodes, which associates some value to each node. Binary trees labelled this way are used to implement binary search trees and binary heaps, and are used for efficient searching and sorting. The designation of non-root nodes as left or right child presents matters in some of these applications, even when there is only one child, and it is particularly significant in binary search trees [1] . In mathematics, what is termed binary tree can vary significantly from author to author. Some use the definition commonly used in computer science, but others define it as every non-leaf having exactly two children and don't necessarily order (as left/right) the children either [2] . The basic structure of the binary tree can be summarized as follows [3] .

- Degree of node: The number of children of a node is denoted as the degree of the node.
- Height of the tree: A tree's maximum number of level is denoted as tree height (or depth).
- The i level of binary tree (i ≥ 1) has up to 2^(i − 1) nodes.
- Binary tree of depth k has at most 2^k − 1 nodes (k ≥ 1).
- To any binary tree, if the number of leaf node is a and the number of nodes of degree 2 is b, then a = b + 1.
- The depth of complete binary tree which have n nodes is (log (2^n)) + 1.
- For complete binary tree with n nodes and nodes hierarchically from top to bottom, from left to right are encoded, then to any node a (1 ≤ a ≤ n) has if a = 1, then node a is the root of binary tree and have no parent, if a > 1,then its parent is a/2 (Rounded down). If 2a > n, then node i have no child node, else its left child node is 2a. If 2a + 1 > n, then a have no right child node, else its right child node is 2a + 1.
- If the binary tree which degree is n has 2^n − 1 nodes, then it is called a full binary tree. Full binary tree is also called complete binary tree.
- For complete binary tree, the number of node which is 1 degree is only possible to 1 or 0.

- For any tree, the total number of nodes = the sum of each node number + 1.

WHY USE A BINARY TREE TRAVERSAL AND ITS PRACTICAL APPLICATION

In binary tree, we often need to find the binary tree node that has some certain characteristics, or need to find all the nodes and process them. For example, based on digital image disorder binary tree traversal—A digital image scrambling method based on binary tree traversal, and discussed the periodic scrambling method and inverse transform. The method is simple and easy to operate, and suitable for images of any sizes. And it has good scrambling effect and great scrambling cycle. Under certain attacks, scrambled image can recover the original image. To some extent, it can meet the digital image encryption and hidden robustness requirements, and using binary tree traversal to expand the convex outer surface of the polyhedron (It can help some production construction). All of these require a binary tree traversal. However, the binary tree is a nonlinear structure, and each node may have two trees. So, we need to find rules that can make all nodes of a binary tree are arranged on a linear queue. So binary tree traversal of each node is in accordance with a path to access binary tree, and each note can be visited only once. Thus, the binary tree node is accessed sequentially formed by a linear sequence, whose result is that each node on the binary tree can be accessed more easily [4] .

BINARY TREE'S RECURSIVE TRAVERSAL ALGORITHM AND DESCRIPTION

Since the tree traversal rule is recursive, recursive traversal of a binary tree is very popular and convenient. Thus, according to the child-first traversal of a binary tree rules, there are three recursive traversal orders:

- Preorder: access root node, traverse the left subtree, traverse the right subtree
- Inorder: traverse the left subtree, access root node, traverse the right subtree
- Postorder: traverse the left subtree, traverse the right subtree, access root node

It can be summed up as some rules. First, preorder traversal of the first root node is the root node, while postorder traversal of the last root node is the root node.

Second, the last root node of preorder traversal is the rightmost child node of the right subtree, the last node of inorder traversal is the most right node of the root node right subtree. Third, leftmost root node inorder traversal first node to the root of the left subtree, postorder traversal is the first node as a left subtree the left child node.

From the above rules, we can draw the following inferences. The whole tree sort can be derived through the preorder traversal and the postorder traversal. Inorder traversal and postorder traversal can determine a binary tree. Preorder traversal and postorder traversal cannot determine a binary tree by themselves.

Let’s write a first binary tree’s Preorder traversal, Inorder traversal and Postorder traversal

```
    Public class BinaryTree implements BinaryTTree {
    Public BinaryNode root;
    Public BinaryTree(){this.root=null;}
    Public Boolean isEmpty(){return this.root==null;}
    }
Public void preOrder(){ // Preorder traversal
PreOrder(root);// Call the recursive method to preorder traversal
}
Public void preorder(BinaryNode p){
if(p!=null)
{
System.out.print(p.data.toString()+” ”);//access to root node
preOrder(p.left);// According to preorder traversal traverse left subtree，then recursive call
preorder(p.right);// According to preorder traversal traverse right subtree，then recursive call
}
Public void inOrder(){//inorder traversal
inOrder(root);
}
Public void inOrder(BinaryNode p)
{
If(p!=null)
{
inOrder(p.left);
System.out.print(p.data.toString()+” ”);
inOrder(p.right);
}
}
Public void postOrder(){//postorder traversal
postOrder(root);
}
Public void postOrder(BinaryNode p)
{
If(p!=null)
{
postOrder(p.left);
postOrder(p.right);
System.out.print(p.data.toString()+” ”);
}
}
```

The above algorithm is based on the definition of the root node p to determine the entire recursive method, The root node p will be refined in each recursive And then find a child node p, and child node is priority. If there is child node, then continue to search until there has no child node, we can output the nodes which have searched before in order.

ANOTHER ALGORITHM OF BINARY TREE TRAVERSAL ALGORITHM—NON-RECURSIVE CALLS ALGORITHM

The binary trees Preorder, Inorder and Postorder all belong to recursive algorithm. When the chain store of binary tree structure is given, the programming language with the recursive function can easily achieve the above algorithm. But recursive algorithm must have parameters. And it should distinguish multiple processing ways through different practical parameters. The method described above is regarding the node p as an argument. When p is a pointer which points to a different node, it means different trees. Accordingly, using different ways to call p is a matter of different traversing.

So the binary tree's non-traversal algorithm needs to build a stack to store traversal. Its algorithm is described as follows: Setting an empty stack; Node p from the binary tree root node, when p is not empty or not empty stack, do the following cycle, and finish the binary tree until the stack is empty.

- If p is not empty, showing just arrived p junction, put p junction stack, enter p left subtree.
- If p is empty, but the stack is not empty, and finish the route, we need to return to find another path. The node returns just after the last point, as long as the stack find a node of the p-point he could enter the right subtree.

Thus, we can launch a non-recursive algorithm Binary Tree

1) Non-recursive of Preorder Traversal's Implement

In the following algorithm, binary tree stored by binary linked list, Create an array stack [Pointer] in order to achieve Stack, top in stack is used to indicate the current location of the stack.

```
void inOrder（BiTree bt）
{/* Non-recursive preorder binary tree */
BiTree stack[Point],t;
int top;
if (bt==NULL) return;
top=0;
t=bt;
```

```
while(!(t==NULL&&top==0))
{ while(t!=NULL)
{ Visite(t.data); /* Data field access node */
if (top
{ stack[top]=t;
top++;
}
else { printf(“Stack Overflow”)；
return；
}
t=t.leftchild； /* Pointer to the left child of p */
}
if (top<=0) return; /* stack empty, then over*/
else{ top--;
t=stack[top]; /* Pop the top element from the stack */
Visite(t.data); /* access node data field */
t=t.rightchild; /* Pointer to the right child node p */
}
}
}
```

2) Non-recursive of Inorder Traversal’s Implement

The non-recursive of inorder traversal’s come true, simply preorder traversal non-recursive algorithm in the Visite (t.data) moved to between t = stack [top] and t = t.rightchild.

3) Non-recursive of Postorder Traversal’s Implement

In the following algorithm, array stack [Pointer] is used to achieve the overall structure of the stack. When the pointer variable p points to the current node to be processed, and node t is used to indicate the position of the current stack initial value of −1, sign integer variable amount can confirm that whether the node p has other children.

```
void Tree(Tree bt)
```

```
/* Non-recursive of Postorder traversal bt*/
{ stacktype stack[Point];
Tree p;
int t,sign;
if (bt==NULL) return;
top=-1 /* The default value of -1 means no stack location element within the stack */
p=bt;
while (!(p==NULL && t==-1))
{ if (p!=NULL) /* The first node into the stack */
{ t++;
stack[t].link=t; /*into stack*/
stack[t].flag=1;
p=p.lchild; /* Get the node left child node*/
}
else { p=stack[t].link;
sign=stack[t].flag;
t--;
if (sign==1) /*if exist right child node*/
{top++;
stack[t].link=p;
stack[t].flag=2; /* Marking the second time out of the stack */
p=p.rightchild;
}
else { Visite(p.data); /* If not, the direct access to the node data field values */
p=NULL;
}
}
}
}
```

The nature of the entire non-recursive algorithm is through the establishment of a stack to store each node to traverse down the node element, and it can sequentially output according to the characteristics of the stack. In a recursive algorithm by means of a recursive loop we will find each of the nodes is in the whole recursive loop, but recursion can be removed at any time to view through putting the element r into the stack one by one, Although this way is more cumbersome than the recursive algorithm, it reduces more computing time and system resources. And it is easier to see the whole nature which traverse through the program algorithm.

IMPROVEMENT OF NON-RECURSIVE ALGORITHM

According to the above non-recursive algorithm, it can be seen that before preorder traversal and time complexity is O (n), then preorder compared to O (n^2), algorithm is relatively complicated. So what can be improved so that the complexity and the time when the same preorder it? Let's first look postorder traversal non-recursive algorithm description, in process of postorder traversal non-recursive algorithm, to ensure the left and right child nodes are traversed, and left nodes must be traversed before the right in order to traverse the root node, we usually use the same array and stack on a non-recursive algorithm as described above, but they are more cumbersome. The postorder traversal of binary Tree determines its complexity of non-recursive algorithm design. If we blindly consider the issue from the "left and right root nodes" perspective, the created algorithm is undoubtedly very complex. If we change the angle of thoughts, the binary operation after preorder access order reversed, that is the "root points to the right-to-left node" that we are familiar with and very simple "preorder traversal". The difference is that the preorder traversal of a binary tree is the first access node and then left and right node access, and the "preorder traversal" here is the first visit and then visit the left node and right node. Here we don't discuss the left node and right node of order, but this idea gave us space to think and provides another way of thinking for the design of the non-recursive algorithm of binary tree traversal. So we can use preorder traversal to output result, then use the result to reverse output. Then the result becomes the result of postorder traversal. We just need to set up a non-recursive traversal more than a usual stack to store the previous preorder traversal non-recursive node obtained from before and then output it. Then there is the improved postorder traversal non- recursive algorithm program that I used to write with c language.

```
Status PostOrderTraverse(BiTree T,Status(* visit)(TElemType e))
{
/* According to preorder traversal traversal binary thinking first */
InitStack(C1);InitStack(C2);/* Initialize the stack */
If(T)Push(C1,T);
While (！StackEmpty(C1))
{
Pop(C1,p);
Push(C2,p);
If(p->lchild) Push(C1,p->lchild);// The left node stack
If(p->rchild) Push(C1,p->lchild);// The right node stack
}
/* Output traversal sequence */
While(! StackEmpty(C2))
{
Pop(C2,p);
Visit(p->data);//output
}
Return OK;
```

CONCLUSION

Compared to the non-recursive algorithm, binary tree's recursive traversal algorithm is more simple and clear. Through simple recursive call, we can write the binary tree traversal algorithm very quickly. However, recursive algorithm relies heavily on pointer node, which means the entire recursive algorithm will not continue to go on if the pointer is lost. Besides, a recursive algorithm is computationally intensive so it needs to call itself constantly to narrow the scope of the call, which leads to a low-efficient program. Thus, we think out the non- recursive calls. Although the non-recursive binary tree traversal is cumbersome procedure, it can better to see the whole process of traversal, such as how to use stack to storage node and then output it one by one. And the efficiency of program is high and the calculate amount is small. In this way, the non-recursive algorithm has been optimized and its complexity has been reduced. And the program operates faster than before. Then, whether to select non-recursive or recursive algorithm depends on their own program.

ACKNOWLEDGEMENTS

The work is supported by the project of Zhejiang province education department of China, Grant No.Y201326675.

REFERENCES

1. Makinson, D. (2009) Sets, Logic and Maths for Computing. Springer Science & Business Media, London, 199.
2. Hazewinkel, M., Ed. (2001) “Binary Tree”, Encyclopedia of Mathematics, Springer, London; Also in Print as Hazewinkel, M. (1997). Encyclopaedia of Mathematics. Supplement I. Springer Science & Business Media, London, 124. http://dx.doi.org/10.1007/978-94-015-1288-6
3. Ye, H.Y. (2015) Data Structure (Java Version). 2nd Edition, Electronic Industry Publishing House, Beijing.
4. Xu, F.S., Li, L.C. and Ma, X.R. (2006) Universal Binary Tree Traversal Non-Recursive Algorithm. Fujian Computer, No. 6, 41, 121.
5. Zhang, X.Q. (2012) Improvement of Binary Non-Recursive Algorithm. Journal of Jiamusi University (Natural Science Edition), 31, 926-928.

Chapter

10

GENERATING TREE-LISTS BY FUSING INDIVIDUAL TREE DETECTION AND NEAREST NEIGHBOR IMPUTATION USING AIRBORNE LIDAR DATA

Joonghoon Shin and Hailemariam Temesgen

Department of Forest Engineering, Resources and Management, Oregon State University, Peavy Hall, Corvallis, OR, USA.

ABSTRACT

Individual tree detection (ITD) and the area-based approach (ABA) are combined to generate tree-lists using airborne LiDAR data. ITD based on the Canopy Height Model (CHM) was applied for overstory trees, while

Citation: Shin, J. and Temesgen, H. (2018), “Generating Tree-Lists by Fusing Individual Tree Detection and Nearest Neighbor Imputation Using Airborne LiDAR Data”. Open Journal of Forestry, 8, 500-531. doi: 10.4236/ojf.2018.84032.

ABA based on nearest neighbor (NN) imputation was applied for understory trees. Our approach is intended to compensate for the weakness of LiDAR data and ITD in estimating understory trees, keeping the strength of ITD in estimating overstory trees in tree-level. We investigated the effects of three parameters on the performance of our proposed approach: smoothing of CHM, resolution of CHM, and height cutoff (a specific height that classifies trees into overstory and understory). There was no single combination of those parameters that produced the best performance for estimating stems per ha, mean tree height, basal area, diameter distribution and height distribution. The trees in the lowest LiDAR height class yielded the largest relative bias and relative root mean squared error. Although ITD and ABA showed limited explanatory powers to estimate stems per hectare and basal area, there could be improvements from methods such as using LiDAR data with higher density, applying better algorithms for ITD and decreasing distortion of the structure of LiDAR data. Automating the procedure of finding optimal combinations of those parameters is essential to expedite forest management decisions across forest landscapes using remote sensing data.

Keywords: Tree-List Generation, Individual Tree Detection, Nearest Neighbor Imputation, Parameter Sensitivity, Airborne LiDAR

INTRODUCTION

A tree-list provides detailed data foresters often desire for management and planning such as tree species, diameter at breast height (DBH), tree height (HT), basal area (BA) and stem volume. Field cruising has been commonly used to obtain such data. Field cruising is costly, however, and remote sensing data can be used as auxiliary information to improve the accuracy and precision of estimates in forest inventory.

Among various remote sensing techniques, airborne light detection and ranging (LiDAR) has been increasingly used in forestry applications during the last decade. LiDAR has performed well in estimating forest attributes such as biomass (Næsset & Gobakken, 2008) , diameter distribution (Gobakken & Næsset, 2004) , volume and BA (Lindberg & Hollaus, 2012) . Tree-lists have also been estimated by LiDAR (Lindberg, Holmgren, Olofsson, Wallerman, & Olsson, 2010, 2013) or aerial photographs (Temesgen, LeMay, Froese, & Marshall, 2003) .

In general, there are mainly two approaches using LiDAR data in forestry, the area-based approach (ABA) and the individual tree detection (ITD) approach (Vauhkonen, Maltamo, McRoberts, & Næsset, 2014) . ABA assumes that the vertical height distribution of laser point clouds is related to variables of interest in an area. A host of summary statistics derived from the point cloud are used to predict many forest inventory attributes. Information on the LiDAR point cloud is not fully utilized in ABA, i.e., most of the studies have focused on vertical height distribution in a sample plot and only a few studies using horizontal information obtained from the LiDAR point cloud. Pippuri, Kallio, Maltamo, Peltola, and Packalén (2012) found horizontal texture metrics from a canopy height model (CHM) could be used to predict the spatial pattern of trees, and horizontal landscape metrics from a CHM used to predict the need for first thinning.

In contrast, ITD identifies individual trees and provides estimates of forest attributes based on the identified individual trees. Although many variations exist, ITD commonly uses a rasterized CHM to segment individual trees with horizontal location of treetop and height across the CHM area. Thus, ITD has apparent advantages over ABA regarding utilization of horizontal information in LiDAR point clouds and can be more suitable for tree-level forest inventories than ABA. However, information on understory vegetation is likely to be missed when using ITD (Koch, Kattenborn, Straub, & Vauhkonen, 2014) . This is because rasterizing LiDAR point clouds into CHM means that there is a rounding effect of summarizing all the point clouds within a range of cells into one cell height value mainly focusing on higher point clouds making it difficult to detect or estimate understory vegetation. Additionally, it is well known that LiDAR has weaknesses for detecting or estimating understory vegetation regardless of the approach used because LiDAR data lack information on understory vegetation (lower proportion of point clouds in understory) (Takahashi, Yamamoto, Miyachi, Senda, & Tsuzuku, 2006) .

Many approaches have been proposed to overcome the limitations above. Maltamo, Eerikäinen, Pitkänen, Hyyppä, and Vehmas (2004) combined a theoretical probability distribution function with the tree height distribution estimated from ITD to detect small and suppressed trees. ITD first estimated the height distribution and the number of large trees. For small trees, two approaches have been used including—the complete Weibull distribution with the parameter prediction method and the left-truncated

Weibull distribution with estimation of parameters from the estimated height distribution by ITD. These approaches were tested for the estimation of the height distribution and the number of trees. DBHs for large and small trees were then predicted using the relationship between DBH and LiDAR metrics. Total timber volume and stem density were finally determined by summing the estimates from the two approaches for large and small trees. Lindberg et al. (2010) proposed a methodology to generate a tree-list combining a CHM-based ITD and ABA estimation. To better detect trees that are close to each other or small: 1) the number of trees per segment was estimated using a training dataset in which the number of field-measured trees for each tree crown segment was known, and 2) a candidate tree-list from the ITD was calibrated using the target distributions of HT and DBH estimated by a k-Nearest Neighbor (NN) approach. The combined approach improved the estimation of distributions for DBH and HT, and produced unbiased estimates of forest attributes. In addition to ITD based on CHM, Lindberg et al. (2013) utilized a 3D clustering method to model a tree crown using a priori information on the shape and proportions of tree crowns. The 3D clustering method identified more trees below the tallest canopy layer and with a DBH < 20 cm than ITD based on CHM. Hamraz, Contreras, and Zhang (2017) proposed the use of vertical stratification of point clouds and LiDAR data with high point cloud density (50 points/m^2), which would have more information on understory vegetation than the one with low density, to detect understory trees. The proposed approach improved detecting understory trees without affecting the overall quality of segmentation for overstory trees.

Many parameters affect the performance of tree segmentation by ITD; these can be classified into two parameters, biological and technical. For the biological parameter, Vauhkonen et al. (2012) claimed that the performance of ITD methods depends more on forest structure, stand density, and tree clustering than on detection techniques. For example, an estimated tree segment by ITD could have no, one, or several trees in it (Breidenbach, Næsset, Lien, Gobakken, & Solberg, 2010) , and trees in an understory under a dense upper canopy are hard to detect with LiDAR (Maltamo et al., 2004) . On the other hand, the methods for ITD were reported as the primary parameter affecting the performance of ITD by Kaartinen et al. (2012) . Substantial differences in the percentage of matched and missed trees, and commission error were found among the ITD methods. Also, the accuracy of determining tree location, tree height, and crown delineation changed according to the ITD methods. In contrast, pulse density showed less impact on ITD.

A typical ITD method consists of the following two steps: 1) generating a rasterized CHM with appropriate smoothing and resolution using normalized LiDAR point cloud data, and 2) tree segmentation using a segmentation technique on the rasterized CHM (finding local maxima as treetops and delineating tree crowns) (Yu, Hyyppä, Holopainen, & Vastaranta, 2010) . Therefore, the performance of ITD is affected by the parameters (smoothing and resolution for CHM, and the algorithm used for tree segmentation). In addition to these parameters, Wiggins (2017) reported that excluding trees below a specific height (minimum height cutoff) improved ITD's accuracy for overstory trees. Maltamo, Tokola, and Lehikoinen (2003) noted that a proper value of the truncation parameter of Weibull for DBH distribution, which can be considered the same as a height cutoff, should be further studied. According to McGaughey (2016) and Wiggins (2017) , there might be an optimal parameterization that balances the smoothing of the CHM, resolution of the CHM, and the height cutoff to best identify individual trees, although Koch et al. (2014) and McGaughey (2016) pointed out that the optimal parameterization can vary over large forest areas with diverse and complicated structure. To offset the variation of the optimal parameters, Koch, Heyder, and Weinacker (2006) proposed applying different intensities of smoothing according to HT. This method would prevent under- and over-representation of local HT maxima.

Other than ITD, detailed information on forest resources, such as a tree list or stand table, has been estimated by several methods that can be mainly classified into two categories: 1) diameter distribution modeling, and 2) imputation. In diameter distribution modeling, parameters of some theoretical distributions are estimated to describe the distribution of tree diameters. Three approaches commonly used are the parameter prediction method, parameter recovery method, and quantile prediction method (Temesgen et al., 2003) . Imputation methods directly substitute measured values from sample locations (references) for locations for which a prediction is desired (targets). The distance metric used to identify suitable references and the number of references used in a single imputation (k) are the key considerations to classify the imputation methods such as most similar neighbor, gradient nearest neighbor, or Random Forest NN (RF NN hereafter) (Eskelson et al., 2009) . Temesgen et al. (2003) used a set of proxy variables to represent a tree-list in NN imputations because there is no single variable to represent the tree-list. On the other hand, Strunk et al. (2017)

used plot identities as a response variable in NN imputations in evaluating NN strategies to impute a tree-list.

In our study, we combined ABA and ITD to estimate tree-list using LiDAR data inspired by the ideas from Maltamo et al. (2003) , Maltamo et al. (2004) and Wiggins (2017) . This was for overcoming the weakness of LiDAR data and the ITD method in identifying understory trees, and utilizing the strength of ITD over ABA. Maltamo et al. (2003) combined pattern recognition of single trees with the truncated Weibull distribution to estimate forest characteristics using digital video imagery. Trees were grouped into large (DBH > 17 cm) and small (DBH ≤ 17 cm) trees. The cutoff DBH value (17 cm) was the minimum size of trees that could be detected by the pattern recognition method. The value of 17 cm in DBH was used as a truncation parameter of the left-truncated Weibull. Pattern recognition was applied to large trees (DBH > 17 cm), and the diameter distribution modeling to small trees (DBH ≤ 17 cm), respectively. This idea was improved upon by Maltamo et al. (2004) , who combined ITD based on CHM for large trees and diameter distribution modeling for small trees. HT distribution was modeled using LiDAR metrics as auxiliary variables. Wiggins (2017) examined the effect of height cutoff on the accuracy of LiDAR data for estimating forest structure of taller trees and found that a 12 m height cutoff produced better results in estimating forest structure and spatial pattern.

For ITD, we used watershed segmentation (Vincent & Soille, 1991) for overstory trees (trees taller than a height cutoff) and ABA by NN (k = 1) imputation for understory trees (trees shorter than the height cutoff). While the performances of diameter distribution modeling depended on the results from large tree estimation by the single tree pattern recognition in Maltamo et al. (2003) or the ITD based on CHM in Maltamo et al. (2004) , in this study, we used ITD and ABA independently. They were only linked by a height cutoff when generating a complete tree-list. Whereas Lindberg et al. (2010) estimated a tree-list for all trees by an ITD method and calibrated it, our approach separated a forest stand into overstory and understory trees, then applied different methods to the overstory and understory trees, respectively. We examined the effects of the combination of the three parameters, smoothing of CHM, resolution of CHM and the height cutoff, as well as LiDAR height classification of field plots on estimating tree-lists via ITD. The explanatory power of our approach was also investigated. We evaluated the performance of generating tree-lists in terms of BA, mean HT, stems per hectare (SPH), and distributions of DBH and HT.

METHODS

Study Area

The study area (43.02435°N, 124.056°W) is located in southwestern Oregon with the extent of 647,951 hectares (Figure 1). The elevation of the area ranges approximately from 20 m to 1000 m above sea level in elevation. The range of slopes in the area is 0° to 89.97°. Douglas-fir (Pseudotsuga menziesii) is the dominant tree species in the study area, and other important species are western hemlock (Tsuga heterophylla), red alder (Alnus rubra), Oregon myrtle (Umbellularia californica), bigleaf maple (Acer macrophyllum), tanoak (Notholithocarpus densiflorus), western redcedar (Thuja plicata), and grand fir (Abies grandis).

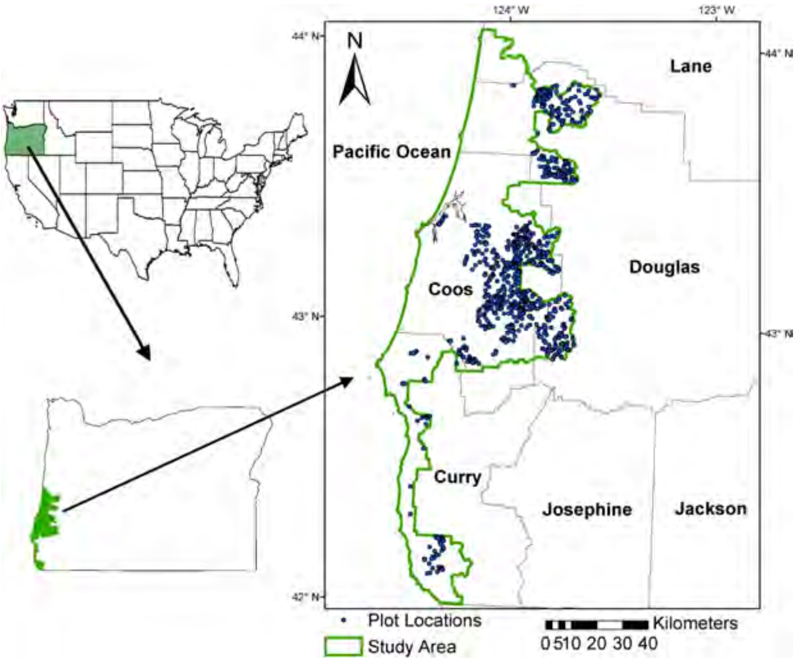


Figure 1. Map of study area and plots.

Airborne LiDAR

Airborne LiDAR data were collected between April 27th, 2008 and April 5th, 2009 using Leica ALS50 Phase II instrumentation. The collection was acquired as logistical constraints and weather allowed. The average pulse density (the average number of pulses returned from surfaces) was 8.10/m^2 for the study area. Table 1 shows the specifications for the LiDAR survey. Laser points with elevations above ground level lower than 1 m and higher than 91.44 m (300 feet) were excluded from the computation because they did not likely represent vegetation of interest (the maximum tree height measured in the field data was 88.4 m).

Table 1. LiDAR survey specifications.

Attribute	Description
Sensor	Leica ALS50 Phase II
Survey altitude	900 m (flown at 900 meters above ground level)
Pulse rate	>105 kHz (>105,000 laser pulse per second)
Pulse mode	Single
Mirror scan rate	52.5 Hz
Field of view	28° (±14° from nadir*)
Roll compensated	Up to 20°
Overlap	100% (50% side-lap)

*Point on the ground vertically beneath the laser sensor on the aircraft.

Field Data

Stratified sampling based on the LiDAR metrics (Hawbaker et al., 2009) was used for field data collection. Only the lands owned by the BLM or the Coquille Tribe in the study area were considered. Then, the non-forested areas were removed. Within this pre-selected area, a set of LiDAR grid metrics (22.86 m by 22.86 m) were calculated from the LiDAR point clouds. Using the principal component analysis, the 80th percentile and standard deviation of the LiDAR height were selected as describing best the variation in forest structure in the pre-selected area. Two thousand cells were randomly selected from the cells with the pre-selected area. Based on these random samples, the range of 80th percentile heights was subdivided into ten classes with a length of 6.10 m, and the range of standard deviations within each height class into three equal-width classes. The maximum height of the uppermost 80th percentile class was increased to 83.52 m to cover the values of the grid cells in the full dataset. A total of 30 bins (10 × 3) were created.

Every grid cell in the pre-defined area was assigned to the bins. Then, 30 primary and 20 alternate plot locations from each bin were randomly selected from the grid cells. 30 plot locations from each bin were measured by field crews from those 50 locations using the primary plot locations unless inconsistencies were found between the LiDAR measured structure and the actual state of the forest. Such inconsistencies were caused by disturbances, such as timber harvesting, fires, or wind throw that occurred after the LiDAR data acquisition. In that case, the next available alternate plot would replace the primary plot. Plot locations overlapping roads, and in tall shrub vegetation near the coast were discarded.

Field sampling was conducted between May 25, 2010 and May 10, 2011. Nested plots with two plot sizes (12.68 m and 5.09 m) were used to measure large (both live trees with a DBH larger than 14 cm and dead trees with a height of 3.05 m or greater and a DBH of 14 cm or greater) and small (only live trees with a height taller than 1.37 m and a DBH less than 14 cm) trees, respectively. Note that only the large tree data were used for this analysis. There was one missing plot, resulting in a total of 899 plots. Table 2 and Table 3 provide a plot-level and tree-level summary of the field measurements. The ten 80th percentile classes for the stratification sampling were used as LiDAR height classes in the current study (from “1” to “10” as height increases) to investigate the effect of LiDAR height classification of field plots on the performance of our proposed approach.

Table 2. Plot-level summary statistics of attributes from the field measurements.

Attribute	**Minimum**	**Maximum**	**Median**	**Mean**	**SD***
BA (m^2/ ha)	0.0	236.5	50.3	61.9	45.9
HT (m)	0.0	63.3	23.4	24.6	10.1
SPH (stems/ha)	0.0	1462.9	316.3	354.1	222.1

*Standard deviation.

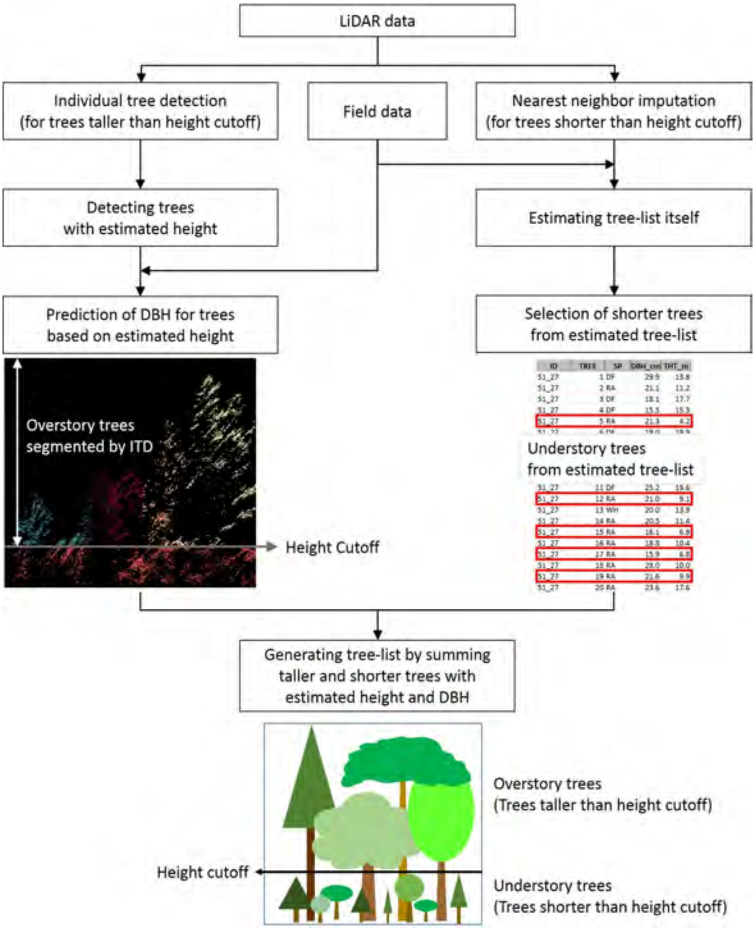
Table 3. Tree-level summary statistics from the field measurements.

Attribute	Minimum	Maximum	Median	Mean	SD*
DBH (cm)	14.0	266.2	26.9	37.4	28.7
HT (m)	0.3	88.4	19.51	23.5	14.2

*Standard deviation.

Generating Tree-Lists

The general steps of our approach are shown in Figure 2. Trees taller than a specified height (a height cutoff) were estimated by ITD using LiDAR data yielding the number and HT of the taller trees. DBHs for the taller trees were predicted based on the estimated HT using the relationship between DBH and HT from field data. For estimating the trees shorter than the height cutoff, tree-lists for target plots were first imputed with the tree-list from reference plots by RF NN imputation using both LiDAR and field data. Then, the shorter trees were selected from the imputed tree-lists. A complete tree-list can be generated by combining those estimated taller and shorter trees. The variables in the complete tree-list were the tree ID, HT, and DBH.


LiDAR data
Individual tree detection
(for trees taller than height cutoff)
Field data
Nearest neighbor imputation
(for trees shorter than height cutoff)
Detecting trees
with estimated height
Estimating tree-list itself
Prediction of DBH for trees
based on estimated height
Selection of shorter trees
from estimated tree-list
Overstory trees
segmented by ITD
Height Cutoff
Understory trees
from estimated tree-list
Generating tree-list by summing
taller and shorter trees with
estimated height and DBH
Overstory trees
(Trees taller than height cutoff)
Height cutoff
Understory trees
(Trees shorter than height cutoff)

ID	TREE	SP	DBH_cm	THT_m
51_27	1	DF	29.9	15.8
51_27	2	RA	21.1	11.2
51_27	3	DF	18.1	17.7
51_27	4	DF	15.5	15.3
51_27	5	RA	21.3	4.2
51_27	11	DF	25.2	19.6
51_27	12	RA	21.0	9.1
51_27	13	WH	20.0	13.9
51_27	14	RA	20.5	11.4
51_27	15	RA	18.1	6.9
51_27	16	RA	18.8	10.4
51_27	17	RA	15.9	6.8
51_27	18	RA	28.0	10.0
51_27	19	RA	21.6	9.9
51_27	20	RA	23.6	17.6

Figure 2. Flowchart of the approach.

Individual Tree Detection

ITD was implemented by the function “TreeSeg” in the FUSION software (McGaughey, 2016) with the argument “ht_threshold” to estimate the tree-list for large trees. This function applies a generalized watershed segmentation algorithm by Vincent and Soille (1991) to a CHM. It should be noted that over-segmentation, known as one of the disadvantages of the watershed algorithm, may be produced with noisy imagery (Romero-Zaliz & Reinoso-Gordo, 2018) . Conceptually, the CHM is inverted, so tree crowns appear as basins. Water fills the basins from local height minima in the CHM by the algorithm, and the basins fill and join with adjacent basins, then watershed edges are established (McGaughey, 2016) . This also can be explained at the pixel level on the CHM. In every CHM pixel above a height threshold, a path is placed by iteratively moving to the neighboring pixel with the largest height value until a local height maximum is reached. A tree crown segment is defined by cells that reach the same local height maximum (Lindberg & Holmgren, 2017) . The “ht_threshold” sets minimum height (height cutoff) for tree segmentation. Fractions of CHM below this height cutoff were excluded in the segmentation process. The other two parameters, the amount of smoothing and the resolution of the CHM, were applied in generating the CHM implemented by the function “CanopyModel” in FUSION. We set levels of those three parameters as follows: 1) 3 levels of smoothing of CHM—no smoothing, median filter using a 3 by 3 neighbor window and median filter using a 5 by 5 neighbor window, 2) 24 resolutions of CHM—0.2, 0.3, …, 2.4, and 2.5 m, 3) 9 percentile height cutoffs on the LiDAR height for each plot—10th, 20th, …, 90th. Because the range of HT is extensive, the LiDAR height percentiles were used as height cutoffs instead of absolute heights as in Wiggins (2017) .

After implementing ITD, we obtained a tree-list above a height cutoff including information on individual tree count, HT, a location of tree, and a number of CHM cells within a tree crown at a combination of smoothing, resolution of CHM, and height cutoff. To predict the DBHs of trees in the estimated tree-lists, an RF regression model for DBH was fitted with the HTs from the field data (16,200 trees). With this model, the DBHs of trees in the estimated tree-lists were predicted using the HTs of those trees. Then, those predicted DBHs were added to the estimated tree-lists. The model was fitted in R version 3.3.3 (R Core Team, 2017) using the R package “randomForest” (Liaw & Wiener, 2002) .

Nearest Neighbor Imputation

To estimate tree-lists for understory trees, we used RF NN imputation instead of diameter distribution modeling because there were many sample plots with multimodal or irregular shapes in diameter distribution and some plots had a small number of trees. NN imputation directly substitutes measured values from references for targets. The type of NN imputation is determined mainly by the distance metric and number of neighbors (k) (Eskelson et al., 2009) . The distance metric measures the similarity between target and reference observations, and the k indicates how many reference observations are used in a single imputation (prediction). Four distance metrics, Euclidean, Mahalanobis, most similar neighbor and RF (Breiman, 2001) , were tested. RF appeared the best for BA, SPH and error index (EI; will be defined in the following section), and Euclidean showed the best for HT (this result is not presented in this manuscript). Thus, we selected the RF algorithm as the distance metric and chose k = 1. RF builds multiple classification (or regression) trees, called forests, with bootstrap samples of training data, while selecting predictors randomly for the best split at each node in the trees. Distance in RF NN is computed as one minus the proportion of classification trees where a target observation is in the same terminal node as a reference observation (Crookston & Finley, 2008) . To estimate tree-lists by RF NN imputation, we imputed plot identities as in Strunk et al. (2017) .

To fit an NN model, it is necessary to define response and predictor variables. Predictor variables were derived from LiDAR point clouds at each filed plot location. It is not clear which a single response variable or multiple response variables should be used for estimating tree-lists because many attributes can be extracted from a tree-list. For example, Temesgen et al. (2003) used a set of 22 proxy variables to represent a tree-list. We considered several forest inventory attributes (basal area, stem volume, Lorey's height, quadratic mean diameter, stems per ha) simultaneously to select appropriate predictor variables for estimating tree-lists via RF NN imputation. "Best subsets" was used as a variable selection method producing the best three predictors for each forest inventory attribute.

From the best predictors for each forest inventory attribute, we obtained a total of 11 predictors after removing duplicates. The selected predictors are shown in Table 4. Like leave-one-out validation, the target plot was excluded from training data when modeling. Nine different tree-lists for each height cutoff were generated from the estimated tree-lists by subtracting trees above the corresponding height cutoff. The variable selection and imputation modeling were implemented in R version 3.3.3 (R Core Team, 2017) using R packages “yaImpute” (Crookston & Finley, 2008) and “randomForest” (Liaw & Wiener, 2002) .

Table 4. Selected predictor variables for RF NN imputation.

Metrics	**Min**	**Max**	**Mean**	**SD**	**Description**
sqrt_mean (m)	2.4	63.1	27.7	13.5	LiDAR height quadratic mean
CHM_SD (m)	1.1	30.1	10.2	6.2	Height standard deviation of rasterized CHM
Vol_3D (m^3)	768.9	30,258.3	12,461.5	6645.6	Volume of the region between rasterized CHM and ground
AShape.4 (m^3)	792.7	20,231.1	8627.4	3938.8	3D alpha shape with alpha value of 4
mode_30th (m)	1.0	53.7	12.8	11.9	LiDAR height mode from the point clouds less than LiDAR height 30th percentile
SD_30th (m)	0.1	17.6	4.9	3.7	LiDAR height standard deviation from the point clouds less than LiDAR height 30th percentile
sqrt_10 (m)	1.8	8.8	5.6	1.2	LiDAR height quadratic mean from the point clouds under 10 m
p.a.2 (%)	8.3	100.0	89.9	16.3	Percentage of first returns above height of 2 m
p.u.5 (%)	0.0	98.6	15.5	20.5	Percentage of first returns under height of 5 m
p.a.15 (%)	0.0	99.6	64.0	32.7	$\frac{\text{Number of total first returns above 15 m}}{\text{Number of total first returns above 2 m}} \times 100$
p.a.10th (%)	12.0	100.0	86.0	16.2	$\frac{\text{Number of total first returns above LiDAR height } 10^{th} \text{ percentile}}{\text{Number of total first returns above 2m}} \times 100$

Performance Measures

Bias and root mean squared error (RMSE) (Walther & Moore, 2005) for mean HT, BA and SPH were computed as follows:

$$\text{bias} = \frac{\sum_{i=1}^{n} (\hat{y}_i - y_i)}{n} \qquad (1)$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}\left(\hat{y}_i - y_i\right)^2}{n}} \qquad (2)$$

where $\hat{y}_i$ is the prediction at the ith plot, y_i is the field-measured value at the ith plot, and n is the number of total sample plots.

Large trees would produce greater uncertainty in estimation than small ones because the larger trees have greater values of HT, DBH, etc. To see the effect of several parameters on tree-list estimation free from the influence of greater value, relative bias (RBias) and relative RMSE (RRMSE) were also calculated for each LiDAR height class by the equations below:

$$RBias(\%) = \frac{\sum_{i=1}^{n}\left(\hat{y}_{ih} - y_{ih}\right)}{n_h} \times \frac{100}{\overline{y}_h} \qquad (3)$$

$$RRMSE(\%) = \sqrt{\sum_{i=1}^{n}\left(\hat{y}_{ih} - y_{ih}\right)^2} \times \frac{100}{\overline{y}_h} \qquad (4)$$

where $\hat{y}_{ih}$ is the prediction at the ith plot in the hth LiDAR height class, y_{ih} is the field-measured value at the ith plot in the hth LiDAR height class, $\overline{y}_h$ is the average of field-measured values at in the hth LiDAR height class, h is the number of LiDAR height classes, and n_h is the number of sample plots in the hth LiDAR height class.

The error index (EI) (Reynolds, Burk, & Huang, 1988) was used to evaluate the size distributions of DBH and HT, respectively. EI measures the proportions of absolute deviation between the predicted and field-measured number of trees to the total number of field-measured trees over the entire distribution. EI for a plot was computed as:

$$EI(\%) = \frac{\sum_{i=1}^{k}\left|n_{pi} - n_{oi}\right|}{N} \times 100 \qquad (5)$$

where n_{pi} and n_{oi} are the predicted and observed numbers of trees, respectively, in DBH or HT class i. k is the number of DBH or HT classes. N is the total number of field-measured trees. The bin widths for classifying DBH and HT were 10 cm and 5 m, respectively.

The coefficient of determination measures (R^2) the proportion of variance in a response variable that is explained by predictor variables. It shows that how well a model's predictions fit the observed values of the response variable, which means the actual explanatory power of the model. The R^2 is calculated as:

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (\hat{y}_i - y_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2} \qquad (6)$$

where $\hat{y}_i$ is the prediction at the ith plot, y_i is the field-measured value at the ith plot, $\bar{y}$ is the average of field-measured values of the total sample plots, and n is the number of total sample plots.

RESULTS

Effects of Smoothing, Resolution, and Height Cutoff on Tree-List Estimation

All the resolutions with pixel size less than 1 m produced too large of estimates of SPH and yielded unreasonable estimates of other attributes regardless of the amount of smoothing and the height cutoff. Hence, resolutions with pixel sizes less than 1 m were dropped from the analysis. The amount of smoothing in CHM had a relatively small effect on tree-list estimation compared to the other parameters. The smoothing generally decreased the variability of estimation among the resolutions at a given height cutoff or the height cutoffs at a given resolution. For this reason, we show the performance only from the smoothing of 3 by 3 neighbor window.

Most cases of the combinations of resolution and height cutoff resulted in the underestimation of SPH (Figure 3(a)). Unbiased SPH estimations were found around 1.1 m to 2.0 m in CHM resolution with the various height cutoffs. Generally, a higher cutoff had a smaller absolute bias compared with the absolute bias from a lower cutoff. In terms of precision, Figure 3(b) shows that a higher cutoff had a relatively consistent RMSE along with resolutions in CHM, which means that higher cutoffs were less affected by resolution for SPH estimation than lower cutoffs as also shown in Figure 3(a). The combinations of the finer resolutions (1.2 ~ 1.3 m) and the lower height cutoffs (p20 and p30) provided the lowest RMSEs. For overstory trees, the patterns of performance measures were similar to the patterns from the combined approach, but the best RMSEs were always found at height cutoff p90. For understory trees, bias and RMSE increased as height cutoff increased except for the bias at height cutoff p90.

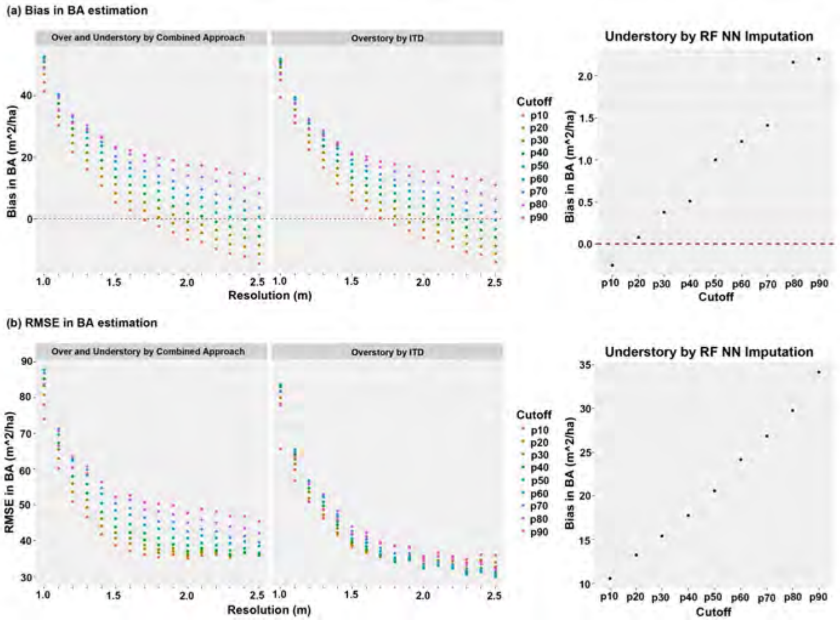


Figure 3. (a) Bias and (b) RMSE in SPH estimation: the left graph is for overstory and understory trees via the combined approach by resolution of CHM and height cutoff; the middle graph is for overstory trees via ITD by amount of smoothing, resolution of CHM and height cutoff; and the right graph is for understory trees via RF NN by height cutoff (the horizontal dashed line indicates unbiased estimates).

For BA estimation, bias decreased as resolution decreased as shown in Figure 4(a). Unbiased BA estimation was achieved for the combination of several cutoffs from p10 to p60 and resolutions with pixel sizes larger than 1.7 m. RMSE in BA estimation also decreased as resolution decreased (Figure 4(b)). Lower cutoffs yielded lower RMSE. The lowest RMSEs appeared for resolutions around 1.8 ~ 2.0 m. For overstory trees, the differences in RMSE between height cutoffs at a given resolution were smaller than the differences for the combined approach except for 1.0 m resolution. For understory trees, bias and RMSE increased as height cutoff increased, and all height cutoffs overestimated SPH.

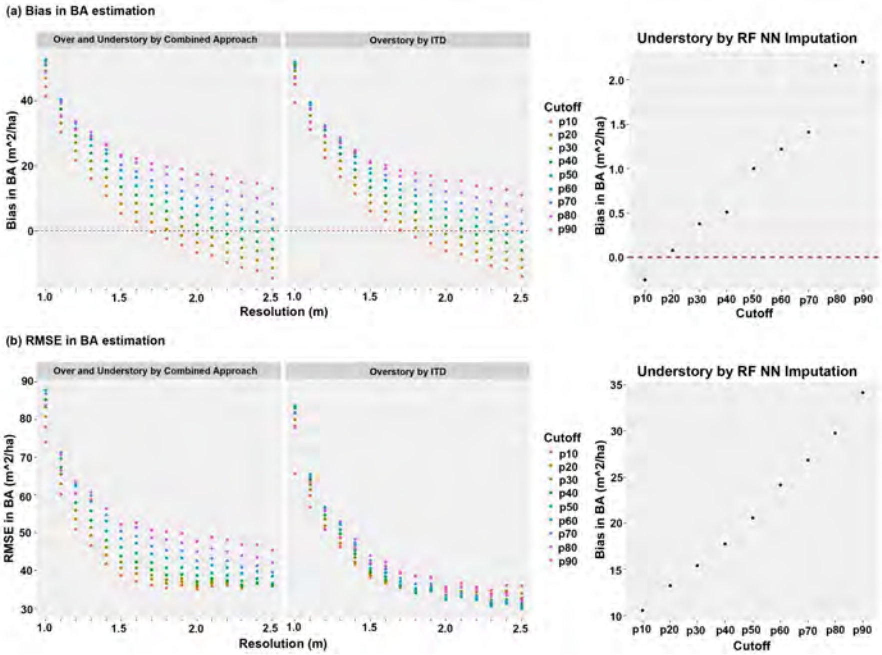


Figure 4. (a) Bias and (b) RMSE in BA estimation: the left graph is for overstory and understory trees via the combined approach by resolution of CHM and height cutoff; the middle graph is for overstory trees via ITD by amount of smoothing, resolution of CHM and height cutoff; and the right graph is for understory trees via RF NN by height cutoff (the horizontal dashed line indicates unbiased estimates).

HT estimation had better performance than the other attributes. The pattern for HT estimation was different from the other attributes. The best accuracy in HT estimation was found with the cutoff at p50 or p60 for any resolution. The poorest accuracy in HT estimation appeared only for the cutoff p10, which had a worse bias for HT estimation as resolution decreased. HT estimation became unbiased as resolution decreased except with cutoffs p10 and p20 (Figure 5(a)). Height cutoffs showing better RMSEs were p50 and p60 with the middle and higher resolutions, and p80 in the lower resolutions at any smoothing level. RMSE increased as resolution decreased especially for cutoffs p10, p20, and p30 (Figure 5(b)). Bias and RMSE of HT estimation for overstory trees only by ITD increased as the resolution decreased. For understory trees, Bias and RMSE for HT estimation also increased as height cutoff increased.

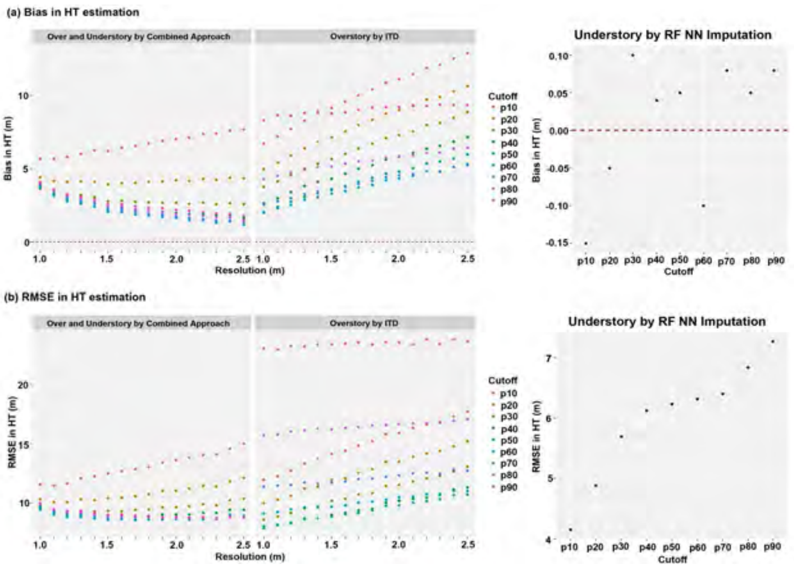


Figure 5. (a) Bias and (b) RMSE in HT estimation: the left graph is for overstory and understory trees via the combined approach by resolution of CHM and height cutoff; the middle graph is for overstory trees via ITD by amount of smoothing, resolution of CHM and height cutoff; and the right graph is for understory trees via RF NN by height cutoff (the horizontal dashed line indicates unbiased estimates).

For the lower resolutions, the lower cutoffs showed better DBH distribution estimation than the higher cutoffs, while it was the opposite with the higher resolutions (Figure 6(a)). This pattern was also observed in HT distribution estimation. The best DBH distribution was found with cutoffs p30 and p40 and lower resolutions while cutoff p90 had the best DBH distribution for the higher resolutions. The HT distribution estimation, in most cases, had the better result with the lower cutoffs than the higher cutoffs (Figure 6(b)). The cutoff p50 had the best performance in most cases, except p90 for 1 and 1.1 m resolutions, and p30 for 1.3 ~ 1.5 m resolutions. The resolutions with medium pixel sizes were better for estimating the HT distribution. For overstory trees, EI for DBH decreased as resolution decreased, and the lowest height cutoff p10 always yielded the best DBH distribution estimation at every resolution. DBH distribution estimation for overstory trees was poorer than for both overstory and understory trees. The best height cutoff for HT distribution of overstory trees estimation increased as resolution decreased. For understory trees, EIs for DBH and HT were reduced as height cutoff increased except for cutoff p80. Contrary to HT estimation for both overstory and understory trees by the combined approach, the best height cutoffs in the estimation of the HT distribution for overstory trees by ITD was for higher cutoffs from p60 to p80 except for resolutions higher than 1.4 m.

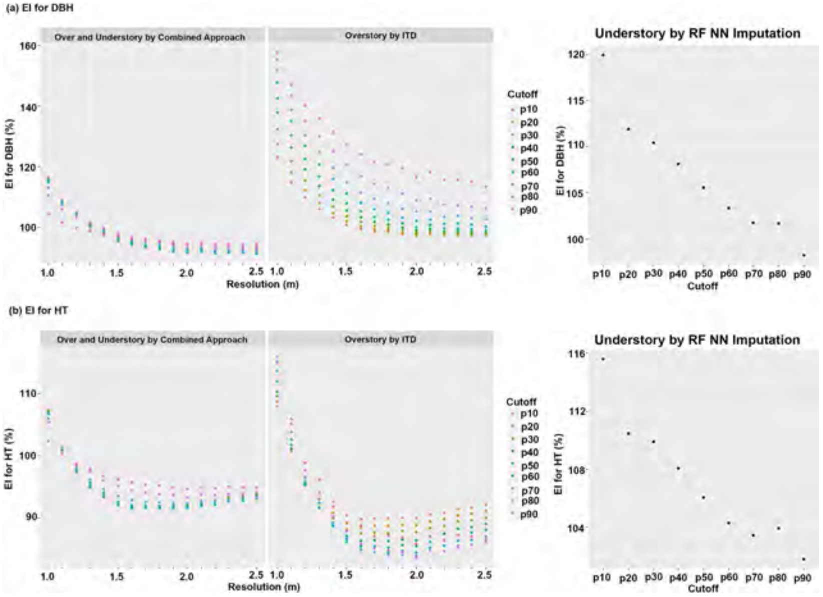


Figure 6. (a) EI for DBH and (b) EI for HT: the left graph is for overstory and understory trees via the combined approach by resolution of CHM and height cutoff; the middle graph is for overstory trees via ITD by amount of smoothing, resolution of CHM and height cutoff; and the right graph is for understory trees via RF NN by height cutoff.

Compared to the combined approach for all trees or the ITD for overstory trees, NN imputation produced much lower biases for understory trees' SPH, BA, and HT (Figures 3-5). The smallest biases for understory trees for SPH, BA and HT estimation were found at cutoffs p10, p20, and p40, respectively. The smallest RMSEs in the three attributes were observed only at cutoff p10.

Effects of Classification of Field Plots by LiDAR Height on Tree-List Estimation

The absolute and relative performance measures separated by LiDAR height class were calculated for each forest attribute estimated. The smallest group, class 1, showed distinct properties in those performances. For the absolute measures, such as bias and RMSE, lower LiDAR height classes, especially the lowest class, generally yielded comparable or better performances for BA and SPH than the higher classes. However, based on the relative measures, the lowest class had much poorer results. Similar patterns were found in EIs for DBH and HT as well. The effect of the amount of smoothing in CHM by LiDAR height class was relatively small. The performances by height cutoff in a given resolution were averaged for this section because it is better to show the general effect of height class on tree-list estimation performance. For SPH estimation (Figure 7), bias decreased as resolution decreased for every height class, but the resolutions showing unbiasedness varied among height classes. Lower height classes had larger variability in bias among resolutions than higher height class. Height class 1 had much larger RBias at higher resolutions than the other height classes. Larger RMSE occurred in height classes 1 through 6, and the largest RMSE was found in height class 3. RRMSE in height class 1 was largest at every resolution. Relatively larger RRMSEs at higher resolutions were observed in the taller height classes.

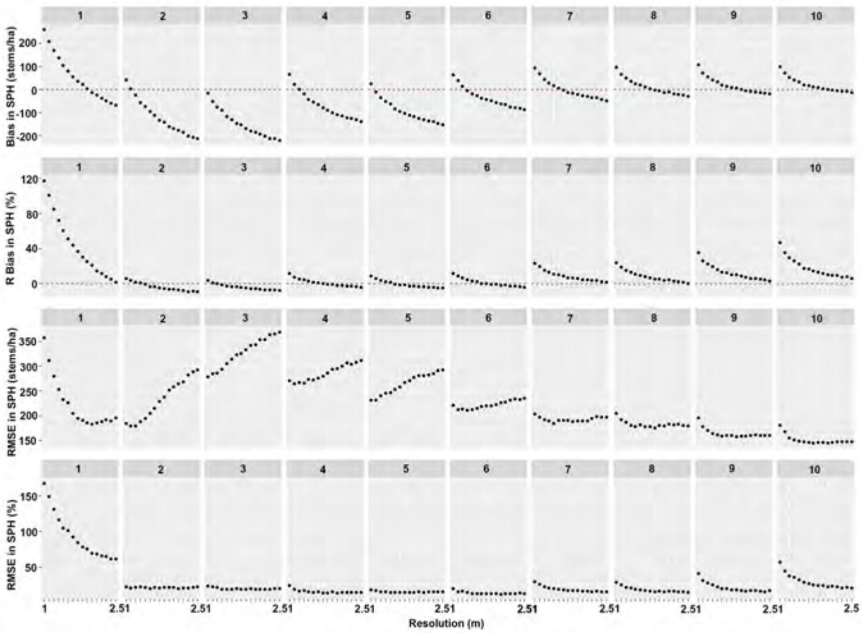


Figure 7. Bias, RBias, RMSE, and RRMSE for SPH estimation via the combined approach by LiDAR height class and resolution of CHM: the values of each performance by height cutoff in a given resolution are averaged.

In BA estimation (Figure 8), biases in the taller height classes were generally larger than biases in the shorter height classes. This pattern was similar for RBias except for height class 1. RBias in height class 1 was larger than the other height classes at resolutions less than or equal to 2.3 m. Lower height classes generally had smaller RMSE than higher height classes, but height class 1 had a much larger RRMSE than the other height classes. Figure 9 shows the performance measures for HT estimation by height class. The pattern of HT estimation among height classes was different from the pattern of SPH and BA estimation. Height class 1 had comparable or better performance in bias, RBias, and RMSE. The primary difference in bias and RBias between class 1 and the other classes was that class 1 mainly underestimated HT while the other classes overestimated. RRMSE for HT in height class 1 had slightly larger values than RRMSE from other height classes. Estimated distributions of DBH and HT for height class 1 were much poorer than the distributions for the other classes. Except class 1, lower height classes showed better performance in EIs for both DBH and HT than higher height classes. Lower resolution generally had lower EIs (Figure 10).

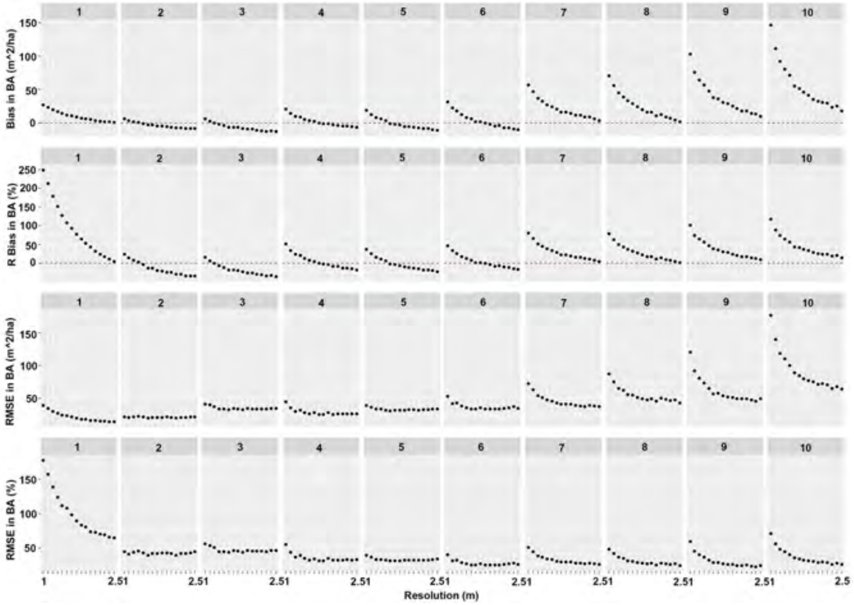


Figure 8. Bias, RBias, RMSE, and RRMSE for BA estimation via the combined approach by LiDAR height class and resolution of CHM: the values of each performance by height cutoff in a given resolution are averaged.

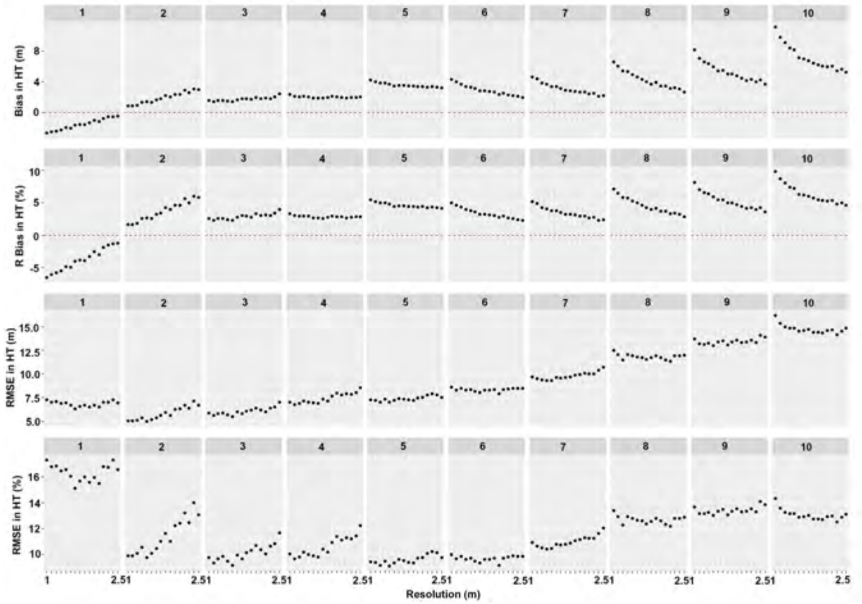


Figure 9. Bias, RBias, RMSE, and RRMSE for HT estimation via the combined approach by LiDAR height class and resolution of CHM: the values of each performance by height cutoff in a given resolution are averaged.

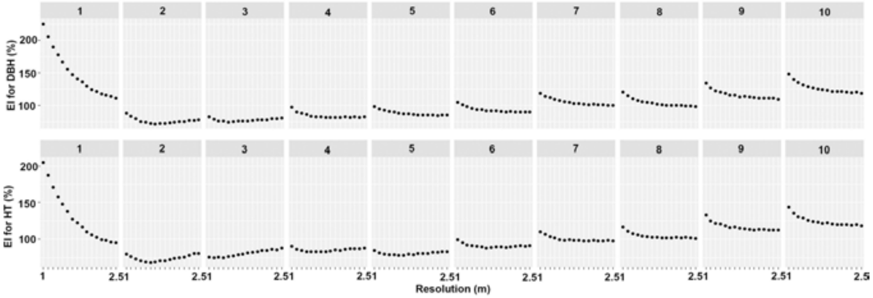


Figure 10. EIs for DBH and HT estimation via the combined approach by LiDAR height class and resolution of CHM: the values of each performance by height cutoff in a given resolution are averaged.

Explanatory Power of Individual Tree Detection for Overstory Trees and Random Forest Nearest Neighbor Imputation for Understory Trees

Tables 5-7 show R^2s for SPH, BA and HT estimation for trees over a given height cutoff (overstory trees) via ITD by resolution of CHM and height cutoff with smoothing using a 3 by 3 window. For SPH estimation (Table 5), the best R^2 was found at resolutions between 1.2 m and 1.7 m for each height cutoff. Height cutoff p90 yielded the largest R^2, 0.501, and the best R^2 decreased as the height cutoff decreased. The lowest height cutoff p10 had negative R^2 at all the resolutions. BA estimation by ITD showed poor explanatory power for overstory (Table 6). Most combinations of resolutions and height cutoffs had negative R^2s, and the best R^2 was 0.338 with the resolution 2.0 m and the height cutoff p10. Larger height cutoffs, from p70 to p90 provided negative R^2 at every resolution. In HT estimation (Table 7), the 1.0 m resolution yielded the best R^2 at every height cutoff except p90. The middle height cutoffs, p50 or p60, had better R^2 than the other height cutoffs. Inferior explanatory power was found at height cutoffs p10 and p90. The explanatory power for HT estimation generally decreased as resolution increased.

Table 5. Explanatory power (R^2) of SPH via ITD for trees taller than given height cutoffs with the smoothing of 3 by 3 neighbor window.

Height cutoff	Resolution (m)															
	1.0	**1.1**	**1.2**	**1.3**	**1.4**	**1.5**	**1.6**	**1.7**	**1.8**	**1.9**	**2.0**	**2.1**	**2.2**	**2.3**	**2.4**	**2.5**
p10	−0.100	−0.020	−0.009	−0.039	−0.104	−0.199	−0.267	−0.343	−0.418	−0.514	−0.566	−0.644	−0.710	−0.779	−0.832	−0.904
p20	−0.115	0.048	0.106	0.107	0.069	−0.008	−0.059	−0.124	−0.192	−0.280	−0.326	−0.399	−0.458	−0.523	−0.574	−0.641
p30	−0.134	0.084	0.170	0.196	0.180	0.122	0.086	0.029	−0.027	−0.110	−0.147	−0.216	−0.267	−0.332	−0380	−0.439
p40	−0.139	0.106	0.215	0.262	0.262	0.223	0.196	0.156	0.104	0.033	0.000	−0.062	−0.103	−0.167	−0.210	−0.261
p50	−0.144	0.124	0.253	0.311	0.319	0.297	0.285	0.252	0.207	0.142	0.113	0.054	0.018	−0.044	−0.081	−0.129
p60	−0.131	0.138	0.274	0.347	0.364	0.356	0.357	0.333	0.296	0.234	0.211	0.158	0.127	0.066	0.039	−0.011
p70	−0.057	0.198	0.321	0.386	0.409	0.403	0.418	0.400	0.369	0.319	0.301	0.252	0.227	0.171	0.148	0.101
p80	0.048	0.264	0.371	0.421	0.452	0.448	0.464	0.460	0.434	0.396	0.390	0.345	0.325	0.276	0.255	0.217
p90	0.236	0.376	0.431	0.464	0.497	0.480	0.500	0.501	0.483	0.454	0.455	0.423	0.417	0.372	0.353	0.330

Table 6. Explanatory power (R^2) of BA via ITD for trees taller than given height cutoffs with the smoothing of 3 by 3 neighbor window.

Height cutoff	**Resolution (m)**															
	1.0	**1.1**	**1.2**	**1.3**	**1.4**	**1.5**	**1.6**	**1.7**	**1.8**	**1.9**	**2.0**	**2.1**	**2.2**	**2.3**	**2.4**	**2.5**
p10	−2.421	−1.008	−0.443	−0.197	0.033	0.178	0.240	0.283	0.326	0.284	0.338	0.294	0.289	0.322	0.271	0.280
p20	−2.947	−1.320	−0.665	−0.378	−0.102	0.078	0.168	0.220	0.279	0.242	0.314	0.270	0.273	0.318	0.263	0.286
p30	−3.593	−1.717	−0.972	−0.628	−0.300	−0.065	0.049	0.118	0.197	0.166	0.265	0.217	0.232	0.287	0.229	0.266
p40	−4.306	−2.159	−1.303	−0.904	−0.516	−0.219	−0.088	0.011	0.106	0.084	0.203	0.161	0.182	0.256	0.197	0.253
p50	−5.072	−2.663	−1.706	−1.244	−0.794	−0.432	−0.273	−0.141	−0.035	−0.052	0.095	0.055	0.089	0.181	0.132	0.202
p60	−6.058	−3.323	−2.242	−1.740	−1.201	−0.752	−0.561	−0.387	−0.253	−0.257	−0.068	−0.108	−0.051	0.060	0.011	0.098
p70	−6.510	−3.724	−2.598	−2.104	−1.525	−1.037	−0.832	−0.616	−0.488	−0.488	−0.254	−0.295	−0.233	−0.118	−0.133	−0.040
p80	−7.105	−4.372	−3.169	−2.752	−2.116	−1.574	−1.377	−1.092	−0.990	−0.928	−0.641	−0.707	−0.615	−0.479	−0.501	−0.354
p90	−6.662	−4.746	−3.638	−3.239	−2.737	−2.125	−2.020	−1.748	−1.648	−1.613	−1.263	−1.386	−1.245	−1.045	−1.084	−0.872

Table 7. Explanatory power (R^2) of HT via ITD for trees taller than given height cutoffs with the smoothing of 3 by 3 neighbor window.

Height cutoff	**Resolution (m)**															
	1.0	**1.1**	**1.2**	**1.3**	**1.4**	**1.5**	**1.6**	**1.7**	**1.8**	**1.9**	**2.0**	**2.1**	**2.2**	**2.3**	**2.4**	**2.5**
p10	0.073	0.022	−0.052	−0.156	−0.223	−0.298	−0.347	−0.427	−0.504	−0.621	−0.639	−0.733	−0.794	−0.844	−0.952	−1.040
p20	0.514	0.481	0.447	0.385	0.364	0.300	0.283	0.231	0.192	0.113	0.110	0.054	0.013	−0.014	−0.070	−0.131
p30	0.700	0.681	0.661	0.629	0.619	0.578	0.567	0.536	0.523	0.469	0.463	0.434	0.407	0.389	0.349	0.307
p40	0.784	0.770	0.760	0.738	0.736	0.718	0.709	0.695	0.685	0.650	0.639	0.625	0.620	0.597	0.569	0.554
p50	0.804	0.797	0.788	0.772	0.771	0.762	0.751	0.744	0.733	0.710	0.708	0.694	0.686	0.676	0.655	0.652
P60	0.775	0.768	0.756	0.748	0.751	0.745	0.738	0.728	0.723	0.713	0.702	0.699	0.687	0.691	0.667	0.671
p70	0.676	0.673	0.670	0.659	0.663	0.658	0.652	0.641	0.637	0.628	0.621	0.611	0.608	0.613	0.592	0.598
p80	0.453	0.450	0.432	0.424	0.414	0.419	0.411	0.405	0.393	0.392	0.387	0.382	0.383	0.379	0.367	0.357
p90	0.028	0.033	0.010	0.014	−0.001	−0.001	−0.008	−0.015	−0.001	−0.021	−0.017	−0.003	−0.039	−0.006	−0.038	−0.024

Table 8 shows the explanatory power of RF NN imputation for trees under a given height cutoff (understory trees). For HT estimation, the R^2s were around 0.5. However, the R^2s for BA and SPH estimation were much poorer than the R^2s for HT estimation or even had negative values. For understory trees for each forest inventory attribute, the scatter plots of observed vs. predicted via RF NN imputation did not show any anomaly. The lower height cutoff we used, the more observations with zero values we had. The prediction results for those observations with zero values were inferior for every height cutoff.

Table 8. Performance measures of RF NN imputation by inventory attributes for trees shorter than given height cutoffs.

Height cutoff	SD SPH *	Bias$_{SPH}$	RMSE$_{SPH}$	R SPH 2	SD BA *	Bias$_{BA}$	RMSE$_{BA}$	R BA 2	SD HT *	Bias$_{HT}$	RMSE$_{HT}$	R HT 2
p10	89.20	0.73	87.99	0.03	9.13	−0.25	10.56	−0.34	6.09	−0.15	4.15	0.54
p20	115.32	1.72	110.63	0.08	11.35	0.08	13.26	−0.37	7.46	−0.05	4.88	0.57
p30	133.73	5.01	129.52	0.06	13.77	0.38	15.41	−0.25	8.11	0.10	5.69	0.51
p40	146.14	6.07	145.21	0.01	15.95	0.51	17.79	−0.25	8.58	0.04	6.12	0.49
p50	156.35	7.83	158.25	−0.03	18.42	1.00	20.59	−0.25	8.89	0.05	6.23	0.51
p60	166.25	8.42	168.72	−0.03	22.11	1.22	24.12	−0.19	9.05	−0.10	6.32	0.51
p70	177.05	9.04	179.29	−0.03	25.18	1.41	26.86	−0.14	9.20	0.08	6.40	0.52
p80	187.56	10.80	193.97	−0.07	29.47	2.16	29.78	−0.02	9.24	0.05	6.84	0.45
p90	197.84	8.78	204.64	−0.07	35.78	2.20	34.17	0.09	9.55	0.08	7.27	0.42

*Standard deviation of field-measured inventory attribute under given height cutoffs.

DISCUSSION

No single combination of smoothing, resolution and height cutoff was found to produce the best results for all performance measures (Table 9). Koch et al. (2014) and McGaughey (2016) also reported similar findings. Similarly, ITD's performance varied depending on the algorithm used to delineate trees in the CHM (Kaartinen et al., 2012) . Differences in performance between the lowest LiDAR height class and the other classes were found based on both absolute and relative performance measures. Kaartinen et al. (2012) reported that the HT class did not generally impact the accuracy of HT estimation, but greater uncertainty was observed for ITD methods capable of finding small trees. According to Hopkinson et al. (2005) , vegetation classes with short height, such as low shrub and aquatic vegetation, yielded the largest relative errors in canopy height estimation, whereas tall vegetation classes showed the largest absolute errors. The low level of penetration of LiDAR returns into the sub-canopy surface might be an essential reason for the high relative bias for low shrub and aquatic vegetation. For aquatic vegetation, it was also believed that the weak laser backscatter from the saturated ground caused the high relative bias. These results were very similar to ours although the smallest height class in our research almost exclusively consisted of trees.

Table 9. Best performance for each assessment by estimation method.

Method	**Target**	**Bias$_{SPH}$**	**RMSE$_{SPH}$**	**Bias$_{BA}$**	**RMSE$_{BA}$**	**Bias$_{HT}$**	**RMSE$_{HT}$**	**EI$_{DBH}$**	**EI$_{HT}$**
Combined	All	0.3079	212.2541	0.0233	35.0225	1.0967	8.4800	91.0333	91.2693
		3/2.1/p90*	No/1.3/p30	5/2.0/p30	3/2.0/p10	No/2.3/p50	No/2.0/p50	No/2.3/p40	No/1.9/p50
ITD	Overstory	0.5497	89.4945	0.0189	29.7892	1.7553	7.7367	96.6311	83.4631
		5/1.2/p50*	3/1.7/p90	No/1.8/p20	No/2.4/p60	No/1.0/p60	No/1.0/p40	No/2.3/p10	No/1.9/p70
NN	Understory	0.7256	87.9857	0.0829	10.5628	0.0420	4.1457	98.2339	101.8002
		p10†	p10	p20	p10	p40	p10	p90	p90
NN	All	−0.3958	217.6176	0.1438	36.2711	−0.4132	8.3843	91.6049	96.4568

*The first argument indicates the amount of smoothing, the second resolution in CHM, and the third percentile height cutoff for the combined method. †This represents percentile height cutoff.

As we reported above, resolutions with pixel sizes less than 1.0 m were dropped in the analysis because it yielded unreasonably large SPH estimations.

Pouliot, King, Bell, and Pitt (2002) claimed that in high-resolution imagery, tree detection and crown delineation became more complicated. This is because high-resolution imagery can display very detailed objects such as branches causing tree crowns to deviate from the conic shape. Thus, more tree crowns could be estimated at higher image resolutions. Conversely, in low-resolution imagery, it is more challenging to identify crown boundaries because they become less distinct. Another reason for our large SPH estimation might be data pits, which are height irregularities in a CHM. The function ‘CanopyModel’ in FUSION used for generating CHMs in our study fills pixels without LiDAR point clouds using an eight-way search and a distance-weighted average (McGaughey, 2016) . However, it might be difficult to avoid irregularities in height on a CHM if laser pulses used for our LiDAR data acquisition penetrated deeply into tree crowns causing large height variations within individual tree crowns (Persson, Holmgren, & Soderman, 2002) . Image smoothing with various filters using mean, median, or Gaussian approaches have been applied to reduce data pits (Persson et al., 2002; Yu, Hyyppä, Vastaranta, Holopainen, & Viitala, 2011) . In our study, the smoothing did not work well at resolutions with pixel sizes larger than 1.0 m although the smoothing using a 5 by 5 window showed smaller SPH estimation than no smoothing and the smoothing with a 3 by 3 window. A pit-free CHM proposed by Khosravipour, Skidmore, Isenburg, Wang, and Hussin (2014) was found to improve the accuracy of tree detection based on either high or low-density LiDAR data; however, this approach could help solve our large SPH estimation at the finer resolutions.

The ratio of average crown diameter to image pixel size was proposed as a guide to determine an optimal image resolution for tree detection and crown delineation using digital camera imagery (Pouliot et al., 2002) . With a small crown diameter to pixel size ratio, it is hard to have distinct crown boundaries in an image, resulting in under-segmentation. However, a large crown diameter to pixel size ratio might cause high variability within a crown in an image resulting in over-segmentation. Although our data from field surveys do not have information on crown diameter, there might be significant variations in the tree crowns considering the diversity of forest stands in our study area. This might be one of the reasons why the high CHM resolutions overestimated SPH in our study. Barnes et al. (2017) found that no single CHM resolution produced the best performance of ITD for both healthy and diseased larch trees, claiming that not only the tree crown size but also the maximum tree height governed an optimal size of CHM resolution. The performance of ITD with high-resolution CHMs (0.15 m) was best for plots with low maximum height (<20 m), and the performance with low-resolution CHMs (0.5 m) was best for plots with high maximum height (>30 m).

LiDAR point cloud density might be related to the optimal CHM resolution as with the tree crown diameter. With LiDAR data of high point cloud density, high CHM resolution could yield high within-crown variations on a CHM. Inversely, with LiDAR data of low point cloud density, low CHM resolution could produce less distinct crown boundaries making it difficult to identify tree crowns. The high CHM resolutions should have yielded good performance in that the LiDAR data used for this study had low point cloud density. However, the high diversity of forest stands in the study area might add more within-crown variations. Even though an optimal resolution of CHM was set based on the crown diameter to CHM resolution ratio, it should be noted that the performance of ITD was still affected by LiDAR point cloud density for trees with small DBH (<20 cm) as reported in Khosravipour et al. (2014) .

The results of large SPH estimation are quite different from previous studies. Stere?czak, Będkowski, and Weinacker (2008) reported that the

0.25 and 0.5 m resolutions in CHMs were better than the 1.0 m resolution for estimating SPH through individual tree delineation based on a similar method to Heurich and Weinacker (2004) . It was found that the number of detected trees decreased as the resolution of CHM decreased (Stere?czak et al., 2008) , and this was also observed in our work, excluding height cutoffs p10 and p20. Smreček et al. (2018) showed very similar results to ours for SPH estimation based on ITD. At the highest resolution (0.5 m), the number of trees identified was hugely overestimated; the number of trees identified decreased as the CHM resolution decreased from 0.5 m to 2.0 m, as was the case in our study. The optimal resolutions for tree identification were 1.0 and 1.5 m depending on the sample plot. Smreček et al. (2018) claimed that this was because the CHM with 0.5 m resolution was too detailed. We observed many estimated trees from ITD with extreme small areas compared to their estimated heights. Those trees should have been removed from the estimated tree-list using an appropriate criterion. With this filtering process, overestimation at high resolutions would be decreased.

Most combinations of parameters resulted in underestimating SPH. According to Lindberg et al. (2010) , ITD underestimates SPH because ITD often misses trees below dominant trees or recognizes trees close to each other as one tree. It was expected that there would be more underestimation as pixel size increased. The larger pixel size we have, the more aggregated information we would get, so lower resolution also would result in underestimating SPH. For this reason, estimates of BA decreased as resolution decreased. The approach of Lindberg et al. (2010) could give an improvement for estimating overstory trees for our study. Considering that most of the combinations for overstory trees by ITD produced negative biases in SPH estimation in our study (Figure 3(a)), estimating of the number of trees per segment would improve the negative biases in SPH estimation by increasing the number of detected trees.

Performance measures for HT estimation were better than measures for the other variables tested. This might be because LiDAR directly measures heights of target objects, so there is less uncertainty in height estimation than other attribute estimation. According to Stere?czak et al. (2008) , there was no difference between the three resolutions (0.25, 0.5, and 1.0 m) in CHM for HT estimation. For understory trees, biases in HT estimation less than 0.15 m in absolute value were produced by RF NN at every height cutoff. The higher the height cutoff applied, the larger the RMSE obtained. This is attributed to the fact that RF NN will have more and larger trees to estimate with higher height cutoffs.

While RF NN imputation showed better performance in estimating SPH than the combined approach and tree segmentation (Figures 3-6), this does not mean that RF NN imputation is better than the combined approach or ITD. It is because the target trees for those two methods are different from each other (tall trees above a height cutoff for ITD and short trees below the height cutoff for RF NN imputation). Therefore, the values dealt with in RF NN imputation were smaller than ITD. Based on relative measures not included on this manuscript, RF NN imputation was generally better in RBias, comparable in EIs, and worse in RRMSE.

The errors for BA and mean HT estimation in taller height classes were larger than in shorter height classes contradicting the fact that airborne LiDAR has difficulty in detecting understory vegetation. This might be because large trees have larger DBH and HT than small trees. To offset this potential issue, relative performance measures such as RBias and RRMSE were calculated. These relative measures revealed that the performance of estimation in shorter height classes was poorer than for the trees in taller height classes. Stere?czak et al. (2008) found a similar phenomenon for young stands.

There was also no single combination of the three parameters tested for explanatory powers that proved best overall. While HT estimation was good, estimation of BA and SPH were poor. Especially, BA estimation was very poor. The negative R^2 indicates (Tables 5-8) our results were worse than the mean value of the data. However, the combinations of parameters for each forest attribute could be a partial guide of generating tree-lists for the forest attributes. First, for overstory trees, R^2 for SPH had larger values with the finer resolution and the higher height cutoff. On the contrary to SPH, R^2 for BA had larger values with the coarser resolution and the lower height cutoff. R^2 for HT had larger values with the finer resolution and the middle height cutoff. For understory trees, R^2 for SPH had larger values with lower height cutoff, and R^2 for BA with higher height cutoff. BA estimation for overstory trees via ITD had more uncertainty sources than the other attributes, including SPH estimation and subsequent prediction of DBH for each detected individual tree (estimated HT used to predict DBH provided additional uncertainty source to the DBH prediction). These uncertainty sources might partially explain the poor performance in BA estimation. Utilizing the limited information in LiDAR data might affect the poor performance for the explanatory powers. We used CHM-based ITD; this method has limitation summarizing LiDAR point clouds within a range of cell into one cell height value regardless of generating a pit-free CHM.

Instead, 3D ITD methods have been recently studied using information in LiDAR as much as possible (Kandare, Ørka, Chan, & Dalponte, 2016) . However, the 3D ITD methods required more complex algorithms to implement, and also processing time could be a new parameter to consider (Pirotti, Kobal, & Roussel, 2017) .

It is well known that it is difficult to estimate characteristics of understory vegetation. Eskelson, Madsen, Hagar, and Temesgen (2011) used beta regression to estimate percent shrub cover, and it yielded poor explanatory power. Rahman and Gorte (2008) developed a tree filtering technique to separate dominant tree and undergrowth vegetation, but it was found difficult to separate undergrowth vegetation very close to a tree using the filtering. Liu, Shen, Zhao, and Xu (2013) suggested a method to extract individual tree crowns from airborne LiDAR in residential areas showing promising applications, but also reported that small trees were omitted if there were an only small number of points representing them in the dataset. Our results for understory trees via RF NN were not good (Table 8). To improve NN estimation with LiDAR data having low point cloud density, we investigated many LiDAR metrics such as metrics from LiDAR point clouds under several height cutoffs as Wing et al. (2012) proposed to estimate understory vegetation cover with airborne LiDAR. Some of the metrics from understory point clouds were selected for NN imputation (Table 4). However, it did not greatly improve the performance of NN imputation compared to NN imputation without those metrics (not presented here). This might fundamentally be because our LiDAR data lacked information on understory vegetation.

In NN imputation, one of the critical parameters is the selection of a number of neighbors for imputation modeling or distance metrics used to measure the similarity between the reference and target plot using auxiliary variables (Eskelson et al., 2009) . While their result varied among different forest types, Strunk et al. (2017) reported that k = 3 and Mahalanobis distance metric produced better performance over other NN strategies in estimating tree-lists. In this study, we only used k = 1 and RF as a distance metric in NN modeling. Combination of the two parameters needs to be examined for understory vegetation. In addition to these two factors, implementing variable selection procedure for NN imputation to each LiDAR height class could have the potential to improve NN modeling performance.

Compared to the results of HT estimation, results related to DBH estimation such as BA and EI for DBH showed poorer performance. It is known that predicting tree-level DBH from height-derived metrics has considerable variability (Matti Maltamo & Gobakken, 2014) . Kaartinen et al. (2012) reported that estimation of DBH based on HT and crown size would have considerable uncertainty because allometric equations used for estimating DBH are sensitive to errors in input data such as the size of tree crown or HT. Another potential reason is the dead trees in the field data. The Pearson's correlation coefficients between the field-measured HT and DBH for live and dead trees are 0.771 and 0.212, respectively. Even though the dead trees account for only 8.8% of a total number of field-measured trees, appropriate handling for dead trees would give opportunities to improve estimating tree-lists.

The scanning angle is another parameter to consider for LiDAR projects (Gatziolis & Andersen, 2008) . If the scanning angle increases, it facilitates changes in pulse propagation direction and increases the distance the pulse moves through the canopy. The change in pulse direction and the increased distance are related to LiDAR data artifacts such as returns below the ground. Therefore, with a wide scanning angle, LiDAR data might have more data artifacts than with a narrow-angle. Additionally, these data artifacts could increase when data acquisition is carried out on a slope, as an off-nadir scanning angle increases on the slope (Gatziolis & Andersen, 2008) .

39.4% of our field plots had slopes more than 30˚ based on digital terrain models from the study site. Khosravipour, Skidmore, Wang, Isenburg, and Khoshelham (2015) showed that normalized LiDAR point clouds could distort tree locations detected from CHM and height estimation depending on the steepness of slope and crown shape. For the slope of more than 30˚ 44.6% of correctly detected trees with wider and irregular crown shapes were affected by the horizontal and vertical displacements. They suggested using a non-normalized CHM to avoid the adverse effect of the distortion by steep slopes, especially in a heterogeneous forest with multiple species. The slope was also found to affect the ABA approach by distorting heights of LiDAR point clouds (Hansen et al., 2017) . They proposed two methods, Procrustean transformation and histogram matching, to counter the distortion of LiDAR point clouds on slope terrain for extracting LiDAR metrics. These point cloud distortions by slope terrain could worsen our results for both overstory and understory estimations.

Another issue is that there was the time lag between LiDAR acquisition and field surveys. This might have the potential source of error, particularly for younger fast-growing stands. Also, there were seasonal differences in the LiDAR acquisition dates (e.g., April through June in the spring, June through August in the summer, and September and October in the fall). According to Gatziolis and Andersen (2008) , the seasonal differences can induce considerable variability in canopy penetrability by LiDAR pulses especially for deciduous forests (e.g., leaf-on and leaf-off conditions) and weather-related limitations. The variability in canopy penetrability might increase uncertainty in modeling forest attributes, and the weather-related limitations could make it difficult to keep the quality of LiDAR data consistent over our whole study area. Time windows, part of LiDAR data acquisition considerations in Gatziolis and Andersen (2008) , should be carefully planned according to project objectives.

CONCLUSION

We proposed an approach to combine ITD and ABA to generate a tree-list using airborne LiDAR data and field measured data. The approach aimed to compensate for the disadvantage of LiDAR data and ITD in estimating understory trees, and to keep the strength of ITD in estimating overstory trees in tree-level. The selected parameters, smoothing, resolution and height cutoff, were examined to determine how they affected the performance of the proposed approach. There was no single combination of the three parameters that provides the best estimation results for all the forest attributes in this study. For each attribute, the best results depended on different combinations of those parameters. This is concurrent with what Koch et al. (2014) and McGaughey (2016) reported. However, our study provided the ranges and patterns of the selected parameters that yielded better performance results for each forest attribute, which could be a partial guide of estimating tree-lists using airborne LiDAR. It would be practical and useful to determine how to automatically find the optimal combinations of those parameters across the forest landscape using remote sensing data. In addition to the three parameters tested in the present study, the automation for the optimal combinations would require considering additional parameters such as forest types, tree species, tree-size parameters (tree crown width or maximum tree height) and topography.

There are several topics for further study to improve the combined approach. A denser point cloud data would have more information on both overstory and understory vegetation in a forest, thus could increase the combined approach's performance. The algorithm used to generate a CHM and to delineate trees on the CHM is another critical parameter in ITD.

Comparison of different algorithms for processing the CHM is an active area of research. Estimating the number of trees per crown segment would help obtain unbiased SPH estimation. A point cloud based ITD method could lead to improvement by utilizing more information in LiDAR data. A minimum crown area by ITD should be examined so that tiny crown would not degrade the quality of the predicted tree-lists. The effect of slope on CHM generation and LiDAR metrics extraction need to be considered for better estimation. Fusing ITD and ABA to predict overstory and understory vegetation shown in this research indicates that forest analysts can benefit from the predictive abilities of the imputation approach and the quality information provided by LiDAR. In that, the approach presented herein can be sufficient for strategic inventory purposes.

Acronyms

ABA:	Area-based approach
BA:	Basal area
CHM:	Canopy height model
DBH:	Diameter at breast height
SPH:	Stems per hectare
EI:	Error index
HT:	Tree height
ITD:	Individual tree detection
NN:	Nearest neighbor
LiDAR:	Light detection and ranging
R^2:	Coefficient of determination
RBias:	Relative bias
RF:	Random forest
RMSE:	Root mean squared error
RRMSE:	Relative root mean squared error

REFERENCES

1. Barnes, C., Balzter, H., Barrett, K., Eddy, J., Milner, S., & Suárez, J. (2017). Individual Tree Crown Delineation from Airborne Laser Scanning for Diseased Larch Forest Stands. Remote Sensing, 9, 231. https://doi.org/10.3390/rs9030231
2. Breidenbach, J., Næsset, E., Lien, V., Gobakken, T., & Solberg, S. (2010). Prediction of Species Specific Forest Inventory Attributes Using a Nonparametric Semi-Individual Tree Crown Approach Based on Fused Airborne Laser Scanning and Multispectral Data. Remote Sensing of Environment, 114, 911-924. https://doi.org/10.1016/j.rse.2009.12.004
3. Breiman, L. (2001). Random Forests. Machine Learning, 45, 5-32. https://doi.org/10.1023/A:1010933404324
4. Crookston, N. L., & Finley, A. O. (2008). Yaimpute: An R Package for Knn Imputation. Journal of Statistical Software, 23, 16. https://doi.org/10.18637/jss.v023.i10
5. Eskelson, B. N. I., Madsen, L., Hagar, J. C., & Temesgen, H. (2011). Estimating Riparian Understory Vegetation Cover with Beta Regression and Copula Models. Forest Science, 57, 212-221.
6. Eskelson, B. N. I., Temesgen, H., Lemay, V., Barrett, T. M., Crookston, N. L., & Hudak, A. T. (2009). The Roles of Nearest Neighbor Methods in Imputing Missing Data in Forest Inventory and Monitoring Databases. Scandinavian Journal of Forest Research, 24, 235-246. https://doi.org/10.1080/02827580902870490
7. Gatziolis, D., & Andersen, H.-E. (2008). A Guide to Lidar Data Acquisition and Processing for the Forests of the Pacific Northwest (Gen. Tech. Rep. PNW-GTR-768) (32 p). Portland, OR: US Department of Agriculture, Forest Service, Pacific Northwest Research Station. https://doi.org/10.2737/PNW-GTR-768
8. Gobakken, T., & Næsset, E. (2004). Estimation of Diameter and Basal Area Distributions in Coniferous Forest by Means of Airborne Laser Scanner Data. Scandinavian Journal of Forest Research, 19, 529-542. https://doi.org/10.1080/02827580410019454
9. Hamraz, H., Contreras, M. A., & Zhang, J. (2017). Vertical Stratification of Forest Canopy for Segmentation of Understory Trees within Small-Footprint Airborne Lidar Point Clouds. ISPRS Journal

of Photogrammetry and Remote Sensing, 130, 385-392. https://doi.org/10.1016/j.isprsjprs.2017.07.001

10. Hansen, E., Ene, L., Gobakken, T., Ørka, H., Bollandsås, O., & Næsset, E. (2017). Countering Negative Effects of Terrain Slope on Airborne Laser Scanner Data Using Procrustean Transformation and Histogram Matching. Forests, 8, 401. https://doi.org/10.3390/f8100401
11. Hawbaker, T. J., Keuler, N. S., Lesak, A. A., Gobakken, T., Contrucci, K., & Radeloff, V. C. (2009). Improved Estimates of Forest Vegetation Structure and Biomass with a Lidar-Optimized Sampling Design. Journal of Geophysical Research: Biogeosciences, 114, G00E04. https://doi.org/10.1029/2008JG000870
12. Heurich, M., & Weinacker, H. (2004). Automated Tree Detection and Measurement in Temperate Forests of Central Europe Using Laser Scanning Data. Freiburg: The ISPRS Working Group on Laser-Scanners for Forest and Landscape Assessment.
13. Hopkinson, C., Chasmer, L. E., Sass, G., Creed, I. F., Sitar, M., Kalbfleisch, W., & Treitz, P. (2005). Vegetation Class Dependent Errors in Lidar Ground Elevation and Canopy Height Estimates in a Boreal Wetland Environment. Canadian Journal of Remote Sensing, 31, 191-206. https://doi.org/10.5589/m05-007
14. Kaartinen, H., Hyyppä, J., Yu, X., Vastaranta, M., Hyyppä, H., Kukko, A., Wu, J.-C. et al. (2012). An International Comparison of Individual Tree Detection and Extraction Using Airborne Laser Scanning. Remote Sensing, 4, 950. https://doi.org/10.3390/rs4040950
15. Kandare, K., Ørka, H. O., Chan, J. C.-W., & Dalponte, M. (2016). Effects of Forest Structure and Airborne Laser Scanning Point Cloud Density on 3d Delineation of Individual Tree Crowns. European Journal of Remote Sensing, 49, 337-359. https://doi.org/10.5721/EuJRS20164919
16. Khosravipour, A., Skidmore, A. K., Isenburg, M., Wang, T., & Hussin, Y. A. (2014). Generating Pit-Free Canopy Height Models from Airborne Lidar. Photogrammetric Engineering & Remote Sensing, 80, 863-872. https://doi.org/10.14358/PERS.80.9.863
17. Khosravipour, A., Skidmore, A. K., Wang, T., Isenburg, M., & Khoshelham, K. (2015). Effect of Slope on Treetop Detection Using a Lidar Canopy Height Model. ISPRS Journal of Photogrammetry and Remote Sensing, 104, 44-52. https://doi.org/10.1016/j.isprsjprs.2015.02.013

18. Koch, B., Heyder, U., & Weinacker, H. (2006). Detection of Individual Tree Crowns in Airborne Lidar Data. Photogrammetric Engineering & Remote Sensing, 72, 357-363. https://doi.org/10.14358/PERS.72.4.357

19. Koch, B., Kattenborn, T., Straub, C., & Vauhkonen, J. (2014). Segmentation of Forest to Tree Objects. In M. Maltamo, E. Næsset, & J. Vauhkonen (Eds.), Forestry Applications of Airborne Laser Scanning: Concepts and Case Studies (pp. 89-112). Dordrecht: Springer Netherlands. https://doi.org/10.1007/978-94-017-8663-8_5

20. Liaw, A., & Wiener, M. (2002). Classification and Regression by Randomforest. R News, 2, 18-22.

21. Lindberg, E., & Hollaus, M. (2012). Comparison of Methods for Estimation of Stem Volume, Stem Number and Basal Area from Airborne Laser Scanning Data in a Hemi-Boreal Forest. Remote Sensing, 4, 1004-1023. https://doi.org/10.3390/rs4041004

22. Lindberg, E., & Holmgren, J. (2017). Individual Tree Crown Methods for 3D Data from Remote Sensing. Current Forestry Reports, 1-13. https://doi.org/10.1007/s40725-017-0051-6

23. Lindberg, E., Holmgren, J., Olofsson, K., Wallerman, J., & Olsson, H. (2010). Estimation of Tree Lists from Airborne Laser Scanning by Combining Single-Tree and Area-Based Methods. International Journal of Remote Sensing, 31, 1175-1192. https://doi.org/10.1080/01431160903380649

24. Lindberg, E., Holmgren, J., Olofsson, K., Wallerman, J., & Olsson, H. (2013). Estimation of Tree Lists from Airborne Laser Scanning Using Tree Model Clustering and K-Msn Imputation. Remote Sensing, 5, 1932-1955. https://doi.org/10.3390/rs5041932

25. Liu, J., Shen, J., Zhao, R., & Xu, S. (2013). Extraction of Individual Tree Crowns from Airborne Lidar Data in Human Settlements. Mathematical and Computer Modelling, 58, 524-535. https://doi.org/10.1016/j.mcm.2011.10.071

26. Maltamo, M., & Gobakken, T. (2014). Predicting Tree Diameter Distributions. In M. Maltamo, E. Næsset, & J. Vauhkonen (Eds.), Forestry Applications of Airborne Laser Scanning: Concepts and Case Studies (pp. 177-191). Dordrecht: Springer Netherlands. https://doi.org/10.1007/978-94-017-8663-8_9

27. Maltamo, M., Eerikäinen, K., Pitkänen, J., Hyyppä, J., & Vehmas, M. (2004). Estimation of Timber Volume and Stem Density Based

on Scanning Laser Altimetry and Expected Tree Size Distribution Functions. Remote Sensing of Environment, 90, 319-330. https://doi.org/10.1016/j.rse.2004.01.006

28. Maltamo, M., Tokola, T., & Lehikoinen, M. (2003). Estimating Stand Characteristics by Combining Single Tree Pattern Recognition of Digital Video Imagery and a Theoretical Diameter Distribution Model. Forest Science, 49, 98-109.
29. McGaughey, R. J. (2016). Fusion/Ldv: Software for Lidar Data Analysis and Visualization, Forest Service. Pacific Northwest Research Station, United States Department of Agriculture.
30. Næsset, E., & Gobakken, T. (2008). Estimation of Above- and Below-Ground Biomass across Regions of the Boreal Forest Zone Using Airborne Laser. Remote Sensing of Environment, 112, 3079-3090. https://doi.org/10.1016/j.rse.2008.03.004
31. Persson, A., Holmgren, J., & Soderman, U. (2002). Detecting and Measuring Individual Trees Using an Airborne Laser Scanner. Photogrammetric Engineering and Remote Sensing, 68, 925-932.
32. Pippuri, I., Kallio, E., Maltamo, M., Peltola, H., & Packalén, P. (2012). Exploring Horizontal Area-Based Metrics to Discriminate the Spatial Pattern of Trees and Need for First Thinning Using Airborne Laser Scanning. Forestry, 85, 305-314. https://doi.org/10.1093/forestry/cps005
33. Pirotti, F., Kobal, M., & Roussel, J. R. (2017). A Comparison of Tree Segmentation Methods Using Very High Density Airborne Laser Scanner Data. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 42, 285-290. https://doi.org/10.5194/isprs-archives-XLII-2-W7-285-2017
34. Pouliot, D. A., King, D. J., Bell, F. W., & Pitt, D. G. (2002). Automated Tree Crown Detection and Delineation in High-Resolution Digital Camera Imagery of Coniferous Forest Regeneration. Remote Sensing of Environment, 82, 322-334. https://doi.org/10.1016/S0034-4257(02)00050-0
35. R Core Team (2017). R: A Language and Environment for Statistical Computing. Vienna: R Foundation for Statistical Computing. http://www.R-project.org/
36. Rahman, M. Z. A., & Gorte, B. (2008). Tree Filtering for High Density Airborne Lidar Data. In Proceedings of Silvilaser 2008: 8th

International Conference on LiDAR Applications in Forest Assessment and Inventory (pp. 544-553). Edinburgh: Heriot-Watt University.

37. Reynolds, M. R., Burk, T. E., & Huang, W.-C. (1988). Goodness-of-Fit Tests and Model Selection Procedures for Diameter Distribution Models. Forest Science, 34, 373-399.

38. Romero-Zaliz, R., & Reinoso-Gordo, J. F. (2018). An Updated Review on Watershed Algorithms. In C. Cruz Corona (Ed.), Soft Computing for Sustainability Science (pp. 235-258). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-62359-7_12

39. Smrecek, R., Michnová, Z., Sackov, I., Danihelová, Z., Levická, M., & Tucek, J. (2018). Determining Basic Forest Stand Characteristics Using Airborne Laser Scanning in Mixed Forest Stands of Central Europe. iForest—Biogeosciences and Forestry, 11, 181-188.

40. Stereńczak, K., Bedkowski, K., & Weinacker, H. (2008). Accuracy of Crown Segmentation and Estimation of Selected Trees and Forest Stand Parameters in Order to Resolution of Used Dsm and Ndsm Models Generated from Dense Small Footprint Lidar Data. Beijing: The ISPRS Congress.

41. Strunk, J., Gould, P., Packalen, P., Poudel, K., Andersen, H.-E., & Temesgen, H. (2017). An Examination of Diameter Density Prediction with K-Nn and Airborne Lidar. Forests, 8, 444. https://doi.org/10.3390/f8110444

42. Takahashi, T., Yamamoto, K., Miyachi, Y., Senda, Y., & Tsuzuku, M. (2006). The Penetration Rate of Laser Pulses Transmitted from a Small-Footprint Airborne Lidar: A Case Study in Closed Canopy, Middle-Aged Pure Sugi (Cryptomeria Japonica D. Don) and Hinoki Cypress (Chamaecyparis Obtusa Sieb. Et Zucc.) Stands in Japan. Journal of Forest Research, 11, 117-123. https://doi.org/10.1007/s10310-005-0189-0

43. Temesgen, H., LeMay, V. M., Froese, K. L., & Marshall, P. L. (2003). Imputing Tree-Lists from Aerial Attributes for Complex Stands of South-Eastern British Columbia. Forest Ecology and Management, 177, 277-285. https://doi.org/10.1016/S0378-1127(02)00321-3

44. Vauhkonen, J., Ene, L., Gupta, S., Heinzel, J., Holmgren, J., Pitkänen, J., Maltamo, M. et al. (2012). Comparative Testing of Single-Tree Detection Algorithms under Different Types of Forest. Forestry: An International Journal of Forest Research, 85, 27-40. https://doi.org/10.1093/forestry/cpr051

45. Vauhkonen, J., Maltamo, M., McRoberts, R. E., & Næsset, E. (2014). Introduction to Forestry Applications of Airborne Laser Scanning. In M. Maltamo, E. Næsset, & J. Vauhkonen (Eds.), Forestry Applications of Airborne Laser Scanning: Concepts and Case Studies (pp. 1-16). Dordrecht: Springer Netherlands. https://doi.org/10.1007/978-94-017-8663-8_1

46. Vincent, L., & Soille, P. (1991). Watersheds in Digital Spaces: An Efficient Algorithm Based on Immersion Simulations. IEEE Transactions on Pattern Analysis and Machine Intelligence, 13, 583-598. https://doi.org/10.1109/34.87344

47. Walther, B. A., & Moore, J. L. (2005). The Concepts of Bias, Precision and Accuracy, and Their Use in Testing the Performance of Species Richness Estimators, with a Literature Review of Estimator Performance. Ecography, 28, 815-829. https://doi.org/10.1111/j.2005.0906-7590.04112.x

48. Wiggins, H. L. (2017). The Influence of Tree Height on Lidar's Ability to Accurately Characterize Forest Structure and Spatial Pattern across Reference Landscapes. Master of Science, Missoula, MT: University of Montana. http://scholarworks.umt.edu/etd/11025

49. Wing, B. M., Ritchie, M. W., Boston, K., Cohen, W. B., Gitelman, A., & Olsen, M. J. (2012). Prediction of Understory Vegetation Cover with Airborne Lidar in an Interior Ponderosa Pine Forest. Remote Sensing of Environment, 124, 730-741. https://doi.org/10.1016/j.rse.2012.06.024

50. Yu, X., Hyyppä, J., Holopainen, M., & Vastaranta, M. (2010). Comparison of Area-Based and Individual Tree-Based Methods for Predicting Plot-Level Forest Attributes. Remote Sensing, 2, 1481. https://doi.org/10.3390/rs2061481

51. Yu, X., Hyyppä, J., Vastaranta, M., Holopainen, M., & Viitala, R. (2011). Predicting Individual Tree Attributes from Airborne Laser Point Clouds Based on the Random Forests Technique. ISPRS Journal of Photogrammetry and Remote Sensing, 66, 28-37. https://doi.org/10.1016/j.isprsjprs.2010.08.003

Chapter

A RECURSIVE APPROACH TO THE KAUFFMAN BRACKET

11

Abdul Rauf Nizami, Mobeen Munir, Umer Saleem, and Ansa Ramzan

Division of Science and Technology, University of Education, Lahore, Pakistan

ABSTRACT

We introduce a simple recursive relation and give an explicit formula of the Kauffman bracket of two-strand braid link $\widehat{x_1^n}$. Then, we give general formulas of the bracket of the sequence of links of three-strand braids $\alpha(n) = x_1x_2x_1x_2\cdots$. Finally, we give an interesting result that the Kauffman bracket of the three-strand braid link $\widehat{x_1^m x_2^n}$ is actually the product of the brackets of the two- strand braid links $\widehat{x_1^m}$ and $\widehat{x_1^n}$. Moreover, a recursive relation for $\left\langle \widehat{x_1^a x_2^b x_1^c x_2^d} \right\rangle$ is also given.

Keywords: Recursive Relation, Kauffman Bracket, Braid Link

Citation: Nizami, A., Munir, M., Saleem, U. and Ramzan, A. (2014), "A Recursive Approach to the Kauffman Bracket". Applied Mathematics, 5, 2746-2755. doi: 10.4236/am.2014.517262.

INTRODUCTION

The Kauffman bracket polynomial was introduced by L. H. Kauffman in 1987 [1] in concern with link invariants. The bracket polynomial soon became popular due to its connections with the Jones polynomial, dichromatic polynomial, and the Potts model. While the HOMPLY polynomial and the bracket polynomial are distinct with different topological properties, there is a very beautiful relationship between them due to F. Jaeger [2] , and it is also observed in a special case by Reshetikhin [3] .

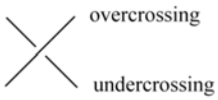
The Kauffman bracket (polynomial) is actually not a link invariant because it is not invariant under the first Reidemeister move. However, it has many applications and it can be extended to a popular link invariant, the Jones polynomial. In the present work we shall confine ourselves to the Kauffman bracket to avoid this work from unnecessary length and to leave it for applications.

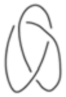
This paper is organized as follows: In Section 2 we shall give the basic ideas about knots, braids, and the Kauffman bracket. In Section 3 we shall present the main results.

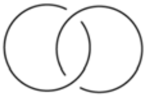
BASIC NOTIONS

Links

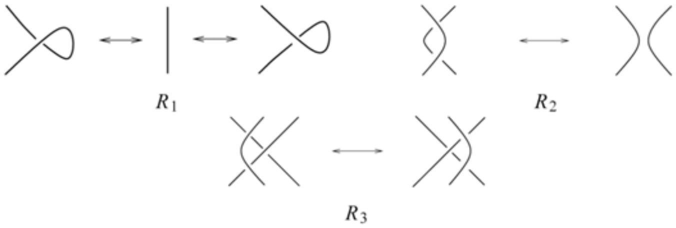
A link is a disjoint union of circles embedded in $\mathbb{R}^3$. A one-component link is called a knot. Links are usually studied via projecting them on a plan; a projection with extra information of overcrossing and undercrossing is called the link diagram.




Trefoil knot


Hopf link

Two links are isotopic if and only if one of them can be transformed to the other by a diffeomorphism of the ambient space onto itself. A fundamental result by Reidemeister [4] about the isotopic link diagrams is: Two unoriented links L_1 and L_2 are equivalent if and only if a diagram of L_1 can be transformed into a diagram of L_1 by a finite sequence of ambient isotopies of the plane and the local (Reidemeister) moves of the following three types:



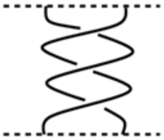
The set of all links that are equivalent to a link *L* is called a class of *L*. By a link *L* we shall always mean the class of *L*.

The main question of knot theory is Which two links are equivalent and which are not? To address this question one needs a knot invariant, a function that gives one value on all links that belong to a single class and gives different values (but not always) on knots that belong to different classes. The present work is basically concerned with this question.

Braids

Braids were first studied by Emil Artin in 1925 [5] [6] , which now play an important role in knot theory, see [7] -[9] for detail.

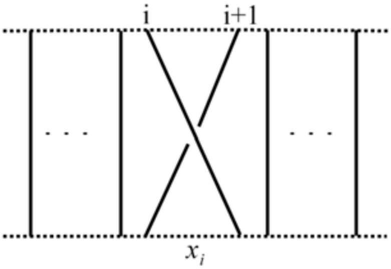
An n-strand *braid* is a set of *n* non intersecting smooth paths connecting *n* points on a horizontal plane to *n* points exactly below them on another horizontal plane in an arbitrary order. The smooth paths are called strands of the braid.



A 2-strand braid

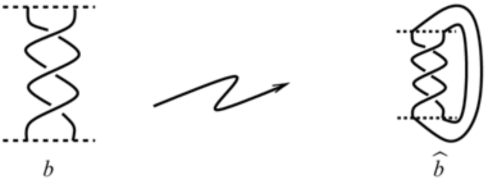
The product *ab* of two n-strand braids is defined by putting the braid *b* above the braid *a* and then gluing their common end points.

A braid with only one crossing is called elementary braid. The ith elementary braid x_i on *n* strands is:



A useful property of elementary braids is that every braid can be written as a product of elementary braids. For instance, the above 2-strand braid is $x_i^{-3} = \left(x_i^{-1}\right)\left(x_i^{-1}\right)\left(x_i^{-1}\right)$.

The closure of a braid *b* is the link $\hat{b}$ obtained by connecting the lower ends of *b* with the corresponding upper ends.

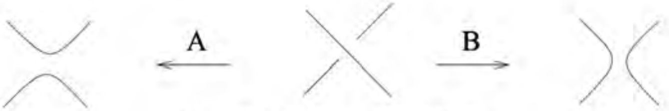


An important result by Alexander [10] connecting knots and braids is: Each link can be represented as the closure of a braid. This result motivated knot theorists to study braids to solve problems of knot theory.

Remark 2.1 In the last section, all the concerned links will be closures of products of elementary braids.

The Kauffman Bracket

Before the definition it is better to understand the two types of splitting of a crossing, the A-type and the B-type splittings:



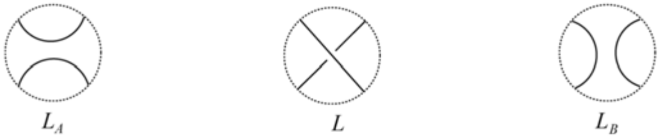
A- and *B*-type splittings

In the following, the symbols ○ and ⊔ represent respectively the unknot and the disconnected sum.

Definition2.2TheKauffmanbracketisthefunction $\langle\cdot\rangle : \text{Links} \to \mathbb{Z}\left[a, a^{-1}\right]$ defined by the axioms:

$$\langle L\rangle = a\langle L_A\rangle + a^{-1}\langle L_B\rangle$$
$$\langle L \sqcup \bigcirc\rangle = \left(-a^2 - a^{-2}\right)\langle L\rangle$$
$$\langle\bigcirc\rangle = 1.$$

Here L, L_A, and L_B are three links which are isotopic everywhere except at one crossing where the look as in the figure:

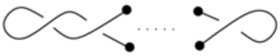


Proposition 2.3 The Kauffman polynomial is invariant under second and third Reidemeister moves but not under the first Reidemeister move [11] .

Main Results

In this section we shall introduce a recursive relation for the Kauffman bracket, shall give an explicit formula of $\left\langle\widehat{x_1^n}\right\rangle$, and shall express $\left\langle\widehat{x_1^m x_2^n}\right\rangle$ as the product of $\left\langle\widehat{x_1^m}\right\rangle$ and $\left\langle\widehat{x_1^n}\right\rangle$.

First of all we give the Kauffman bracket of the k-twist unknot U_k:



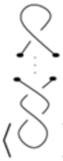
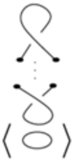
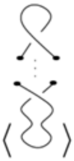
Unknot with k twists

Lemma 3.1 The Kauffman bracket of the k-twist unknot is

$$\langle U_k\rangle = (-1)^k a^{3k}.$$

Proof. We prove it by induction on k:

The case $k = 0$ holds by definition as U_0 is the unknot without any crossings. Now, with the assumption that the result holds for an arbitrary k, we have

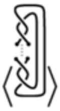
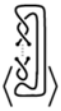
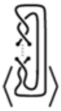
$\langle U_{k+1}\rangle = \Big\langle$  $\Big\rangle = a\Big\langle$  $\Big\rangle + a^{-1}\Big\langle$  $\Big\rangle$

$$\begin{aligned} &= a(-a^{2}-a^{-2})\langle U_k\rangle + a^{-1}\langle U_k\rangle \\ &= -a^{3}\langle U_k\rangle - a^{3}[(-1)^{k}a^{3k}] \\ &= (-1)^{k+1}a^{3(k+1)}.\ \square \end{aligned}$$

Theorem 3.2 (A recursive relation) The following relation holds for any $n \geq 2$:

$$\left\langle \widehat{x_1^{n}} \right\rangle = (-1)^{n-1}a^{3n-2} + a^{-1}\left\langle \widehat{x_1^{n-1}} \right\rangle. \tag{3.1}$$

Proof. We prove it using directly the definition and Lemma 3.1:

$\langle \widehat{x_1^{n}} \rangle = \Big\langle$  $\Big\rangle = a\Big\langle$  $\Big\rangle + a^{-1}\Big\langle$  $\Big\rangle$

$$= a\langle U_{n-1}\rangle + a^{-1}\langle \widehat{x_1^{n-1}}\rangle$$

$$= a[(-1)^{n-1}a^{3(n-1)}] + a^{-1}\langle \widehat{x_1^{n-1}}\rangle$$

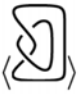
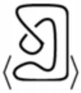
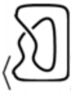
$$= (-1)^{n-1}a^{3n-2} + a^{-1}\langle \widehat{x_1^{n-1}}\rangle.\square$$

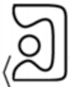
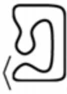
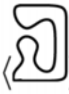
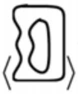
From this recursive relation, we get the explicit formula for the 2-strand braid link $\widehat{x_1^{n}}$:

Proposition 3.3 The Kauffman bracket of the link $\widehat{x_1^{n}}, n \geq 2,$ is

$$\left\langle \widehat{x_1^{n}} \right\rangle = -a^{-n-2} + \sum_{k=1}^{n-1}(-1)^{n+k-2}a^{(3n+2)-4k}.$$

Proof. We prove it by induction on n.
For $n = 2$, we have

$\langle \widehat{x_1^2} \rangle = \langle$  $\rangle = a\,\langle$  $\rangle + a^{-1}\langle$  $\rangle$

$= a\Big[a\,\langle$  $\rangle + a^{-1}\langle$  $\rangle\Big] + a^{-1}\Big[a\,\langle$  $\rangle + a^{-1}\langle$  $\rangle\Big]$

$$= a[a(-a^2-a^{-2})+a^{-1}(1)]+a^{-1}[a(1)+a^{-1}(-a^2-a^{-2})]$$
$$= -a^4-a^{-4},$$

which satisfies the recursive relation.

With the assumption that the relation holds for an arbitrary *n*, we, using Theorem 3.2, get

$$\left\langle \widehat{x_1^{n+1}} \right\rangle = (-1)^{(n+1)-1} a^{3(n+1)-2} + a^{-1}\left\langle \widehat{x_1^{n}} \right\rangle$$
$$= (-1)^n a^{3(n+1)-2} + a^{-1}\Big[(-1)^{n-1} a^{3n-2} + (-1)^{n-2} a^{3n-6}$$
$$+(-1)^{n-3} a^{3n-10} + \cdots + (-1)^{2n-3} a^{-n+6} - a^{-n-2}\Big]$$
$$= (-1)^n a^{3(n+1)-2} + \Big[(-1)^{n-1} a^{3n-3} + (-1)^{n-2} a^{3n-7}$$
$$+(-1)^{n-3} a^{3n-11} + \cdots + (-1)^{2n-3} a^{-n+5} - a^{-n-3}\Big]$$
$$= (-1)^n a^{3(n+1)-2} + (-1)^{n-1} a^{3(n+1)-6} + (-1)^{n-2} a^{3(n+1)-10}$$
$$+(-1)^{n-3} a^{3(n+1)-14} + \cdots + (-1)^{2(n+1)-3} a^{-(n+1)+6} - a^{-(n+1)-2}.$$

This completes the proof.

In the following we give the Kauffman bracket polynomial of the closure of the braid $\alpha(n) = x_1x_2x_1x_2\cdots$ (*n* factors); this sequence contains the powers of the Garside element $\Delta = x_1x_2x_1 = x_2x_1x_1 :\ \alpha(3k) = \Delta^k$.

Proposition 3.4 The Kauffman bracket of $\alpha(n) = x_1x_2x_1\cdots(n\text{-times})$ satisfy the recurrence relations:

$$\left\langle \Delta^{2k} \right\rangle = a^6 \left\langle \Delta^{2(k-1)} \right\rangle - a^{16-6k} - a^{8-6k} + a^{4-6k} + a^{-4-6k}$$

$$\left\langle \Delta^{2k} x_1 \right\rangle = a^6 \left\langle \Delta^{2(k-1)} x_1 \right\rangle - a^{15-6k} - a^{7-6k} + a^{3-6k} + a^{-5-6k}$$

$$\left\langle \Delta^{2k} x_1 x_2 \right\rangle = a^6 \left\langle \Delta^{2(k-1)} x_1 x_2 \right\rangle - a^{14-6k} - a^{6-6k} + a^{2-6k} + a^{-6-6k}$$

$$\left\langle \Delta^{2k+1} \right\rangle = a^6 \left\langle \Delta^{2k-1} \right\rangle - a^{13-6k} - a^{5-6k} + a^{1-6k} + a^{-7-6k}$$

$$\left\langle \Delta^{2k+1} x_2 \right\rangle = a^6 \left\langle \Delta^{2k-1} x_2 \right\rangle - a^{12-6k} - a^{4-6k} + a^{-6k} + a^{-8-6k}$$

$$\left\langle \Delta^{2k+1} x_2 x_1 \right\rangle = a^6 \left\langle \Delta^{2k-1} x_2 x_1 \right\rangle - a^{11-6k} - a^{3-6k} + a^{-1-6k} + a^{-9-6k}$$

Proof. Simply, apply the definition for different values of *k*, and write recursively each next bracket in terms of the previous one.

Lemma 3.5 The Kauffman brackets for *k* = 0 are:

$$\left\langle \Delta^{0} \right\rangle = a^4 + 2 + a^{-4}$$

$$\left\langle \Delta^{0} x_1 \right\rangle = a^{-1} + a^{-5}$$

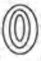
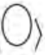
$$\left\langle \Delta^{0} x_1 x_2 \right\rangle = a^{-6}$$

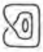
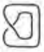
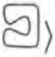
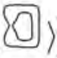
$$\left\langle \Delta \right\rangle = a + a^{-7}$$

$$\left\langle \Delta x_2 \right\rangle = -a^4 + 1 + a^{-8}$$

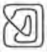
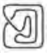
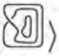
$$\left\langle \Delta x_2 x_1 \right\rangle = a^7 - a^3 + a^{-1} + a^{-9}$$

Proof. The proofs of first three cases are given (proofs of remaining cases are similar):

$\left\langle \Delta^{0} \right\rangle = \langle$  $\rangle = (-a^2 - a^{-2})^2 \langle$  $\rangle = a^4 + 2 + a^{-4}$

$\left\langle \Delta^{0} x_1 \right\rangle = \langle$  $\rangle = (-a^2 - a^{-2}) \langle$  $\rangle = (-a^2 - a^{-2}) \big[a \langle$  $\rangle + a^{-1} \langle$  $\rangle \big]$

$\qquad = (-a^2 - a^{-2}) \left[a + a^{-1}(-a^2 - a^{-2}) \right] = a^{-1} + a^{-5}$

$\left\langle \Delta^{0} x_1 x_2 \right\rangle = \langle$  $\rangle = a \langle$  $\rangle + a^{-1} \langle$  $\rangle$

$\qquad = a\langle x_1 \rangle + a^{-1}(-a^2 - a^{-2})\langle x_1 \rangle = a^{-6}$ □

Theorem 3.6 For any $k \geq 0$ the Kauffman bracket of $\alpha(n) = x_1 x_2 x_1 \cdots (n\text{-times})$ is given by:

$$\left\langle \Delta^{2k} \right\rangle = 2a^{6k} + a^{-6k+4} + a^{-6k-4}$$

$$\left\langle \Delta^{2k} x_1 \right\rangle = -a^{6k+3} + a^{6k-1} + a^{-6k+3} + a^{-6k-5}$$

$$\left\langle \Delta^{2k} x_1 x_2 \right\rangle = -a^{6k+2} + a^{-6k+2} + a^{-6k-6}$$

$$\left\langle \Delta^{2k+1} \right\rangle = a^{-6k+1} + a^{-6k-7}$$

$$\left\langle \Delta^{2k+1} x_2 \right\rangle = -a^{6k+4} + a^{-6k} + a^{-6k-8}$$

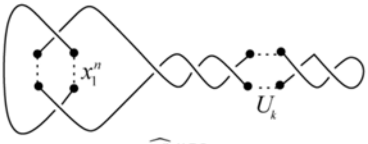
$$\left\langle \Delta^{2k+1} x_2 x_1 \right\rangle = a^{6k+7} - a^{6k+3} + a^{-6k-1} + a^{-6k-9}.$$

Proof. We prove it by induction on k. The case $k = 0$ is covered by Lemma 3.5, and the inductive step can be checked with Proposition 3.4.

For instance,

$$\begin{aligned}\left\langle \Delta^{2k+2} \right\rangle &= a^{6} \left\langle \Delta^{2k} \right\rangle - a^{10-6k} - a^{2-6k} + a^{-2-6k} + a^{-10-6k} \\ &= a^{6}\left[2a^{6k} + a^{4-6k} + a^{-4-6k}\right] - a^{10-6k} - a^{2-6k} + a^{-2-6k} + a^{-10-6k} \\ &= 2a^{6+6k} + a^{10-6k} + a^{2-6k} - a^{10-6k} - a^{2-6k} + a^{-10-6k} + a^{-2-6k} \\ &= 2a^{6(k+1)} + a^{4-6(k+1)} + a^{-4-6(k+1)}. \quad \square\end{aligned}$$

In connected sum $\widehat{x_1^n} \# U_k$ of the braid link $\widehat{x_1^n}$ with the trivial knot U_k has the diagram:



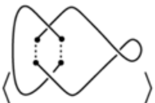
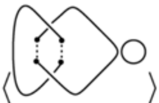
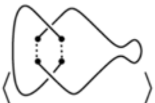
$\widehat{x_1^n} \# U_k$

Lemma 3.7

$$\left\langle \widehat{x_1^n} \# U_k \right\rangle = (-1)^k a^{3k} \left\langle \widehat{x_1^n} \right\rangle.$$

Proof. We prove it by induction on k:

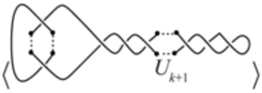
For $k = 1$, we have

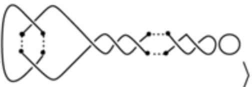
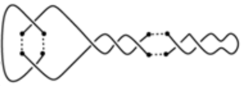
$\langle \widehat{x_1^n} \# U_1 \rangle = \langle$  $\rangle = a\,\langle$  $\rangle + a^{-1}\,\langle$  $\rangle$

$$= a(-a^2 - a^{-2})\langle \widehat{x_1^n} \rangle + a^{-1}\langle \widehat{x_1^n} \rangle$$

$$= -a^3 \langle \widehat{x_1^n} \rangle.$$

Now, with the assumption that the result holds for an arbitrary k, we have

$\langle \widehat{x_1^n} \# U_{k+1} \rangle = \langle$  $\rangle$

$= a\,\langle$  $\rangle + a^{-1}\,\langle$  $\rangle$

$$= a(-a^2 - a^{-2})\langle \widehat{x_1^n} \# U_k \rangle + a^{-1}\langle \widehat{x_1^n} \# U_k \rangle$$

$$= -a^3 \langle \widehat{x_1^n} \# U_k \rangle$$

$$= -a^3 [(-1)^k a^{3k} \langle \widehat{x_1^n} \rangle]$$

$$= (-1)^{k+1} a^{3(k+1)} \langle \widehat{x_1^n} \rangle,$$

as required.

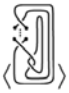
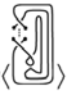
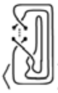
The following result confirms that the Kauffman bracket of $\left\langle \widehat{x_1^m x_2^n} \right\rangle$ is actually the product $\left\langle \widehat{x_1^m} \right\rangle \left\langle \widehat{x_1^n} \right\rangle$.

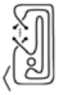
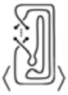
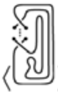
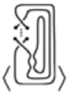
Theorem 3.8 For any m, $n \geq 2$,

$$\left\langle \widehat{x_1^m x_2^n} \right\rangle = \left\langle \widehat{x_1^m} \right\rangle \left\langle \widehat{x_1^n} \right\rangle.$$

Proof. We prove it by induction on n:

When $n = 2$,

$\left\langle \widehat{x_1^m x_2^2} \right\rangle = \langle$  $\rangle = a\langle$  $\rangle + a^{-1}\langle$  $\rangle$

$= a^2\langle$  $\rangle + \langle$  $\rangle + \langle$  $\rangle + a^{-2}\langle$  $\rangle$

$$= a^2(-a^2 - a^{-2})\left\langle \widehat{x_1^m} \right\rangle + \left\langle \widehat{x_1^m} \right\rangle + \left\langle \widehat{x_1^m} \right\rangle + a^{-2}(-a^2 - a^{-2})\left\langle \widehat{x_1^m} \right\rangle$$

$$= \left\langle \widehat{x_1^m} \right\rangle \left[-a^4 - 1 + 1 + 1 - 1 - a^{-4} \right]$$

$$= \left\langle \widehat{x_1^m} \right\rangle \left[-a^4 - a^{-4} \right] = \left\langle \widehat{x_1^m} \right\rangle \left\langle \widehat{x_1^2} \right\rangle. \tag{3.2}$$

Suppose the result holds for $n = k$, that is $\left\langle \widehat{x_1^m x_2^k} \right\rangle = \left\langle \widehat{x_1^m} \right\rangle \left\langle \widehat{x_1^k} \right\rangle$.

Now, using Lemma 3.7, we have

$$\left\langle \widehat{x_1^m x_2^{k+1}} \right\rangle = a\left\langle \widehat{x_1^m \# U_k} \right\rangle + a^{-1}\left\langle \widehat{x_1^m x_2^k} \right\rangle$$

$$= a\left[(-1)^k a^{3k} \left\langle \widehat{x_1^m} \right\rangle \right] + a^{-1}\left\langle \widehat{x_1^m} \right\rangle \left\langle \widehat{x_1^k} \right\rangle$$

$$= \left\langle \widehat{x_1^m} \right\rangle \left[(-1)^k a^{3k+1} + a^{-1}\left\langle \widehat{x_1^k} \right\rangle \right]$$

$$= \left\langle \widehat{x_1^m} \right\rangle \left[(-1)^k a^{3k+1} + (-1)^{k-1} a^{3k-3} + (-1)^k a^{3k-7} \right.$$
$$\left. + (-1)^{k+1} a^{3k-11} + \cdots + (-1)^{2k-2} a^{-k+5} - a^{-k-3} \right]$$

$$= \left\langle \widehat{x_1^m} \right\rangle \left[(-1)^{k-2} a^{3(k+1)-2} + (-1)^{k-1} a^{3(k+1)-6} \right.$$
$$\left. + (-1)^k a^{3(k+1)-10} + \cdots + (-1)^{2k-2} a^{-(k+1)+6} - a^{-(k+1)-2} \right]$$

$$= \left\langle \widehat{x_1^m} \right\rangle \left\langle \widehat{x_1^{k+1}} \right\rangle.$$

This completes the proof.

Corollary 3.9

$$\left\langle \widehat{x_1^m x_2^n} \right\rangle = \left\langle \widehat{x_1^n x_2^m} \right\rangle.$$

Proof. It is obvious: $\left\langle \widehat{x_1^m x_2^n} \right\rangle = \left\langle \widehat{x_1^m} \right\rangle \left\langle \widehat{x_1^n} \right\rangle = \left\langle \widehat{x_1^n} \right\rangle \left\langle \widehat{x_1^m} \right\rangle = \left\langle \widehat{x_1^n x_2^m} \right\rangle.$

Corollary 3.10

$$\deg \left\langle \widehat{x_1^m x_2^n} \right\rangle = 3(m+n)-4$$

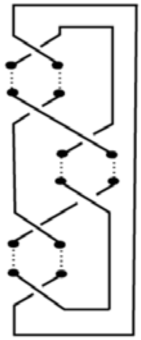
and

$$\operatorname{span} \left\langle \widehat{x_1^m x_2^n} \right\rangle = 4(m+n).$$

Proof. The result follows immediately from Theorem 3.8 as $\deg \left\langle \widehat{x_1^n} \right\rangle = 3n-2$ and span $\left\langle \widehat{x_1^n} \right\rangle = -n-2.$

For the following, let us fix the notation L_{abc} for the link with the understanding that the link contains a, b,

and c crossings of type x_1, x_2, and x_1, respectively, and that $L_{abc} \neq \widehat{x_1^a x_2^b}\, x_1^a$.



L_{abc}

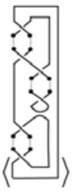
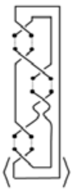
Proposition 3.11 The Kauffman bracket of the link L_{abc} is

$$\langle L_{abc} \rangle = \left\langle \widehat{x_1^a} \right\rangle \left\langle \widehat{x_1^b} \right\rangle \left\langle \widehat{x_1^c} \right\rangle + \left(a^{-b+2} + a^{-b-2}\right) \left\langle \widehat{x_1^a} \right\rangle \left\langle \widehat{x_1^c} \right\rangle + a^{-b} \left\langle \widehat{x_1^{a+c}} \right\rangle$$

Proof. We prove it by induction on b:
For $b = 1$, we have

$$\begin{aligned}\left\langle L_{a1c}\right\rangle &= \left\langle \widehat{x_1^a}\right\rangle\left\langle \widehat{x_1^1}\right\rangle\left\langle \widehat{x_1^c}\right\rangle + \left(a+a^{-3}\right)\left\langle \widehat{x_1^a}\right\rangle\left\langle \widehat{x_1^c}\right\rangle + a^{-1}\left\langle \widehat{x_1^{a+c}}\right\rangle \\ &= a^{-3}\left\langle \widehat{x_1^a}\right\rangle\left\langle \widehat{x_1^c}\right\rangle + \left(a+a^{-3}\right)\left\langle \widehat{x_1^a}\right\rangle\left\langle \widehat{x_1^c}\right\rangle + a^{-1}\left\langle \widehat{x_1^{a+c}}\right\rangle \\ &= a\left\langle \widehat{x_1^a}\right\rangle\left\langle \widehat{x_1^c}\right\rangle + a^{-1}\left\langle \widehat{x_1^{a+c}}\right\rangle.\end{aligned}$$

Now, with the assumption that the result holds for an arbitrary k, we have $b = k + 1$

$\left\langle L_{a(k+1)c}\right\rangle = a\langle$  $\rangle + a^{-1}\langle$  $\rangle$

$$\begin{aligned} &= a\left\langle \widehat{x_1^a}\#U_k\#\widehat{x_1^c}\right\rangle + a^{-1}\left\langle \widehat{x_1^a x_2^k x_1^c}\right\rangle \\ &= a\left\langle \widehat{x_1^a}\#U_k\right\rangle\left\langle \widehat{x_1^c}\right\rangle + a^{-1}\left\langle \widehat{x_1^a x_2^k x_1^c}\right\rangle \\ &= a(-1)^k a^{3k}\left\langle \widehat{x_1^a}\right\rangle\left\langle \widehat{x_1^c}\right\rangle + a^{-1}\left(\left\langle \widehat{x_1^a}\right\rangle\left\langle \widehat{x_1^k}\right\rangle\left\langle \widehat{x_1^c}\right\rangle + \left(a^{-k+2}+a^{-k-2}\right)\left\langle \widehat{x_1^a}\right\rangle\left\langle \widehat{x_1^c}\right\rangle + a^{-k}\left\langle \widehat{x_1^{a+c}}\right\rangle\right) \\ &= (-1)^k a^{3k+1}\left\langle \widehat{x_1^a}\right\rangle\left\langle \widehat{x_1^c}\right\rangle + a^{-1}\left\langle \widehat{x_1^a}\right\rangle\left\langle \widehat{x_1^k}\right\rangle\left\langle \widehat{x_1^c}\right\rangle + \left(a^{-k+1}+a^{-k-3}\right)\left\langle \widehat{x_1^a}\right\rangle\left\langle \widehat{x_1^c}\right\rangle + a^{-k-1}\left\langle \widehat{x_1^{a+c}}\right\rangle \\ &= \left((-1)^k a^{3k+1} + (-1)^{k+1} a^{3k-3} + \cdots - a^{-k+5} - a^{-k-3}\right)\left\langle \widehat{x_1^a}\right\rangle\left\langle \widehat{x_1^c}\right\rangle \\ &\quad + \left(a^{-k-1+2}+a^{-k-1-2}\right)\left\langle \widehat{x_1^a}\right\rangle\left\langle \widehat{x_1^c}\right\rangle + a^{-k-1}\left\langle \widehat{x_1^{a+c}}\right\rangle \\ &= \left\langle \widehat{x_1^a}\right\rangle\left\langle \widehat{x_1^{k+1}}\right\rangle\left\langle \widehat{x_1^c}\right\rangle + \left(a^{-(k+1)+2}+a^{-(k+1)-2}\right)\left\langle \widehat{x_1^a}\right\rangle\left\langle \widehat{x_1^c}\right\rangle + a^{-(k+1)}\left\langle \widehat{x_1^{a+c}}\right\rangle\end{aligned}$$

as required.

Proposition 3.12 The Kauffman bracket of the link $\widehat{x_1^a x_2^b x_1^c x_2^d}$ is

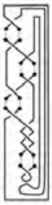
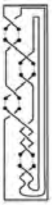
$$\left\langle \widehat{x_1^a x_2^b x_1^c x_2^d} \right\rangle = \left(\sum_{i=1}^{d} (-1)^{d+i} a^{3d-4i+2} \right) \left\langle \widehat{x_1^a} \right\rangle \left\langle \widehat{x_1^b} \right\rangle \left\langle \widehat{x_1^c} \right\rangle$$
$$+ \left((-1)^{d+1} a^{-b+3d} + a^{-b-d} \right) \left\langle \widehat{x_1^a} \right\rangle \left\langle \widehat{x_1^c} \right\rangle$$
$$+ \left(\sum_{i=1}^{d} (-1)^{d+i} a^{3d-4i+2-b} + a^{-d} \left\langle \widehat{x_1^b} \right\rangle \right) \left\langle \widehat{x_1^{a+c}} \right\rangle.$$

Proof. We prove it by induction on *d*:

For *d* = 1, we have

$$\left\langle \widehat{x_1^a x_2^b x_1^c x_2^1} \right\rangle = \left(\sum_{i=1}^{1} (-1)^{d+i} a^{3-4i+2} \right) \left\langle \widehat{x_1^a} \right\rangle \left\langle \widehat{x_1^b} \right\rangle \left\langle \widehat{x_1^c} \right\rangle + \left((-1)^{d+1} a^{-b+3} + a^{-b-1} \right) \left\langle \widehat{x_1^a} \right\rangle \left\langle \widehat{x_1^c} \right\rangle$$
$$+ \left(\sum_{i=1}^{1} (-1)^{1+i} a^{3-4i+2-b} + a^{-1} \left\langle \widehat{x_1^b} \right\rangle \right) \left\langle \widehat{x_1^{a+c}} \right\rangle$$
$$= a \left\langle \widehat{x_1^a} \right\rangle \left\langle \widehat{x_1^b} \right\rangle \left\langle \widehat{x_1^c} \right\rangle + \left(a^{-b+3} + a^{-b-1} \right) \left\langle \widehat{x_1^a} \right\rangle \left\langle \widehat{x_1^c} \right\rangle$$
$$+ \left(a^{-b+1} + a^{-1} \left\langle \widehat{x_1^b} \right\rangle \right) \left\langle \widehat{x_1^{a+c}} \right\rangle.$$

Now, with the assumption that the result holds for *d* = *k*, we have

$\left\langle \widehat{x_1^a x_2^b x_1^c x_2^{k+1}} \right\rangle = a\langle$  $\rangle + a^{-1}\langle$  $\rangle = a(L_{abc}\#U_k) + a^{-1} \left\langle \widehat{x_1^a x_2^b x_1^c x_2^k} \right\rangle$

$$= a\left[(-1)^k a^{3k} \left(\left\langle \widehat{x_1^a} \right\rangle \left\langle \widehat{x_1^b} \right\rangle \left\langle \widehat{x_1^c} \right\rangle + \left(a^{-b+2} + a^{-b-2} \right) \left\langle \widehat{x_1^a} \right\rangle \left\langle \widehat{x_1^c} \right\rangle + a^{-b} \left\langle \widehat{x_1^{a+c}} \right\rangle \right) \right]$$
$$+ a^{-1} \left[\left(\sum_{i=1}^{k} (-1)^{k+i} a^{3k-4i+2} \right) \left\langle \widehat{x_1^a} \right\rangle \left\langle \widehat{x_1^b} \right\rangle \left\langle \widehat{x_1^c} \right\rangle + \left((-1)^{k+1} a^{-b+3k} + a^{-b-k} \right) \left\langle \widehat{x_1^a} \right\rangle \left\langle \widehat{x_1^c} \right\rangle \right.$$
$$\left. + \left(\sum_{i=1}^{k} (-1)^{k+i} a^{3k-4i+2-b} + a^{-k} \left\langle \widehat{x_1^b} \right\rangle \right) \left\langle \widehat{x_1^{a+c}} \right\rangle \right]$$

$$\begin{aligned}
&=(-1)^{k}a^{3k+1}\left\langle\widehat{x_1^a}\right\rangle\left\langle\widehat{x_1^b}\right\rangle\left\langle\widehat{x_1^c}\right\rangle+\left((-1)^{k}a^{3k+3-b}+(-1)^{k}a^{3k-1-b}\right)\left\langle\widehat{x_1^a}\right\rangle\left\langle\widehat{x_1^c}\right\rangle\\
&\quad+(-1)^{k}a^{3k+1-b}+\sum_{i=1}^{k}(-1)^{k+i}a^{3k-4i+1}\left\langle\widehat{x_1^a}\right\rangle\left\langle\widehat{x_1^b}\right\rangle\left\langle\widehat{x_1^c}\right\rangle\\
&\quad+\left((-1)^{k+1}a^{3k-1-b}+a^{-k-1-b}\right)\left\langle\widehat{x_1^a}\right\rangle\left\langle\widehat{x_1^c}\right\rangle+\left(\sum_{i=1}^{k}(-1)^{k+i}a^{3k-4i+1-b}+a^{-k-1}\left\langle\widehat{x_1^b}\right\rangle\right)\left\langle\widehat{x_1^{a+c}}\right\rangle\\
&=\left(\sum_{i=1}^{k+1}(-1)^{(k+1)+i}a^{3(k+1)-4i+2}\right)\left\langle\widehat{x_1^a}\right\rangle\left\langle\widehat{x_1^b}\right\rangle\left\langle\widehat{x_1^c}\right\rangle+\left((-1)^{(k+1)+1}a^{-b+3(k+1)}+a^{-b-(k+1)}\right)\left\langle\widehat{x_1^a}\right\rangle\left\langle\widehat{x_1^c}\right\rangle\\
&\quad+\left(\sum_{i=1}^{k+1}(-1)^{(k+1)+i}a^{3(k+1)-4i+2-b}+a^{-(k+1)}\left\langle\widehat{x_1^b}\right\rangle\right)\left\langle\widehat{x_1^{a+c}}\right\rangle,
\end{aligned}$$

Abdul Rauf Nizami, Mobeen Munir, Umer Saleem, Ansa Ramzan as was required.

REFERENCES

1. Kauffman, L.H. (1987) State Models and the Jones Polynomial. Topology, 26, 395-407. http://dx.doi.org/10.1016/0040-9383(87)90009-7
2. Jaeger, F. (1990) A Combinatorial Model for the Homy Polynomial. European Journal of Combinatorics, 11, 549-555.
3. Reshetikhin, N.Y. (1988) Quantized Universal Enveloping Algebras, the Yang-Baxter Equation and Invariants of Links, I and II. LOMI Reprints E-4-87 and E-17-87, Steklov Institute, Leningrad, USSR.
4. Reidemeister, K. (1948) Knot Theory. Chelsea Publ and Co., New York.
5. Artin, E. (1925) Theorie der Zöpfe. Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg, 4, 27-72. http://dx.doi.org/10.1007/BF02950718
6. Artin, E. (1947) Theory of Braids. Annals of Mathematics, 48, 101-126. http://dx.doi.org/10.2307/1969218
7. Birman, J.S. (1974) Braids, Links, and Mapping Class Groups. Princeton University Press, Princeton.
8. Manturov, V.O. (2004) Knot Theory. Chapman and Hall/CRC, Boca Raton. http://dx.doi.org/10.1201/9780203402849
9. Murasugi, K. (1996) Knot Theory and Its Applications. BirkhäUser, Boston.
10. Alexander, J. (1923) Topological Invariants of Knots and Links. Transactions of the American Mathematical Society, 20, 275-306.
11. Adams, C.C. (1994) The Knot Book. W H Freeman and Company, New York.

Chapter

A NOVEL MULTIWAY SPLITS DECISION TREE FOR MULTIPLE TYPES OF DATA

12

Zhenyu Liu,[1,2] Tao Wen,[1,2] Wei Sun,[2] and Qilong Zhang[1]

[1]College of Computer Science and Engineering, Northeastern University, Shenyang 110819, China

[2]Department of Computer Science and Technology, Dalian Neusoft University of Information, Dalian, Liaoning, China

ABSTRACT

Classical decision trees such as C4.5 and CART partition the feature space using axis-parallel splits. Oblique decision trees use the oblique splits based on linear combinations of features to potentially simplify the boundary structure. Although oblique decision trees have higher generalization accuracy, most oblique split methods are not directly conducive to the categorical data and are computationally expensive. In this paper, we propose a multiway splits decision tree (MSDT) algorithm, which adopts feature weighting and

Citation: Zhenyu Liu, Tao Wen, Wei Sun, and Qilong Zhang, "A Novel Multiway Splits Decision Tree for Multiple Types of Data", Mathematical Problems in Engineering, volume 2020, article ID 7870534, https://doi.org/10.1155/2020/7870534.

clustering. This method can combine multiple numerical features, multiple categorical features, or multiple mixed features. Experimental results show that MSDT has excellent performance for multiple types of data.

INTRODUCTION

Despite the great success of deep neural network (DNN) model in image processing, speech recognition, and other fields in recent years, decision trees have competitive performance compared to DNN scheme, such as the advantage of interpretability, less parameters, and good robustness to noise, and can be applied to large-scale data sets with less computational cost. Therefore, the decision tree is still one of the hotspots in the field of machine learning today [1–3]. The research mainly focused on the construction method of decision trees, split criterion [4], decision trees ensemble [5, 6], mixing with other learners [7–9], decision trees for semisupervised learning [10], and so on.

Despite practical success, the optimal construction of decision trees has been theoretically proven to be NP-complete [11]. In order to avoid the local optimal solution, some researchers adopted evolutionary algorithms to build decision trees [12–14]. However, due to the time complexity, the most popular algorithms, such as ID3 [15], C4.5 [16], and CART [17], and their various modifications [18] are greedy by nature and construct the decision tree in a top-down, recursive manner. Besides, they only act on one dimension at a time and thus result in an axis-parallel split. In the induction of decision tree, if the candidate features are numerical, a suitable cut point needs to be searched. Instances in the training set are divided into the left node or the right node according to the following formula:

$$x_l \leq \theta_l, \qquad (1)$$

where x_l denotes the value of the instance on the feature A_l and θ_l is the cut point.

Axis-parallel trees have the advantages of fast induction and strong comprehensibility. However, in the case of highly correlated features, a very bad situation may arise. Figure 1 gives an illustration. The parallel splits will be carried out many times with a stair case-like structure, which leads to the complexity of the decision tree structure.

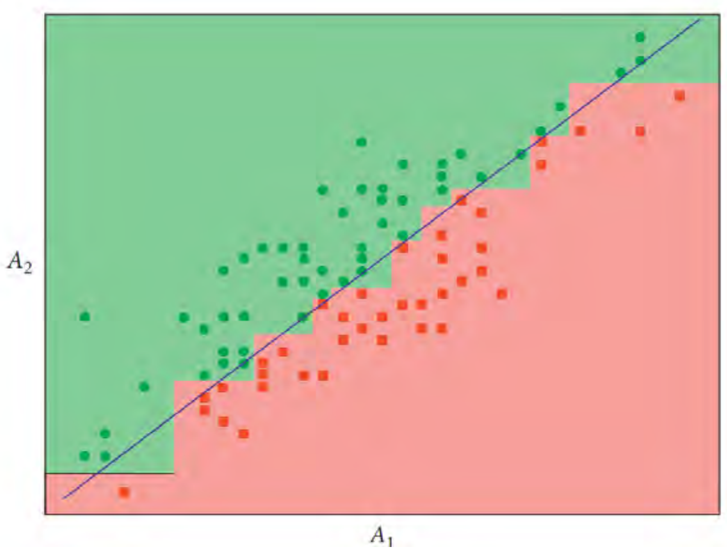


Figure 1. Axis-parallel splits and oblique split.

To solve the problem of parallel decision trees, some researchers introduced oblique decision trees. In such oblique decision trees, the nonleaf node tests the linear combination of features, i.e.,

$$\sum_{i=1}^{p} a_l x_l \leq \theta, \qquad (2)$$

where a_l represents the coefficient for the lth feature, θ is the threshold, and p is the number of features. In Figure 1, the instances of the two classes can be completely separated by one oblique split. Therefore, it is generally believed that the oblique splits can often produce smaller decision trees and better generalization performance for the same data.

It is much more difficult to search the optimal oblique hyperplanes than the optimal axis-parallel hyperplanes. To solve this problem, numerous techniques have been applied, for example, hill-climbing [17], simulated annealing [19], and genetic algorithm [20]. Among them, a large amount of research work has been done on reducing the risk of falling into local optimal solution, such as Simulated Annealing Decision Tree (SADT) [19], which used the simulated annealing algorithm; OC1 [21] method combined the ideas of CART-LC [17] and SADT.

In searching oblique hyperplanes, thousands of candidates have been tried in both simulated annealing algorithm and genetic algorithm, resulting in low time efficiency. So many researchers used linear discriminant analysis, linear regression, perceptron, SVM, and other methods to find suitable oblique hyperplanes. Fisher's decision tree (FDT) [22] takes advantage of dimensionality reduction of Fisher's linear discriminant and uses the decomposition strategy of decision trees to come up with an oblique decision tree. FDT is only applicable to binary classification problems. Based on ADTree [23], Hong et al. [24] proposed the multivariate ADTree. Paper [24] presented and discussed the different variations of ADTree (Fisher's ADTree, Sparse ADTree, and Regularized Logistic ADTree). Wickramarachchi et al. [25] explored a decision tree algorithm (HHCART). HHCART uses a series of Householder matrices to reflect the training data during tree construction. Shah and Sastry [26] defined separability of instances as the split criterion that optimized their evaluation function at each node and then presented the Alopex Perceptron Decision Tree algorithm for learning a decision tree. Menze et al. [27] presented an oblique tree forest method, which used LDA and ridge regression to conduct oblique splits.

In the above oblique methods, the trees with fewer nodes and better accuracy can be obtained. However, there are also some deficiencies, mainly including three aspects.

Inability to Directly Employ the Methods for Categorical Data

The oblique splits use the linear combination of features. Therefore, the categorical features need to be converted into one or more numerical features [28]. This transformation may bring new biases to the classification problems, thus reducing the generalization ability of the models.

High Time Cost

The oblique splits always require complex matrix calculation when using linear discriminant analysis, ridge regression, or other methods. Although these methods are more efficient than simulated annealing and genetic algorithm, they still pay more cost than the axis-parallel methods, such as C4.5.

Some Methods Cannot Be Suitable for Multiclassification Problems

Generally, the oblique split methods conduct the binary splits. Although the binary tree can also be directly used for multiclassification problems, some binary splits rely on class label, such as FDA, original SVM, etc., which makes some algorithms like FDT in [22] limited to binary classification problems. In addition, some models need to convert multiclassification problems into binary ones [7].

In order to overcome the above shortcomings, this paper proposes a multiway splits decision tree for multiple types of data (numerical, categorical, and mixed data). The specific characteristics of this method are as follows:

- Categorical features are handled directly.
- The time complexity is similar to that of the axis-parallel split algorithms.
- It is not necessary to convert multiclassification problems into binary ones by using the multiway splits directly.

The remainder of the paper is organized as follows. In Section 2, we review RELIEF-F and *k*-means algorithms briefly. Section 3 presents our algorithm and discusses its time complexity. Section 4 presents and analyzes the compared experimental results with other decision trees. The last section gives the conclusion of this paper.

PRELIMINARIES

The proposed decision tree method needs to weight the features by RELIEF-F algorithm and split the nodes by the weighted *k*-means algorithm. Therefore, this section reviews the two algorithms and their variations.

RELIEF-F Algorithms

The RELIEF algorithm [29] is popular to feature selection. It estimates the weights of features according to the correlation between individual feature and class label. RELIEF randomly samples an instance R from the training set and then searches its two nearest neighbors H and M: H is from the same class (called near Hit) and M is from different class (called near Miss). If the distance between R and H on feature A is less than the distance between R and M, RELIEF will increase A’s weight. On the contrary, RELIEF will decrease the weight.

In fact, RELIEF’s estimate W(A) of feature A is an approximation of the following difference of probabilities:

$$W(A) = P(\text{different value of } A|\text{nearest instance from different class}) - P(\text{different value of } A|\text{nearest instance from same class}), \tag{3}$$

where P(·|·) represents the conditional probability.

RELIEF algorithm only deals with binary classification problems. Kononeill addressed an algorithm called RELIEF-F for multiclassification problems [30].

The algorithm picks m instances. For each instance R, its k_{nn} nearest neighbors are searched in each class

The weight W(A) is calculated as follows:

$$W(A) = W(A) - \sum_{j=1}^{k_{\text{nn}}} \text{diff}\frac{(A, R, H_j)}{(mk_{\text{nn}})} + \sum_{T \notin \text{class}(R)} \frac{\left[p(T)/1 - p(\text{class}(R)) \sum_{j=1}^{k_{\text{nn}}} \text{diff}\left(A, R, M_j(T)\right)\right]}{(mk_{\text{nn}})}, \tag{4}$$

where represents the proportion of class instances to the total instances and represents the th nearest neighbor to in class . calculates the difference between two instances R_1 and R_2 on the feature as follows:

where p(T) represents the proportion of class T instances to the total instances and $M_j(T)$ represents the jth nearest neighbor to R in class T. diff(A, R_1, R_2) calculates the difference between two instances R_1 and R_2 on the feature A as follows:

$$\text{diff}(A, R_1, R_2) = \begin{cases} \dfrac{|R_1[A] - R_2[A]|}{\max(A) - \min(A)}, & \text{if } A \text{ is numerical,} \\ 0, & \text{if } A \text{ is categorical And } R_1[A] = R_2[A], \\ 1, & \text{if } A \text{ is categorical And } R_1[A] \neq R_2[A]. \end{cases} \tag{5}$$

k-Means, *k*-Modes, and *k*-Prototypes

The *k*-means is widely used in real world applications due to its simplicity and efficiency.

Let D be a set of n instances. D is characterized by a set of p features and needs to be clustered into k clusters $C_1, C_2, \ldots, C_k$. First, randomly pick some instances as the centers of the initial k clusters $\mu_1, \mu_2, \ldots, \mu_k$, and then calculate the cluster label for each instance x_i as follows:

$$\text{label}_i = \arg\min_{1 \le j \le k} \left\| \mathbf{x}_i - \boldsymbol{\mu}_j \right\|. \tag{6}$$

After all the instances are partitioned, each cluster center will be updated by the following formula:

$$\boldsymbol{\mu}_\mathbf{j} = \frac{1}{n_j} \sum_{i=1}^{n_j} \mathbf{x}_\mathbf{i}, \quad \mathbf{x}_\mathbf{i} \in C_j. \tag{7}$$

Repeat formulas (6) and (7) until the variable E in formula (8) converges to the local optimal solution or the preset number of iterations is reached:

$$E = \sum_{j=1}^{k} \sum_{i=1}^{n_j} \left\| \mathbf{x}_\mathbf{i} - \boldsymbol{\mu}_\mathbf{j} \right\|^2, \quad \mathbf{x}_\mathbf{i} \in C_j. \tag{8}$$

However, the classical *k*-means is only worked on the numerical data. The *k*-modes and *k*-prototypes are variants of *k*-means for categorical and mixed data, respectively [31]. When *k*-modes processes the categorical variables, the center of each cluster is represented by modes. When calculating the distance between instance and cluster center, the distance on each feature is calculated by formula (5) and then accumulated.

It is straightforward to integrate the k-means and kmodes into the k-prototypes. dis(x_i, μ_j) is the distance between instance x_i and cluster center μ_j as follows:

$$\text{dis}\left(\mathbf{x}_\mathbf{i}, \boldsymbol{\mu}_\mathbf{j}\right) = (1 - \gamma) \cdot \text{dis\_}n\left(\mathbf{x}_\mathbf{i}, \boldsymbol{\mu}_\mathbf{j}\right) + \gamma \cdot \text{dis\_}c\left(\mathbf{x}_\mathbf{i}, \boldsymbol{\mu}_\mathbf{j}\right), \tag{9}$$

where dis_n(x_i, μ_j) represents the distance on the numerical variables and dis_c(x_i, μ_j) represents the distance on the categorical variables, respectively. γ is used to adjust the proportion of dis_n(x_i, μ_j) and dis_c(x_i, μ_j), $\gamma \in [0, 1]$.

OUR PROPOSED ALGORITHM

Our proposed MSDT has three differences with most oblique methods: (i) MSDT does not use greedy methods to pursue maximum impurity reduction, (ii) MSDT uses a combination of multiple variables to do multiway splits for nonleaf nodes, and (iii) MSDT treats categorical features in a similar way to numerical features.

Multiway Splits

Most oblique methods conduct binary splits, while the proposed algorithm performs multiway splits; that is, in one split, multiple hyperplanes are generated simultaneously, and the feature space is divided into several disjoint regions. Ho [32] categorized the linear split methods into three types, axis-parallel linear splits, oblique linear splits, and piecewise linear splits, while our method falls into the third. Piecewise linear split methods find k anchors in feature space, and each instance is clustered according to the nearest neighbor anchor. Figure 2 shows the 5-way splits of the two-dimensional feature space.

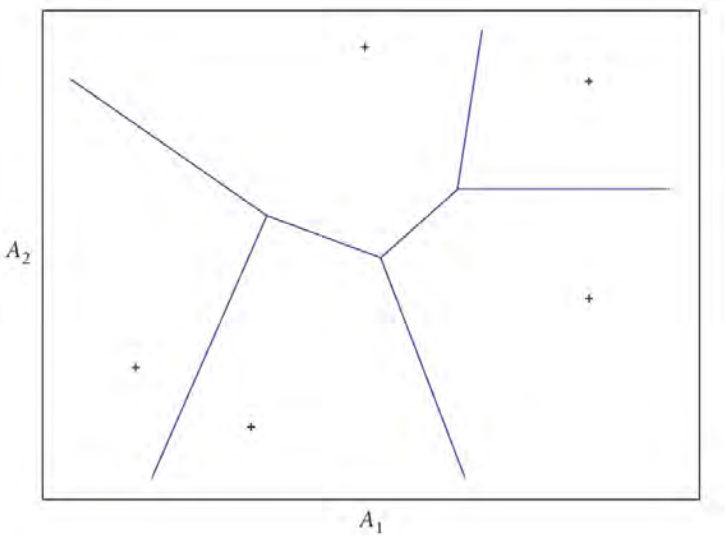


Figure 2. Piecewise linear splits.

Location of Anchor

Finding suitable split hyperplanes is the key problem in most decision tree induction algorithms. Under piecewise linear splits, the problem of finding appropriate hyperplanes is equivalent to that of finding appropriate anchors. Usually, anchor selection can use the class centroids, or cluster centers generated by some clustering algorithms. In MSDT, we first use RELIEF-F to weight features and then use *k*-means with weighted distance to cluster instances.

Why Do We Use k-Means?

If the instances are linearly separable, it is obviously more efficient to use simply the class centroids than cluster centers as anchors. However, when the instances of some classes are distributed in different regions of the feature space, the class centroids may no longer be suitable for being anchors. For example, in Figure 3, the circular instances are distributed in two different areas. If the solid line that is perpendicular to the line between the two class centroids is used to separate the instances, the effect is obviously not satisfactory. The instances in Figure 3 are obviously distributed into two clusters. If the instances are divided by the dotted line that is a perpendicular bisector of the two cluster centers, at least the circular instances on the right side of the figure can be distinguished.

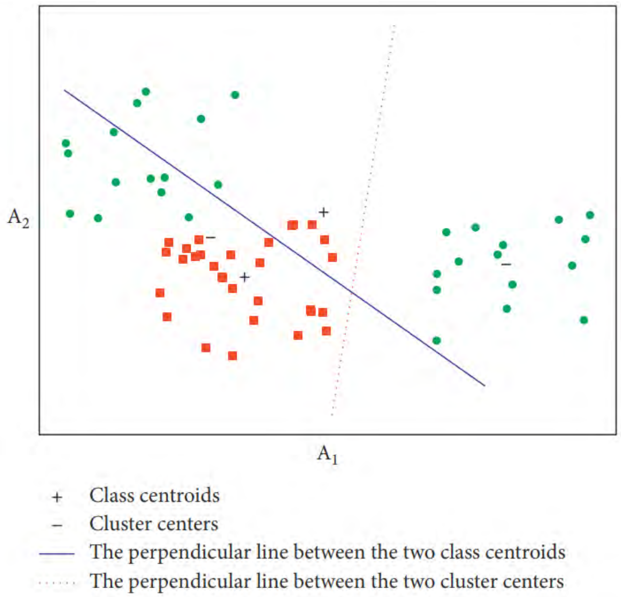


Figure 3. Split results of class centroids as anchors and cluster centers as anchors.

The split method proposed is based on clustering assumption. The clustering assumption states that the samples belonging to the same cluster belong to the same class. *k*-means methods partition instances according to some (dis)similarity measures; hence, the leaf nodes of MSDT can be regarded as some prototypes, and the class of a test instance depends on which prototype the instance is more similar to.

The univariate decision trees can produce a comprehensible classification mode, due to the knowledge representation method—a decision tree is a graphical representation and can be easily converted into a set of rules written in a natural language. Some researchers believe that multivariate decision trees are not able to convert into the comprehensible rules. The other researchers think that multivariate tree with fewer nodes is easy to understand. MSDT is easy to understand due to two reasons. One is that MSDT has fewer nodes compared to univariate decision trees. The other one is that the similarity with the prototype is easy to understand by the users and it can replace the rules generated by the univariate decision tree.

Why Do We Weight Features?

The original *k*-means is an unsupervised clustering algorithm, which is suitable for unlabeled data. And the optimization goal is to minimize (8). The goal of split is to reduce the class impurity of current node as much as possible. Note that the two goals are not the same. Therefore, we estimate the correlations between features and label to weight features. When calculating the distance from an instance to a cluster center, we give a larger weight to the feature strongly related to the label that enlarges the contribution of the feature to the distance. Otherwise, we give a smaller weight that reduces the contribution of the uncorrelated feature to the distance. In this way, the optimization goal of *k*-means algorithm is close to that of node split.

Figure 4 shows an example to illustrate the effectiveness of feature weighting. The solid line comes from unweighted features, and the dotted line comes from weighted features when the weight of A_1 is 0.05 and the weight of A_2 is 0.95. It is obvious that some instances have been corrected.

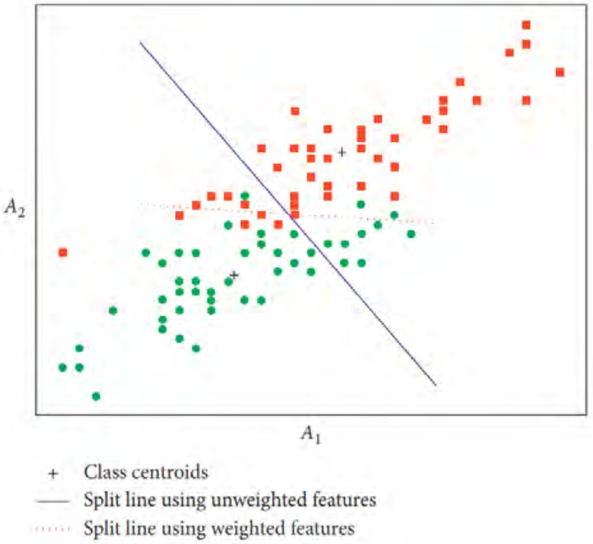


Figure 4. Split results of unweighted and weighted features.

To further illustrate the role of feature weighting, we use dataset *iris* to carry out a simple experiment: 150 samples of dataset *iris* come from three classes, and each class has 50 samples. We directly use *k*-means algorithm to cluster and obtain 10 misclassified samples. The specific results are shown in Table 1.

Table 1. Split results of unweighted features for the *iris* dataset.

	Setosa	**Versicolor**	**Virginica**
Child node 1	50	0	0
Child node 2	0	44	4
Child node 3	0	6	46

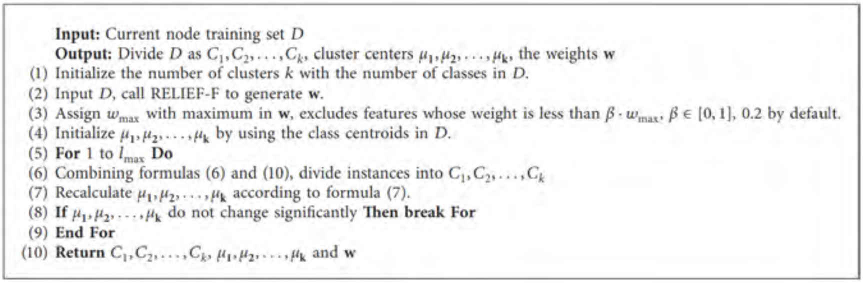
Then, we use the RELIEF-F algorithm to calculate the weights of four features, which are 0.09, 0.14, 0.34, and 0.39, respectively. In the process of *k*-means clustering, the distances between instances and cluster centers are calculated by (10), where *p* indicates the number of features and w_l indicates the weight of the *l*th feature. We obtain 6 misclassified samples, and the specific results are shown in Table 2.

$$\left\|\mathbf{x_i} - \boldsymbol{\mu}_\mathbf{j}\right\| = \sqrt{\sum_{l=1}^{p} w_l \cdot \left(x_{i,l} - \mu_{j,l}\right)^2}. \qquad (10)$$

Table 2. Split results of weighted features for the *iris* dataset.

	Setosa	Versicolor	Virginica
Child node 1	50	0	0
Child node 2	0	48	4
Child node 3	0	2	46

Our proposed split method is shown in Algorithm 1, which will be used to split nodes for numerical data.



```
     Input: Current node training set $D$
     Output: Divide $D$ as $C_1, C_2, \ldots, C_k$, cluster centers $\mu_1, \mu_2, \ldots, \mu_k$, the weights $\mathbf{w}$
 (1) Initialize the number of clusters $k$ with the number of classes in $D$.
 (2) Input $D$, call RELIEF-F to generate $\mathbf{w}$.
 (3) Assign $w_{\max}$ with maximum in $\mathbf{w}$, excludes features whose weight is less than $\beta \cdot w_{\max}$, $\beta \in [0, 1]$, 0.2 by default.
 (4) Initialize $\mu_1, \mu_2, \ldots, \mu_k$ by using the class centroids in $D$.
 (5) For 1 to $l_{\max}$ Do
 (6) Combining formulas (6) and (10), divide instances into $C_1, C_2, \ldots, C_k$
 (7) Recalculate $\mu_1, \mu_2, \ldots, \mu_k$ according to formula (7).
 (8) If $\mu_1, \mu_2, \ldots, \mu_k$ do not change significantly Then break For
 (9) End For
(10) Return $C_1, C_2, \ldots, C_k$, $\mu_1, \mu_2, \ldots, \mu_k$ and $\mathbf{w}$
```

Algorithm 1. Multi_split.

In the fifth step of Algorithm 1, 1_{max} represents the maximum number of iterations. In the experiments, we set it to 6 by default. The reason for setting such a small value is mainly to consider the time efficiency of the algorithm. In addition, the purpose of clustering is to split nodes. Even if the clustering algorithm does not converge, the partition results can still be accepted.

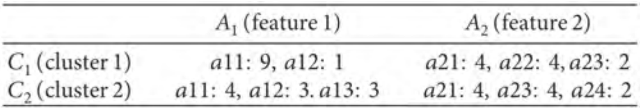
Categorical Feature

As mentioned in the previous subsection, the split method can be directly applied to numerical features. For categorical features, RELIEF-F algorithm can still be used to weight features. However, in the process of clustering, the representation of cluster center and the distance from instance to cluster center need to be redefined.

The *k*-modes extends the *k*-means by replacing the means of numerical variables with the modes of the categorical variables. Yet it is less precise to calculate the distance. What is more, choosing different modes may cause opposite conclusion while there are several modes for a feature.

Here is an example. Suppose there are two clusters C_1 and C_2 described by two categorical features A_1 and A_2, and each cluster contains 10 instances as is shown in Table 3. The modes of C_1 and C_2 for A_1 are a11, which makes A_1 useless for distinguishing the distances between instances and the clusters. There are two modes for A_2 in C_1 and C_2, respectively. Suppose that there is an instance q = (a11, a21); if μ_1 = (a11, a21) is selected as the center of C_1 and μ_2 = (a11, a23) for C_2, distance between q and μ_1 is 0 and distance between q and μ_2 is 1; hence, q is nearer to C_1. If μ_1 = (a11, a22) is selected as the center of C_1 and μ_2 = (a11, a21) for C_2, distance between q and μ_1 is 1 and distance between q and μ_2 is 0; hence, q is nearer to C_2.

Table 3. The distribution of values.



	A_1 (feature 1)	A_2 (feature 2)
C_1 (cluster 1)	a11: 9, a12: 1	a21: 4, a22: 4, a23: 2
C_2 (cluster 2)	a11: 4, a12: 3. a13: 3	a21: 4, a23: 4, a24: 2

To avoid the less precision and the ambiguity of distance measure on the modes, we use the probability estimation of each categorical feature value to represent the cluster center and define a function to calculate the distance from instance to cluster center.

Let D be a set of categorical data described by p categorical features. Number of instances in D is n and instances are partitioned into k clusters. There are $d^{(i,j)}$ with different values $\omega_1, \omega_2, \ldots, \omega_{d\{i,j\}}$ for the lth feature A_l of the jth cluster C_j, $l \in \{1, 2, \ldots, p\}$, $j \in \{1, 2, \ldots, k\}$.

Definition 1. C_{j,x_l} represents the set of instances with value of x_l on the feature$A_l$inC_j,where $x_l \in \{\omega_1, \omega_2, \ldots, \omega_{d\{l,j\}}\} x_1 \in \{\omega_1, \omega_2, \ldots, \omega_{d\{l,j\}}\}$. The condition probability is estimated as follows:

$$P(x_l|j) = \frac{|C_{j,x_l}|}{|C_j|}, \qquad (11)$$

$S_{j,l}$ is the summary of all values of A_l in C_j, defined as follows:

$$S_{j,l} = \{P(\omega_1|j), P(\omega_2|j), \ldots, P(\omega_d|j)\}. \qquad (12)$$

Definition 2. The center of C_j is represented by the following vector:

$$\mu_j = (S_{j,1}, S_{j,2}, \ldots, S_{j,p}). \qquad (13)$$

Definition 3. diff(A_l, ω, $S_{j,l}$) represents the distance between value ω and $S_{j,l}$ for A_l:

$$\mathrm{diff}\left(A_l, \omega, S_{j,l}\right) = \begin{cases} 1 - P(\omega|j), & \text{if } \omega \in \{\omega_1, \omega_2, \ldots, \omega_d\}, \\ 1, & \text{others.} \end{cases} \tag{14}$$

Definition 4. dis_c(x_i, μ_j) represents the weighted distance between instance x_i and center μ_j:

$$\mathrm{dis\_}c\left(\mathbf{x_i}, \boldsymbol{\mu}_\mathbf{j}\right) = \sum_{l=1}^{p}\left(w_l \cdot \mathrm{diff}\left(A_l, x_{i,l}, S_{j,l}\right)\right). \tag{15}$$

According to formula (15), in the above example, the weights of two features are 1. The distances between instance q = (a11, a21) and two cluster centers (μ_1 and μ_2) in Table 3 are 0.7 = 0.1 + 0.6 and 1.2 = 0.6 + 0.6, respectively. It means that q is closer to C_1, which is in accordance with the human's intuition.

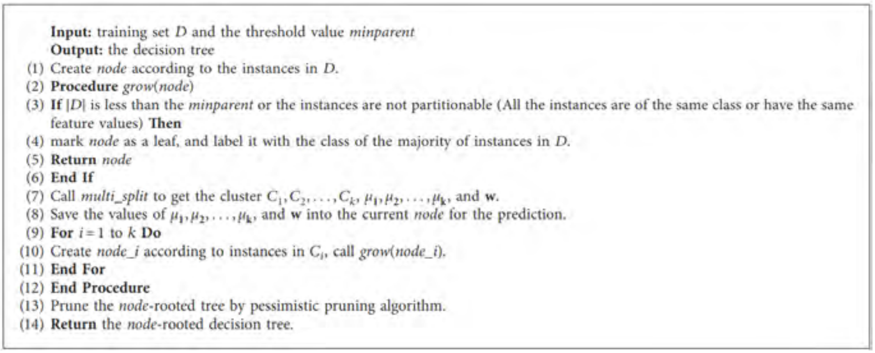
To cluster categorical data, we use formula (13) to replace formula (7) in step 4 and step 7 of Algorithm 1 and formula (15) to formula (10) in step 6.

Mixed Features Data

For mixed data, the vector of cluster center consists of two parts: one is the means of numerical features and the other is the vector as shown in (13). In this case, we use (9) to calculate the distance from instance to cluster center, where dis_n and dis_c are obtained by (10) and (15), respectively. As the ratio of numerical and categorical features differs by the datasets, we choose γ in (9) that makes the most reduction of GINI index, where γ ∈ {0.1, 0.2, . . . , 0.9}.

MSDT and Time Complexity Analysis

The multi_split function is prompted for node splits. Algorithm 2 describes the construction process of MSDT.



```
Input: training set $D$ and the threshold value minparent
Output: the decision tree
(1) Create node according to the instances in $D$.
(2) Procedure grow(node)
(3)     If $|D|$ is less than the minparent or the instances are not partitionable (All the instances are of the same class or have the same feature values) Then
(4)         mark node as a leaf, and label it with the class of the majority of instances in $D$.
(5)         Return node
(6)     End If
(7)     Call multi_split to get the cluster $C_1, C_2, \ldots, C_k$, $\mu_1, \mu_2, \ldots, \mu_k$, and $\mathbf{w}$.
(8)     Save the values of $\mu_1, \mu_2, \ldots, \mu_k$, and $\mathbf{w}$ into the current node for the prediction.
(9)     For $i = 1$ to $k$ Do
(10)        Create node_i according to instances in $C_i$, call grow(node_i).
(11)    End For
(12) End Procedure
(13) Prune the node-rooted tree by pessimistic pruning algorithm.
(14) Return the node-rooted decision tree.
```

Algorithm 2. MSDT.

In step 2 of Algorithm 1, RELIEF-F is used to get the weights. Time complexity of RELIEF-F is $O(m \cdot p \cdot n \cdot \log_2 k_m)$, where *p* is the feature number, *n* is the instance number, *m* is the sampling number, and k_{nn} is the nearest neighbor number. In this paper, *m* is set $\log_2$ n, k_{nn} is set 1, and $\log_2$, k_{nn} is negligible, so the time complexity of RELIEF-F in this paper is $O(p \cdot n \cdot \log_2 n)$.

Steps 4 to 9 of Algorithm 1 are the clustering process, and the time complexity is $O(I \cdot p \cdot n \cdot k)$, where *k* is cluster number and I is iteration number. When we use Algorithm 1 to split nodes, the max iterations Imax is 6; it means that time complexity may reach $O(6 \cdot p \cdot n \cdot k)$ in the worst case.

Considering the above two parts, the time complexity of Algorithm 1 is $O((6k + \log_2 n) \cdot p \cdot n)$. Compared with the time complexity of the classical axis-parallel splits, there is an extra *k*. When *k* is large, this algorithm is lower efficiency than the axis-parallel algorithms. Compared with binary splits, if the node numbers of the decision trees are the same, the operations in *k*-way splits are obviously less than in binary splits.

OC1 [21] is a classic oblique decision tree, whose time complexity is $O(p \cdot n^2 \cdot \log_2 n)$ in the worst case. In [25], the time complexities of HHCART(A) and HHCART(D) are $O((p + n \cdot \log_2 n) \cdot p^2 \cdot k)$ and $O((p + \log_2 n) \cdot p \cdot n \cdot k)$, respectively. In [22], the speed of FDT for splitting node is close to or even better than that of axis-parallel split method. The time complexity of this method is $O(p^2 \cdot n)$. Unfortunately, it can only be applied in binary classification problems.

In summary, when *k* is small, the efficiency of the proposed split method is close to classical axis-parallel split methods, and it is better than most oblique split methods.

EXPERIMENTS

In this section, we use experimental results to demonstrate the effectiveness and performance of our proposed algorithm. In the first part, the experiments are used to illustrate the effectiveness of clustering, feature weighting, and the novel distance calculation method for categorical feature. The second parts compare MSDT with classical decision trees and another two oblique trees. Finally, we use a larger dataset covertype to compare with two axis-parallel trees.

Datasets

As shown in Table 4, the 20 UCI datasets [33] are used to evaluate the proposed algorithm, where the number of instances, the number of classes, and feature types (numerical data 1–10, categorical data 11–15, and mixed data 16–20) are varied and are sufficiently representative to demonstrate the performance of MSDT. In column Features with $p_{(n)}$ & $p_{(c)}$, $p_{(n)}$ and $p_{(c)}$ is number of numerical features and categorical features, respectively. Abalone is treated as a 3-category classification problem (grouping classes 1–8, 9 and 10, and 11 on).

Table 4. Datasets.

No.	Name	Abb.	Instances	Features	Classes
1	Blood Transfusion Service Center	Blood	748	4	2
2	Glass identification	Glass	214	9	6
3	Image segmentation	Image	2130	19	7
4	Iris	Iris	150	4	3
5	Letter recognition	Letter	20000	16	26
6	Multiple features-mor	MF-mor	2000	6	10
7	Waveform database generator (version 1)	Wav1	5000	21	3
8	Waveform database generator (version 2)	Wav2	5000	40	3

9	Wine	Wine	178	13	3
10	Yeast	Yeast	1484	8	10
11	Balance scale	Balance	625	4	3
12	Car evaluation	Car	1728	6	4
13	Chess (King-Rook vs. King-Pawn)	Chess	3196	36	2
14	Hayes-Roth	Hayes	160	4	3
15	MONK's problems	MONK	432	6	2
16	Abalone	Abalone	4177	7&1	3
17	Contraceptive method choice	CMC	1473	2&7	3
18	Flags	Flags	194	10&19	8
19	Teaching assistant evaluation	TAE	151	1&4	3
20	Zoo	Zoo	101	15&1	7

Comparison of Different Piecewise Linear Split Methods

The piecewise linear split methods can be summarized as two steps. First, find appropriate anchors. Then, divide instances according to the nearest anchor. On the basis of this approach, our proposed algorithm is improved in three aspects: feature weighting, clustering, and special categorical feature processing. This section combines these three changes into multiple functions and compares the performances in multiple types of data. These functions are shown in Table 5.

Table 5. Split function description.

Function name	**Function description**		
	Feature weighting	**Clustering**	**Cluster center and distance of categorical feature**
Fun 0	✗	✗	*k*-modes
Fun 1	✓	✗	*k*-modes
Fun 2	✗	✓	*k*-modes
Fun 3	✓	✓	*k*-modes
Fun 4	✗	✗	Definitions 1–4
Fun 5	✓	✗	Definitions 1–4
Fun 6	✗	✓	Definitions 1–4
Fun 7	✓	✓	Definitions 1–4

The pessimistic pruning algorithm is adopted after the decision trees are generated. In addition, the average results of all experiments are obtained by 10 repetitions of 10-fold cross-validation.

Numerical Data

In terms of numerical data, the proposed algorithm uses weighted *k*-means to optimize cluster center position. In order to demonstrate the role of clustering and feature weighting, we implement four different node split functions to generate decision trees. Fun0 directly uses the center of each class as the anchor. Instances are divided according to the nearest anchor. Euclidean distance is used for distance calculation. Fun1 also uses the center of each class as the anchor. However, in the process of selecting the nearest anchor for each instance, RELIEF-F is firstly used to calculate the weight of each feature. Then remove features whose weights are less than 1/5 of the maximum. Finally, the distance is calculated according to formula (10). Fun2 uses the center of each class as the initial cluster center of *k*-means and the outputs of *k*-means as the partition results. Fun3 combines Fun1 with Fun2 and is our proposed algorithm for numerical data.

Table 6 gives the classification accuracy of the 4 functions in 10 datasets, and the best entry in each row is bolded. As can be seen, Fun3 gets the best accuracy on 9 of 10 datasets and the average improvement is 4.16% higher than Fun0. In particular, the accuracy increases by more than 8% on Glass and Letter. The average accuracy of 10 datasets shows that Fun1 is about 1.07% higher than Fun0, and Fun3 is 1.39% higher than Fun2. The results show that feature weighting improves the classification performance. Fun2 is about 2.77% higher than Fun0, and Fun3 is 3.09% higher than Fun1. The reason for improvement is using clustering.

Table 6. Comparison of the accuracy for different splitting functions on numerical data (%).

	Fun 0	**Fun 1**	**Fun 2**	**Fun 3**
Blood	76.44	76.78	77.02	**77.53**
Glass	61.68	63.46	66.92	**70.47**
Image	91.68	92.56	94.79	**96.01**
Iris	91.40	**96.00**	93.07	**96**
Letter	72.96	78.18	89.61	**91.77**
MF-mor	69.77	70.00	70.48	**71.13**

Wav1	**82.30**	81.76	80.92	**82.30**
Wav2	80.58	80.35	79.34	**80.65**
Wine	**95.73**	93.60	95.56	95.11
Yeast	52.63	53.19	55.14	**55.84**
Average	77.52	78.59	80.29	**81.68**

Categorical and Mixed Data

On the categorical and mixed data, we implement eight different split functions to generate decision trees. Fun0 directly uses the center of each class as the anchor. For categorical features, modes are used to replace the means of numerical features as the component of anchors. When calculating the distance between instance and anchor, the distance on each feature is calculated by formula (5) and then summed. The difference between Fun1 and Fun0 is that the weight of each feature is calculated by RELIEF-F. Then remove features whose weights are less than 1/5 of the maximum. The distance between the instance and the anchor is obtained by formula (15) and formula (9). Fun2 adds clustering process on Fun0. *k*-modes and *k*-prototypes are used for categorical data and mixed data, respectively. Fun3 combines Fun1 and Fun2. Fun4-7 corresponds to Fun0-3 respectively. On the categorical features, the calculation of cluster centers and distances adopts the method described in Section 3.3 (formulas (13) and (15), respectively). Fun7 is our proposed algorithm for categorical and mixed data.

Table 7 gives the classification accuracy of the 8 functions in 5 categorical datasets (*Balance*, *Car*, *Chess*, *Hayes*, and *MONK*) and 5 mixed datasets (*Abalone*, *CMC*, *Flags*, *TAE*, and *Zoo*), and the best entry in each row is bolded. Except for *CMC* and *Zoo*, Fun7 obtains the best accuracy, and the average is 11.77% higher than Fun0. As can be seen, Fun1 is better than Fun0, Fun3 is better than Fun2, Fun5 is better than Fun4, and Fun7 is better than Fun6. The average improvement is 5.37%. This is the contribution of feature weighting. It is shown that Fun2 is better than Fun0, Fun3 is better than Fun1, Fun6 is better than Fun4, and Fun7 is better than Fun5. The average improvement is 4.45%. The reason is the use of clustering. Meanwhile, we can see that Fun4-7 is averagely 1.48% better than Fun0-3. This improvement is the statistical distribution of feature values instead of modes.

Table 7. Comparison of the accuracy for different splitting functions on categorical and mixed data.

	Fun 0	**Fun 1**	**Fun 2**	**Fun 3**	**Fun 4**	**Fun 5**	**Fun6**	**Fun 7**
Balance	72.51	70.75	73.42	73.47	68.94	71.04	69.23	**76.45**
Car	74.88	86.11	76.89	89.48	81.92	83.76	92.73	**96.45**
Chess	80.05	95.66	90.42	95.96	71.51	93.68	92.62	**99.17**
Hayes	71.00	76.88	72.00	77.88	59.25	71.56	66.69	**79.75**
MONK	73.98	85.23	83.94	88.03	88.80	97.41	88.17	**99.63**
Abalone	59.12	59.44	61.33	61.46	50.56	55.66	61.66	**62.84**
CMC	45.19	44.31	**49.04**	45.66	43.13	45.10	44.60	47.95
Flags	47.01	57.06	50.31	60.05	47.73	59.12	48.76	**62.42**
TAE	47.75	49.07	59.47	53.97	56.56	58.41	61.85	**63.77**
Zoo	95.64	**96.73**	95.74	96.53	95.54	96.04	95.74	96.34
Average	66.71	72.12	71.26	74.25	66.39	73.18	72.21	**78.48**

Comparison with Other Decision Trees

In order to verify the performance of our proposed algorithm, we selected four decision trees: J48 (WEKA's implementation of C4.5), CART$_{SL}$ (scikit-learn's implementation of optimal CART), OC1, and HHCART(A). Since CART$_{SL}$ and OC1 do not support categorical features, we convert categorical features to numerical features using the One Hot method. The 10 repetitions of 10-fold cross-validation were used in our experiments to report the average accuracy and tree size of 5 classifiers on the test set. Friedman test and Nemenyi test will be used to analyze the algorithm difference.

The accuracy over the numerical datasets by each method is shown in Table 8. As can be seen, MSDT gets the best accuracy on 5 of 10 datasets and the average accuracy is 81.68%. It is 1.91%, 4.42%, 1.15%, and 1.81% higher than other four trees, respectively. In order to further demonstrate the differences of the classifiers, the Friedman test is used. We use the averages of the ranks of 5 classifiers on 10 datasets to calculate $F_F = 7.129032$. Here, with 5 algorithms and 10 datasets, F_F follows the F − distribution with 4 and 36 degrees of freedom, and the critical value is F(4, 36) = 2.634. So, we reject the null hypothesis; namely, there are significant differences among the five classifiers. Nemenyi method is used for post hoc test. Critical interval (*CD*) is obtained by the following formula:.

$$\mathrm{CD} = q_{\alpha}\sqrt{\frac{k(k+1)}{6N}}, \qquad (16)$$

where k is the number of algorithms and N is the number of datasets. When k = 5, N = 10 and significance α = 0.05, q_α = 2.728, the calculated critical interval CD = 1.92899. In the case of these conclusions, the MSDT and OC1 have obvious performance advantages compared with CARTSL.

Table 8. The accuracy of five classifiers on numerical data.

Dataset	**Accuracy % (rank)**				
	J48	**CART$_{SL}$**	**OC1**	**HHCART(A)**	**MSDT**
Blood	**78.09±0.50(1)**	70.51±0.70(5)	77.34±0.56(3)	76.50±0.45(4)	77.53±0.68(2)
Glass	66.73±2.84(3)	66.59±2.06(4)	68.43±2.80(2)	64.02±**3.14(1)**	
Image	96.55±0.26(3)	96.32±0.39(4)	96.78±0.33(2)	**97.22±0.25(1)**	96.01±0.40(5)
Iris	95.13±0.73(3)	95.07±0.61(4)	94.67±0.62(5)	95.33±0.63(2)	**96.00±0.00(1)**
Letter	88.09±0.23(5)	88.19±0.17(4)	89.72±0.20(2)	88.23±0.22(3)	**91.77±0.16(1)**
MF-mor	**72.01±0.62(1)**	65.11±0.43(5)	70.15±0.48(3)	69.83±0.63(4)	71.13±0.43(2)
Wav1	76.50±0.32(4)	75.38±0.52(5)	80.07±0.33(2)	79.83±0.33(3)	**82.30±0.32(1)**
Wav2	75.29±0.61(4)	74.27±0.43(5)	79.68±0.56(2)	79.01±0.45(3)	**80.65±0.42(1)**
Wine	93.54±0.88(3)	90.00±1.46(5)	92.54±0.98(4)	94.47±0.86(2)	**95.11±1.23(1)**
Yeast	55.77±0.93(3)	51.14±1.01(5)	**55.90±1.06(1)**	54.28±1.11(4)	55.84±0.93(2)
Average	79.77(3)	77.26(4.6)	80.53(2.6)	79.87(3.1)	**81.68(1.7)**

The accuracy over the categorical and mixed datasets by each method is shown in Table 9. MSDT gets the best accuracy on 4 of 10 datasets and the average accuracy is 78.48%. It is 3.62%, 1.1%, 1.56%, and 1.88% higher than four other trees, respectively. We use the averages of the ranks of 5 classifiers on 10 datasets to calculate $F_F = 1.48951$. Here, the critical value is $F(4, 36) = 2.634$. So, we cannot reject the null hypothesis; namely, there is no significant difference among the five classifiers. In the case of these conclusions, on categorical and mixed data, the advantages of three multivariate decision trees over two univariate decision trees are not so obvious. Especially in OC1, one categorical feature is transformed into multiple numerical features by One Hot method, which greatly increases the dimension of feature space. In the new feature space, the data becomes very sparse, and OC1 cannot find a suitable split hyperplanes.

Table 9. The accuracy of five classifiers on categorical and mixed data.

Dataset	**Accuracy % (rank)**				
	J48	**CART$_{SL}$**	**OC1**	**HHCART(A)**	**MSDT**
Balance	64.00±1.18(5)	76.91±0.82(3)	80.36±1.49(2)	**82.82 1.12(1)**	76.45±2.05(4)
Car	92.52±0.63(5)	**97.31±0.22(1)**	96.27±0.76(4)	96.91±0.53(2)	96.45±0.23(3)
Chess	99.41±0.10(2)	**99.54±0.06(1)**	98.58±0.15(5)	99.23±0.17(3)	99.17±0.21(4)
Hayes	75.13±1.55(5)	**83.00±1.52(1)**	79.26±1.56(3)	76.37±1.16(4)	79.75±2.14(2)
MONK	96.11±1.54(3)	91.94±1.51(5)	93.37±1.53(4)	98.57±1.58(2)	**99.63±0.85(1)**
Abalone	61.30±0.45(2)	57.52±0.54(5)	60.05±0.34(3)	59.29±0.43(4)	**62.84±0.28(1)**
CMC	**51.61±0.66(1)**	47.40±0.47(4)	46.92±1.46(5)	47.83±0.73(3)	47.95±1.38(2)
Flags	**64.34±3.34(1)**	61.96±1.03(3)	60.35±3.62(5)	61.73±1.49(4)	62.42±2.04(2)
TAE	51.43±4.67(5)	62.91±1.87(3)	63.10±3.97(2)	53.26±4.57(4)	**63.77±3.79(1)**
Zoo	92.77±0.01(3)	95.35±0.00(2)	90.98±0.33(4)	89.99±0.02(5)	**96.34±0.45(1)**
Average	74.86(3.2)	77.38(2.8)	76.92(3.7)	76.60(3.2)	**78.48(2.1)**

The tree size over 20 datasets by each method is shown in Table 10. In terms of the complexity of model structure, the average number of nodes in three multivariate decision tree is lower than the other two univariate decision trees. We use the averages of the ranks of 5 classifiers on 20 datasets to calculate $F_F = 3.35294$. Here, with 5 algorithms and 20 datasets, F_F follows the *F-distribution* with 4 and 76 degrees of freedom, and the critical value is $F(4, 76) = 2.492$. So, we reject the null hypothesis. Nemenyi method is used for post hoc test. Critical interval is obtained; $CD = 1.364$. In the case of these conclusions, the MSDT has obvious performance advantages compared with J48.

Table 10. The tree size of five classifiers.

Dataset	**Tree size (rank)**				
	J48	**CART$_{SL}$**	**OC1**	**HHCART(A)**	**MSDT**
Blood	12.60±0.87(2)	179.64±1.00(5)	**7.88±0.75(1)**	16.56±0.88(3)	32.72±2.50(4)
Glass	46.68±1.66(5)	44.93±0.56(3)	32.65±0.78(2)	**27.73±0.67(1)**	46.55±2.15(4)
Image	80.28±1.28(4)	69.22±0.79(2)	**57.62±0.93(1)**	70.43±1.46(3)	161.59±4.57(5)
Iris	8.40±0.24(4)	8.56±0.14(5)	5.10±0.08(2)	5.60±0.12(3)	**4.00±0.00(1)**
Letter	2326.72±9.04(5)	2098.23±9.34(3)	2285.22±9.81(4)	**1795.23±8.91(1)**	1893.41±11.64(2)
MF-mor	162.40±5.54(2)	462.94±1.89(5)	**153.32±3.55(1)**	212.97±4.74(4)	174.07±1.29(3)
Wav1	546.74±7.62(5)	484.45±2.61(4)	470.26±6.17(3)	465.34±6.62(2)	**178.12±5.12(1)**
Wav2	585.42±9.45(5)	453.02±1.73(2)	481.54±8.05(4)	480.38±9.17(3)	**126.18±10.69(1)**
Wine	9.54±0.31(4)	10.47±0.62(5)	9.25±0.30(3)	7.83±0.27(2)	**4.00±0.72(1)**
Yeast	328.70±4.12(4)	463.38±2.33(5)	260.47±9.81(2)	303.25±6.40(3)	**101.52±9.00(1)**

Balance	42.05±1.31(2)	140.83±1.47(4)	159.93±2.36(5)	57.82±1.42(3)	**31.13±5.48(1)**
Car	170.45±1.27(4)	97.18±1.11(2)	**91.46±1.19(1)**	186.54±2.28(5)	141.60±1.32(3)
Chess	54.66±1.07(4)	52.15±0.64(2)	53.70±0.96(3)	**49.62±1.38(1)**	106.96±2.90(5)
Hayes	27.13±0.55(4)	22.09±0.33(3)	20.73±0.47(2)	29.75±0.51(5)	**18.63±0.94(1)**
MONK	39.14±1.83(3)	59.09±6.33(5)	56.28±5.92(4)	20.93±3.41(2)	**18.54±1.08(1)**
Abalone	589.00±18.45(3)	965.09±2.66(5)	679.45±13.25(4)	401.11±9.52(2)	**270.67±19.72(1)**
CMC	247.16±6.21(3)	611.94±3.87(5)	538.95±8.36(4)	**108.92±4.78(1)**	139.64±6.33(2)
Flags	62.63±2.07(5)	45.01±0.52(3)	39.87±1.28(2)	54.28±2.11(4)	**26.86±1.74(1)**
TAE	93.01±3.75(4)	56.40±0.53(2)	**50.57±0.66(1)**	68.82±0.72(3)	114.40±1.76(5)
Zoo	11.00±0.00(3)	**9.65±0.00(1)**	12.07±0.23(5)	10.53±0.07(2)	11.08±0.57(4)
Average	272.19(3.75)	316.71(3.55)	273.32(2.7)	218.68 (2.65)	**180.08 (2.35)**

Comparison on Big Data

The data set *covertype* comes from UCI [35], which is a 7-classification problem, which includes 581012 instances and 54 features. 10 of 54 features are numerical, and the remainders are Boolean. MSDT and J48 regard Boolean as categorical features, and CART$_{SL}$ is regarded as numerical features. The 10 repetitions of 10-fold cross-validation are used. Table 11 provides the accuracy of the three classifiers, the size of the tree, and the time to build the tree.

Table 11. The performance of three classifiers on covertype

	J48	**CART$_{SL}$**	**MSDT**
Accuracy (%)	94.58 ± 0.09	94.30 ± 0.11	94.64 ± 0.12
Tree size	28348.11 ± 274.83	51728.89 ± 347.59	25736.90 ± 235.86
Time (s)	71.61 ± 1.52	45.58 ± 0.56	150.12 ± 2.66

The three classifiers achieve similar accuracies on *covertype*. In terms of tree size, MSDT has the least number of nodes. The running time provided in Table 11 is the time to build a tree and does not include the time consumed by loading data and testing. J48 runs slower than CART$_{SL}$, which does not mean that there is a significant difference in time complexity between the two algorithms. The difference may be caused by the different development language. There are two reasons why MSDT gets the most expensive time consumption. One is that the time complexity of our proposed method is higher than that of the axis-parallel methods when dividing a node. The other one is that the axis-parallel methods mainly perform relational operations, for instance, "<." Our method needs to calculate a large number of distances, which requires arithmetic operations of real number. Although multiway splits can reduce the number of times to split nodes, the time consumed by our method is about 2 to 3 times that of the axis-parallel methods from the experimental results.

CONCLUSION

The decision trees generated by the oblique splits often have better generalization ability and fewer nodes. However, most oblique split methods are time-consuming and cannot be directly used for categorical data, and some of these methods can only be used for binary classification problems. Our proposed algorithm MSDT uses feature weighting and clustering to multiway splits of nonleaf nodes, which can be directly applied to multiclassification problems. Meanwhile, it has a time complexity similar to that of the axis-parallel algorithms. In addition, we give the representation of cluster center and the distance from instance to cluster center, which enables clustering to be used in categorical and mixed data. Experimental results show that MSDT has a good generalization accuracy on multiple types of data.

ACKNOWLEDGMENTS

This research was supported by the National Nature Science Foundation of China under Grants 61772101, 61170169, and 61602075 and in part by the Ph.D. Scientific Research Starting Foundation of Liaoning Province under Grant 20180540084.

REFERENCES

1. A. C. Sick-Samuels, K. E. Goodman, G. Rapsinski et al., “A decision tree using patient characteristics to predict resistance to commonly used broad-spectrum antibiotics in children with gram-negative bloodstream infections,” *Journal of the Pediatric Infectious Diseases Society*, vol. 9, no. 2, p. 142, 2019.
2. S. Guney and A. Atasoy, “Freshness classification of horse mackerels with E-Nose system using hybrid binary decision tree structure,” *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 34, no. 3, pp. 1–17, 2020.
3. Z. Liu, L. Wang, X. Li, and X. Ji, “Optimize x265 rate control: an exploration of lookahead in frame bit allocation and slice type decision,” *IEEE Transactions on Image Processing*, vol. 28, no. 5, pp. 2558–2573, 2019.
4. Y. Wang, S.-T. Xia, and J. Wu, “A less-greedy two-term Tsallis Entropy Information Metric approach for decision tree classification,” *Knowledge-Based Systems*, vol. 120, pp. 34–42, 2017.
5. Z.-H. Zhou and J. Feng, “Deep forest: towards an alternative to deep neural networks,” in *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI’17)*, pp. 3553–3559, Melbourne, Australia, August 2017.
6. X. Guan, J. Liang, Y. Qian, and J. Pang, “A multi-view OVA model based on decision tree for multi-classification tasks,” *Knowledge-Based Systems*, vol. 138, pp. 208–219, 2017.
7. L. Zhang and P. N. Suganthan, “Oblique decision tree ensemble via multisurface proximal support vector machine,” *IEEE Transactions on Cybernetics*, vol. 45, no. 10, p. 1, 2015.
8. R. Katuwal, P. N. Suganthan, and L. Zhang, “An ensemble of decision trees with random vector functional link networks for multi-class classification,” *Applied Soft Computing*, vol. 70, pp. 1146–1153, 2018.
9. R. Katuwal and P. N. Suganthan, “Enhancing multi-class classification of random forest using random vector functional neural network and oblique decision surfaces,” in *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, Rio de Janeiro, Brazil, July 2018.

10. Z. Liu, T. Wen, W. Sun, and Q. Zhang, "Semi-supervised self-training feature weighted clustering decision tree and random forest," *IEEE Access*, vol. 8, pp. 128337–128348, 2020.
11. L. Hyafil and R. L. Rivest, "Constructing optimal binary decision trees is NP-complete," *Information Processing Letters*, vol. 5, no. 1, pp. 15–17, 1976.
12. H. E. L. Cagnini, R. C. Barros, and M. P. Basgalupp, "Estimation of distribution algorithms for decision-tree induction," in *Proceedings of the IEEE Congress on Evolutionary Computation*, IEEE, San Sebastian, Spain, June 2017.
13. R. Rivera-Lopez and J. Canul-Reich, "Construction of near-optimal axis-parallel decision trees using a differential-evolution-based approach," *IEEE Access*, vol. 6, pp. 5548–5563, 2018.
14. M. P. Basgalupp, R. C. Barros, A. C. P. L. F. de Carvalho, A. A. Freitas, and D. D. Ruiz, "LEGAL-tree: a lexicographic multi-objective genetic algorithm for decision tree induction," in *Proceedings of the ACM Symposium on Applied Computing*, ACM, Honolulu, HI, USA, March 2009.
15. J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, no. 1, pp. 81–106, 1986.
16. J. R. Quinlan, "C4.5: programs for machine learning," *Machine Learning*, vol. 16, no. 3, pp. 235–240, 1994.
17. W. Buntine, "Learning classification trees," *Statistics and Computing*, vol. 2, no. 2, pp. 63–73, 1992.
18. A. Cherfi, A. K. Nouira, and A. Ferchichi, "Very fast C4.5 decision tree algorithm," *Applied Artificial Intelligence*, vol. 32, no. 2, pp. 119–137, 2018.
19. R. S. Bucy and R. S. Diesposti, "Decision tree design by simulated annealing," *ESAIM: Mathematical Modelling and Numerical Analysis*, vol. 27, no. 5, pp. 515–534, 1993.
20. R. C. Barros, M. P. Basgalupp, A. C. P. L. F. de Carvalho et al., "A survey of evolutionary algorithms for decision-tree induction," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 3, pp. 291–312, 2012.
21. S. K. Freitas, S. Kasif, and S. Salzberg, "A system for induction of oblique decision trees," *Journal of Artificial Intelligence Research*, vol. 2, no. 1, pp. 1–32, 1996.

22. A. López-Chau, J. Cervantes, L. López-García, and F. G. Lamont, “Fisher’s decision tree,” *Expert Systems with Applications*, vol. 40, no. 16, pp. 6283–6291, 2013.

23. Y. Freund and L. Mason, “The alternating decision tree learning algorithm,” in *Proceeding of the International Conference on Machine Learning*, Bled, Slovenia, June 1999.

24. K. S. Hong, P. L. Ooi, C. K. Ye et al., “Multivariate alternating decision trees,” *Pattern Recognition*, vol. 50, no. C, pp. 195–209, 2016.

25. D. C. Wickramarachchi, B. L. Robertson, M. Reale, C. J. Price, and J. Brown, “HHCART: an oblique decision tree,” *Computational Statistics & Data Analysis*, vol. 96, pp. 12–23, 2015.

26. S. Shah and P. S. Sastry, “New algorithms for learning and pruning oblique decision trees,” *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, vol. 29, no. 4, pp. 494–505, 1999.

27. B. H. Menze, B. M. Kelm, D. N. Splitthoff, U. Koethe, and F. A. Hamprecht, “On oblique random forests,” in *Proceedings of the European Conference on Machine Learning & Knowledge Discovery in Databases*, Bled, Slovenia, September 2011.

28. R. Gnanadesikan, *Methods For Satistical Data Analysis of Multivariate Observations*, Wiley, Hoboken, NJ, USA, 1977.

29. K. Kira and L. A. Rendell, “A Practical approach to feature selection,” in *Proceedings of the Ninth International Workshop on Machine Learning (ML 1992)*, pp. 249–256, Morgan Kaufmann Publishers Inc., Aberdeen, Scotland, July 1992.

30. I. Kononenko, “Estimating attributes: analysis and extension of relief,” in *Proceeding of the 1994 European Conference on Machine Learning*, pp. 171–182, Catania, Italy, April 1994.

31. Z. Huang, “Extensions to the *k*-means algorithm for clustering large data sets with categorical values,” *Data Mining and Knowledge Discovery*, vol. 2, no. 3, pp. 283–304, 1998.

32. T. K. Ho, “The random subspace method for constructing decision forests,” *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 20, no. 8, pp. 832–844, 1998.

33. M. Lichman, “UCI machine learning repository,” 2013, http://archive.ics.uci.edu/ml.

Section 4: Algebraicity and Boolean Algebras

Chapter

13

ON THE DEFORMATION THEORY OF STRUCTURE CONSTANTS FOR ASSOCIATIVE ALGEBRAS

B. G. Konopelchenko

Dipartimento di Fisica, Universita del Salento and INFN, Sezione di Lecce, 73100 Lecce, Italy

ABSTRACT

An algebraic scheme for constructing deformations of structure constants for associative algebras generated by deformation driving algebras (DDAs) is discussed. An ideal of left divisors of zero plays a central role in this construction. Deformations of associative three-dimensional algebras with the DDA being a three-dimensional Lie algebra and their connection with integrable systems are studied.

Citation: Citation: B. G. Konopelchenko, "On the Deformation Theory of Structure Constants for Associative Algebras", Advances in Mathematical Physics, vol. 2010, Article ID 389091, 21 pages, 2010. https://doi.org/10.1155/2010/389091.

INTRODUCTION

An idea to study deformations of structure constants for associative algebras goes back to the classical works of Gerstenhaber [1, 2]. As one of the approaches to deformation theory he suggested "to take the point of view that the objects being deformed are not merely algebras, but essentially algebras with a fixed basis" and to treat "the algebraic set of all structure constants as parameter space for deformation theory" [2].

Thus, following this approach, one chooses the basis P_0, P_1,..., P_N for a given algebra A, takes the structure constants C^n_{jk} defined by the multiplication table

$$\mathbf{P}_j\mathbf{P}_k = \sum_{n=0}^{N} C^n_{jk}\mathbf{P}_n, \quad j,k = 0,1,\ldots,N, \tag{1.1}$$

and looks for their deformations $C^n_{jk}(x)$, where $(x) = (x^1,\ldots,x^M)$ is the set of deformation parameters, such that the associativity condition

$$\sum_{m=0}^{N} C^m_{jk}(x)C^n_{ml}(x) = \sum_{m=0}^{N} C^m_{kl}(x)C^n_{jm}(x) \tag{1.2}$$

or similar equation is satisfied.

A remarkable example of deformations of this type with M = N + 1 has been discovered by Witten [3] and Dijkgraaf et al. [4]. They demonstrated that the function F which defines the correlation functions $\langle\Phi_j\Phi_k\Phi_l\rangle = \partial^3F/\partial x^j\partial x^k\partial x^l$ and so forth in the deformed two-dimensional topological field theory obeys the associativity equation (1.2) with the structure constants given by

$$C^l_{jk} = \sum_{m=0}^{N} \eta^{lm}\frac{\partial^3 F}{\partial x^j\partial x^k\partial x^m}, \tag{1.3}$$

where the constants are $\eta^{lm} = (g^{-1})^{lm}$ and $g_{lm} = \partial^3F/\partial x^0\partial x^l\partial x^m$ where the variable x^0 is associated with the units element. Each solution of the WDVV equations (1.2) and (1.3) describes a deformation of the structure constants of the N + 1- dimensional associative algebra of primary fields Φ_j.

The interpretation and formalization of the WDVV equation in terms of Frobenius manifolds proposed by Dubrovin [5, 6] provides us with a method to describe class of deformations of the so-called Frobenius algebras. An extension of this approach to general algebras and corresponding F-manifolds has been given by Hertling and Manin [7]. The beautiful and rich theory of Frobenius and F-manifolds has various applications from the singularity theory to quantum cohomology (see, e.g., [6, 8, 9]).

An alternative approach to the deformation theory of the structure constants for commutative associative algebras has been proposed recently in [10–14]. Within this method the deformations of the structure constants are governed by the so-called central system (CS). Its concrete form depends on the class of deformations under consideration and CS contains, as particular reductions, many integrable systems like WDVV equation, oriented associativity equation, and integrable dispersionless, dispersive, and discrete equations (Kadomtsev-Petviashvili equation, etc.). The common feature of the coisotropic, quantum, discrete deformations considered in [10–14] is that for all of them elements p_j of the basis and deformation parameters x_j form a certain algebra (Poisson, Heisenberg, etc.). A general class of deformations considered in [13] is characterized by the condition that the ideal $J = \langle f_{jk} \rangle$ generated by the elements $f_{jk} = -p_j p_k + \sum_{l=0}^{N} C_{jk}^{l}(x) p_l$ representing the multiplication table (1) is the Poisson ideal. It was shown that this class contains a subclass of so-called integrable deformations for which the CS has a simple and nice geometrical meaning.

In the present paper we will discuss a purely algebraic formulation of such integrable deformations. We will consider the case when the algebra generating deformations of the structure constants, that is, the algebra formed by the elements p_j of the basis and deformation parameters x_k(deformation driving algebra (DDA)), is a Lie algebra. The basic idea is to require that all elements $f_{jk} = -p_j p_k + \sum_{l=0}^{N} C_{jk}^{l}(x) p_l$ are left divisors of zero and that they generate the ideal J of left divisors of zero. This requirement gives rise to the central system which governs deformations generated by DDA. This central system of equations for structure constants differs, in general, from the associativity condition. So, deformed algebras form families of commutative but not necessarily associative algebras.

Here we will study the deformations of the structure constants for the three-dimensional algebra in the case when the DDA is given by one of the three-dimensional Lie algebras. Such deformations are parametrized by a single deformation variable x. Depending on the choice of DDA and identification of p_1, p_2, and x with the elements of DDA, the corresponding CS takes the form of the system of ordinary differential equations or the system of discrete equations (multidimensional mappings). In the first case the CS contains the third-order ODEs from the Chazy-Bureau list as the particular examples. This approach provides us also with the Lax form of the above equations and their first integrals.

The paper is organized as follows. General formulation of the deformation theory for the structure constants is presented in Section 2. Quantum, discrete, and coisotropic deformations are discussed in Section 3. Three-dimensional Lie algebras as DDAs are analyzed in Section 4. Deformations generated by general DDAs are studied in Section 5. Deformations driven by the nilpotent and solvable DDAs are considered in Sections 6 and 7, respectively.

DEFORMATIONS OF THE STRUCTURE CONSTANTS GENERATED BY DDA

So, we consider a finite-dimensional commutative algebra A with (or without) unit element P_0 in the fixed basis composed by the elements $\mathbf{P}_0, \mathbf{P}_1, \ldots, \mathbf{P}_N$. The multiplication table (1) defines the structure constants C^l_{jk}. The commutativity of the basis implies that $C^l_{jk} = C^l_{kj}$. In the presence of the unit element one has $C^l_{j0} = \delta^l_j$ where δ^l_j is the Kronecker symbol.

Following Gerstenhaber's suggestion [1, 2] we will treat the structure constants C^l_{jk} as the objects to deform and will denote the deformation parameters by $x^1, x^2, \ldots, x^M$. For the undeformed structure constants the associativity conditions (1.2) are nothing else than the compatibility conditions for the table of multiplication (1.1). In the construction of deformations we should first specify a “deformed” version of the multiplication table and then require that this realization is self-consistent and meaningful.

Thus, to define deformations one has the following.

- We associate a set of elements $p_0, p_1, \ldots, p_N, x^1, x^2, \ldots, x^M$ with the elements of the basis $\mathbf{P}_0, \mathbf{P}_1, \ldots, \mathbf{P}_N$ and deformation parameters $x^1, x^2, \ldots, x^M$.

- We consider the Lie algebra *B* of the dimension N + M with the basis elements $e_1,\dots,e_{N+M}$ obeying the commutation relations:

$$[e_\alpha, e_\beta] = \sum_{\gamma=1}^{N+M} C_{\alpha\beta\gamma} e_\gamma, \quad \alpha,\beta = 1,2,\dots,N+M. \tag{2.1}$$

- We identify the elements $p_1,\dots,p_N,x^1,x^2,\dots,x^M$ with the elements $e_1,\dots,e_{N+M}$, thus defining the deformation driving algebra (DDA). Different identifications define different DDAs. We assume that the element p_0 commutes with all elements of DDA and we put $p_0 = 1$. The commutativity of the basis in the algebra *A* implies the commutativity between p_j, and in this paper we assume the same property for all x^k. So, we will consider the DDAs defined by the commutation relations of the type

$$[p_j, p_k] = 0, \quad \left[x^j, x^k\right] = 0, \quad \left[p_j, x^k\right] = \sum_l \alpha_{jl}^k x^l + \sum_l \beta_j^{kl} p_l, \tag{2.2}$$

where α_{jl}^k and β_j^{kl} are some constants.

- We consider the elements

$$f_{jk} = -p_j p_k + \sum_{l=0}^{N} C_{jk}^l(x) p_l, \quad j,k = 1,\dots,N \tag{2.3}$$

of the universal enveloping algebra U(B) of the algebra DDA(*B*). These f_{jk} “represent” the table (1) in U(B).

- 5We require that all f_{jk} are left zero divisors and have a common right zero divisor.

In this case f_{jk} generate the left ideal J of left zero divisors. We remind that non-zero elements a and b are called left and right divisors of zero if ab = 0 (see e.g., [15]).

Definition 2.1. The structure constants $C_{jk}^l(x)$ are said to define deformations of the algebra *A* generated by given DDA if all f_{jk} are left zero divisors with common right zero divisor.

To justify this definition we first observe that the simplest possible realization of the multiplication table (1) in U(B) given by the equations $f_{jk} = 0$, j, k = 1,...,N is too restrictive in general. Indeed, for instance, for the Heisenberg algrebra B [12] such equations imply that $[p_l, C_{jk}^m(x)] = \partial C_{jk}^m / \partial x^l = 0$ and, hence, all C_{jk}^m are constants. So, one should look for a weaker realization of the multiplication table. A condition that all f_{jk} are left zero divisors is a natural candidate. The condition of compatibility of the corresponding equations $f_{jk} \cdot \Psi_{jk} = 0$, j, k = 1,...,N where Ψ_{jk} are right zero divisors requires that the l.h.s. of these equations and, hence, Ψ_{jk} should have a common divisor (see, e.g., [15]). We restrict ourselves to the case when $\Psi_{jk} = \Psi \cdot \Phi_{jk}, \ j,k = 1,\ldots,N$ where Φ_{jk} are invertible elements of U(B). In this case one has the set of equations

$$f_{jk} \cdot \Psi = 0, \quad j,k = 0,1,\ldots,N; \tag{2.4}$$

that is, all left zero divisors f_{jk} have common right zero divisor Ψ.

These conditions impose constraints on $C^m_{jk}(x)$. To clarify these constraints we will use the associativity of U(B). First we observe that due to the relations (2.2) one has the identity ($p_0 = 1$)

$$\left[p_l, C^m_{jk}(x)\right] = \sum_{t=0}^{N} \Delta^{mt}_{jk,l}(x) p_t, \tag{2.5}$$

where $\Delta^{mt}_{jk,l}(x)$ are certain functions of $x^1,\ldots,x^M$ only. Then, taking into account (2.2) and associativity of U(B), one obtains

$$(p_j p_k) p_l - p_j (p_k p_l) = \sum_{s,t=0}^{N} K^{st}_{klj} \cdot f_{st} + \sum_{t=0}^{N} \Omega^t_{klj}(x) \cdot p_t, \quad j,k,l = 0,1,\ldots,N, \tag{2.6}$$

where

$$\begin{aligned} K^{st}_{klj} &= \frac{1}{2}(\delta^s_k \delta^t_l + \delta^t_k \delta^s_l) p_j - \frac{1}{2}\left(\delta^s_k \delta^t_j + \delta^t_k \delta^s_j\right) p_l + \frac{1}{2}\left(\delta^s_j C^t_{kl} + \delta^t_j C^s_{kl}\right) \\ &\quad - \frac{1}{2}\left(\delta^s_l C^t_{kj} + \delta^t_l C^s_{kj}\right) + \Delta^{st}_{kl,j} - \Delta^{st}_{kj,l}, \\ \Omega^t_{klj}(x) &= \sum_s C^s_{jk} C^t_{ls} - \sum_s C^s_{lk} C^t_{js} + \sum_{s,n} \left(\Delta^{sn}_{kj,l} - \Delta^{sn}_{kl,j}\right) C^t_{sn}. \end{aligned} \tag{2.7}$$

Thus, the identity (2.6) gives

$$\sum_{s,t=0}^{N} K^{st}_{klj} \cdot f_{st} + \sum_{t=0}^{N} \Omega^t_{klj}(x) \cdot p_t = 0, \quad j,k,l = 0,1,\ldots,N. \tag{2.8}$$

Due to the relations (2.4), (2.8) implies that

$$\left(\sum_{t=0}^{N} \Omega^t_{klj}(x) \cdot p_t\right) \Psi = 0. \tag{2.9}$$

These equations are satisfied if

$$\Omega^t_{klj}(x) = \sum_s C^s_{jk} C^t_{ls} - \sum_s C^s_{lk} C^t_{js} + \sum_{s,n} \left(\Delta^{sn}_{kj,l} - \Delta^{sn}_{kl,j} \right) C^t_{sn} = 0, \quad j,k,l,t = 0,1,\ldots,N. \tag{2.10}$$

This system of equations plays a central role in our approach. If Ψ has no left zero divisors linear in p_j, the relation (2.10) is the necessary condition for existence of a common right zero divisor for f_{jk} since U(B) has no zero elements linear in p_j (see e.g., [16]).

At N ≥ 3 it is also a sufficient condition. Indeed, if $C^m_{jk}(x)$ are such that (2.10) is satisfied, then

$$\sum_{s,t=0}^{N} K^{st}_{klj} \cdot f_{st} = 0, \quad j,k,l = 0,1,\ldots,N. \tag{2.11}$$

Generically, it is the system of $(1/2)N^2(N-1)$ linear equations for $N(N+1)/2$ unknowns f_{st} with noncommuting coefficients K^{st}_{klj}. At N ≥ 3 for generic (nonzeros, nonzero divisors) $K^{st}_{klj}(x,p)$ the system (2.11) implies that

$$\alpha_{jk} f_{jk} = \beta_{lm} f_{lm}, \quad j,k,l,m = 1,\ldots,N, \tag{2.12}$$

$$\gamma_{jk} f_{jk} = 0, \quad j,k = 1,\ldots,N, \tag{2.13}$$

where α_{jk}, β_{lm}, and γ_{jk} are certain elements of U(B) (see e.g., [17, 18]). Thus, all f_{jk} are right zero divisors. They are also left zero divisors. Indeed, due to Ado's theorem (see e.g., [16]) finite-dimensional Lie algebra B and, hence, U(B) are isomorphic to matrix algebras. For the matrix algebras zero divisors (matrices with vanishing determinants) are both right and left zero divisors [15]. Then, under the assumption that all α_{jk} and β_{lm} are not zero divisors, the relations (2.12) imply that the right divisor of one of f_{jk} is also the right zero divisor for the others.

At N = 2 one has only two relations of the type (2.12) and a right zero divisor of one of f_{11}, f_{12}, f_{22} is the right zero divisor of the others. We note that it is not easy to control assumptions mentioned above. Nevertheless, (2.4) and (2.10) certainly are fundamental one for the whole approach.

We will refer to the system (2.10) as the Central System (CS) governing deformations of the structure constants of the algebra *A* generated by a given DDA. Its concrete form depends strongly on the form of the brackets $[p_t, C^l_{jk}(x)]$ which are defined by the relations (2.2) for the elements of the basis of DDA. For stationary solutions $(\Delta^{mt}_{jk,l} = 0)$ the CS (2.10) is reduced to the associativity conditions (1.2).

QUANTUM, DISCRETE, AND COISOTROPIC DEFORMATIONS

Coisotropic, quantum, and discrete deformations of associative algebras considered in [10–14] represent particular realizations of the above general scheme associated with different DDAs.

For the *quantum deformations* one has M = N and the deformation driving algebra is given by the Heisenberg algebra [12]. The elements of the basis of the algebra *A* and deformation parameters are identified with the elements of the Heisenberg algebra in such a way that

$$[p_j, p_k] = 0, \quad \left[x^j, x^k\right] = 0, \quad \left[p_j, x^k\right] = \hbar\delta^k_j, \quad j,k = 1,\ldots,N, \tag{3.1}$$

where ħ is the real constant (Planck's constant in physics). For the Heisenberg DDA

$$\Delta^{mt}_{jk,l} = \hbar\delta^t_0 \frac{\partial C^m_{jk}(x)}{\partial x^l}, \tag{3.2}$$

and consequently

$$\Omega^n_{klj}(x) = \hbar\frac{\partial C^n_{jk}}{\partial x^l} - \hbar\frac{\partial C^n_{kl}}{\partial x^j} + \sum_{m=0}^{N}\left(C^m_{jk}C^n_{ml} - C^m_{kl}C^n_{jm}\right) = 0, \quad j,k,l,n = 0,1,\ldots,N. \tag{3.3}$$

Quantum CS (3.3) governs deformations of structure constants for associative algebra driven by the Heisenberg DDA. It has a simple geometrical meaning of vanishing Riemann curvature tensor for torsionless Christoffel symbols Γ^l_{jk} identified with the structure constants $(C^l_{jk} = \hbar\Gamma^l_{jk})$ [12].

In the representation of the Heisenberg algebra (3.1) by operators acting in a linear space H left divisors of zero are realized by operators with nonempty kernel. The ideal J is the left ideal generated by operators f_{jk} which have nontrivial common kernel or, equivalently, for which equations

$$f_{jk}|\Psi\rangle = 0, \quad j,k = 1,2,\ldots,N \tag{3.4}$$

have nontrivial common solutions $|\Psi\rangle \subset H$. The compatibility condition for (3.4) is given by the CS (3.3). The common kernel of the operators f_{jk} forms a subspace H_Γ in the linear space H. So, in the approach under consideration the multiplication table (1) is realized only on H_Γ, but not on the whole *H*. Such type of realization of the constraints is well known in quantum theory as Dirac's recipe for quantization of the first-class constraints [19]. In quantum theory context equation (3.4) serves to select the physical subspace in the whole Hilbert space. Within the deformation theory one may refer to the subspace H_Γ as the "structure constants" subspace. In [12] the recipe (3.4) was the starting point for construction of the quantum deformations.

Quantum CS (3.3) contains various classes of solutions which describe different classes of deformations. An important subclass is given by isoassociative deformations, that is, by deformations for which the associativity condition (1.2) is valid for all values of deformation parameters. For such quantum deformations the structure constants should obey the following equations:

$$\frac{\partial C^n_{jk}}{\partial x^l} - \frac{\partial C^n_{kl}}{\partial x^j} = 0, \quad j,k,l,n = 1,\ldots,N. \tag{3.5}$$

These equations imply that $C^n_{jk} = \partial^2\Phi^n / \partial x^j \partial x^k$ where Φ^n are some functions while the associativity condition (1.2) takes the following form:

$$\sum_{m=0}^{N} \frac{\partial^2\Phi^m}{\partial x^j \partial x^k} \frac{\partial^2\Phi^n}{\partial x^m \partial x^l} = \sum_{m=0}^{N} \frac{\partial^2\Phi^m}{\partial x^l \partial x^k} \frac{\partial^2\Phi^n}{\partial x^m \partial x^j}. \tag{3.6}$$

It is the oriented associativity equation introduced in [5, 20]. Under the gradient reduction $\Phi^n = \sum_{l=0}^{N} \eta^{nl} (\partial F / \partial x^l)$ equation (3.7) becomes the WDVV equations (1.2) and (1.3).

Non-isoassociative deformations for which the condition (3.5) is not valid are of interest too. They are described by some well-known integrable soliton equations [12]. In particular, there are Boussinesq equation among them for N = 2 and Kadomtsev-Petviashvili (KP) hierarchy for the infinite-dimensional algebra of polynomials in the Faa' de Bruno basis [12]. In the latter case the deformed structure constants are given by

$$C^l_{jk} = \delta^l_{j+k} + H^k_{j-l} + H^j_{k-l}, \quad j,k,l = 0,1,2,\ldots \tag{3.7}$$

With

$$H^j_k = \frac{1}{\hbar} P_k\left(-\hbar\tilde{\partial}\right) \frac{\partial \log \tau}{\partial x^j}, \quad j,k = 1,2,3,\ldots, \tag{3.8}$$

where τ is the famous tau-function for the KP hierarchy and

$P_k(-\hbar\tilde{\partial}) \quad \doteqdot P_k(-\hbar(\partial/\partial x^1),(-1/2)\hbar(\partial/\partial x^2),-(1/3)\hbar(\partial/\partial x^3),\ldots)$ where $P_k(t_1,t_2,t_3,\ldots)$ are Schur polynomials defined by the generating formula $\exp(\sum_{k=1}^{\infty}\lambda^k t_k)=\sum_{k=0}^{\infty}\lambda^k P_k(t)$.

Discrete deformations of noncommutative associative algebras are generated by the DDA with M = N and commutation relations

$$[p_j,p_k]=0,\quad \left[x^j,x^k\right]=0,\quad \left[p_j,x^k\right]=\delta_j^k p_j,\quad j,k=1,\ldots,N. \tag{3.9}$$

In this case

$$\Delta_{jk,l}^{mt}=\delta_l^t(T_l-1)C_{jk}^m(x),\quad j,k,l,m,t=0,1,2,\ldots,N, \tag{3.10}$$

where for an arbitrary function φ(x) the action of T_j is defined by $T_j\varphi(x^0,\ldots,x^j,\ldots,x^N)=\varphi(x^0,\ldots,x^j+1,\ldots,x^N)$. The corresponding CS is of the form

$$C_lT_lC_j-C_jT_jC_l=0,\quad j,l=0,1,\ldots,N, \tag{3.11}$$

where the matrices C_j are defined as $(C_j)_k^l = C_{jk}^l,\ j,k,l = 0,1,\ldots,N$. The discrete CS (3.11) governs discrete deformations of associative algebras. The CS (3.11) contains, as particular cases, the discrete versions of the oriented associativity equation, WDVV equation, Boussinesq equation, and discrete KP hierarchy and Hirota-Miwa bilinear equations for KP τ-function [13].

For *coisotropic deformations* of commutative algebras [10, 11] again M = N, but the DDA is the Poisson algebra with p_j and x^k identified with the Darboux coordinates, that is,

$$\{p_j,p_k\}=0,\quad \left\{x^j,x^k\right\}=0,\quad \left\{p_j,x^k\right\}=-\delta_j^k,\quad j,k=0,1,\ldots,N, \tag{3.12}$$

where {, } is the standard Poisson bracket. The algebra U(B) is the commutative ring of functions and divisors of zero are realized by functions with zeros. So, the functions f_{jk} should be functions with common set Γ of zeros. Thus, in the coisotropic case the multiplication table (1) is realized by the following set of equations [10]:

$$f_{jk}=0,\quad j,k=0,1,2,\ldots,N. \tag{3.13}$$

The compatibility condition for these equations is (see e.g., [10])

$$\{f_{jk},f_{nl}\}|_{\Gamma}=0,\quad j,k,l,n=1,2,\ldots,N. \tag{3.14}$$

The set Γ is the coisotropic submanifold in $R^{2(N+1)}$. The condition (3.14) gives rise to the following system of equations for the structure constants:

$$[C,C]^m_{jklr} \div \sum_{s=0}^{N}\left(C^m_{sj}\frac{\partial C^s_{lr}}{\partial x^k}+C^m_{sk}\frac{\partial C^s_{lr}}{\partial x^j}-C^m_{sr}\frac{\partial C^s_{jk}}{\partial x^l}-C^m_{sl}\frac{\partial C^s_{jk}}{\partial x^r}+C^s_{lr}\frac{\partial C^m_{jk}}{\partial x^s}-C^s_{jk}\frac{\partial C^m_{lr}}{\partial x^s}\right)=0 \quad (3.15)$$

while the equations $\Omega^n_{klj}(x)=0$ have the form of associativity conditions (1.2):

$$\Omega^n_{klj}(x)=\sum_{m=0}^{N}\left(C^m_{jk}(x)C^n_{ml}(x)-C^m_{kl}(x)C^n_{jm}(x)\right)=0. \quad (3.16)$$

Equations (3.15) and (3.16) form the CS for coisotropic deformations [10]. In this case C^l_{jk} is transformed as the tensor of the type (1,2) under the general transformations of coordinates x^j, and the whole CS of (3.15) and (3.16) is invariant under these transformations [14]. The bracket $[C,C]^m_{jklr}$ has appeared for the first time in [21] where the so-called differential concomitants were studied. It was shown in [16] that this bracket is a tensor only if the tensor C^l_{jk} obeys the algebraic constraint (3.16). In [7] the CS of (3.15) and (3.16) has appeared implicitly as the system of equations which characterizes the structure constants for F-manifolds. In [10] it has been derived as the CS governing the coisotropic deformations of associative algebras.

The CS of (3.15) and (3.16) contains the oriented associativity equation, the WDVV equation, dispersionless KP hierarchy, and equations from the genus zero universal Whitham hierarchy as the particular cases [10, 11]. Yano manifolds and Yano algebroids associated with the CS of (3.15) and (3.16) are studied in [14].

We would like to emphasize that for all deformations considered above the stationary solutions of the CSs obey the global associativity condition (1.2).

THREE-DIMENSIONAL LIE ALGEBRAS AS DDA

In the rest of the paper we will study deformations of associative algebras generated by three-dimensional real Lie algebra L. The complete list of such algebras contains 9 algebras (see e.g. [16]). Denoting the basis elements by e_1, e_2, e_3, one has the following nonequivalent cases:

(1) abelian algebra L_1,

(2) general algebra L_2: $[e_1,e_2]=e_1,\ [e_2,e_3]=0,\ [e_3,e_1]=0,$

(3) nilpotent algebra $L_3 : [e_1, e_2] = 0,\ [e_2, e_3] = e_1,\ [e_3, e_1] = 0,$,

(4)–(7) four nonequivalent solvable algebras: $[e_1, e_2] = 0,\ [e_2, e_3] = \alpha e_1 + \beta e_2,\ [e_3, e_1] = \gamma e_1 + \delta e_2$ with $\alpha\delta - \beta\gamma \neq 0,$

(8)-(9) simple algebras L_8 = so (3) and L_9 = so (2,1).

In virtue of the one-to-one correspondence between the elements of the basis in DDA and the elements p_j, x^k an algebra L should have an abelian subalgebra and only one of its elements may play the role of the deformation parameter x. For the original algebra *A* and the algebra *B* one has two options.

(1) *A* is a two-dimensional algebra without unit element and *B* = *L*.

(2) *A* is a three-dimensional algebra with the unit element and $B = L_0 \oplus L$ where L_0 is the algebra generated by the unity element p_0.

After the choice of *B* one should establish a correspondence between p_1, p_2, x and e_1, e_2, e_3 defining DDA. For each algebra L_k there are obviously, in general, six possible identifications if one avoids linear superpositions. Some of them are equivalent. The incomplete list of nonequivalent identifications is as follows

- algebra $L_1 : p_1 = e_1,\ p_2 = e_2,\ x = e_3$; DDA is the commutative algebra with

$$[p_1, p_2] = 0, \quad [p_1, x] = 0, \quad [p_2, x] = 0, \tag{4.1}$$

- algebra L_2:

case (a) $p_1 = -e_2,\ p_2 = e_3,\ x = e_1;$ the corresponding DDA is the algebra L_{2a} with the commutation relations:

$$[p_1, p_2] = 0, \quad [p_1, x] = x, \quad [p_2, x] = 0, \tag{4.2}$$

case (b) $p_1 = e_1,\ p_2 = e_3,\ x = e_2;$ the corresponding DDA L_{2b} is defined by

$$[p_1, p_2] = 0, \quad [p_1, x] = p_1, \quad [p_2, x] = 0, \tag{4.3}$$

- algebra L_3: $p_1 = e_1,\ p_2 = e_2,\ x = e_3;$ DDA L_3 is

$$[p_1, p_2] = 0, \quad [p_1, x] = 0, \quad [p_2, x] = p_1, \tag{4.4}$$

- solvable algebra L_4 with $\alpha = 0,\ \beta = 1,\ \gamma = -1,\ \delta = 0 : p_1 = e_1,\ p_2 = e_2,\ x = e_3;$ DDA L_4 is

$$[p_1, p_2] = 0, \quad [p_1, x] = p_1, \quad [p_2, x] = p_2, \tag{4.5}$$

- solvable algebra L_5 at $\alpha = 1,\ \beta = 0,\ \gamma = 0,\ \delta = 1 : p_1 = e_1,\ p_2 = e_2,\ x = e_3;$ DDA L_5 is

$$[p_1, p_2] = 0, \quad [p_1, x] = p_1, \quad [p_2, x] = -p_2. \tag{4.6}$$

For the second choice of the algebra $B = L_0 \oplus L$ mentioned above the table of multiplication (1.1) consists of the trivial part $\mathbf{P}_0\mathbf{P}_j = \mathbf{P}_j\mathbf{P}_0 = \mathbf{P}_j, j = 0, 1, 2$ and the nontrivial part:

$$\begin{aligned}
\mathbf{P}_1^2 &= A\mathbf{P}_0 + B\mathbf{P}_1 + C\mathbf{P}_2,\\
\mathbf{P}_1\mathbf{P}_2 &= D\mathbf{P}_0 + E\mathbf{P}_1 + G\mathbf{P}_2,\\
\mathbf{P}_2^2 &= K\mathbf{P}_0 + M\mathbf{P}_1 + N\mathbf{P}_2.
\end{aligned} \tag{4.7}$$

For the first choice B = K the multiplication table is given by (4.7) with A = D = K = 0.

It is convenient also to arrange the structure constants A, B, . . . , N into the matrices C_1, C_2 defined by $(C_j)^l_k = C^l_{jk}$. One has

$$C_1 = \begin{pmatrix} 0 & A & D \\ 1 & B & E \\ 0 & C & G \end{pmatrix}, \quad C_2 = \begin{pmatrix} 0 & D & K \\ 0 & E & M \\ 1 & G & N \end{pmatrix}. \tag{4.8}$$

In terms of these matrices the associativity conditions (1.2) are written as

$$C_1C_2 = C_2C_1. \tag{4.9}$$

Simple algebras L_8 and L_9 do not contain two commuting elements to be identified with p_1 and p_2, and, hence, they cannot be DDA. Deformations generated by algebras L_6 and L_7 will be considered elsewhere.

DEFORMATIONS GENERATED BY GENERAL DDAS

(1) Commutative DDA (4.1) does not force any deformation of structure constants. So, we begin with the three-dimensional commutative algebra *A* and DDA L_{2a} defined by the commutation relations (4.2). These relations imply that for an arbitrary function φ(x)

$$[p_j, \varphi(x)] = \Delta_j \varphi(x), \quad j = 1, 2, \tag{5.1}$$

where $\Delta_1 = (x\partial/\partial x)$, $\Delta_2 = 0$. Consequently, one has the following CS:

$$\Omega^n_{klj}(x) = \Delta_l C^n_{jk} - \Delta_j C^n_{kl} + \sum_{m=0}^{2}\left(C^m_{jk}C^n_{lm} - C^m_{kl}C^n_{jm}\right) = 0, \quad j, k, l, n = 0, 1, 2. \tag{5.2}$$

In terms of the matrices C_1 and C_2 defined above this CS has a form of the Lax equation:

$$x\frac{\partial C_2}{\partial x} = [C_2, C_1]. \tag{5.3}$$

The CS (5.3) has all remarkable standard properties of the Lax equations (see e.g. [20, 21]): it has three independent first integrals:

$$I_1 = \operatorname{tr} C_2, \qquad I_2 = \frac{1}{2}\operatorname{tr}(C_2)^2, \qquad I_3 = \frac{1}{3}\operatorname{tr}(C_2)^3, \tag{5.4}$$

and it is equivalent to the compatibility condition of the linear problems:

$$\begin{aligned} C_2\Phi &= \lambda\Phi, \\ x\frac{\partial\Phi}{\partial x} &= -C_1\Phi, \end{aligned} \tag{5.5}$$

where Φ is the column with three components and λ is a spectral parameter. Though the evolution in x described by the second linear problem (5.5) is too simple, nevertheless the CS (5.2) or (5.3) has the meaning of the isospectral deformations of the matrix C_2 that is typical to the class of integrable systems (see e.g. [22, 23]).

CS (5.3) is the system of six equations for the structure constants D, E, G, L,M, N with free A, B, C:

$$\begin{aligned} D' &= DB + KC - AE - DG, \\ K' &= DE + KG - AM - DN, \\ E' &= MC - EG - D, \\ M' &= E^2 + MG - BM - EN - K, \\ G' &= GB + NC - CE - G^2 + A, \\ N' &= GE - CM + D, \end{aligned} \tag{5.6}$$

where $D' = x\partial D/\partial x$ and so forth. Here we will consider only simple particular cases of the CS (5.6). First it corresponds to the constraint A = 0, B = 0, C = 0, that is, to the nilpotent P_1. The corresponding solution is

$$\begin{aligned} &D = \frac{\beta}{\ln x}, \quad E = -\beta + \frac{\gamma}{\ln x}, \quad G = \frac{1}{\ln x}, \quad K = \alpha\beta + 2\beta^2 + \delta\ln x - \frac{\beta\gamma}{\ln x}, \\ &\qquad M = \alpha\gamma + 3\beta\gamma + \mu\ln x - \delta(\ln x)^2 - \frac{\gamma^2}{\ln x}, \quad N = \alpha + \beta - \frac{\gamma}{\ln x}, \end{aligned} \tag{5.7}$$

where α, β, γ, δ, μ are arbitrary constants. The three integrals for this solution are

$$I_1 = \alpha, \quad I_2 = \frac{1}{2}\alpha^2 + 3\beta^2 + 2\alpha\beta + \mu,$$
$$I_3 = \frac{1}{3}\left((\alpha+\beta)^3 - \beta^3\right) + (\alpha+\beta)(\mu + \beta(\alpha + 2\beta)) - \gamma\delta. \tag{5.8}$$

The second example is given by the constraint B = 0, C = 1, G = 0 for which the quantum CS (3.3) is equivalent to the Boussinesq equation [12]. Under this constraint the CS (5.6) is reduced to the single equation:

$$E'' - 6E^2 + 4\alpha E + \beta = 0, \tag{5.9}$$

and the other structure constants are given by

$$A = 2E - \alpha, \quad B = 0, \quad C = 1, \quad D = \gamma - \frac{1}{2}E', \quad G = 0,$$
$$K = -E^2 + \alpha E + \frac{1}{2}\beta, \quad M = \gamma + \frac{1}{2}E', \quad N = \alpha - N, \tag{5.10}$$

where α, β, γ are arbitrary constants. The corresponding first integrals are

$$I_1 = \alpha, \quad I_2 = \frac{1}{2}\left(\beta + \alpha^2\right), \quad I_3 = \frac{1}{3}\alpha^3 + \gamma^2 + \frac{1}{2}\alpha\beta - \frac{1}{4}(E')^2 + E^3 - \alpha E^2 - \frac{1}{2}\beta E. \tag{5.11}$$

Integral I_3 reproduces the well-known first integral of (5.9). Solutions of (5.9) are given by elliptic integrals (see e.g., [24]). Any such solution together with the formulae (5.10) describes deformation of the three-dimensional algebra *A* driven by DDA L_{2a}.

Now we will consider deformations of the two-dimensional algebra *A* without unit element according to the first option mentioned in the previous section. In this case the CS has the form (5.3) with the 2 × 2 matrices

$$C_1 = \begin{pmatrix} B & E \\ C & G \end{pmatrix}, \quad C_2 = \begin{pmatrix} E & M \\ G & N \end{pmatrix} \tag{5.12}$$

or in components

$$E' = MC - EG,$$
$$M' = E^2 + MG - BM - EN,$$
$$G' = GB + NC - CE - G^2,$$
$$N' = GE - CM. \tag{5.13}$$

In this case there are two independent integrals of motion:

$$I_1 = E + N, \qquad I_2 = \frac{1}{2}\left(E^2 + N^2 + 2MG\right). \tag{5.14}$$

The corresponding spectral problem is given by (5.5). Eigenvalues of the matrix C_2, that is, $\lambda_{1,2} = (1/2)(E + N \pm \sqrt{(E-N)^2 + 4GM})$ are invariant under deformations and $C_2 = (1/2)I_1^2 - I_2$. We note also an obvious invariance of (5.6) and (5.13) under the rescaling of *x*.

The system of (5.13) contains two arbitrary functions B and C. In virtue of the possible rescaling $P_1 \to \mu_1 P_1$, $P_2 \to \mu_2 P_2$ of the basis for the algebra *A* with two arbitrary functions μ_1, μ_2, one has four nonequivalent choices (1) B = 0, C = 0, (2) B = 1, C = 0 (3) B = 0, C = 1, and (4) B = 1, C = 1.

In the case B = 0, C = 0 (nilpotent P_1) the solution of the system (5.13) is

$$B = 0, \quad C = 0, \quad E = \frac{\beta}{\ln x}, \quad G = \frac{1}{\ln x}, \quad M = \gamma \ln x - \frac{\beta^2}{\ln x} + \alpha\beta, \quad N = -\frac{\beta}{\ln x} + \alpha, \tag{5.15}$$

where α, β, γ are arbitrary constants. For this solution the integrals are equal to $I_1 = \alpha$, $I_2 = \gamma + (1/2)\alpha^2$, and $\lambda_{1,2} = (1/2)(\alpha + \sqrt{\alpha^2 + 4\gamma})$.

At B = 1, C = 0 the system (5.13) has the following solution:

$$B = 1, \quad C = 0, \quad E = \frac{\gamma}{x+\beta}, \quad G = \frac{x}{x+\beta},$$

$$M = \delta + \left(\alpha\gamma + \beta\delta - \frac{\gamma^2}{\beta}\right)\frac{1}{x} + \frac{\gamma^2}{\beta(x+\beta)}, \quad N = -\frac{\gamma}{x+\beta} + \alpha, \tag{5.16}$$

where α, β, γ, δ are arbitrary constants. The integrals are $I_1 = \alpha, I_2 = \delta + (1/2)\alpha^2$. The formulae (5.15) and (5.16) provide us with explicit deformations of the structure constants.

In the last two cases the CS (5.13) is equivalent to the simple third-order ordinary differential equations. At B = 0, C = 1 with additional constraint $I_1 = 0$ one gets

$$G''' + 2G^2G' + 4(G')^2 + 2GG'' = 0 \tag{5.17}$$

while at B = 1, C = 1, and $I_1 = 0$ the system (5.13) becomes

$$G''' + 2G^2G' + 4(G')^2 + 2GG'' - G' = 0. \tag{5.18}$$

The second integral for these ODEs is

$$I_2 = -\frac{1}{2}G^4 + \frac{1}{2}(G')^2 - 2G^2G' - GG'' + \frac{1}{2}BG^2. \tag{5.19}$$

Equation (5.17) with $G' = \partial G/\partial y$ is the Chazy V equation from the well-known Chazy-Bureau list of the third-order ODEs having Painlevé property [25, 26]. The integral (5.19) is known too (see e.g. [27]).

The appearance of the Chazy V equation among the particular cases of the system (5.13) indicates that for other choices of B and C the CS (5.13) may be equivalent to the other notable third-order ODEs. It is really the case. Here we will consider only the reduction C = 1 with $I_1 = N + E = 0$. In this case the system (5.13) is reduced to the following equation:

$$G''' + 2G^2G' + 4(G')^2 + 2GG'' - 2G'\Phi - G\Phi' = 0, \qquad (5.20)$$

where $\Phi = B' + (1/2)B^2$. The second integral is

$$I_2 = -\frac{1}{2}G^4 + \frac{1}{2}(G')^2 - 2G^2G' - GG'' + \Phi G^2, \qquad (5.21)$$

and $\lambda_{1,2} = \pm\sqrt{I_2/2}$.

Choosing particular B or Φ, one gets equations from the Chazy-Bureau list. Indeed, at Φ = 0 one has the Chazy V equation (5.17). Choosing $\Phi = G'$, one gets the Chazy VII equation:

$$G''' + 2G^2G' + 2(G')^2 + GG'' = 0. \qquad (5.22)$$

At B = 2G (5.20) becomes the Chazy VIII equation:

$$G''' - 6G^2G' = 0. \qquad (5.23)$$

Choosing the function Φ such that

$$\left(6\Phi e^{(1/3)G}\right)' = 2G^2G' + (G')^2 + 4GG'', \qquad (5.24)$$

one gets the Chazy III equation:

$$G''' - 2GG'' + 3(G')^2 = 0. \qquad (5.25)$$

In the above particular cases the integral I_2 (5.21) is reduced to those given in [27].

All Chazy equations presented above have the Lax representation (5.3) with $E = -N = -(1/2)(G' + G^2 + GB$, $M = -(1/2)(G'' + 3GG' + G^3 + G^2B + (GB)')$, C = 1, and the proper choice of B.

Solutions of all these Chazy equations provide us with the deformations of the structure constants (5.12) for the two-dimensional algebra *A* generated by the DDA L_{2a}.

(2) Now we pass to the DDA L_{2b}. The commutation relations (4.3) imply that

$$[p_1, \varphi(x)] = (T-1)\varphi(x)\cdot p_1, \quad [p_2, \varphi(x)] = 0, \tag{5.26}$$

where φ(x) is an arbitrary function and Tφ(x) = φ(x + 1). Using (5.26), one finds the corresponding CS:

$$\sum_{m=0}^{2}\Big((\Delta_l+1)C^m_{jk}(x)\cdot C^n_{lm}(x) = (\Delta_j+1)C^m_{kl}(x)\cdot C^n_{jm}(x)\Big), \quad j,k,l,n = 0,1,2, \tag{5.27}$$

where Δ_1 = T − 1, Δ_2 = 0. In terms of the matrices C_1 and C_2, this CS is

$$C_1TC_2 = C_2C_1. \tag{5.28}$$

For nondegenerated matrix C_1 one has

$$TC_2 = C_1^{-1}C_2C_1. \tag{5.29}$$

The CS (5.29) is the discrete version of the Lax equation (5.3) and has similar properties. It has three independent first integrals:

$$I_1 = \operatorname{tr} C_2, \qquad I_2 = \frac{1}{2}\operatorname{tr}(C_2)^2, \qquad I_3 = \frac{1}{3}\operatorname{tr}(C_2)^3, \tag{5.30}$$

and it represents itself the compatibility condition for the linear problems:

$$\begin{aligned}\Phi C_2 &= \lambda\Phi,\\ T\Phi &= \Phi C_1.\end{aligned} \tag{5.31}$$

Note that det C_2 is the first integral too.

The CS (5.28) is the discrete dynamical system in the space of the structure constants. For the two-dimensional algebra *A* with matrices (5.12) it is

$$\begin{aligned}&BTE + ETG = EB + MC,\\ &BTM + ETN = E^2 + MG,\\ &CTE + GTG = BG + CN,\\ &CTM + GTN = EG + NG,\end{aligned} \tag{5.32}$$

where B and C are arbitrary functions. For nondegenerated matrix C_1, that is, at BG − CE ≠ 0, one has the resolved form (5.29), that is,

$$\begin{aligned}TE &= \frac{GM-EN}{BG-CE}C, \qquad TG = B + \frac{BN-CM}{BG-CE}C,\\ TM &= \frac{GM-EN}{BG-CE}G, \qquad TN = E + \frac{BN-CM}{BG-CE}G.\end{aligned} \tag{5.33}$$

This system defines discrete deformations of the structure constants.

NILPOTENT DDA

For the nilpotent DDA L_3, in virtue of the defining relations (4.5), one has

$$[p_1, \varphi(x)] = 0, \qquad [p_2, \varphi(x)] = \frac{\partial\varphi}{\partial x}\cdot p_1 \tag{6.1}$$

or

$$[p_j, \varphi(x)] = \frac{\partial\varphi}{\partial x}\cdot\sum_{k=1}^{2} a_{jk}p_k, \tag{6.2}$$

where $a_{21} = 1$, $a_{11} = a_{12} = a_{22} = 0$. Using (6.2), one gets the following CS:

$$\sum_{q=1}^{2} a_{lq}\sum_{m=0}^{2} C^n_{qm}\frac{\partial C^m_{jk}}{\partial x} - \sum_{q=1}^{2} a_{jq}\sum_{m=0}^{2} C^n_{qm}\frac{\partial C^m_{kl}}{\partial x} + \sum_{m=0}^{2}\left(C^m_{jk}C^n_{lm} - C^m_{kl}C^n_{jm}\right) = 0, \quad j,k,l,n = 0,1,2. \tag{6.3}$$

In the matrix form it is

$$C_1\frac{\partial C_1}{\partial x} = [C_1, C_2]. \tag{6.4}$$

For invertible matrix C_1

$$\frac{\partial C_1}{\partial x} = C_1^{-1}[C_1, C_2]. \tag{6.5}$$

This system of ODEs has three independent first integrals:

$$I_1 = \operatorname{tr} C_1, \qquad I_2 = \frac{1}{2}\operatorname{tr}(C_1)^2, \qquad I_3 = \frac{1}{3}\operatorname{tr}(C_1)^3, \tag{6.6}$$

and it is equivalent to the compatibility condition for the linear system:

$$\begin{aligned} C_1\Phi &= \lambda\Phi, \\ C_1\frac{\partial\Phi}{\partial x} &+ C_2\Phi = 0. \end{aligned} \tag{6.7}$$

So, as in the previous section the CS (6.4) describes isospectral deformations of the matrix C_1. This CS governs deformations generated by L_3.

For the two-dimensional algebra *A* without unit element the CS is given by (6.4) with the matrices (5.12). First integrals in this case are $I_1 = B + G, I_2 = (1/2)(B^2 + G^2 + 2CE)$ and $\det C_1 = (1/2)I_1^2 - I_2$. Since $\det C_1$ is a constant on the solutions of the system, then at $\det C_1 \neq 0$ one can always introduce the variable y defined by $x = y \det C_1$ such that CS (6.5) takes the form

$$\begin{aligned} B' &= EBG + ENC - GMC - CE^2, \\ E' &= GBM + GEN - ECM - MG^2, \\ C' &= BCE + BG^2 + MC^2 - CEG - BNC - GB^2, \\ G' &= CMG + CE^2 - CEN - BGE, \end{aligned} \tag{6.8}$$

where $B' = \partial B/\partial y$ and so forth and M, N are arbitrary functions. At $\det C_1 = BG - CE = 1$ this system becomes

$$\begin{aligned} B' &= E + C(EN - GM), \\ E' &= M + G(EN - GM), \\ C' &= G - B + C(MC - BN), \\ G' &= -E - C(EN - GM). \end{aligned} \tag{6.9}$$

Choosing M = N = 0, one gets

$$B' = E, \quad E' = 0, \quad C' = G - B, \quad G' = -E. \tag{6.10}$$

The solution of this system is

$$E = \alpha, \quad B = \alpha y + \beta, \quad G = -\alpha y + \gamma, \quad C = -y^2 + (\gamma - \beta)y + \delta, \tag{6.11}$$

where α, β, γ, δ are arbitrary constants subject to the constraint βγ − αδ = 1. First integrals for this solution are $I_1 = \beta + \gamma, I_2 = (1/2)(\beta^2 + \gamma^2 + 2\alpha\delta)$.

With the choice M = 0, N = 1 and under the constraint $I_1 = B + G = 0$ the system (6.8) takes the form

$$B' = (1 + C)E, \quad E' = -BE, \quad C' = -(2 + C)B. \tag{6.12}$$

This system can be written as a single equation in the different equivalent forms. One of them is

$$(E')^2 + \alpha E^4 - 2E^3 + E^2 = 0, \tag{6.13}$$

where α is an arbitrary constant and

$$B^2 = -1 - \alpha E^2 + 2E, \quad C = \alpha E - 2, \quad G = -B. \tag{6.14}$$

The second integral is equal to −1.

Solutions of (6.13) can be expressed through the elliptic integrals. Solutions of (6.13) and the formulae (6.14) define deformations of the structure constants driven by DDA L_3.

SOLVABLE DDAS

(1) For the solvable DDA L_4 the relations of (4.5) imply that

$$[p_j, \varphi(x)] = (T-1)\varphi(x)p_j, \quad j = 1, 2, \qquad (7.1)$$

where φ(x) is an arbitrary function and T is the shift operator Tφ(x) = φ(x + 1). With the use of (7.1) one arrives at the following CS:

$$C_1TC_2 = C_2TC_1. \qquad (7.2)$$

For nondegenerated matrix C_1 (7.2) is equivalent to the equation $T(C_2C_1^{-1}) = C_1^{-1}C_2$ or

$$TU = C_1^{-1}UC_1, \qquad (7.3)$$

where $U \doteqdot C_2C_1^{-1}$. Using this form of the CS, one promptly concludes that the CS (7.2) has three independent first integrals:

$$I_1 = \mathrm{tr}\left(C_2C_1^{-1}\right), \qquad I_2 = \frac{1}{2}\,\mathrm{tr}\left(C_2C_1^{-1}\right)^2, \qquad I_3 = \frac{1}{3}\,\mathrm{tr}\left(C_2C_1^{-1}\right)^3, \qquad (7.4)$$

and it is representable as the commutativity condition for the linear system:

$$\begin{aligned} &\Phi C_2C_1^{-1} = \lambda\Phi, \\ &T\Phi = \Phi C_1. \end{aligned} \qquad (7.5)$$

For the two-dimensional algebra *A* one has the CS (7.2) with the matrices (5.12). It is the system of four equations for six functions:

$$\begin{aligned} &BTE + ETG = ETB + MTC, \\ &BTM + ETN = ETE + MTG, \\ &CTE + GTG = GTB + NTC, \\ &CTM + GTN = GTE + NTG. \end{aligned} \qquad (7.6)$$

Choosing B and C as free functions and assuming that BG − CE ≠ 0, one can easily resolve (7.6) with respect to TE, TG, TM, TN. For instance, with B = C = 1 one gets the following four-dimensional mapping:

$$TE = M - E\frac{M-N}{E-G}, \quad TG = 1 + \frac{M-N}{E-G},$$
$$TM = N + (N-G)\frac{M-N}{E-G} - G\left(\frac{M-N}{E-G}\right)^2,$$
$$TN = M + (1-E)\frac{M-N}{E-G} + \left(\frac{M-N}{E-G}\right)^2. \tag{7.7}$$

(2) In a similar manner one finds the CS associated with the solvable DDA L_5. Since in this case

$$[p_1, \varphi(x)] = (T-1)\varphi(x)p_1, \qquad [p_2, \varphi(x)] = \left(T^{-1}-1\right)\varphi(x)p_2, \tag{7.8}$$

the CS takes the form

$$C_1TC_2 = C_2T^{-1}C_1. \tag{7.9}$$

For nondegenerated C_2 it is equivalent to

$$TV = C_2VC_2^{-1}, \tag{7.10}$$

where $V \doteqdot T^{-1}C_1 \cdot C_2$. Similar to the previous case the CS has three first integrals:

$$I_1 = \mathrm{tr}(C_1TC_2), \qquad I_2 = \frac{1}{2}\,\mathrm{tr}\,(C_1TC_2)^2, \qquad I_3 = \frac{1}{3}\,\mathrm{tr}\,(C_1TC_2)^3, \tag{7.11}$$

and it is equivalent to the compatibility condition for the linear system:

$$\left(T^{-1}C_1\right)C_2\Phi = \lambda\Phi,$$
$$T\Phi = C_2\Phi. \tag{7.12}$$

Note that the CS (7.9) is of the form (3.11) with $T_1 = T$, $T_2 = T^{-1}$. Thus, the deformations generated by L_5 can be considered as the reductions of the discrete deformations (3.11) under the constraint $T_1T_2C^n_{jk} = C^n_{jk}$.

A class of solutions of the CS (7.9) is given by

$$C_j = g^{-1}T_jg, \tag{7.13}$$

where g is 3 × 3 matrix and $T_0 = 1$, $T_1 = T$, $T_2 = T^{-1}$. Since $C^n_{jk} = C^n_{kj}$, one has $T_jg^m_l = T_lg^m_j$ and hence $g^m_j = T_j\Phi^m$ where Φ^0, Φ^1, Φ^2 are arbitrary functions. So, this subclass of deformations are defined by three arbitrary functions.

To describe the isoassociative deformations for which $C_1(x)C_2(x) = C_2(x)C_1(x)$ for all x these functions should obey the systems of equations:

$$\sum_{l,t=0}^{2} T_j T_t \Phi^n \cdot \left(g^{-1}\right)_l^t \cdot T_k T_m \Phi^l = \sum_{l,t=0}^{2} T_k T_t \Phi^n \cdot \left(g^{-1}\right)_l^t \cdot T_j T_m \Phi^l, \quad j,k,n,m = 0,1,2. \tag{7.14}$$

It is a version of the discrete oriented associativity equation.

ACKNOWLEDGMENT

The author is very grateful to the referees for careful reading of the manuscript and various useful and fruitful remarks.

REFERENCES

1. M. Gerstenhaber, “On the deformation of rings and algebras,” *Annals of Mathematics*, vol. 79, pp. 59–103, 1964.
2. M. Gerstenhaber, “On the deformation of rings and algebras. II,” *Annals of Mathematics*, vol. 84, pp. 1–19, 1966.
3. E. Witten, “On the structure of the topological phase of two-dimensional gravity,” *Nuclear Physics B*, vol. 340, no. 2-3, pp. 281–332, 1990.
4. R. Dijkgraaf, H. Verlinde, and E. Verlinde, “Topological strings in d<1,” *Nuclear Physics B*, vol. 352, no. 1, pp. 59–86, 1991.
5. B. Dubrovin, “Integrable systems in topological field theory,” *Nuclear Physics B*, vol. 379, no. 3, pp. 627–689, 1992.
6. B. Dubrovin, “Geometry of 2D topological field theories,” in *Integrable Systems and Quantum Groups (Montecatini Terme, 1993)*, vol. 1620 of *Lecture Notes in Mathematics*, pp. 120–348, Springer, Berlin, Germany, 1996.
7. C. Hertling and Y. I. Manin, “Weak Frobenius manifolds,” *International Mathematics Research Notices*, no. 6, pp. 277–286, 1999.
8. Y. I. Manin, *Frobenius Manifolds, Quantum Cohomology, and Moduli Spaces*, vol. 47 of *American Mathematical Society Colloquium Publications*, American Mathematical Society, Providence, RI, USA, 1999.
9. C. Hertling and M. Marcolli, Eds., *Frobenius Manifolds, Quantum Cohomology and Singularities*, vol. E36 of *Aspects of Mathematics*, Friedrich Vieweg & Sohn, Wiesbaden, Germany, 2004.
10. B. G. Konopelchenko and F. Magri, “Coisotropic deformations of associative algebras and dispersionless integrable hierarchies,” *Communications in Mathematical Physics*, vol. 274, no. 3, pp. 627–658, 2007.
11. B. G. Konopelchenko and F. Magri, “Dispersionless integrable equations as coisotropic deformations: generalizations and reductions,” *Theoretical and Mathematical Physics*, vol. 151, no. 3, pp. 803–819, 2007.
12. B. G. Konopelchenko, “Quantum deformations of associative algebras and integrable systems,” *Journal of Physics A*, vol. 42, no. 9, Article ID 095201, 18 pages, 2009.

13. B. G. Konopelchenko, “Discrete integrable systems and deformations of associative algebras,” *Journal of Physics A*, vol. 42, no. 45, Article ID 454003, 35 pages, 2009.

14. B. G. Konopelchenko and F. Magri, “Yano manifolds, F-manifolds and integrable systems,” unpublished.

15. B. L. van der Waerden, *Algebra*, Springer, New York, NY, USA, 1971.

16. N. Bourbaki, *Groupes et Algebres de Lie*, Hermann, Paris, France, 1972.

17. J. Dieudonné, “Les déterminants sur un corps non commutatif,” *Bulletin de la Société Mathématique de France*, vol. 71, pp. 27–45, 1943.

18. I. M. Gelfand and V. S. Retakh, “Determinants of matrices over noncommutative rings,” *Functional Analysis and Its Applications*, vol. 25, no. 2, pp. 91–102, 1991.

19. P. A. M. Dirac, *Lectures on Quantum Mechanics*, Yeshiva University, New York, NY, USA, 1964.

20. A. Losev and Y. I. Manin, “Extended modular operads,” in *Frobenius Manifolds, Quantum Cohomology and Singularities*, C. Hertling and M. Marcoli, Eds., vol. E36 of *Aspects Mathematics*, pp. 181–211, Vieweg, Wiesbaden, Germany, 2004.

21. K. Yano and M. Ako, “On certain operators associated with tensor fields,” *K dai Mathematical Seminar Reports*, vol. 20, pp. 414–436, 1968.

22. S. Novikov, S. V. Manakov, L. P. Pitaevskiĭ, and V. E. Zakharov, *Theory of Solitons*, Contemporary Soviet Mathematics, Plenum, New York, NY, USA, 1984.

23. M. J. Ablowitz and H. Segur, *Solitons and the Inverse Scattering Transform*, vol. 4 of *SIAM Studies in Applied Mathematics*, SIAM, Philadelphia, Pa, USA, 1981.

24. H. F. Baker, *Abelian Functions*, Cambridge Mathematical Library, Cambridge University Press, Cambridge, UK, 1995.

25. J. Chazy, “Sur les équations différentielles du troisième ordre et d’ordre supérieur dont l’intégrale générale a ses points critiques fixes,” *Acta Mathematica*, vol. 34, no. 1, pp. 317–385, 1911.

26. F. J. Bureau, “Differential equations with fixed critical point. II,” *Annali di Matematica Pura ed Applicata. Serie Quarta*, vol. 66, pp. 1–116, 1964.
27. C. M. Cosgrove, “Chazy classes IX-XI of third-order differential equations,” *Studies in Applied Mathematics*, vol. 104, no. 3, pp. 171–228, 2000.

Chapter

THE BOOLEAN ALGEBRA AND CENTRAL GALOIS ALGEBRAS

14

George Szeto and Lianyong Xue

Department of Mathematics, Bradley University, USA

ABSTRACT

Let B be a Galois algebra with Galois group G, $J_g = \{b \in B \mid b\ x = g(x)\ b$ for all $x \in B\}$ for $g \in G$, and $BJ_g = Be_g$ for a central idempotent e_g. Then a relation is given between the set of elements in the Boolean algebra $(B_a, \leq)$ generated by $\{0, e_g \mid g \in G\}$ and a set of subgroups of G, and a central Galois algebra Be with a Galois subgroup of G is characterized for an $e \in B_a$.

INTRODUCTION

Galois theory of rings have been intensively studied [1, 3, 4, 5, 6, 7]. Let B be a Galois algebra with Galois group G and $J_g = \{b \in B \mid bx = g(x)b$ for all

Citation: George Szeto, Lianyong Xue, "The Boolean algebra and central Galois algebras", International Journal of Mathematics and Mathematical Sciences, vol. 28, Article ID 184907, 6 pages, 2001. https://doi.org/10.1155/S0161171201007104.

$x \in B\}$ for each $g \in G$. In [4], it was shown that $BJ_g = Be_g$ for some central idempotent e_g of B. Let B_a be the Boolean algebra generated by $\{0, e_g \mid g \in G\}$. In [7], the following structure theorem for B was given: there exist $\{e_i \in B_a \mid i = 1,2,...,m$ for some integer $m\}$ and some subgroups H_i of G such that $B = \oplus \sum_{i=1}^{m} Be_i \oplus B(1 - \sum_{i=1}^{m} e_i)$ where Be_i is a central Galois algebra with Galois group H_i for each $i = 1,2,...,m$ and $B(1 - \sum_{i=1}^{m} e_i) = C(1 - \sum_{i=1}^{m} e_i)$ which is a commutative Galois algebra with Galois group induced by and isomorphic with G in case $1 \neq \sum_{i=1}^{m} e_i,$ where C is the center of B. We observe that (1) $e_i = \prod_{h \in H_i} e_h$ which is a nonzero monomial in B_a for a maximal subset H_i of G, (2) H_i is a subgroup of G, and (3) Be_i is a central Galois algebra with Galois group H_i. In the present paper, we will discuss a general case: what kind of elements e in B_a and subgroups H_e give a central Galois algebra Be with Galois group H_e? We will show that (1) for any nonzero monomial $e = \prod_{g \in S} e_g$ of B_a for some subset S of G, let $H_e = \{g \in G \mid e \leq e_g$, that is, $ee_g = e\}$; then H_e is a subgroup of G, (2) when $H_e \neq \{1\}$, Be is a central Galois algebra with Galois group H_e if and only if e is a nonzero minimal element in B_a (i.e., Be is one of the components of B as given in [7, Theorem 3.8]), (3) for a nonzero monomial $e = \Pi_{g \in S}\, e_g$ of B_a for some subset S of G, let $T_e = \{g \in G \mid e = e_g\}$; then T_e is a subgroup of G if and only if $e = 1$, and (4) let $H_1 = \{g \in G \mid e_g = 1\}$. Then $e_g = 0$ for each g ⍰ H_1 if and only if B is either a central Galois algebra with Galois group H_1 or a commutative Galois algebra with Galois group G. Thus, $\{Be \mid e$ is a nonzero minimal element in $B_a\}$ are the only central Galois algebras with Galois group H_e arising from nonzero monomials e in B_a, and when $B_a = \{0,1\}$, B is a central Galois algebra with Galois group H_1 and the center C is a commutative Galois algebra with Galois group G/H_1. This fact generalizes the DeMeyer theorem for a Galois algebra with an indecomposable center C (see [1, Theorem 1]).

DEFINITIONS AND NOTATIONS

Let B be a ring with 1, C the center of B, G an automorphism group of B of order n for some integer n, and B^G the set of elements in B fixed under each element in G. B is called a Galois extension of B^G with Galois group G if there exist elements $\{a_i, b_i$ in B, $i = 1,2,...,m\}$ for some integer m such that $\sum_{i=1}^{m} a_i g(b_i) = \delta_{1,g}$ for each $g \in G$. B is called a Galois algebra over R if B is a Galois extension of R which is contained in C, and B is called a central Galois ex-tension if B is a Galois extension of C. Throughout this paper, we assume that B is a Galois algebra with Galois group G. Let $J_g = \{b \in B \mid bx = g(x)b$ for all $x \in B\}$ and $J_g^{(A)} = \{b \in A \mid bx = g(x)b \text{ for all } x \in A\}$ for each $g \in G$, where $A \subset B$. In [4], it was shown that $BJ_g = Be_g$ for some central idempotent e_g of B. We denote by B_a the Boolean algebra generated by $\{0, e_g \mid g \in G; \leq\}$, where $e \leq e'$ if $ee = e$.

THE MONOMIALS AND SUBGROUPS

Let e be a nonzero monomial of B_a, $e = \Pi_{g \in S} e_g$ for a subset S of G. We have two subsets of G, $H_e = \{g \in G \mid e \leq e_g\}$ and $T_e = \{g \in G \mid e = e_g\}$. We are going to show that H_e is a subgroup of G, and that T_e is a subgroup of G if and only if $e = 1$. Let K be a subgroup of G. Then K is called a nonzero subgroup of G if $\Pi_{k \in K} e_k \neq 0$, and K is called a maximal nonzero subgroup of G if $K \subset K'$, where K' is a nonzero subgroup of G such that $\Pi_{k \in K} e_k = \Pi_{k \in K'} e_k$, then $K = K'$. We note that each nonzero subgroup is contained in a unique maximal nonzero subgroup of G. We will show that there exists a one-to-one correspondence between the following three sets: (1) the set of nonzero monomials in B_a, (2) the set of maximal nonzero subgroups of G, and (3) the set of Galois extensions in B generated by a nonzero monomial e with a maximal Galois subgroup of G.

Lemma 3.1

Let e be a nonzero monomial in B_a and $H_e = \{g \in G \mid e \leq e_g\}$. Then H_e is a subgroup of G.

Proof

For any $g, h \in H_e$, $e \leq e_g$, and $e \leq e_h$. Hence $e \leq e_g e_h$. But $J_g J_h \subset J_{gh}$, so $BJ_g J_h \subset BJ_{gh}$. Therefore $Be_g e_h \subset Be_{gh}$. Thus $e_g e_h \leq e_{gh}$; and so $e \leq e_g e_h \leq e_{gh}$. This implies that $gh \in H_e$. Noting that G is finite, we conclude that H_e is a subgroup of G.

Theorem 3.2

There exists a one-to-one correspondence between the set of nonzero monomials in B_a and the set of maximal nonzero subgroups of G.

Proof

Define $f : e \rightarrow H_e$ for a nonzero monomial e in B_a, where H_e is given in Lemma 3.1. By Lemma 3.1, H_e is a subgroup of G. Also, by the definition of H_e, it is easy to see that H_e is a maximal nonzero subgroup of G. Thus f is well defined. Next we show that f is one to one. Let e and e' be two nonzero monomials in B_a such that $f(e) = f(e')$, that is, $H_e = H_{e'}$. Then $e = \Pi_{h\in H_e} e_h = \Pi_{h\in H_{e'}} e_h = e'$. Thus f is one to one. Moreover, let K be a maximal nonzero subgroup of G. Then $e = \Pi_{k\in K} e_k \neq 0$ and $K = \{g \in G \mid e \leq e_g\}$ by the definition of a maximal nonzero subgroup of G. Thus $f(e) = K$. Therefore f is a bijection.

Let $N(H_e)$ be the normalizer of H_e in G for a nonzero monomial e in B_a. We next show that Be is a Galois extension with a maximal Galois subgroup $G(e)$ where $G(e) = \{g \in G \mid g(e) = e\}$, and $G(e) = N(H_e)$. Consequently, we can establish a one-to-one correspondence between the set of maximal nonzero subgroups of G and the set of Galois extensions in B generated by a nonzero monomial e with a maximal Galois subgroup of $N(H_e)$.

Lemma 3.3

For a nonzero monomial e in B_a, let $G(e) = \{g \in G \mid g(e) = e\}$. Then, (1) *$G(e) = N(H_e)$, where $N(H_e)$ is the normalizer of H_e in G, and* (2) *Be is a Galois extension with a maximal Galois subgroup of* $G(e)|_{Be} \cong G(e)$.

Proof

(1) For any $g \in N(H_e)$, since $Be = B\Pi_{h\in H_e} e_h = B\Pi_{h\in H_e} J_h, g(Be) = g(B\Pi_{h\in H_e} J_h)$ $= B\Pi_{h\in H_e} J_{ghg^{-1}} = B\Pi_{h\in gH_eg^{-1}} J_h =$ $B\Pi_{h\in H_e} J_h = Be$ (for $gHg^{-1} = H$). Hence $g(e) = e$; and so $g \in G(e)$. Conversely, for any $g \in G(e)$,

$$Be = g(Be) = g(B\Pi_{h\in H_e} e_h) = g(B\Pi_{h\in H_e} J_h) = B\Pi_{h\in H_e} J_{ghg^{-1}} = B\Pi_{h\in H_e} e_{ghg^{-1}}. \tag{3.1}$$

Thus $e = \Pi_{h\in H_e} e_{ghg^{-1}}$. Therefore $e \leq e_{ghg^{-1}}$; and so $ghg^{-1} \in H_e$ for each $h \in H_e$. This implies that $g \in N(H_e)$.

(2) Since B is a Galois algebra with Galois group G and $e \in C^{G(e)}$, Be is a Galois extension with a maximal Galois subgroup of

$G(e)|_{Be} \cong G(e)$ (see [7, proof of Lemma 3.7]). Moreover, let $g \in G$ but $g \notin G(e)$. Then $g(e) \neq e$. Thus g is not an automorphism of Be; and so $G(e)$ is the maximal Galois group contained in G for Be.

Theorem 3.4

There exists a one-to-one correspondence between the set of maximal nonzero subgroups of G and the set of Galois extensions in B generated by a nonzero monomial e with a maximal Galois subgroup $G(e)|_{Be} \cong G(e)$ *such that* $G(e) = N(H_e)$.

Proof

Let $\alpha : e \rightarrow Be$ for each nonzero monomial e in B_a. Then, by Lemma 3.3, Be is a Galois extension in B generated by e with a maximal Galois subgroup $G(e)|_{Be} \cong G(e)$ such that $G(e) = N(H_e)$. Clearly, α is a bijection from the set of nonzero monomials in B_a to the set of Galois extensions Be for a nonzero monomial e in B_a with a maximal Galois subgroup $G(e)|_{Be} \cong G(e)$ which is $N(H_e)$. Thus Theorem 3.4 is an immediate consequence of Theorem 3.2.

In the following, we show that the set $T_e = \{g \in G \mid e = e_g\}$ for a nonzero monomial e in B_a is not a subgroup of G unless $e = 1$.

Theorem 3.5

Let e be a nonzero monomial in B_a *and* $T_e = \{g \in G \mid e = e_g\}$. *Then* T_e *is a subgroup of G if and only if* $e = 1$.

Proof

Assume T_e is a subgroup of G. Then $1 \in T_e$; and so $e = e_1 = 1$. Conversely, assume $e = 1$. Then $T_e = T_1 = \{g \in G \mid 1 = e_g\}$. But the condition that $1 = e_g$ is equivalent to that $1 \leq e_g$, so $T_e = T_1 = H_1$ where H_1 is given in Lemma 3.1. Hence by Lemma 3.1, T_e is a subgroup of G.

CENTRAL GALOIS ALGEBRAS

In Section 3, Lemma 3.1 proves that for a nonzero monomial $e \in B_a$, H_e $(= \{g \in G \mid e \leq e_g\})$ is a subgroup of G. In [7], it was shown that if H is a maximal subset of G such that $\Pi_{h \in H} J_h \neq \{0\}$, then H is a subgroup of G.

We will show that the maximal subset H is exactly H_e for a minimal nonzero monomial $e \in B_a$. Thus Be is a central Galois algebra with Galois group H_e (see [7, Theorem 3.6]). Next is a characterization of the central Galois algebra Be with Galois group H_e for a nonzero monomial $e \in B_a$.

Theorem 4.1

Let e be a nonzero monomial in B_a such that $H_e \neq \{1\}$. The following statements are equivalent:

- Be is a central Galois algebra with Galois group $\mathrm{H_e}$.
- $eJ_g = \{0\}$ *for each* $g \notin H_e$.
- *e is a minimal nonzero monomial in B_a.*

Proof.

(1)⇒(2). Since B is a Galois algebra over a commutative ring R with Galois group $G, B = \oplus \sum_{g \in G} J_g$ (see [4, Theorem 1]). Hence

$$Be = \oplus \sum_{g \in G} eJ_g = \left(\oplus \sum_{h \in H_e} eJ_h \right) \oplus \left(\oplus \sum_{g \notin H_e} eJ_g \right). \tag{4.1}$$

By hypothesis, Be is a central Galois algebra with Galois group H_e, so $Be = \oplus \sum_{h \in H_e} J_h^{(Be)}$. But by [7, Lemma 3.3], $J_h^{(Be)} = eJ_h$ for each $h \in H_e$; and so $Be = \oplus \sum_{h \in H_e} eJ_h$. Thus $\oplus \sum_{g \notin H_e} eJ_g = \{0\}$, that is, $eJ_g = \{0\}$ for each $g \notin H_e$.

(2)⇒(1). Since $Be = \oplus \sum_{g \in G} eJ_g = (\oplus \sum_{h \in H_e} eJ_h) \oplus (\oplus \sum_{g \notin H_e} eJ_g)$ and $eJ_g = \{0\}$ for each $g \notin H_e, Be = \oplus \sum_{h \in H_e} eJ_h$. By [7, Lemma 3.3] again, $J_h^{(Be)} = eJ_h$ for each $h \in H_e$. Hence $Be = \oplus \sum_{h \in H_e} J_h^{(Be)}$, where $J_h^{(Be)} J_{h^{-1}}^{(Be)} = (eJ_h)(eJ_{h^{-1}}) = eJ_h J_{h^{-1}} = eC$ which is the center of Be. Moreover, B is a Galois R-algebra, so it is a separable R-algebra. Thus, Be is a separable algebra over Re (see [2, Proposition 1.11, page 46]). Therefore, Be is a central Galois algebra over Ce (see [3, Theorem 1]).

(3)⇒(2). Since e is a minimal nonzero monomial in B_a, for each $g \in G$, either $e \leq e_g$ or $ee_g = 0$. Since $e \leq e_g$ for each $g \in H_e$, we have that $ee_g = 0$ for each $g \notin H_e$. Therefore, $BeJ_g = Bee_g = \{0\}$; and so $eJ_g = \{0\}$ for each $g \notin H_e$.

(2)⇒(3). Suppose e is not a minimal nonzero monomial in B_a. Then there exists a $g \in G$ such that $0 < ee_g < e$. By the definition of H_e, $e = \Pi_{h \in H_e} e_h$; and so $ee_h = e$ for each $h \in H_e$. Hence $g \notin H_e$. Therefore, $BeJ_g = Bee_g \neq \{0\}$. This implies that $eJ_g \neq \{0\}$ for some $g \notin H_e$. This contradicts hypothesis (2). Thus statement (3) holds.

When e is a minimal nonzero monomial in B_a, Theorem 4.1 shows that Be is a central Galois algebra with Galois group H_e. Hence the order of H_e is a unit in Be (see [4, Corollary 3]). Moreover, by Lemma 3.3, Be is a Galois extension with Galois group $G(e)$ which is $N(H_e)$, so we have a structure of Be.

Theorem 4.2

For a minimal nonzero monomial e in $\mathrm{B_a}$, Be is a central Galois algebra with Galois group $\mathrm{H_e}$ and Ce is a commutative Galois algebra with Galois group $\mathrm{G(e)/H_e}$.

Proof

Since e is a minimal nonzero monomial in B_a, Be is a central Galois algebra with Galois group H_e by Theorem 4.1. Hence $|H_e|$, the order of H_e, is a unit in Ce. Moreover, by Lemma 3.3, Be is a Galois extension with Galois group $G(e)$ which is $N(H_e)$, so H_e is a normal subgroup of $G(e)$. Let $\{a_i, b_i \mid i = 1,2,...,m\}$ be a $G(e)$-Galois system for Be. Then, $\sum_{i=1}^{m} a_i g(b_i) = \delta_{1,g} e$ for each $g \in G(e)$. Let $x_i = (1/|H_e|) \sum_{h \in H_e} h(a_i)$ and $y_i = \sum_{h \in H_e} h(b_i)$. Then, x_i and y_i are invariant under each element in H_e. Hence, $x_i, y_i \in Ce$ since $(Be)^{H_e} = Ce$. It is straightforward to verify that $\{x_i, y_i\}$ is a $G(e)/H_e$-Galois system for Ce.

Theorem 4.1 characterizes a central Galois algebra Be for a minimal nonzero monomial $e \in B_a$. Next we want to characterize a central Galois algebra $B1$ for the maximal monomial 1 in B_a.

Theorem 4.3

Let $H_1 = \{h \in G \mid e_h = 1\}$. Then $e_g = 0$ for each $g \notin H_1$ if and only if B is either a central Galois algebra with Galois group H_1 or a commutative Galois algebra with Galois group G.

Proof

(⇒) Case 1. $H_1 \neq \{1\}$. Since $e_g = 0$ for each $g \notin H_1$, $J_g = \{0\}$ for each $g \notin H_1$. Hence, by (2)⇒(1) in Theorem 4.1, B (= $B1$) is a central Galois algebra with Galois group H_1. Case 2. $H_1 = \{1\}$. By hypothesis, $e_g = 0$ for each $g \neq 1$ in G, so $B = \oplus \sum_{g \in G} J_g = J_1 = C$. Thus B is a commutative Galois algebra with Galois group G.

(⇐) Assume B is a central Galois algebra with Galois group H_1. Then $H_1 \neq \{1\}$. Hence, by (1)⇒(2) in Theorem 4.1, $J_g = 1J_g = \{0\}$ for each $g \notin H_1$. Thus $e_g = 0$ for each $g \notin H_1$. Next, assume B is a commutative Galois algebra with Galois group G. Then $J_g = \{0\}$ for each $g \neq 1$ in G (see [3, Proposition 2]). Hence $e_g = 0$ for each $g \neq 1$ in G. Therefore $H_1 = \{1\}$ and $e_g = 0$ for each $g \notin H_1$.

As a consequence of Theorem 4.3, the DeMeyer theorem (see [1, Theorem 1]) for central Galois algebras with a connected center is generalized.

Corollary 4.4

Let B be a Galois algebra with Galois group G. If $B_a = \{0,1\}$, then B is a central Galois algebra with Galois group H_1 and C is a commutative Galois algebra with Galois group G/H_1.

Proof

Since $B_a = \{0 ,1\}$, $e_g = 0$ for each $g \notin H_1$; and so the corollary holds.

We conclude the present paper with an example of a Galois algebra B such that $B_a = \{0,1\}$, but its center C is not indecomposable.

Example 4.5

Let $R[i, j, k]$ be the quaternion algebra over the real field R, $B = R[i, j, k] \oplus R[i, j, k]$, and $G = \{1, g_i, g_j, g_k, g, gg_i, gg_j, gg_k\}$, where $g_i(a_1, a_2) = (ia_1i^{-1}, ia_2i^{-1})$, $g_j(a_1, a_2) = (ja_1j^{-1}, ja_2j^{-1})$, $g_k(a_1, a_2) = (ka_1k^{-1}, ka_2k^{-1})$, and $g(a_1, a_2) = (a_2, a_1)$ for all (a_1, a_2) in B. Then,

(1) B is a Galois extension with a G-Galois system: $\{a_1 = (1, 0), a_2 = (i, 0), a_3 = (j, 0), a_4 = (k, 0), a_5 = (0, 1), a_6 = (0, i), a_7 = (0, j), a_8 = (0, k); b_1 = (1/4)(1, 0), b_2 = -(1/4)(i, 0), b_3 = -(1/4)(j, 0), b_4 = -(1/4)(k, 0), b_5 = (1/4)(0, 1), b_6 = -(1/4)(0, i), b_7 = -(1/4)(0, j), b_8 = -(1/4)(0, k)\}$.

(2) $B^G = \{(r, r) \mid r \in R\} \cong R$.

(3) By (1) and (2), B is a Galois algebra over R with Galois group G.

(4) $J_1 = C = R \oplus R$, $J_{gi} = (Ri) \oplus (Ri)$, $J_{gj} = (Rj) \oplus (Rj)$, $J_{gk} = (Rk) \oplus (Rk)$, and $J_g = Jgg_i = Jgg_j = Jgg_k = \{0\}$.

(5) $BJ_1 = BJ_{gi} = BJ_{gj} = BJ_{gk} = B1$ and $BJ_g = BJ_{ggi} = BJ_{ggj} = BJ_{ggk} = \{0\}$. Hence $e_1 = e_{gi} = e_{gj} = e_{gk} = 1$ and $e_g = e_{ggi} = e_{ggj} = e_{ggk} = 0$. Thus $B_a = \{0,1\}$.

(6) $H_1 = \{1, g_i, g_j, g_k\}$ and B is a central Galois algebra with Galois group H_1.

(7) $C = R \oplus R$ which is a commutative Galois algebra with Galois group $G/H_1 \cong \{1, g\}$.

ACKNOWLEDGEMENT

This paper was written under the support of a Caterpillar Fellowship at Bradley University. The authors would like to thank the Caterpillar Inc. for the support.

REFERENCES

1. F. R. DeMeyer, Galois theory in separable algebras over commutative rings, Illinois J. Math. 10 (1966), 287–295. MR 33#149. Zbl 216.34001.
2. F. R. DeMeyer and E. Ingraham, Separable Algebras over Commutative Rings, Lecture Notes in Mathematics, vol. 181, Springer-Verlag, Berlin, 1971. MR 43#6199. Zbl 215.36602.
3. M. Harada, Supplementary results on Galois extension, Osaka J. Math. 2 (1965), 343–350. MR 33#151. Zbl 178.36903.
4. T. Kanzaki, On Galois algebra over a commutative ring, Osaka J. Math. 2 (1965), 309–317. MR 33#150. Zbl 163.28802.
5. G. Szeto and L. Xue, On three types of Galois extensions of rings, Southeast Asian Bull. Math. 23 (1999), no. 4, 731–736. CMP 1 810 837. Zbl 945.16023.
6. _______, On characterizations of a center Galois extension, Int. J. Math. Math. Sci. 23 (2000), no. 11, 753–758. MR 2001c:16061.
7. _______, The structure of Galois algebras, J. Algebra 237 (2001), no. 1, 238–246. CMP 1 813 896.

Chapter

ON ADDITION OF SETS IN BOOLEAN SPACE

15

Vladimir Leontiev[1], Garib Movsisyan[2], and Zhirayr Margaryan[3]

[1]Moscow State University, Moscow, Russia.
[2]BIT Group, Moscow, Russia.
[3]Yerevan State University, Yerevan, Armenia.

ABSTRACT

In many problems of combinatory analysis, operations of addition of sets are used (sum, direct sum, direct product etc.). In the present paper, as well as in the preceding one [1], some properties of addition operation of sets (namely, Minkowski addition) in Boolean space B^n are presented. Also, sums and multisums of various "classical figures" as: sphere, layer, interval etc. are considered. The obtained results make possible to describe multisums by such characteristics of summands as: the sphere radius, weight of layer, dimension of interval etc. using the methods presented in [2], as

Citation: Leontiev, V., Movsisyan, G. and Margaryan, Z. (2016), "On Addition of Sets in Boolean Space". Journal of Information Security, 7, 232-244. doi: 10.4236/jis.2016.74019.

well as possible solutions of the equation $X + Y = A$, where $X, Y, A \subseteq B^n$, are considered. In spite of simplicity of the statement of the problem, complexity of its solutions is obvious at once, when the connection of solutions with constructions of equidistant codes or existence the Hadamard matrices is apparent. The present paper submits certain results (statements) which are to be the ground for next investigations dealing with Minkowski summation operations of sets in Boolean space.

Keywords: Hadamard Matrices, Minkowski Addition, Multiset, Cardinality, Multisum, Interval, Quadrate, Boolean Space, Stabilizer, Additive Channel

Sum of Sets According to Minkowski

If $x = (x_1 x_2 \cdots x_n)$, $y = (y_1 y_2 \cdots y_n)$ are points in B^n, where B^n, is a Boolean space, then:

$$x + y = ((x_1 \oplus y_1)(x_2 \oplus y_2)\cdots(x_n \oplus y_n)),$$

where $\oplus$ is the mod 2 addition operation.

This addition operation for members of B^n can be extended in subsets of B^n.

In other words, if $X, Y \in 2^{B^n}$, then:

$$X + Y = \{x + y; x \in X, y \in Y\}. \qquad (1)$$

Thus, the sum of subsets $X + Y$ is consisted of sums of points belonging to X and Y, respectively.

Examples.

1. if $X \in 2B^n$, $y \in B^n$, then $\{X + y\}$ is the “shift” of the set X to the point y, and $|X + Y| = |X|$.
2. if X is a subset in B^n, then $X + X = X$.
3. $X + B^n = B^n$ for any $X \in 2B^n$.

Also, $\{X + Y\}$ can be interprated as union of “shifts” of the sets X onto points of the sets Y.

The family $(2^{B^n}, +)$, with an introduced Minkowski addition operation “+” forms a monoid with the neutral element $\{0^n\}$, which is one member set having the zero element of B^n.

The following inequality is valid:

$$\max\{|X|,|Y|\} \le |X+Y| \le |X|\cdot|Y|.$$

Both limits are achievable here. The following statements describe the sets in which these limits are achieved.

Definition [2]. The pair (*X*, *Y*) is called additive if for any $x_i, x_j \in X$ and $y_s, y_r \in Y$ the following is valid:

$$x_i + x_j \ne y_s + y_r.$$

Statement 1. The upper limit is achieved if *f* (*X*, *Y*) is an additive pair.

Corollary. If $|X+Y| = |X|\cdot|Y|$, then $|X_0+Y_0| = |X_0|\cdot|Y_0|$, for all $X_0 \subseteq X, Y_0 \subseteq Y$.

We consider an arbitrary subgroup $G \subseteq B^n$ and the action of this subgroup on the family $2B^n$:

$$gX = \{x+g, x \in X\},$$

where $g \in G$. Thus, G acts on $2B^n$ with shifts transferring the subset into its "shift".

Definition [3]. A stabilizer of the set X with respect to the group G is the union of "shifts" G_X from G, conserving X, i.e. $gX = X$ for all $g \in G_X$.

Statement 2. The lower limit is achieved if there exists $z \in B^n$, for which $X + z \subseteq G_y$ or $Y + z \subseteq G_x$.

Corollary. If $|X+Y| = |X|$, then $|X+Z| = X$ for all $Z \subseteq Y$.

Now let $X \subseteq B^n$ and $X = \{v_1, v_2, \cdots, v_m\}$.

Example.

1. If $G = \{v, 0\}$ is a group of shifts, then for $X = \{v_1, v_2, \cdots, v_m\}$ the following is valid:

$$g_1X = \{v_1+v, v_2+v, \cdots, v_m+v\}.$$

In this case X has a non-obvious stabilizer if all constituents of X can be partitioned into the pairs $(x, x+v)$, $x \in X$, i.e. $X = \{(v_i, v_i+v)\}$ with respect to $i = 1, 2, \cdots, m/2$. For $m = 4$ we get $X = \{(v_1, v_1+v), (v_2, v_2+v)\}$. It is clear that in this case $X + v = X$ and $v \in G_X$. Thus, all subsets of X, having a non-obvious stabilizers, are described above.

In the general form the stabilizer G_X for an arbitrary group G and an arbitrary set $X \subseteq B^n$ can be described in the following terms [3] .

Statement 3. The constituent $g \in G_X$ if the set X can be partitioned into the pairs (v_i, v_j) in such a way that $v_i + v_j = g$ for all pairs which are included in the partition.

This statement can be obtained by analogical consideration for $G = B^n$ as in [4].

From the above statement one can construct the following algorithm for building the stabilizer G_X of an arbitrary set X for the subgroup $G \subseteq B^n$, acting on $2B^n$. And at same time $|X| = 2m$.

1. First we build the multiset $C = X + X$.
2. Then we choose all the pairs in C having the multiplicity m.
3. Then we build all partitions in A out of these pairs.
4. If $\{P_X\}$ is the set of all partitions of X having the same weights in pairs $x \in C$, then

$$G_X = \{x\} \cup \{0\}.$$

Example.

1. Let $G = B^4$. $X = \{v_1 = (0\,0\,1\,1), v_2 = (1\,0\,1\,0), v_3 = (1\,1\,1\,0), v_4 = (0\,1\,1\,1)\}$.

Then:

$$v_1 + v_2 = (1\,0\,0\,1), v_1 + v_3 = (1\,1\,0\,1), v_1 + v_4 = (0\,1\,0\,0),$$
$$v_3 + v_4 = (1\,0\,0\,1), v_2 + v_4 = (1\,1\,0\,1), v_2 + v_3 = (0\,1\,0\,0).$$

This means that all pairs $(v_1, v_2), (v_3, v_4), (v_1, v_3), (v_2, v_4), (v_1, v_4), (v_2, v_3)$ have the multiplicity 2 in the sum $X + X$. Then we have:

$$X = \{v_1, v_2\} \cup \{v_3, v_4\} = \{v_1, v_3\} \cup \{v_2, v_4\} = \{v_1, v_4\} \cup \{v_2, v_3\}.$$

The sum of the pairs in each of the solutions is the same. Hence, the following set:

$$G_X = \{(v_1 + v_2), (v_1 + v_3), (v_1 + v_4), 0\} = \{(1\,0\,0\,1), (1\,1\,0\,1), (0\,1\,0\,0), (0\,0\,0\,0)\}$$

is a stabilizer for X.

Below we present the simple properties of the operation "+"—it is addition in the sense of Minkowski, as was mentioned above—which can be taken as properties of an algebraic system with basic set $2B^n$ and those for operations of addition, union of sets, set intersection etc.

1. Assosiativity:

$$X+(Y+Z)=(X+Y)+Z$$

2. Commutativity:

$$X+Y=Y+X$$

3. Distributivity with respect to union:

$$(X\cup Y)+Z=(X+Z)\cup(Y+Z)$$

4. $$\bigcup_{i=1}^{m}(X+Y_i)=X+\bigcup_{i=1}^{m}Y_i.$$

There are finitely many other relations connecting constituents of the algebraic system described above.

Sum of Spheres in B^n

Let $\rho(x,y)=\|x+y\|$ be the Hamming distance between the points *x*, *y* ∈ B^n and $S_t(v)$ be the set of the points of the sphere of the radius t with the centre at the point $v \in B^n$. In other words, $S_t(v)$ is the sphere of the radius t having the point v as its centre. And at that, $S_n(v)=B^n$ for all $a \in B^n$.

Statement 4 [1].

$$S_t(v)=S_t(0)+v. \qquad (2)$$

Statement 5.

$$\overline{S}_t(v)=S_{n-(t+1)}(\overline{v}) \text{ for } t\le n-1$$

Here $\overline{S}_t(v)$ is the set complement of the sphere $S_t(v)$ in B^n and $\overline{v}$ is the logic “negation” of the binary set v. We assume that $S_t(v)=\varnothing$, for $t < 0$.

Proof. Let us note that if $x\in\overline{S}_t(v)$, then $\rho(x,v)\ge t+1$. As:

$$\rho(x,v)\ge\rho(x,\overline{v})=n,$$

then for $x\in\overline{S}_t(v)$ we have:

$$\rho(x,\overline{v})=n-\rho(x,v)\le n-(t+1),$$

or:

$x \in S_{n-(t+1)}(\overline{v})$ for $t \le n-1$.

and if $t = n$, then $\overline{S}_t(a) = \varnothing$.

Example.

1. We consider the sphere $S_1(0)$. Then $\overline{S}_1(0) = S_{n-2}(11\cdots1)$.

Formula (2) in the preceding statement allows the following generalization connected with addition.

Let $M \in 2^{B^n}$ and $S_p(M)$ be the set of points belonging to the union of spheres of the radii p with the centres at the points M, that is:

$$S_p(M) = \bigcup_{x \in M} S_p(x)$$

$S_p(M)$ is the "generalized" sphere of the radius p having its centre at the point M.

Statement 6 [1] . The following presentation is valid:

$$S_p(M) = M + S_p(0).$$

Corollary. For $M_1, M_2 \subseteq B^n$ the following take place:

1. $S_p(M_1 + M_2) = S_p(M_1) + M_2 = S_p(M_2) + M_1 = S_p(0) + M_1 + M_2.$
2. $S_p(S_g(M_1)) = S_{p+g}(M_1).$

Statement7 [1] . The following relation is valid:

$$S_p(M_1) + S_g(M_2) = S_{p+g}(M_1 + M_2)$$

for $p + g \le n;$

and the next one is valid:

$$S_p(M_1) + S_g(M_2) = B^n$$

for $p + g \ge n$ . .

Corollary. For $M_1, M_2 \subseteq B^n$ the following is valid:

$$S_p(M_1) + S_g(M_2) = M_1 + M_2 + S_{p+g}(($$

The Sum of Facets in B^n

A facet, or sub-cube, or interval in B^n is the set of points satisfying the following condition [5] [6] :

$$J = \{u \le x \le v\} \ ,$$

where (≤) is a coordinate-wise partial order relation in B^n:

$x \le y \rightleftarrows x_i \le y_i$, $i = \overline{1,n}$, where $x = (x_1 x_2 \cdots x_n)$, $y = (y_1 y_2 \cdots y_n)$.

In other words, an interval can be given by a word of the length n in the alphabet $\{0,1,c\}$, the letters of which are ordered linearly: $0 < 1 < c$.

Indeed, if:

$$J = \{\alpha_1 \alpha_2 \cdots \alpha_n \le x \le \beta_1 \beta_2 \cdots \beta_n\},$$

then the code λ(J) of the interval J is built in the following way.

Let $\lambda(J) = (\lambda_1 \lambda_2 \cdots \lambda_n)$. Then:

$$\lambda_i = \begin{cases} \alpha_i = \beta_i, & \text{if } \alpha_i = \beta_i \\ c, & \text{if } \alpha_i < \beta_i \end{cases}$$

Examples.

1. If $J = \{0\,1\,0\,0 \le x \le 0\,1\,1\,1\}$, then $\lambda(J) = (0\,1\,c\,c)$.
2. If $J = B^n = \{0\,0 \cdots 0 \le x \le 1\,1 \cdots 1\}$, then $\lambda(B^n) = (c\,c \cdots c)$.

If $\lambda(J) = (\lambda_1 \lambda_2 \cdots \lambda_n)$ is the code of the interval J, then all points of the interval J are obtained from the code λ(J) by replacing the letters in an arbitrary way by zeros or units.

Let λ_1(J) and λ_2(J) be the numbers of letters 1 and c, respectively included in λ(J) which is the code of the interval J. it is clear that λ_2(J) is the dimension of J, i.e. $\lambda_2(\lambda) = \dim J$ and $|J| = 2^{\dim J}$.

If the operation "⋆" is introduced in the alphabet A by the following Caley table [7]:

$\star$	0	1	c
0	0	1	c
1	1	0	c
c	c	c	c

then the sum of the intervals J of the system defined above as a sum of subsets is the interval the code of which is calculated by the codes of items (addends) using the above Caley table.

Statement 7 [1] . The sum J_1 + J_2 is an interval with the code $\lambda(J_1+J_2)=\lambda(J_1)+\lambda(J_2)$ and dimension $\lambda_2(J_1+J_2)=\lambda_2(J_1)+\lambda_2(J_2)-\lambda_2(J_1\cap J_2)$

.

Examples.

1. If $J_1=\{(010),(011)\},J_2=(100)$, then $\lambda(J_1+J_2)=(11c)$.

On the other hand, we get by definition:

$J_1+J_2=\{(010)+(100),(011)+(100)\}=\{(110),(111)\}$,

i.e. $J_1+J_2=\{(11c),c\in\{0,1\}\}$.

2. If $J_1=B^n$, then $J_1+J_2=B^n$ for any interval J_2.

Statement 8 [1]. $\rho(J_1,J_2)=\lambda_1(J_1+J_2)$, where ρ is the Hausdorff distance between the sets [8] .

Thus, the distance between the intervals J_1 and J_2 is the number of occurrence of letters 1 in the code of their sum.

Sum of Layers in B^n

Let $B_p^n=\{x\in B^n,\|x\|=p\}$ be the p-th layer of an -dimensional cube or sphere of the radius p and the centre at zero [9] [10] .

By definition $B_p^n+B_q^n$ is the sum of layers in B^n, consisting of the union of sums of the points one of which has the weight and the second has the weight q. It is clear that the symmetrical group S_n operates on each layer in the following manner:

if $g\in S_n$, then $g(x_1x_2\cdots x_n)=x_{g(1)}x_{g(2)}\cdots x_{g(n)}$.

Hence, g permutes the coordinates of the point , leaving its Hamming weight unchanged.

At the same time the relation $g(x+y)=g(x)+g(y)$ is valid for $g\in S_n,x,y\in B^n$.

Thus, each layer B_p^n is a transitive set or an orbit of operation of the group S$_n$ on the cube B^n.

Let $|p-g|=a\,,\quad p+g=b$.

Statement 9 [1] . The following formula is valid:

$$B_p^n + B_q^n = \bigcup_{2r\le \min\{2n-b,b\}-a} B_{a+2r}^n \qquad (3)$$

For not large values of the layer the following table of addition is valid:

$+$	B_0^n	B_1^n	B_2^n
B_0^n	B_0^n	B_1^n	B_2^n
B_1^n	B_1^n	$B_0^n \cup B_2^n$	$B_1^n \cup B_3^n$
B_2^n	B_2^n	$B_1^n \cup B_3^n$	$B_0^n \cup B_2^n \cup B_4^n$

Note that Formula (3) can be rewritten for any number of terms, using the above-mentioned property of distributivity.

Indeed, using (3), we get:

$$B_p^n + B_q^n + B_s^n = \bigcup_{2r\le \min\{2n-b,b\}-a} B_{a+2r}^n + B_s^n = \bigcup_{2r\le \min\{2n-b,b\}-a} \left(B_{a+2r}^n + B_s^n\right),$$

which makes possible to use (3) again.

Example.

1. Let us find the sum $B_1^n + B_2^n + B_4^n$. We have:

$$B_1^n + B_2^n + B_4^n = \left(B_1^n \cup B_3^n\right) + B_4^n = \left(B_1^n + B_4^n\right) \cup \left(B_3^n + B_4^n\right)$$
$$= \left(B_3^n + B_5^n\right) \cup \left(B_1^n \cup B_3^n \cup B_5^n \cup B_7^n\right) = B_1^n \cup B_3^n \cup B_5^n \cup B_7^n.$$

NB. As each layer B_p^n is a sphere of the radius p and the centre at zero point, then all the preceding formulae are rules of ‘sphere’ addition.

Sum of Subsets in B^n

If we take subspaces in B^n as terms of the sum X + Y, we will get a well-known object. Indeed, if X, Y is a subspace in B^n, then (X + Y) is a subspace, too, and we have:

$$\dim\left(X+Y\right) = \dim X + \dim Y - \dim\left(X \cap Y\right),$$

in terms of cardinality:

$$|X+Y| = \frac{|X|\cdot|Y|}{|X\cap Y|}.$$

Thus, “theory of addition of subspaces” being a well-developed part of linear algebra, makes possible to answer many questions concerning the subject problem.

Sum of Spheres in B^n

The k-dimensional interval we denote by J^k.

According to statement 6, we have:

$$J^k + S_t^n(0) = \bigcup_{x\in J_k} S_t^n(x),$$

i.e. $J^k + S_t^n(0)$ is the union of all spheres of the radii t with centres at the points in the interval J^k, or:

$$J^k + S_t^n(0) = \bigcup_{x\in S_t^n(0)} \left(J^k + x\right).$$

Let $t_1 = \min(t, n-k)$.

Statement 10. $J^k + S_t^n(0) = S_{t_1}^{n-k}\left(J^k\right)$.

For the cardinality of the set $J^k + S_t^n(0)$ the following is true:

Corollary. $J^k + S_t^n(0) = 2^k S_{t_1}^{n-k}$, where $S_{t_1}^{n-k}$ is the cardinality of the sphere of the radius t$_1$ in B^{n-k}.

Proof. If $\lambda\left(J^k\right) = \{(cc\cdots c0\cdots 0)\}$, for any point the following is valid:

$$z = \left(\alpha_1\alpha_2\cdots\alpha_k, \beta_1\beta_2\cdots\beta_{n-k}\right) \in \bigcup_{x\in J^k} S_t^n(x),$$

if $\|\beta_1\cdots\beta_{n-k}\| \le t$. Indeed, in this case belongs to the sphere of the radius t with the centre at $(\alpha_1\alpha_2\cdots\alpha_k, 0\cdots 0)$. Inversely, if $z = \left(\alpha_1\alpha_2\cdots\alpha_k, \beta_1\cdots\beta_{n-k}\right)$ and $\|\beta_1\cdots\beta_{n-k}\| \ge t+1,$ then $\rho(z,y) \ge t+1$ for any point y in the interval J^k, that is:

$$z \notin \bigcup_{x\in J^k} S_t^n(x).$$

Therefore, $\left|S_t^n(y)+J^k\right| = 2^k\left(1+\binom{n-k}{1}+\cdots+\binom{n-k}{t_1}\right) = 2^k S_{t_1}^{n-k}.$

Sum of a Layer and an Interval in B^n

Analogous to the preceding statement and corollary we get the sum of the sets $B_t^n + J^k$.

Statement 11. The following relation is valid:

$$B_t^n + J^k = S_{t_1}^{n-k}\left(J^k\right)\setminus S_{p-k-1}^{n-k}\left(J^k\right).$$

Corollary. The cardinality of the set $B_t^n + J^k$ is calculated as follows:

$$\left|B_t^n + J^k\right| = 2^k\left(S_{t_1}^{n-k} - S_{p-k-1}^{n-k}\right).$$

Sum of a Sphere and a Layer in B^n

Statement 12. The following is valid:

$$B_p^n + S_q^n(M) = S_{l_1}^n(M)\setminus S_{l_2}^n(M),$$

where $l_1 = \min(p+q,n), l_2 = \max(0,p-q)-1.$

Proof. We have from statements 9 and 6:

$$B_p^n + S_q^n(M) = B_p^n + \bigcup_{i=0}^{q} B_i^n + M = \bigcup_{i=0}^{q}\left(B_p^n + B_i^n\right) + M$$

$$= \bigcup_{i=\max(0,p-q)}^{\min(p+q,n)} B_i^n + M = S_{l_1}^n(0)\setminus S_{l_2}^n(0) + M = S_{l_1}^n(M)\setminus S_{l_2}^n(M).$$

Q.E.D.

EQUATION IN SETS

Let $\left(2^{B^n}+\right)$ be the monoid of all subsets with operation of addition (1) in B^n as was defined above. This monoid is of certain interest both in classical discrete analysis [8] and for a number of problems connected with theory of information [4] .

The ‘simplest’ equation in sets is as follows:

$$X + Y = A \qquad (4)$$

where $X, Y, A \in 2^{B^n}$.

It is clear that Equation (4) always has the trivial solution $X = \{0\}, Y = A.$

Examples.

1. If $A = B^n$, then one can choose B^n for X, and any subset of B^n for Y.
2. If A is a subspace of B^n, then A + A = A and, therefore, Equation (4) has the solution X = Y = A.
3. $\{(11)\}+\{(01),(10)\}=\{(10),(01)\};\{(10)\}+\{(00),(01)\}=\{(10),(11)\}$.

Now, let:

$$\|X+Y\| = \min\left\{\|x\|, x \in X+Y\right\}.$$

Then $\rho(X,Y) = \|X+Y\|$; consequently, the Hausdorff distance between the sets X and Y:

$$\rho(X,Y) = \min_{\substack{x\in X\\ y\in Y}} \rho(x,y)$$

is expressed by the norm of the sum of these solutions.

On the other hand, if:

$$R(X,Y) = \left\{\rho(x,y); x \in X, y \in Y\right\},$$

then $R(X,Y)$ is the reciprocal spectrum of the distance between the points of the sets X and Y and:

$$R(X,X) = \left\{\|x+y\|, x \in X, y \in X\right\},$$

that is, $R(X,X) = R(X)$ is the spectrum of the distance between the points of the set X, or rather, the spectrum of X.

Thus, the set X + X describes, to a considerable extent, the set of distances between the points of X or the spectrum of X.

In an additive channel of communication [4] the class of equivalence has one to one presentation by transitive sets of certain ‘generating’ channels. The problem is to order these transitive sets through cardinalities of ‘generating’ channels. We need the following numerical parameters, which depend on solutions of Equation (4) and on the right hand side of A.

Let $N(A) = \{(X,Y), X+Y=A\}$.

We introduce the following parameters:

$$m(A) = \min_{(X,Y)\in N(A)} |X \cup Y|, \quad \bar{m}(A) = \begin{cases} |A \cup \{0\}|, \text{ if } N(A) = \varnothing \\ \min\limits_{(X,X)\in N(A)} |X| \end{cases}$$

$$M(A) = \max_{(X,Y)\in N(A)} |X \cup Y|, \quad \bar{M}(A) = \begin{cases} |A \cup \{0\}|, \text{ if } N(A) = \varnothing \\ \max\limits_{(X,X)\in N(A)} |X| \end{cases}$$

Introduction of such definitions as $\bar{m}(A)$ and $\bar{M}(A)$ is explained by the fact that the equation X + X = A can sometimes have no solution (for instance, for $|A| = 3, |A| = 5;$ or for $0 \notin A$), though the equation X + X = A always has a solution.

Then, for the minimal and maximal cardinality set $X \cup Y$, where $(X,Y) \in N(A)$, we get respective boundary values, which make possible to narrow the region $N(A)$, i.e. the region of the set of solutions of Equation (4) (we shall see this below).

It is not hard to prove that:

$$m(A) \le \bar{m}(A) \le \bar{M}(A) \le |A \cup \{0\}| \le M(A). \tag{5}$$

As every solution (X, X) of the equation X + X = A is a solution for (4), then we present the following useful statement which makes possible to obtain solutions of the equation X + X = Y from solutions of the Equation (4), under certain limitations.

Statement 13. If (X_0, Y_0) is a solution of the equation X + Y = A, then $(X_0 \cup Y_0)$ is a solution of the equation X + X = A, iff $(X_0 + X_0) \subseteq A$ and $(Y_0 + Y_0) \subseteq A$.

Statement 14. For the subspace $A \subseteq B^n$ the following is valid:

(a) $m(A) = \bar{m}(A);$

(b) $M(A) = \bar{M}(A) = 2^{\dim A}$.

Proof. It follows from (5) that it is sufficient to prove for (a) that:

$$m(A) \ge \bar{m}(A).$$

Let (X_0, Y_0) is a solution of the equation X + Y = A, for which:

$$m(A) = |X_0 \cup Y_0|. \qquad (6)$$

On the other hand, it follows from Statement 13 that $(X_0 \cup Y_0)$ is a solution of the equation X + X = A and, consequently, $|X_0 \cup Y_0| \ge \overline{m}(A)$. Taking into account this and (6), we get:

$$m(A) \ge \overline{m}(A).$$

The proof for the case (b) is analogical.

Statement 15. The following estimations are valid:

1. $m(A) \le 2^{\left[\frac{k}{2}\right]} + 2^{\left]\frac{k}{2}\right[} - 2,$ for the subspaces $A \subseteq B^n$ for $\dim\ A = k \ge 3;$
2. $m(A) \ge \left]\frac{1}{2}\left((8|A|-7)^{\frac{1}{2}}+1\right)\right[.$ for $A \subseteq B^n$.
3. If $A \subseteq B^n$ is a subspace, then equality in $m(A) \ge \left]\frac{1}{2}\left((8|A|-7)^{\frac{1}{2}}+1\right)\right[$ takes place if dim A = 1, 2 or 4.

Proof. Items 1 and 2 of this statement were proved in [1] , and we prove only item 3.

Necessity. We assume that:

$$m(A) = \left]\frac{1}{2}\left((8|A|-7)^{\frac{1}{2}}+1\right)\right[, \qquad (7)$$

and that the pair (X_0, Y_0) is a solution for the equation X + Y = A.

It follows from (7) that:

$$m(A)(m(A)-1) = 2(2^k-1), k = \dim A. \qquad (8)$$

According to the statement, we have that the set $(X_0 \cup Y_0)$ is a solution for the equation X + X = A, as well. We consider a Boolean matrix $n \times m(A),$ having points from $(X_0 \cup Y_0)$ in its rows. We denote by k_i the number of units in the i-thcolumn of this matrix. As A is a subspace, then the following equality is true: $k_i\left(m(A)-k_i\right)=2^{k-1}$, i.e. $k_i=2^{s_i}$ and $m(A)-k_i=2^{k-1-s_i}$. Consequently, $m(A)=2^{s_i}+2^{k-1-s_i}$. This and (8) give:

$$2^{2(k-1-s_i)}+2^{2s_i}=2^k+2^{k-1-s_i}+2^{s_i}-2. \qquad (9)$$

For $s_i\geq 2$ and $k-1-s_i\geq 2$ this equation has no solution for every k. Consequently, $s_i\leq 1$ or $k-1-s_i\leq 1$. Now it is easy to find the solution of Equation (9): $k=1,2$ or 4.

Sufficiency. Let $\{e_1,e_2,\cdots,e_k\}$ be the basis for the space A$_k$. We consider the following sets:

$X_1=\{0,e_1\}\subseteq A_1;$

$X_2=\{0,e_1,e_2\}\subseteq A_2;$

$X_4=\{0,e_1,e_2,e_3,e_4,e_1+e_2+e_3+e_4\}\subseteq A_4.$

As $|X_k|\geq m(A_k)\geq \left]\frac{1}{2}\left(\left(8|A_k|-7\right)^{\frac{1}{2}}+1\right)\right[$, then for $k\in\{1,2,4\}$ we have:

$$|X_k|=m(A_k)=\left]\frac{1}{2}\left(\left(8|A_k|-7\right)^{\frac{1}{2}}+1\right)\right[.$$

The statement is proved.

Examples.

1. The pair (X_0,Y_0), where $X_0=Y_0=S_t^n(0)$ is a solution of the equation: $X+Y=B^n\setminus\{1^n\}$, for $n=2t+1$.
2. The pair (X_1,Y_1), where $X_1=S_t^n(0),Y_1=S_t^n(0)\bigcup x,x\in B^n\setminus S_t^n(0)$ is a solution of the equation: $X+Y=B^n$, for $n=2t+1$.

If we keep to these examples, then we can assume that there exists some monotonous dependence of the function *m*(*A*) on the cardinality A. But one can manage to find the possible connection between the right hand side of Equation (4) and the function *m*(*A*) for the case if A is the halfspace.

Corollary. For the halfspace A$_1$, A$_2$ the inequality $\dim A_1\geq\dim A_2$ is valid if $m(A_1)\geq m(A_2)$.

The “seemingly obvious” hypothesis that the upper limit of *m*(*A*) is reached for all $k=\dim A$ is refuted by the following examples.

Examples.

1. Let $A = B^5$. In this case there is no solution of Equation (4), satisfying the condition: $|X| = 9$. Consequently, since for k = 5 the following is valid:

$$2^{\left[\frac{k}{2}\right]} + 2^{\left]\frac{k}{2}\right[} - 2 = 10 \geq m(A),$$

$m(A) = 10,$ and the upper limit is reached in this example.

2. Let $A = B^7 \cdot X$ = {(0 0 0 0 0 0 0), (0 0 0 0 1 0 0), (0 0 0 0 1 1 0), (0 0 0 1 1 0 1), (0 0 1 0 0 0 1), (0 0 1 0 1 1 1), (0 1 0 0 0 0 0), (0 1 0 0 1 0 0), (0 1 1 0 1 0 0), (0 1 1 0 1 0 1), (0 1 1 1 0 1 1), (0 1 1 1 1 0 0), (0 1 1 1 1 1 0), (1 0 0 0 0 1 0), (1 0 0 1 1 0 1), (1 0 0 1 1 1 0), (1 0 1 0 0 0 0), (1 0 1 1 0 1 1), (1 1 0 0 1 0 1),(1 1 0 1 1 0 0)}.

We have: $X + X = A, m(A) = 22 < 24 = 2^{\left[\frac{k}{2}\right]} + 2^{\left]\frac{k}{2}\right[} - 2.$ Consequently, the upper limit is not reached in this example.

Statement 16 [11] . If $A \setminus \{0\} \subseteq B_k^n$, then the solution of the equation X + X = A is an equidistant code with a distance between any two points equal k, and $\bar{M}(A) \leq n+1$. At the same time $\bar{M}(A) = n+1$ if there exists a Hadamard matrix of the order n + 1 [12] .

Consequently, the problem of constructing of an equidistant code with the distance k having the minimal cardinality can be formulated in terms of solvability of the equation X + X = A.

Definition. The set $A \in 2^{B^n}$ is called a quadrate if the following equation:

$$X + X = A \tag{10}$$

is solvable.

It is clear that a quadrate always contains the zero point.

Example.

1. If A is a halfspace in B^n, then, as it was mentioned above, A + A = A and, therefore, A is a quadrate. If $A \setminus \{0\} \subseteq B_{2r+1}^n,\ r \in \{0, 1, \cdots\}$, then A is a quadrate if $|A| = 2,\ i.e.\ A = \{0, x; x \in B_{2r+1}^n\}$.

The notion of 'quadrate' is connected with problems of equivalence of additive channels [4] where description of the class of equivalence is connected with finding of all solutions of the following equation:

$X + X = A$.

Let:

$$M(n,d) = \max_{N(A)\neq\varnothing}\left\{\bar{M}(A); A\setminus\{0\}\subseteq\bigcup_{i=d}^{n}B_i^n\right\}.$$

We denote by $A(n,d)$ the cardinality of the maximal code with the minimal distance d [6] .

Statement 17. $M(n,d) = A(n,d)$.

From this and taking into account the known estimations $A(n,d)$ (the upper limit; see [6]) we get:

Statement18.The following inequality is valid:

$$M(n,d)\leq\frac{2^n}{\sum_{i=0}^{t}\binom{n}{i}}, t = \left]\frac{d-1}{2}\right[.$$

At the same time equality takes place if there exists a perfect code in B^n with the minimal distance d.

Consequently, the problem of constructing the code of maximal cardinality ? in particular, a perfect code ? is reduced to finding the solution of maximal cardinality for Equation (10) among all quadrates of the union of layers $B_d^n\cup B_{d+1}^n\cup\cdots\cup B_n^n$.

Statement 19 [1] . If A, B is a quadrate, then (A + B) is a quadrate too.

Corollary. The preceding statement is valid for any number of summands. Now let $GL_2(n)$ be a group of invertible matrices having components in the field $F_2=\{0,1\}$.

Definition. The set of matrices $G_A\subseteq GL_2(n)$ is called stabilizer of the set $A\in 2^{B^n}$ if all matrices in G_Δ conserve A, i.e. $gA = A$, where $g\in G_A$.

At the same time, if $A=\{v_1, v_2,\cdots,v_m\}$, then $gA=\{gv_1, gv_2,\cdots,gv_m\}$.

Statement 20. Let G_A be a stabilizer of the set $A\in 2^{B^n}$ and $G_A=\{g\}$. Then the pair X, gX is the solution of the equation X + X = A, as well.

MULTISETS

The second definition of addition of sets from 2B^n is connected with multiplicity of containing each member into the sum A + B [4] .

Definition. A multisum of two sets $A,B\in 2^{B^n}$ is called multiset:

$$A+B=\{\alpha*(x+y),x\in A,y\in B\}, \tag{11}$$

in which each member $(x+y)$ is counted as many times as it comes in sum (11), and α is the multiplicity of the member $(x+y)$.

Examples.

1. If $A=\{(0\,1),(1\,0)\}\in 2^{B^2}$, then $A+A=\{2*(0\,0),2*(1\,1)\}$.
2. If $A_1=B_1^3=\{(0\,0\,1),(0\,1\,0),(1\,0\,0)\}$, $A_2=B_2^3=\{(1\,1\,0),(1\,0\,1),(0\,1\,1)\}$, then $A_1+A_2=\{3*(1\,1\,1),2*(1\,0\,0),2*(0\,1\,0),2*(0\,0\,1)\}$.

It is clear that by definition $|A_1+A_2|=|A_1|\cdot|A_2|$, in which the cardinality of the multiset is the sum of the multiplicities of its members.

In particular, the following expression is valid:

$$\bigcup_{x\in B^n}\{C+x\}=|C|*B^n,$$

where C is an arbitrary subset in B^n and $|C|$ is the multiplicity of the constituent $y\in B^n$.

It follows from this that:

$$\left|\bigcup_{x\in B^n}\{C+a\}\right|=|C|\cdot 2^n.$$

Let $a=|g-p|,b=g+p.$

Statement 21. For the multiset $B_g^n+B_p^n$ the following formula is valid:

$$B_g^n+B_p^n=\bigcup_{2r\le\min\{2n-b,b\}-a}\alpha(a+2r)*B_{a+2r}^n, \tag{12}$$

where $\alpha(a+2r)=\binom{a+2r}{r}\binom{n-a-2r}{\min(g,p)-r}$ is the multiplicity of the member of the multiset $B_p^n+B_g^n$ with the weight $(a+2r)$.

Proof. Let z be any member of the multiset $B_p^n + B_g^n$. Since: $|p-g| = a \le \|z\| \le \min(2n-b,b)$ and $\|z\|-a$ is an even number, then $\|z\|$ always is presentable in the form: $\|z\| = a+2r, r \ge 0$.

We assume (without violating generality) that: $x = (x_1x_2), y = (y_1y_2), z = (z_1z_2)$, where $x_1 \in B_{a+r}^{a+2r}, y_1 \in B_r^{a+2r}, z_1 \in B_{a+2r}^{a+2r}$, $x_2, y_2 \in B_{\min(p,g)-r}^{n-a-2r}, z_2 \in B_0^{n-a-2r}$ and $\rho(x_1,y_1) = a+2r$, $\rho(x_2,y_2) = 0$.

Hence, we have:

$$\left|\{(x_1,y_1); x_1+y_1=z_1\}\right| = \binom{a+2r}{r};$$

$$\left|\{(x_2,y_2); x_2=y_2\}\right| = \binom{n-a-2r}{\min(p,g)-r},$$

that is:

$$\left|\{(x,y); x+y=z\}\right| = \binom{a+2r}{r}\binom{n-a-2r}{\min(p,g)-r}.$$

From this, taking into account Statement 9, we get Formula (12).

Corollary. For $g \le p \le n$ we have:

a) $$\binom{n}{g}\binom{n}{p} = \sum_{2r=0}^{\min\{2n-b,b\}-a}\binom{n}{a+2r}\binom{a+2r}{r}\binom{n-a-2r}{g-r};$$

b) $$\binom{n}{p}\binom{n}{g} = \sum_{i=1}^{n-p}\binom{n}{p-g+2i}\binom{p-g+2i}{i}\binom{n-p+g-2i}{g-i};$$

if $p+g \ge n$, and:

c) $$\binom{n}{p}\binom{n}{g} = \sum_{i=0}^{g}\binom{n}{p-g+2i}\binom{p-g+2i}{i}\binom{n-p+g-2i}{g-i};$$

if $p+g \le n$.

Statement 22. For the multiset $S_t^n(v)+B_p^n$ the following is valid:

$$S_t^n(v)+B_p^n = \bigcup_{2r=\max x(0,p-t)-l}^{\min(p+t,n)} \alpha(l+2r)*\left(B_{l+2r}^n+v\right),$$

where $v \in B^n$; and at the same time:

$$\alpha(l+2r) = \sum_{i=0}^{t}\sum_{\substack{r\\ l=|i-p|}}\binom{l+2r}{r}\binom{n-l-2r}{\min(i,p)-r}$$

is the multiplicity of the members of $\left(B_{l+2r}^{n}+v\right)$.

Corollary. For $n \geq p, t \geq 0$ the following is valid:

a) $\binom{n}{p}\sum_{i=0}^{t}\binom{n}{i}=\sum_{i=0}^{\min(p,t)}\sum_{r=0}^{i}\binom{n}{p+r}\binom{p+r}{i}\binom{i}{r}+\sum_{i=p+1}^{t}\sum_{r=0}^{p}\binom{n}{i+r}\binom{i+r}{p}\binom{p}{r};$

b) $\binom{n}{p}\binom{n}{g}=\sum_{i=0}^{g}\binom{n}{p+i}\binom{p+i}{g}\binom{g}{i}.$

Statement 23. For the multiset $S_{t_1}^{n}(v_1)+S_{t_2}^{n}(v_2)$ the following equality is valid:

$$S_{t_1}^{n}(v_1)+S_{t_2}^{n}(v_2)=\bigcup_{m=0}^{t_1+t_2}\alpha_m*\left(B_m^n+(v_1+v_2)\right),$$

where $v_1, v_2 \in B^n$ and at the same time:

$$\alpha_m=\sum_{i=0}^{t_1}\binom{m}{i}\sum_{j=0}^{\min(t_1-i,t_2-m+i)}\binom{n-m}{j}$$

is the multiplicity of the members $x \in \left(B_m^n+(v_1+v_2)\right)$.

Corollary. For $n \geq t_1, t_2 \geq 0$ the following equality is valid:

$$\sum_{i=0}^{t_1}\binom{n}{i}\sum_{j=0}^{t_2}\binom{n}{j}=\sum_{m=0}^{t_1+t_2}\binom{n}{m}\sum_{i=0}^{t_1}\binom{m}{i}\sum_{j=0}^{\min(t_1-i,t_2-m+i)}\binom{n-m}{j}$$

Statement 24. For the multiset $B_p^n+J^k$ the following formula is valid:

$$B_p^n+J^k=\bigcup_{x\in B^n}\alpha(x)*x,$$

where $\alpha(x)=\binom{k}{p-\lambda_1\left(J^k+x\right)}$ is the multiplicity of x.

Corollary. For $n \geq p \geq 0$ the following is valid:

$$\binom{n}{p}=\sum_{i=0}^{p}\binom{n-k}{i}\binom{k}{p-i}$$

Statement 25. For the multiset $S_t^n(v)+J^k$ the following formula is valid:

$$S_t^n(v)+J^k=\bigcup_{x\in B^n}\alpha(x)*(x+v),$$

where $v \in B^n$, and at the same time:

$$\alpha(x) = \sum_{i=0}^{t-\lambda_1\left(J^k+x\right)} \binom{k}{i}$$

is the multiplicity of $x+v$.

Corollary. For $n \geq k \geq 0, n \geq t \geq 0$ the following is valid:

$$\sum_{i=0}^{t}\binom{n}{i} = \sum_{i=0}^{t}\binom{n-k}{i}\sum_{j=0}^{t-i}\binom{k}{j}.$$

Statement 26. For the multiset $J^{k_1} + J^{k_2}$ the following is valid:

$$J^{k_1} + J^{k_2} = \alpha\left(J^{k_1}, J^{k_2}\right) * J^{k_3},$$

where $k_3 = k_1 + k_2 - \lambda_2\left(J^{k_1} \cap J^{k_2}\right)$, J^{k_3} is the interval with the code: $\lambda\left(J^{k_1} + J^{k_2}\right)$, and $\alpha\left(J^{k_1}, J^{k_2}\right) = \lambda_2\left(J^{k_1} \cap J^{k_2}\right)$ is the multiplicity of the members of J^{k_3}.

Finally, we define the operation “/”, that is, subtraction for multisets.

Let $X = \{\alpha(x) * x; x \in B^n\}, Y = \{\alpha(y) * y; y \in B^n\}$.

Definition. $X \setminus Y = \{\alpha(z) * z;$ where $\alpha(z) = \max\{\alpha(x) - \alpha(y), 0\}, z = x = y \in B^n\}$.

Example. We consider the multisets:

$$X = B_p^n + J^k = \left\{\binom{k}{p-\lambda_1\left(J^k+x\right)} * x, x \in B^n\right\},$$

$$Y = B_p^n + J^k = \left\{\alpha(y) * y, y \in B^n, \alpha(y) = \begin{cases} 1; & \text{if } \binom{k}{p-\lambda_1(J^k+y)} \geq 1 \\ 0; & \text{otherwise} \end{cases}\right\}.$$

From Statements 22 and 12 we get:

$$X \setminus Y = \left\{\left(\binom{k}{p-\lambda_1\left(J^k+z\right)} - 1\right) * z, z \in B^n\right\}.$$

REFERENCES

1. Leontiev, V.K., Movsisyan, G.L. and Margaryan, Zh.G. (2016) Algebra and Geometry of Sets in Boolean Space. Open Journal of Discrete Mathematics (OJDM), 6, 25-40.
2. Movsisyan, G.L. (2013) Dirichlet Regions and Perfect Codes in Additive Channel. Open Journal of Discrete Mathematics (OJDM), 3, 137-142.
3. Sachkow, W.N. (1977) Combinatory Methods of Discrete Mathematics. Nauka, Moscow. (In Russian)
4. Leontiev, V.K., Movsisyan, G.L. and Osipyan, A. (2014) Classification of the Subsets and the Additive Channels. Open Journal of Discrete Mathematics (OJDM), 4, 67-76.
5. Leontiev, V.K. (2001) Selected Problems of Combinatorial Analysis. Bauman Moscow State Technical University, Moscow. (In Russian)
6. Leontiev, V.K. (2015) Combinatorics and Information. Moscow Institute of Physics and Technology (MIPT), Moscow. (In Russian)
7. Lang, S. (1968) Algebra. Moscow, Mir. (In Russian)
8. Nigmatulin, R.G. (1991) Complexity of Boolean Functions. Nauka, Moscow, p. 240. (In Russian)
9. Movsisyan, G.L. (1982) Perfect Codes in the Schemes Johnson. Bulletin of MSY, Computing Mathematics and Cybernetics, 1, 64-69. (In Russian)
10. Leontiev, V.K., Movsisyan, G.L. and Margaryan, Zh.G. (2012) Constant Weight of Perfect and D-Representable Codes. Proceedings of the Yerevan State University, Physical and Mathematical Sciences, 16-19.
11. McWilliams, F.J. and Sloane, N.J.A. (1977) The Theory of Error-Correcting Codes. Parts I and II, North-Holland Publishing Company.
12. Delsarte, P. (1973) Four Fundamental Parameters of a Code and Their Combinatorial Significance. Information and Control, 23, 407-438.

Chapter

ALGEBRA AND GEOMETRY OF SETS IN BOOLEAN SPACE

16

Vladimir Leontiev[1], Garib Movsisyan[2], and Zhirayr Margaryan[3]

[1]Moscow State University, Moscow, Russia.
[2]BIT Group, Moscow, Russia.
[3]Yerevan State University, Yerevan, Armenia.

ABSTRACT

In the present paper, geometry of the Boolean space B^n in terms of Hausdorff distances between subsets and subset sums is investigated. The main results are the algebraic and analytical expressions for representing of classical figures in B^n and the functions of distances between them. In particular, equations in sets are considered and their interpretations in combinatory terms are given.

Citation: Leontiev, V. , Movsisyan, G. and Margaryan, Z. (2016), “Algebra and Geometry of Sets in Boolean Space”. Open Journal of Discrete Mathematics, 6, 25-40. doi: 10.4236/ojdm.2016.62004.

Keywords: Equations on Sets, Hausdorff Distance, Hamming Distance, Generating Function, Minkowski Sum, Sum of Sets

DISTANCE BETWEEN SUBSETS B^N

Let $B=\{0,1\}$, $B^n=\{0,1\}^n$ and *B*^{n} be the set of all words of finite length in the alphabet B. For $X,Y\in 2^{B^n}$ we take:

$$\rho(X,Y)=\min_{\substack{u\in X\\ v\in Y}}\rho(u,v).$$

It is clear that $\rho(X,Y)$ is the Hausdorff distance between the subsets X Y, and $0\le\rho(u,v)\le n$, and $\rho(u,v)=\|u+v\|$ is the Hamming distance between the points: $u=(u_1u_2\cdots u_n),v=(v_1v_2\cdots v_n)\in B^n$, where $u+v=((u_1\oplus v_1)(u_2\oplus v_2)\cdots(u_n\oplus v_n))$ and $\oplus$ is the addition operation with respect to mod 2.

The Hausdorff distance has essential role in many problems of discrete analysis [1] and thus has certain interest. On the other hand, there only are a few essential results concerning distances between the subsets *B*^{n}, and their investigation offers significant difficulties.

First, we present the following simple properties of the Hausdorff distance:

1) $\rho(X,Y)=\rho(Y,X)$;

2) $\rho(X,Y)=0\rightleftarrows X\cap Y\neq\varnothing$;

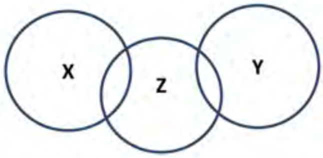
3) Если $X\subseteq X',Y\subseteq Y'$, то $\rho(X,Y)=\rho(X',Y')$;

4) $\rho(X,Y)=\rho(X+a,Y+a)$, for $a\in B^n$.

Let us note that, generally speaking, the Hausdorff distance does not satisfy the triangle inequality:

$$\rho(X,Y)\le\rho(X,Z)+\rho(Z,Y), \qquad (1)$$

which is demonstrated in the following picture:



But inequality (1) holds true if $|Z|=1$.

Distance between Spheres in B^n

Let $S_p^n(x)$ be a sphere of radius p with the center at $x \in B^n$. We take, for an arbitrary subset, $M \subseteq B^n$:

$$S_p^n(M) = \bigcup_{x \in M} S_p^n(x).$$

Thus, we have the following two equvalent interpretations for $S_p^n(M)$:

1) $S_p^n(M)$ is the set of all points in B^n which are at the distance $\le p$ from the set M;
2) $S_p^n(M)$ is the set of all points in B^n, covered by the spheres of the radius p with the centers at points of the set M.

Examples.

1) $S_1^n(S_1^n(a)) = S_2^n(a)$, for an arbitrary point: $a \in B^n$;
2) $S_p^n(B^n) = B^n$, for an arbitrary $p \ge O$;
3) If $S_p^n(M)$ = B^n and $S_{p-1}^n(M) \neq B^n$, then p is the radius of the covering of the set B^n [2] .

Theorem 1. $\rho(S_p^n(M_1), S_q^n(M_2)) = \max\{0, \rho(M_1, M_2) - p - q\}$.

Proof. We consider two cases.

a) $p + q > n.$ Then,

$$S_p^n(M_1) \cap S_q^n(M_2) \neq \varnothing$$

and, consequently,

$$\rho(S_p^n(M_1), S_q^n(M_2)) = 0.$$

b) $p + q \le n.$ Let $x \in S_p^n(M_1), y \in S_q^n(M_2)$. We present them in the form:

$$x = x_1 + x_2, \text{ where } x_1 \in M_1, \|x_2\| \le p;$$

$$y = y_1 + y_2, \text{ where } y_1 \in M_2, \|y_2\| \le q.$$

From here we have:

$$\rho(x, y) = \rho(x_1 + x_2, y_1 + y_2) = \|x_1 + x_2 + y_1 + y_2\|.$$

As $\|x_1 + x_2 + y_1 + y_2\| \ge \|x_1 + y_1\| - \|x_2 + y_2\|,$ consequently we have:

$$\min_{x_1\in M_1, y_1\in M_2}\left\|x_1+x_2+y_1+y_2\right\| \geq \min_{x_1\in M_1, y_1\in M_2}\left(\left\|x_1+y_1\right\|-\left\|x_2+y_2\right\|\right)$$

$$=\min_{x_1\in M_1, y_1\in M_2}\left\|x_1+y_1\right\|-\max_{x_2\leq p, y_2\leq q}\left\|x_2+y_2\right\|.$$

Then, taking into account that:

$$\max_{\|x_2\|\leq p, \|y_2\|\leq q}\left\|x_2+y_2\right\|=p+q, p+q\leq n,$$

we have:

$$\rho\left(S_p^n\left(M_1\right), S_q^n\left(M_2\right)\right)=\rho\left(M_1, M_2\right)-p-q.$$

The theorem is proved.

Let:

$$R\left(r_1, r_2\right)=\max_{\substack{|X|=r_1\\ |Y|=r_2}}\rho\left(X, Y\right).$$

Taking into account that the sphere of the radius p with the center at $(00\cdots0)$ and the sphere of the radius q with the center at $(11\cdots1)$ contain, respectively, as many points as:

$$\sum_{t=0}^{p}\binom{n}{t} \text{ and } \sum_{t=q}^{n}\binom{n}{t},$$

we get the following corollary.

Corrollary. If $q>p$, then:

$$R\left(\sum_{t=0}^{p}\binom{n}{t}, \sum_{t=q}^{n}\binom{n}{t}\right)\geq q-p.$$

The value of the function $R\left(r_1, r_2\right)$ for definite values of r_1, r_2 was calculated in [1] .

Theorem 2. If $q>p$, then:

$$R\left(\sum_{t=0}^{p}\binom{n}{t}, \sum_{t=q}^{n}\binom{n}{t}\right)=q-p.$$

The general form of the standard generating function for the distance between the subsets $X, Y \subseteq B^n$ has the following form:

$$F_{p,q}(z) = \sum_{\substack{|X|=p \\ |Y|=q}} z^{\rho(X,Y)}. \tag{2}$$

The summation in (2) is over all pairs of the subsets (X, Y) with $|X| = p, |Y| = q$.

Let us consider a few examples.

1) $p = q = 1$.

In this case, we have:

$$F_{1,1}(z) = \sum_{x,y \in B^n} z^{\rho(x,y)} = \sum_x \sum_y z^{\rho(x,y)} = \sum_x \sum_{k=0}^{n} \binom{n}{k} z^k = 2^n (1+z)^n.$$

Thus, $F_{1,1}(z) = 2^n (1+z)^n$, which is the well-known function of distribution of distances between the points in the space B^n with the metrics of Hamming.

2) p = 1, and q is an arbitrary positive integer which does not exeed 2^n.

In this case:

$$F_{1,q}(z) = \sum_{\substack{x \in B^n \\ |Y|=q}} z^{\rho(x,Y)} = \sum_x \sum_{|Y|=q} z^{\rho(x,Y)}. \tag{3}$$

As:

$$\rho(x,Y) = \rho(x+a, Y+a),$$

for arbitrary points $a, x \in B^n$ and any subset $Y \subseteq B^n$, then we get from (3):

$$F_{1,q}(z) = 2^n \sum_{|Y|=q} z^{\rho(0,Y)}.$$

Then:

$$\rho(0,Y) = \min_{x}\left\{\rho(0,x) = \|x\|, x \in B^n\right\}. \qquad (4)$$

Consequently, the distance between the zero point and an arbitrary subset Y equals the minimal weight of the points which are in Y.

Hence, $\rho(0,Y) \geq k$, if there are not points with $\|x\| \leq k-1$ in Y. The numbers of subsets Y with $|Y| = q$ and the condition $\rho(0,Y) \geq k$ are found by the following formula:

$$\lambda_q(k) = \binom{2^n - S_{k-1}^n}{q}, \qquad , (5)$$

where $S_r^n = \sum_{i=0}^{r}\binom{n}{i}$

where $S_r^n = \sum_{i=0}^{r}\binom{n}{i}$ is the cardinality of the sphere with the radius r in B^n

From (5), we get the following statement:

Lemma1. If $\lambda^o(q,k)$ is the number of the subsets of cardinality q (in B^n) having the distance to the zero point, then:

$$\lambda^o(q,k) = \lambda_q(k) - \lambda_q(k+1).$$

For $k = 0 : \lambda^o(q,0) = \binom{2^{n-1}}{q-1}.$

Theorem 3. The following formula holds true:

$$F_{1,q}(z) = 2^n\binom{2^{n-1}}{q-1} + 2^n\sum_{k=1}^{n} z^k\left[\binom{2^n - S_{k-1}^n}{q} - \binom{2^n - S_k^n}{q}\right]. \qquad (6)$$

Proof. By definition:

$$F_{1,q}(z) = 2^n \sum_{|Y|=q} z^{\rho(0,Y)}.$$

From this and Lemma 1, taking into account (3) and (4), we get:

$$F_{1,q}(z) = 2^n \left[\binom{2^{n-1}}{q-1} + \sum_{Y} z^{\rho(0,Y)} \right] = 2^n \binom{2^{n-1}}{q-1} + 2^n \sum_{k=1}^{n} \lambda^o (q,k) z^k$$
$$= 2^n \binom{2^{n-1}}{q-1} + 2^n \sum_{k=1}^{n} z^k \left[\binom{2^n - S_{k-1}^n}{q} - \binom{2^n - S_k^n}{q} \right].$$

The theorem is proved.

If $\Phi_{1,p}(z)$ is the generating function of the random value $\xi = \rho(a,Y)$ uniformly distributed on the pairs (a,Y), where $|Y| = q$, then the following holds true.

Corollary 1. The following formula holds true:

$$\Phi_{1,p}(z) = q + \frac{1}{\binom{2^n}{q}} \sum_{k=1}^{n} z^k \left[\binom{2^n - S_{k-1}^n}{q} - \binom{2^n - S_k^n}{q} \right].$$

Corollary 2. The following holds true:

$$M\xi = \frac{1}{\binom{2^n}{q}} \sum_{k=1}^{q} k \left[\binom{2^n - S_{k-1}^n}{q} - \binom{2^n - S_k^n}{q} \right].$$

Corollary 3. The formula for $F_{1,1}(z)$ follows from (6).

Proof. From (6) we get for p = 1:

$$F_{1,1}(z) = 2^n + 2^n \sum_{k=1}^{n} z^k \left[\left(2^n - S_{k-1}^n\right) - \left(2^n - S_k^n\right) \right] = 2^n + 2^n \sum_{k=1}^{n} z^k \binom{n}{k} = 2^n (1+z)^n.$$

Corollary 4. For q = 2, the following formula holds true:

$$M\xi = \frac{n}{2} - \frac{\sqrt{n}}{\sqrt{\pi}} + 0(1).$$

Proof. By definition and from Corollary 2, we derive:

$$\phi_{2,1}^{1}(1) = M\xi = \frac{1}{\binom{2^n}{2}} \sum_{k=1}^{n} k \left[\binom{2^n - S_{k-1}^n}{2} - \binom{2^n - S_k^n}{2} \right]. \tag{7}$$

Transforming the terms in (7), we get:

$$M\xi = \frac{1}{2\binom{2^n}{2}} \sum_{k=1}^{n} k\binom{n}{k}\left[2^{n+1} - 2S_{k+1}^{n} - \binom{n}{k} - 1\right]. \qquad (8)$$

Then, using the following formulas:

$$\sum_{k=1}^{n} k\binom{n}{k} = n2^{n-1}, \sum_{k=1}^{n} k\binom{n}{k} S_{k-1}^{n} = n\sum_{k=1}^{n}\binom{n-1}{k-1} S_{k-1}^{n},$$

$$\sum_{k=1}^{n} k\binom{n}{k}^{2} = \frac{n}{2}\binom{2^n}{n}.$$

Let us "compress" the sum:

$$S = \sum_{k=1}^{n}\binom{n-1}{k-1} S_{k-1}^{n}.$$

By definition, we have:

$$\begin{aligned} S &= \sum_{k=1}^{n}\binom{n-1}{k-1}\sum_{r=0}^{k-1}\binom{n}{r} = \sum_{k=1}^{n}\binom{n-1}{k-1}\left(2^n - \sum_{r=k}^{n}\binom{n}{r}\right) = 2^n\sum_{k=1}^{n}\binom{n-1}{k-1}\sum_{k=1}^{n}\binom{n-1}{k-1}\sum_{r=k}^{n}\binom{n}{r} \\ &= 2^{n-1}\left(2^n - 1\right) - \sum_{k=1}^{n}\binom{n-1}{k-1}\sum_{r=k}^{n}\binom{n}{r} = 2^{n-1}2^n - \sum_{k=1}^{n}\binom{n-1}{k-1}\sum_{r=k}^{n}\binom{n}{r} \\ &= 2^{2n-1}\sum_{k=1}^{n}\binom{n-1}{k-1}\sum_{r=k}^{n}\binom{n}{r}. \end{aligned}$$

Furthermore, if:

$$R = \sum_{k=1}^{n}\binom{n-1}{k-1}\sum_{r=k}^{n}\binom{n}{r},$$

then:

$$\begin{aligned} R &= \sum_{k=1}^{n}\binom{n-1}{k-1}\sum_{r=k}^{n}\underset{u}{coef}\left\{(1+u)^n u^{-n+r-1}\right\} = \underset{u}{coef}\left\{\frac{(1+u)^n}{u^{n+1}}\sum_{k=1}^{n}\binom{n-1}{k-1}\sum_{r=k}^{n}u^r\right\} \\ &= \underset{u}{coef}\left\{\frac{(1+u)^n}{u^{n+1}}\sum_{k=1}^{n}\binom{n-1}{k-1}\frac{u^k - u^{n+1}}{1-u}\right\} \\ &= \underset{u}{coef}\left\{\frac{(1+u)^n}{u^{n+1}(1-u)}\sum_{k=1}^{n}\binom{n-1}{k-1}u^k\right\} - \underset{u}{coef}\left\{\frac{(1+u)^n}{(1-u)}\sum_{k=1}^{n}\binom{n-1}{k-1}\right\}. \end{aligned}$$

Further:

$$\operatorname{coef}\left\{\frac{(1+u)^n u}{u^{n+1}(1-u)}\sum_{k=1}^{n}\binom{n-1}{k-1}u^{k-1}\right\}=\operatorname*{coef}_{u}\left\{\frac{(1+u)^n}{u^n(1-u)}(1+u)^{n-1}\right\}$$
$$=\operatorname*{coef}_{u}\left\{\frac{(1+u)^{2n-1}}{u^{n+1}(1-u)}\right\}=\sum_{r=0}^{n}\binom{2n-1}{n-r}=\sum_{r=0}^{n}\binom{2n-1}{r}=2^{2n-2}$$

And:

$$\operatorname*{coef}_{u}\left\{\frac{(1+u)^n}{1-u}\sum_{k=1}^{n}\binom{n-1}{k-1}\right\}=0.$$

From here it follows that $R=2^{2n-2}$.

Then:

$$\sum_{k=1}^{n}k\binom{n}{k}\left[2^{n+1}-2S_{k-1}^{n}-\binom{n}{k}-1\right]$$
$$=2^{n+1}2^{n-1}n-2\sum_{k=1}^{n}k\binom{n}{k}S_{k-1}^{n}-\sum_{k=1}^{n}k\binom{n}{k}^2-\sum_{k=1}^{n}k\binom{n}{k}=n2^{2n}-2nS-\frac{n}{2}\binom{2n}{n}-n2^{n-1}$$
$$=n2^{2n}-2n\left\{\left(2^{2n-1}-2^n\right)-R\right\}-\frac{n}{2}\binom{2n}{n}-n2^{n-1}=n2^{n+1}-2nR-\frac{n}{2}\binom{2n}{n}-n2^{n-1}$$
$$=\frac{n}{2}2^{2n}-\frac{n}{2}\binom{2n}{n}+3n2^{n-1}.$$

Taking this and (8) into account, we get:

$$M\xi=\frac{n}{2}-\frac{\sqrt{n}}{\sqrt{\pi}}+0(1).$$

And the generating function:

$$F_{p,q}(z)=\sum_{\substack{|X|=p\\|Y|=q}}z^{\rho(X,Y)}$$

can be expressed by the following parameters:

2) if $M\subseteq B^n$, then the family of all subsets of cardinality p, having distances $\geq r$ from M is expressed as $\binom{B^n\setminus S_{r-1}^{n}(M)}{p}$. Indeed,

$S^n_{r-1}(M)$ contains all the points of B^n, having distances $\le r-1$ from the set M.

Hence, the set $B^n \setminus S^n_{r-1}(M)$ does not contain such points; consequently, for an arbitrary subset $X \in \binom{B^n \setminus S^n_{r-1}(M)}{p}$, the expression $\rho(X,M) \ge r$ holds true.

The cardinality of this family is:

$$\binom{\left|B^n \setminus S^n_{r-1}(M)\right|}{p} = \binom{2^n - \left|S^n_{r-1}(M)\right|}{p}.$$

2) The number of all m-element subsets having the distance r from M is:

$$\binom{2^n - \left|S^n_{r-1}(M)\right|}{p} - \binom{2^n - \left|S^n_{r}(M)\right|}{p}.$$

Summarizing all the previous, we get the following statement.

Theorem 4. The following expression is true:

$$F_{p,q}(z) = \sum_{|Y|=q} \sum_{r=1} \left\{ \binom{2^n - \left|S^n_{r-1}(Y)\right|}{p} - \binom{2^n - \left|S^n_{r}(Y)\right|}{p} \right\} z^r.$$

SUM OF SETS IN *B^N*

Let $X, Y \in B^n$; we take:

$$X + Y = \{u + v, u \in X, v \in Y\}$$

The operation "+" is defined in the family 2*B^n* of all subsets of *B^n*, and (2*B^n*, "+") is a monoid with the neutral element $0 = 0^n$ [3] [4] .

Besides, the following inequality holds true:

$$\max\{|X|,|Y|\} \le |X+Y| \le |X||Y|.$$

Here both limits are reachable.

The properties of “+” are as follows:

1) $X+0=X$;
2) $X=Y\rightarrow X+u=Y+u$ for all $u\in B^n$;
3) $(X+Y)+Z=X+(Y+Z)$-associativity;
4) $X+Y=Y+X$ -communacativity;
5) $(X\bigcup Y)+Z=(X+Z)\bigcup(Y+Z)$-distributivity;
6) $(X\bigcap Y)+Z=(X+Z)\bigcap(Y+Z)$;
7) $(X\setminus Y)+Z=(X+Z)\setminus(Y+Z)$.

Examples.

If X is a subspace in B^n, then:

1) $X+X=X$;
2) $X+Y=X$, if $Y\subseteq X$.

Let the following holds true:

$$\|X+Y\|=\min_{\substack{x\in X\\ y\in Y}}\|x+y\|.$$

Then $\rho(X,Y)=\|X+Y\|$.

Thus, there is certain analogy between the norm of the sum of points and the distance between those points, as well as between the norm of the sum of the sets and the distance between those sets.

In the general form, the following statement connecting the operations “∪” и+, is true:

$$(X\bigcup Y)+(U\bigcup V)=(X+U)\bigcup(X+V)\bigcup(Y+U)\bigcup(Y+V).$$

Sum of Facets in B^n and the Distance between Them

A facet or interval in B^n is the set of points $J=\{a\le x\le b\}$, where the partial order $x\le y$ is defined in the classic way [5] [6] :

$$x\le y\rightleftarrows x_i\le y_i, i=\overline{1,n}, \text{ for } x=(x_1\ x_2\cdots x_n), y=(y_1\ y_2\ldots y_n).$$

Every interval J can be written in the form of a word of the length n, in the alphabet $A=\{0,1,c\}$, the letters of which are ordered linearly: $0<1<c$. Examples.

If $n = 4$ and $J = \{(0\,1\,0\,0) \le x \le (0\,1\,1\,1)\}$, then every point of J can be presented by the word $(0\,1\,c\,c)$, which means the following: all the points which are obtained from the word $(0\,1\,c\,c)$ by the substitution either 0 or 1 for a letter of the given word, are contained in the interval J. Consequently, the cardinality of the interval J is $2^2 = 4$, for the given case, i.e. $|J| = 4$. Hence, each interval J has its corresponding code word $\lambda(J)$ in the alphabet A. The number of letters c in the code $\lambda(J)$ is the dimension of the interval J, i.e. is $\dim J$. And the following formula is obvious:

$|J| = 2^{\dim J}$.

If the operation "*" is introduced on the alphabet A:

*	0	1	c
0	0	1	c
1	1	0	c
c	c	c	c

then the sum $J_1 + J_2$ of the intervals J_1 and J_2 is the Minkowski sum:

$J_1 + J_2 = \{u + v, u \in J_1, v \in J_2\}$.

Examples.

1) If $J_1 = \lambda(0\,1\,c), J_2 = \lambda(1\,0\,0)$, then $\lambda(J_1 + J_2) = (1\,1\,c)$, i.e. $J_1 + J_2 = \{(1\,1\,0), (1\,1\,1)\}$, which corresponds to the definition of the sum $J_1 + J_2$.

2) If $J_1 = \lambda(c\,c \cdots c)$, then $J_1 + J_2 = J_1 = B^n$ for every interval J_2 .

The distance between the intervals J_1 and J_2 , having the codes $\lambda(J_1)$ and $\lambda(J_2)$-taking into account the introduced definitions-are calculated in the following way.

Let $\lambda_1(J)$ be the number of occurrences of letter 1 in the code of the interval J.

Statement 1. $\rho(J_1, J_2) = \lambda_1(J_1 + J_2)$.

Thus, the distance between the intervals J_1 and J_2 is the number of occurrences of letter 1 in the code of their sum.

Examples.

1) Let $\lambda(J_1)=(0\,1\,1\,c\,1\,c),\lambda(J_2)=(1\,1\,c\,0\,c\,0)$.

Then $\lambda(J_1+J_2)=(1\,0\,c\,c\,c\,c)$ and $\rho(J_1,J_2)=1.$

Let $J_n^p=\{J_p\}$ be the family of all p-dimensional intervals of B^n.Then $\left|J_n^p\right|=\binom{n}{p}2^{n-p}.$

Let us consider the direct product $J_n^p\times J_n^q$ and introduce uniform distribution on it with the generating function:

$$F_{p,q}(z)=\frac{1}{\left|J_n^p\right|\left|J_n^q\right|}\sum_{J_n^p,J_n^q}z^{\rho\left(J_n^p,J_n^q\right)}.$$

Theorem 5. The following formula is true:

$$F_{p,q}(z)=\frac{1}{\binom{n}{p}2^{n-p}}\frac{1}{2\pi i}\oint_{|u|=r}\frac{(u+2)^q(u+2+1)^{n-q}}{u^{p+1}}\mathrm{d}u,$$

where $r<1$.

Let us consider the matrix $\left\|\alpha_{ij}\right\|$ the rows of which are the codes $\lambda(J_1),\lambda(J_2),\cdots,\lambda(J_m)$ of the intervals from the family $\{J_1,J_2,\cdots,J_m\}$

Lemma 2. The following expression is true:

$$\sum_{i<j}\rho(J_i,J_j)=\sum_{i=1}^{n}k_i^0k_i^1,$$

where $k_i^0k_i^1$ is the number of zeros and, respectively, units in the i-th column of the matrix $\left\|\alpha_{ij}\right\|$.

Proof. According to the definition:

$$\sum_{i,j}\rho(J_i,J_j)=\sum_{i,j}\sum_{k=1}^{n}\delta_{i_j}^k, \tag{9}$$

where $\delta_{i_j}^{k}=\begin{cases}1, & \text{if } \alpha_i^k \neq \alpha_j^k \text{ and } \alpha_i^k \neq c, \alpha_j^k \neq c;\\ 0, & \text{otherwise,}\end{cases}$

and $\lambda(J_i)=\left(\alpha_i^1\ \alpha_i^2 \cdots \alpha_i^n\right)$ is the code of the interval J_i.

It follows from (9):

$$\sum_{i,j}\rho\left(J_i,J_j\right)=\sum_{k=1}^{n}\sum_{i,j}\delta_{i_j}^{k}. \tag{10}$$

The internal sum in (10) equals the number of such pairs $\left(\alpha_i^k,\alpha_j^k\right)$ in which one of $\alpha_i^k - s$ is unit and the other is zero, i.e. is $k_i^1 k_i^0$. The Lemma is proved.

Example.

1) Let $S(3)$ be the family of all edges in B^3. We consider the matrix of their codes:

$$\left\|\alpha_{ij}\right\|=\left\|\begin{matrix} c & 0 & 0\\ c & 0 & 1\\ c & 1 & 0\\ c & 1 & 1\\ \vdots & \vdots & \vdots\\ 0 & c & 0\\ 0 & c & 1\\ 1 & c & 0\\ 1 & c & 1\\ \vdots & \vdots & \vdots\\ 0 & 0 & c\\ 0 & 1 & c\\ 1 & 0 & c\\ 1 & 1 & c \end{matrix}\right\|$$

The total number of the edges in B^3 is B^3 is $\binom{n}{1}2^{n-1}/_{n=3}=12$. Each column of the matrix has the length 12, and all letters of the alphabet $\{0,1,c\}$ occur in equal number, 4 times. Therefore, $k_i^0=k_i^1=4$, $i=1,2,3.$

From here, we get:

$$\sum_{J_i,J_j\in S(3)} \rho\left(J_i,J_j\right)=\sum_{i=1}^{3}4\times 4=48.$$

2) In the general form, if $S(n)$ is the family of all edges in B^n, each column contains 2^{n-1} letters c and $k_i^0=k_i^1=\dfrac{n2^{n-1}-2^{n-1}}{2}=\left(n-1\right)2^{n-2}$.
From here, we get:

$$\sum_{J_i,J_j\in S(n)} \rho\left(J_i,J_j\right)=\left(n-1\right)^2 2^{2n-4}n=\left(n-1\right)^2 n2^{2n-4}. \qquad (11)$$

Thus, the sum of the pairwise distances between the intervals in B^n is calculated by formula (11).

The Sum of Spheres in B^n

In the general form, the following statement holds true.

Lemma 3. The following formula is true:

$$S_p^n\left(M\right)=M+S_p^n\left(0\right).$$

Proof. By definition:

$$S_p^n\left(M\right)=\underset{x\in M}{U}S_p^n\left(x\right)=\underset{x\in M}{U}\left\{x+S_p^n\left(0\right)\right\}=M+S_p^n\left(0\right).$$

Thus, the above introduced parameter $S_p^n\left(M\right)$ of the setMis rather easily expressed in the terms of the operation "+".

Lemma4. $S_p^n\left(S_q^n\left(M\right)\right)=S_{p+q}^n\left(M\right)$ if $p+q\le n$.

Proof. If $v\in S_p^n\left(S_q^n\left(M\right)\right)$, then either v is at the distance $\le p$ from $S_q^n\left(M\right)$ or there is a point $a\in S_q^n\left(M\right)$ such that $\rho\left(v,a\right)\le p$.

Then, from $a\in S_q^n\left(M\right)$, it follows that $\rho\left(a,z\right)\le q$, for all $z\in M$.

From here we get:

$$\rho\left(v,z\right)\le\rho\left(v,a\right)+\rho\left(a,z\right)\le p+q$$

or:

$$v\in S_{p+q}^n\left(M\right).$$

Hence, $S_p^n\left(S_q^n\left(M\right)\right)\subseteq S_{p+q}^n\left(M\right)$.

And if $v \in S^n_{p+q}(M)$, then there is a point $z \in M$ such that $\rho(v,z) \le p+q$
.

Hence, $\rho\left(v, S^n_q(z)\right) \le p$ and, consequently, $v \in S^n_p\left(S^n_q(M)\right)$, that is, $S^n_{p+q}(M) \subseteq S^n_p\left(S^n_q(M)\right)$, and the proof is completed.

Theorem 6. The following expression is true:

$$S^n_p(M_1) + S^n_q(M_2) = S^n_{p+q}(M_1 + M_2) \qquad (12)$$

Proof. We have from Lemma 4:

$$S^n_{p+q}(M_1 + M_2) = S^n_p\left(S^n_q(M_1 + M_2)\right).$$

Then, we have from Lemma 3:

$$\begin{aligned} S^n_{p+q}(M_1 + M_2) &= S^n_p(0) + S^n_q(M_1 + M_2) = S^n_p(0) + S^n_q(0) + M_1 + M_2 \\ &= \left(S^n_p(0) + M_1\right) + \left(S^n_q(0) + M_2\right) = S^n_p(M_1) + S^n_q(M_2). \end{aligned}$$

And the proof is over.

Formula (12) defines the rule of "addition" for arbitrary spheres in the space B^n.

Sum of Layers in B^n

Let $B^n_p = \left\{x \in B^n, \|x\| = p\right\}$ be the p-th layer of the n-dimensional cube, or be the sphere of the radius p with the center at zero [7] [8] .

According to definition, $B^n_p + B^n_q$ is the sum of the layers in B^n or it is the sum of the points with the weights p and q. As $\|x\| - \|y\| \le \|x + y\| \le \|x\| + \|y\|$, then all points from $\left(B^n_p + B^n_q\right)$ have weights from the interval $\{p-q, p+q\}$.

Then, the following statement is true.

Lemma 5. The following expression holds true:

$$\left(B^n_p + B^n_q\right) = \bigcup_{|p-q|+2r \le \min\{2n-(p+q),(p+q)\}} B^n_{a+2r}.$$

Proof. First, let us note the following.

If $x \in B^n_p + B^n_q$, and $\|x\| = m$, then $B^n_m \subseteq B^n_p + B^n_q$. Consequently, every layer B^n_r is invariant with respect to the operation of the symmetric group S_n and, for $g \in S_n$, we have:

$g(x+y) = g(x) + g(y)$, where $x, y \in B^n$.

In standard terms, the symmetric group S_n operates on B^n, and every layer is a transitive set or an orbit of action of the group S_n.

If $x \in B^n_p + B^n_q$, then $x = y + z$, where $y \in B^n_p, z \in B^n_q$. Therefore, for each permutation $g \in S_n$ we have: $g(x) = g(y+z) = g(y) + g(z)$, and we get:

$$\{g(x)\} \subseteq B^n_m \quad \text{и} \quad B^n_m \subseteq B^n_p + B^n_q.$$

Taking this into account, to describe the set $B^n_p + B^n_q$, it is sufficient to describe only the weights of the points which are included into this sum. The minimal weight of these is $d = |p - q|$.

We discuss the following outline:

$z = 111100$

$z_1 = 110000$

$z_2 = 011000$

$z_3 = 001100$

$z_4 = 000110$

$z_5 = 000011$

$$\|z + z_1\| = 2, \|z + z_2\| = 2, \|z + z_3\| = 2, \|z + z_4\| = 4, \|z + z_5\| = 6.$$

Here $n = 6, p = 4, q = 2$ and the “block” of the first 2 units is shifted by a unit in each consecutive word. Thus, we get all weights: $\|z + z_i\| : 2, 4, 6.$ In the general case, the situation is absolutely analogous, and the weights are arranged as follows:

$p - q,\ p - q + 2,\ p - q + 4, \cdots,$

for $p \geq q.$

Here the condition for z holds true:

$$p-q+2r \le \min\left\{p+q, 2n-\left(p+q\right)\right\}.$$

Examples.

1) $B_1^n + B_1^n = B_0^n \cup B_2^n$;
2) $B_2^n + B_2^n = B_0^n \cup B_2^n \cup B_4^n$;
3) $B_2^n + B_3^n = B_1^n \cup B_3^n \cup B_5^n$.

Theorem 7. The following expression is true:

$$B_p^n + S_q^n\left(M\right) = S_{l_1}^n\left(M\right) \setminus S_{l_2}^n\left(M\right),$$

where $l_1 = \min\left(p+q, n\right), l_2 = \max\left(0, p-q\right) - 1.$

Proof. From Lemmas 3 and 5, we have:

$$B_p^n + S_q^n(M) = B_p^n + \bigcup_{i=0}^{q} B_i^n + M = \bigcup_{i=0}^{q}\left(B_p^n + B_i^n\right) + M$$

$$= \bigcup_{i=|p-q|}^{p+q} B_i^n + M = S_{p+q}^n\left(0\right) \setminus S_{|p-q|}^n\left(0\right) + M = S_{p+q}^n\left(M\right) \setminus S_{|p-q|}^n\left(M\right).$$

The proof is over.

Sum of Subspaces in B^n

As usual, let $L\left(X\right)$ be the subspace generated by the vectors from the set X, or be the space "worn" on X.

Statement 2. $L\left(X_1 + X_2\right) = L\left(X_1\right) + L\left(X_2\right),$

and:

$$\dim L\left(X_1 + X_2\right) = \dim L\left(X_1\right) + \dim L\left(X_2\right) - \dim\left(L\left(X_1\right) \cap L\left(X_2\right)\right)$$

Statement 3. Let $X + X \subseteq L\left(X\right)$.

And if:

$$|X| > \frac{|L(X)|}{2}, \tag{13}$$

then the following equality is true:

$$X + X = L\left(X\right).$$

Proof. We assume the contrary, that is, $y \in L(X) \setminus (X + X)$.

Then, we have:

$X + y \subseteq L(X)$ and $(X + y) \cap X = \varnothing$.

Hence, $X \cup (X + y) \subseteq L(X)$. Consequently, $|X| + |X + y| \geq |L(X)|$. That is, $2|X| \leq |L(X)|$.

This contradicts the initial condition and the proof is over.

The following example shows that condition (12) is not necessary.

Example.

Let $X = \{(0000), (1000), (0100), (0010), (0001), (1111)\}$. Then $X + X = L(X)$, although $|X| = 6 < \frac{16}{2} = 8$.

EQUATIONS IN SETS

The “simplest” equation by sets is the following:

$$X + Y = A \qquad (14)$$

where $X, Y, A \in 2^{B^n}$.

Equation (14) always has the trivial solution: $X = \{x\}, Y = A + x$, where $x \in B^n$.

The significance of Equation (14) is explained by the following circumstances.

1) The standard problems of covering and partitioning in the Boolean space B^n [6] can be formulated as problems of describing the set of solutions of Equation (14).
2) For certain additional conditions, the solution of Equation (14) forms a perfect pair (perfect code) in the additive channel of communication [9] .
3) The set of all solutions of Equation (14) coincides with the class of equivalence of the additive channel of communication [3] .

Examples.

1) If $A = B^n$, we can take $(S_p^n(0), S_p^n(0)), p > \frac{n}{2}$ as the solution (X, Y) for Equation (14).

2) If A is a subspace of B^n, then Equation (14) has the following solution: $X = A, Y \subseteq A$.

The following statements are true:

Statement 4. If the equations $X + Y = A, X + Y = C$ are solvable, then the equation $X + Y = A + C$ also is solvable.

Proof. Let the pairs $(X_0, Y_0), (X_1, Y_1)$ be the solutions of the equations $X + Y = A$ and $X + Y = C$, respectively. Then for the pairs $((X_0 + X_1), (Y_0 + Y_1))$ we have $(X_0 + X_1) + (Y_0 + Y_1) = (X_0 + Y_0) + (X_1 + Y_1) = A + C,$ as was required to be proved.

Statement 5. For $|p - q| + 2r \leq \min\{p + q, n\}$, the equation:

$$X + Y = \bigcup_{r=0} B^n_{|p-q|+2r}$$

has the solution $X = B^n_p, Y = B^n_q$.

Statement 6. For $0 \leq p \leq n$ and $M \subseteq B^n$, the equation $X + Y = S^n_p(M)$ has the following solution:

$$X = S^n_{p_1}(M_1), Y = S^n_{p_2}(M_2),$$

where $p_1 + p_2 = p, M_1 + M_2 = M$.

Statement 7. For $0 \leq p, q \leq n, M \subseteq B^n$, the equation $X + Y = S^n_p(M) \setminus S^n_q(M)$ has the following solution:

$X = B^n_{p_1}, Y = S^n_{p_2}(M),\ \ where\ \ p = \min(n, p_1 + p_2), q = \max(0, p_1 - p_2) - 1.$

Statement 8. The sets of solutions (X, Y) of the equations $X + Y = A$ and $(X + M_1) + (Y + M_2) = A + (M_1 + M_2)$ coincide for all $M_1, M_2 \subseteq B^n$

Below, when discussing Equation (14), without violating generality, we may assume $0 \in X \bigcap Y \bigcap A,$ if necessary.

Statement 9. The equation $X + X = A$ has no solution for $|A| > \frac{n(n+1)}{2}, A \setminus \{0\} \subseteq B^n_p$.

Proof. If $A \subseteq B^n_p$ and $X + X = A$, for $a, b \in X$ we have $a + b \in A$ and

$\|a+b\| = p$ or $\rho(a,b) = p$. Thus, X is an equidistant code [2] , therefore, $|X| \le n+1$. Consequently:

$$|A| \le \frac{n(n+1)}{2}.$$

From here it follows that the equation $X + X = A$ has no solution for $|A| > \frac{n(n+1)}{2}$, if $A \setminus \{0\} \subseteq B_p^n$.

Statement 10. The equation $X + X = A$ (in “facets”, i.e. X, A are facets in B^n) is solvable iff the code of the interval A does not contain the letter 1.

Let (X_0, Y_0) be a solution of the equation $X + X = A$. As the following equality:

$$(X_0 \cup Y_0) + (X_0 \cup Y_0) = A$$

holds iff:

$$(X_0 + X_0) \subseteq A \text{ и } (Y_0 + Y_0) \subseteq A,$$

then the following statement is true.

Statement 11. If (X_0, Y_0) is a solution of the equation $X + Y = A$, then $(X_0 \cup Y_0)$ is a solution of the equation $X + X = A$, iff $(X_0 + X_0) \subseteq A$ and $(Y_0 + Y_0) \subseteq A$.

Statement 12. If A is a subspace from B^n, then every subset $X \subseteq A, |X| > \frac{|A|}{2}$, is a solution of the equation $X + X = A$.

In an additive channel of communication [3] an equivalence class has a unique representation by transitive sets of certain “generating” channels. The problem is to order these transitive sets by cardinalities of “generating” channels.

Let $N(A) = \{(X,Y), X + Y = A\}$.

We introduce the following parameters:

$$m(A) = \min_{(X,Y) \in N(A)} |X \cup Y|, \quad \overline{m}(A) = \begin{cases} |A \cup \{0\}|, & \text{if } N(A) = \varnothing \\ \min_{(X,X) \in N(A)} |X| \end{cases}$$

Such definition of $\overline{m}(A)$ is justified, because it is not always that the equation $X + Y = A$ has solutions for instance, if $|A| = 3, |A| = 5,$ or for

$0 \notin A$), though the equation $X + Y = A$ always has a solution.

One can easily prove that:

$$m(A) \leq \overline{m}(A) \leq |A \cup \{0\}|. \tag{15}$$

Statement 13. For the subspace $A \subseteq B^n$, the following is true:

$$m(A) = \overline{m}(A).$$

Proof. It follows from (15) that it is sufficient to prove that the following equality is true:

$$m(A) \geq \overline{m}(A).$$

Let (X_0, Y_0) be a solution of the equation $X + Y = A$, for which:

$$m(A) = |X_0 \cup Y_0|. \tag{16}$$

On the other hand, it follows from Statement 11 that $(X_0 \cup Y_0)$ is a solution of the equation $X + X = A$ and, consequently, $|X_0 \cup Y_0| \geq \overline{m}(A)$. Taking this and (16) into account, we get:

$$m(A) \geq \overline{m}(A).$$

Theorem 8. The following estimations are true:

1) $m(A) \geq \left]\frac{1}{2}\left((8|A|-7)^{\frac{1}{2}}+1\right)\right[$, *for* $A \subseteq B^n$;

2) $m(A) \leq 2^{\left[\frac{k}{2}\right]} + 2^{\left]\frac{k}{2}\right[} - 2,$ for the subspace $A \subseteq B^n$, for dim $A = k \geq 3$.

Proof. We have:

$$A = X + Y \subseteq (X \cup Y) + (X \cup Y).$$

From this and definition of addition of sets we get:

$$|A| \leq \|(X \cup Y) + (X \cup Y)\| \leq \frac{m(A)(m(A)-1)}{2} + 1.$$

Consequently:

$$m(A) \geq \left]\frac{1}{2}\left((8|A|-7)^{\frac{1}{2}}+1\right)\right[.$$

To prove the 2nd estimation, we consider such subspaces $L_1, L_2 \subseteq A$ for which the following is true:

$$L_1 \cap L_2 = \{0\}, \dim L_1 = \left[\frac{k}{2}\right], \dim L_2 = \left]\frac{k}{2}\right[.$$

Let:

$$X = (L_1 \setminus a_1) \cup (L_2 \setminus a_2) \cup (a_1 + a_2),$$

where $a_1 \in L_1, \;\; a_2 \in L_2, a_1 \neq 0, a_2 \neq 0.$

Let us prove that $(X, X) \in N(A)$.

We have:

$$\begin{aligned} X + X &= ((L_1 \setminus a_1) \cup (L_2 \setminus a_2) \cup (a_1 + a_2)) + ((L_1 \setminus a_1) \cup (L_2 \setminus a_2) \cup (a_1 + a_2)) \\ &= ((L_1 \setminus a_1) + (L_1 \setminus a_1)) \cup ((L_2 \setminus a_2) + (L_1 \setminus a_1)) \cup ((a_1 + a_2) + L_1 \setminus a_1) \\ &\quad \cup (L_1 \setminus a_1 + L_2 \setminus_2) \cup ((L_2 \setminus a_2) + (L_2 \setminus a_2)) \cup ((a_1 + a_2) + (L_2 \setminus a_2)) \\ &\quad \cup ((L_1) a_1 + (a_1 + a_2)) \cup ((L_2) a_2 + (a_1 + a_2)) \cup \{0\} \\ &= ((L_1 \setminus a_1) + (L_1 \setminus a_1)) \cup ((L_2 \setminus a_2) + (L_2 \setminus a_2)) \cup ((L_1 \setminus a_1) + (L_2 \setminus a_2)) \\ &\quad \cup ((a_1 + a_2) + (L_1 \setminus a_1)) \cup ((a_1 + a_2) + (L_2 \setminus a_2)). \end{aligned}$$

As (Statement 12) $(L_1 \setminus a_1) + (L_1 \setminus a_1) = L_1, (L_2 \setminus a_2) + (L_2 \setminus a_2) = L_2$ we get:

$$X + X = L_1 \cup L_2 \cup L \setminus ((L_1 + a_2) \cup (L_2 + a_1)) \cup (a_1 + a_2) + (L_1 \setminus a_1)) \cup ((a_1 + a_2) + (L_2 \setminus a_2)).$$

Then, using:

$$(L_1 \setminus a_1) + (a_1 + a_2) = (L_1 + a_2) \setminus a_1; L_2 \setminus a_2 + (a_1 + a_2) = (L_2 + a_1) \setminus a_2,$$

we get:

$$X + X = L_1 \cup L_2 \cup (L \setminus ((L_1 + a_2) \cup (L_2 + a_1)) \cup ((L_1 + a_2) \setminus a_1) \cup ((L_2 + a_1) \setminus a_2) = A.$$

Hence, taking this and Statement 13 into account we get:

$$m(A) = \overline{m}(A) \leq |X \cup X| = |(L_1 \cup L_2) \setminus (a_1 \cup a_2) \cup (a_1 + a_2)| = |L_1| + |L_2| - 3 + 1 = 2^{\left[\frac{k}{2}\right]} + 2^{\left]\frac{k}{2}\right[} - 2.$$

The statement is proved.

Examples.

1) $A = B^4$. We have:

$$m(A) = \overline{m}(A) = 6.$$

2) $A = B^5$. We have:

$$9 \le m(A) \le \overline{m}(A) \le 10,$$

but actually:

$$m(A) = \overline{m}(A) = 10.$$

3) $A = B^7$. We have:

$$17 \le m(A) \le \overline{m}(A) \le 22.$$

We consider the set:

$$\begin{aligned} X = \{&0000000,\ 0000100,\ 0000110,\ 0001101,\ 0010001,\ 0010111, \\ &0100000, 0100100,\ 0110100,\ 0110101,\ 0111011,\ 0111100, \\ &0111110,\ 1000010,\ 1001101,\ 1001110,\ 1010000,\ 1011011, \\ &1100101,\ 1101100\}. \end{aligned}$$

We have: $X + X = B^7, u|X| = 20.$ Hence, $m(A) \le \overline{m}(A) \le 20.$

4) $A_1 = \{000000,\ 100000,\ 010000,\ 001000,\ 000100,\ 000010,\ 000001\}.$
$A_2 = \{100000,\ 010000,\ 001000,\ 000100,\ 000010,\ 000001\}.$

We have:

$$6 \le m(A_1) \le 7 \quad \text{and} \quad 6 \le m(A_2) \le 7.$$

But actually $m(A_1) = m(A_2) = 7.$

Suggestion. For each $A \subseteq B^n$ the following is true:

$$m(A) \le 2^{\left[\frac{k}{2}\right]} + 2^{\left]\frac{k}{2}\right[} - 2, \quad where \quad k = \dim L(A) \ge 3.$$

REFERENCES

1. Nigmatulin, R.G. (1991) Complexity of Boolean Functions. Moscow, Nauka, 240 (in Russian).
2. McWilliams, F.J. and Sloane, N.J.A. (1977) The Theory of Error-Correcting Codes, Parts I and II. North-Holland Publishing Company, Amsterdam.
3. Leontiev, V.K. (2001) Selected Problems of Combinatorial Analysis. Bauman Moscow State Technical University, Moscow, 2001 (in Russian).
4. Leontiev, V.K. (2015) Combinatorics and Information. Moscow Institute of Physics and Technology (MIPT), Moscow, 2015 (in Russian).
5. Leontiev, V.K., Movsisyan, G.L. and Osipyan, A.A. (2014) Classification of the Subsets Bn, and the Additive Channels. Open Journal of Discrete Mathematics (OJDM), 4, 67-76.
6. Leontiev, V.K., Movsisyan, G.L., Osipyan, A.A. and Margaryan, Zh.G. (2014) On the Matrix and Additive Communication Channels. Journal of Information Security (JIS), 5, 178-191.
7. Leontiev, V.K., Movsisyan, G.L. and Margaryan, Zh.G. (2012) Constant Weight Perfect and D-Representable Codes. Proceedings of the Yerevan State University, Physical and Mathematical Sciences, 16-19.
8. Movsisyan, G.L. (1982) Perfect Codes in the Schemes Johnson. Bulletin of MSY, Computing Mathematics and Cybernetics, 1, 64-69 (in Russian).
9. Movsisyan, G.L. (2013) Dirichlet Regions and Perfect Codes in Additive Channel. Open Journal of Discrete Mathematics (OJDM), 3, 137-142.

Chapter

MULTIPATH DETECTION USING BOOLEAN SATISFIABILITY TECHNIQUES

17

Fadi A. Aloul[1] and Mohamed El-Tarhuni[2]

[1]Department of Computer Science and Engineering, American University of Sharjah, United Arab Emirates
[2]Department of Electrical Engineering, American University of Sharjah, United Arab Emirates

ABSTRACT

A new technique for multipath detection in wideband mobile radio systems is presented. The proposed scheme is based on an intelligent search algorithm using Boolean Satisfiability (SAT) techniques to search through the uncertainty region of the multipath delays. The SAT-based scheme utilizes the known structure of the transmitted wideband signal, for example, pseudo-random (PN) code, to effectively search through the entire space by eliminating subspaces that do not contain a possible solution. The paper

Citation: Fadi A. Aloul, Mohamed El-Tarhuni, "Multipath Detection Using Boolean Satisfiability Techniques", Journal of Computer Networks and Communications, vol. 2011, Article ID 365107, 9 pages, 2011,. https://doi.org/10.1155/2011/365107.

presents a framework for modeling the multipath detection problem as a SAT application. It also provides simulation results that demonstrate the effectiveness of the proposed scheme in detecting the multipath components in frequency-selective Rayleigh fading channels.

INTRODUCTION

There has been a growing interest in developing high data rate mobile radio systems to support a wide range of applications such as real-time multimedia services and high-speed internet access. To achieve this goal, wide band transmission schemes are being investigated including single carrier and multicarrier spread spectrum techniques, Ultra-wideband systems, and OFDM-based schemes. Multipath propagation, caused by reflection, refraction, and scattering of radio waves as they pass through the wireless channel, is considered as one of the main challenges in wide band mobile radio communication systems. Multipath propagation results in receiving multiple copies of the transmitted signal. In narrow band transmission schemes, where the multipath components are very close and unresolved by the receiver, severe fading is observed in the received signal strength leading to significant degradation in the bit error rate (BER) performance of the system. On the other hand, in wide band signal transmission, where multipath components could be resolved by the receiver, multipath propagation can be exploited using a RAKE receiver to improve the system BER performance through the diversity gain from the different copies of the received signal. However, for full utilization of the multipath scenario, it is very important for the receiver to first detect the presence of these multipath components and identify their corresponding parameters (time delay, amplitude, and phase).

In spread spectrum systems, a pseudo-random (PN) code is used to spread the message spectrum over a wide bandwidth. At the receiving end, a time-synchronized version of the same PN code is used to despread the signal and recover the original message [1]. Synchronization is very crucial for the proper operation of the system. It can be accomplished by searching a range of delays for the correct multipath delays. The uncertainty range represents the possible delays that the signal may have and is related to the channel memory. The delay range is usually specified as cells that are one-chip or one-half of a chip apart, where a chip is the shortest element in the PN code. The search for the multipath components through these cells, that is, finding the cells that have strong energy and hence multipath components, can either be done in a serial or parallel fashion [2–5].

In serial search, one cell at a time is tested by measuring the signal energy at that cell using a single correlator circuit. If the energy exceeds a preset threshold, then the cell is declared as a multipath cell, either directly or after a verification stage, while if the energy is below the threshold, then it is declared as a no multipath cell. The search advances to the next cell and the process is continued until all cells in the uncertainty range are tested. The other search strategy uses parallel search where the energies of all cells are calculated simultaneously using a bank of parallel correlators and cells with energy above the threshold are declared as multipath cells. Apparently, serial search is slower compared to parallel search as it takes longer time to search all the cells and find the delays. On the other hand, serial search has a much lower reduced complexity (both hardware and processing).

A common drawback of existing schemes is that in searching for the correct cells they do not utilize the inherent structure of the PN code. In the worst case or in a low SNR environment, these schemes need to search all possible cells in the search window, which could be as large as the length of the PN code, in order to find the correct cells. For example, for a PN code with a length of 2047 chips (generated by an 11-stage shift register) the serial and parallel search schemes need to test 2047 cells if the search step is one chip or twice of that if the search step is one-half of a chip. This testing may need to be repeated many times if the multipath components were not detected at the first trial due to noise and fading. In this paper, we propose a PN code acquisition scheme that exploits the structure of the PN code to reduce the number of decisions needed to find correct cells. The proposed scheme is based on using advanced Boolean Satisfiability (SAT) techniques to perform intelligent search of the uncertainty region and hence reduce the number of decisions needed to find the correct cells significantly. This is done by searching only PN code phases that result in minimum difference (minimum distance) between the PN code in the received signal and a locally generated PN code.

Recently, Boolean Satisfiability (SAT) has been shown to be very successful in solving complex problems in various Engineering and Computer Science applications. Such applications include Formal Verification [6], FPGA routing [7], Power Optimization [8, 9], Fault Tolerance [10], and Microprocessor Verification [11]. SAT has also been extended to a variety of applications in Artificial Intelligence including other well-known NP-complete problems such as graph colorability, vertex cover, hamiltonian path, and independent sets [12]. Despite SAT being an NP-Complete problem [13], many researchers have developed powerful SAT solvers that are able of handling problems consisting of thousands of variables and millions of constraints [14–22]. Briefly defined, the SAT problem involves a set of Boolean variables and a set of constraints expressed in product-of-sum form. The goal is to identify an assignment to the variables that would satisfy all constraints or prove that no such assignment exists.

Even though in recent years we have seen a surge in the application of SAT techniques to assist in finding solutions to various Engineering problems, very few researchers reported on the use of SAT-based techniques in mobile communication-related research. In this paper, we propose the formulation of the PN acquisition problem as a SAT instance and use intelligent SAT search engines for multipath detection.

The reminder of this paper is organized as follows. Sections 2 and 3 present the signal model and an overview of SAT, respectively. Section 4 describes the proposed scheme and shows how to formulate the PN code acquisition problem as a SAT instance. Simulation results are presented and discussed in Section 5. Finally, the paper is concluded in Section 6.

SIGNAL MODEL

A direct-sequence spread spectrum system is investigated in this paper. The signal model assumes that a separate pilot signal is transmitted along with the data channel to allow for PN code acquisition and tracking as well as channel estimation. The transmitted signal is given by

$$s(t) = \sqrt{P} \sum_{i=0}^{M-1} \left(d_i W + \sqrt{G_p} V \right) \sum_{k=0}^{N-1} c_k g(t - iT_b - kT_c), \qquad (1)$$

where P is the transmitted power, d_i is a random sequence of information data with $d_i \in \pm 1$, W and V are orthogonal codes with length N (i.e., Walsh codes) used to separate the pilot channel from the data channel, G_p is the pilot channel power gain relative to the data channel, $c_k \in \pm 1$ is the spreading pseudo-random (PN) code, N is the PN code length which is the same as the number of chips per bit, that is, $N = T_b/T_c$, T_b is the bit duration, T_c is the chip duration, and g(t) is the chip pulse shape. M is the number of data bits.

The radio channel is modeled as a frequency-selective Rayleigh fading channel, which is a common model for mobile radio systems, using narrow-band transmission. The received signal is given by

$$u(t) = \sum_{l=1}^{L} \beta_l s(t - \tau_l) + n(t), \tag{2}$$

where L is the number of paths, β_1 is the lth path complex coefficient with Rayleigh amplitude and uniform phase distribution over [0, 2π), τ_1 is the lth path delay that we would like to estimate, and n(t) is an additive white Gaussian noise (AWGN) with zero mean and two-sided power spectral density $N_0/2$ that models the effect of the receiver noise.

To maximize the signal-to-noise ratio, the received base-band signal is first applied to a chip-matched filter to produce the following signal samples at the chip rate:

$$z[k] = \int_{(k-1)T_c}^{kT_c} u(t)g(t)dt. \tag{3}$$

In conventional PN code acquisition schemes, the output of the chip-matched filter is correlated with a locally generated PN code with different offsets that cover the delay uncertainty region (possibly the whole PN code period) as follows:

$$y[i] = \sum_{k=0}^{N-1} z[k-i]c_k; \quad i = 0, 1, \ldots, N-1, \tag{4}$$

where the index indicates the delay offset under test. The correlation results in (4) are used to estimate the energy at different delay offsets and a decision is made on the existence of the multipath delays based on the highest energy values. It is also common to use a preset threshold where only energy values that exceed the threshold are declared as potentially correct multipath components while others are ignored. Note that in some cases, especially for very long PN codes, it is possible to perform the correlation over a fraction of the code length and the upper limit in (4) will be less than N −1.

The main objective of the acquisition system is to maximize the probability of detection while minimizing the probability of false alarm. Based on the outcome of the decision process, we can have one of the following events.

- *Detection:* This event occurs when the energy value exceeds the threshold and the estimated delay matches one of the actual delays of the multipath components in the received signal. We would like to maximize the detection probability to improve the performance of the RAKE receiver in detecting the transmitted data.

- *False Alarm:* This event occurs when the energy value exceeds the threshold but the estimated delay did not match any of the actual delays of the multipath components. We would like to minimize the false alarm probability since the RAKE receiver would be using a signal that has no useful energy to detect the data.
- *Miss:* This event occurs when the energy value is below the threshold but the delay offset has a correct multipath component. We would like to minimize such event since the RAKE receiver will not get all useful energy in detecting the data.

It is also noted that there are other performance criteria for evaluating code acquisition schemes, such as the mean acquisition time and the probability of achieving correct acquisition within a specified period of time [23].

BOOLEAN SATISFIABILITY

The last few years have seen significant advances in Boolean satisfiability (SAT) solving. These advances have lead to a successful deployment of SAT solvers in a wide range of problems in Engineering and Computer Science. Given a set of Boolean variables and a set of constraints expressed in product-of-sum form, the goal of SAT solver is to find a variable assignment that satisfies all constraints or prove that no such assignment exists. The term “Satisfiability” emerges from that fact that we are asked to find a satisfying assignment, while the term “Boolean” comes from the fact that such assignment consists of only *true* or *false* variable states.

The SAT problem is usually expressed in conjunctive normal form (CNF). A CNF formula φ on n binary variables $x_1, \dots , x_n$ is the conjunction (AND) of m clauses $\omega_1, \dots ,\omega_m$ each of which is a disjunction (OR) of one or more literals, where a literal is the occurrence of a variable or its complement. A formula φ maps to a unique n-variable Boolean function $f (x_n, \dots , x_n)$ [24]. Clearly, a function f can be represented by many equivalent CNF formulas. We will refer to a CNF formula as a *clause database* and use “formula” and “CNF formula” interchangeably.

A variable x is said to be *assigned* when its logical value is set to 0 or 1 and *unassigned* otherwise. A literal is a *true* (*false*) literal if it evaluates to 1 (0) under the current assignment to its associated variable, and a *free literal* if its associated variable is *unassigned*. A clause is said to be *satisfied* if at least one of its literals is true, *unsatisfied* if all of its literals are false, *unit* if all but a single literal are set to false, and *unresolved* in the remaining cases. A formula is said to be satisfied if all its clauses are satisfied, and unsatisfied if at least one of its clauses is unsatisfied. In summary, the SAT problem is defined as follows. Given a Boolean formula in CNF, find an assignment of variables that satisfies the formula or prove that no such assignment exists.

In the following example, the CNF formula

$$\varphi = (a \vee b) \cdot \left(\overline{b} \vee c\right) \cdot (\overline{a} \vee c) \qquad (5)$$

consists of 3 variables, 3 clauses, and 6 literals. The assignment {a = 1, b = 0,c = 0} violates the third clause and unsatisfies φ, whereas the assignment t {a = 1, b = 0,c = 1} satisfies φ. Note that a problem with n variables will have 2^N possible assignments for the variables. The above example with 3 variables has 8 possible assignments.

Despite the SAT problem being NP-Complete [13], there have been dramatic improvements in SAT solver technology over the past decade. This has lead to the development of several powerful SAT algorithms that are capable of solving problems consisting of thousands of variables and millions of constraints. Such solvers include GRASP [18], zChaff [17], Berkmin [20], MiniSAT [16], and RSat [21]. In the next three subsections, we describe the basic SAT search algorithm, recent extensions to the SAT solver input, and the use of hardware with SAT.

Backtrack Search

Most modern complete SAT algorithms can be classified as enhancements to the basic Davis-Logemann-Loveland (DLL) backtrack search approach [25]. The DLL procedure performs a search process that traverses the space of 2^N variable assignments until a satisfying assignment is found (the formula is satisfiable), or all combinations have been exhausted (the formula is unsatisfiable). It maintains a *decision tree* to keep track of variable assignments and can be viewed as consisting of three main engines: (1) *Decision* engine that makes *elective* assignments to the variables, (2) *Deduction* engine that determines the consequences of these assignments, typically yielding additional *forced* assignments to, that is, implications of, other variables, and (3) *Diagnosis* engine that handles the occurrence of conflicts, that is, assignments that cause the formula to become unsatisfiable, and backtracks appropriately. An example of a decision tree is shown in Figure 1.

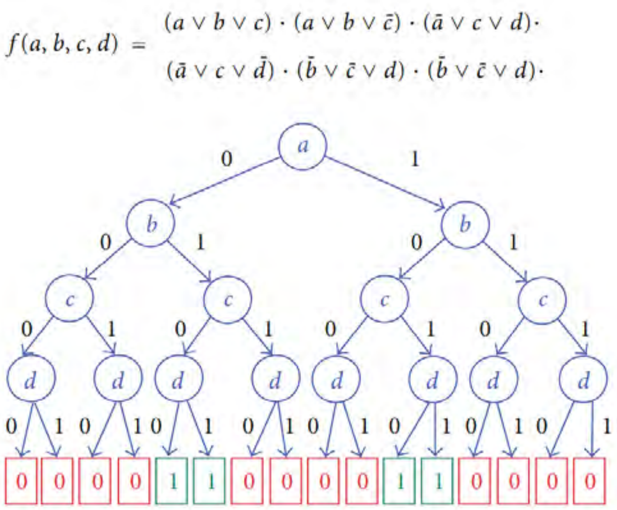

$f(a,b,c,d) = (a \vee b \vee c) \cdot (a \vee b \vee \bar{c}) \cdot (\bar{a} \vee c \vee d) \cdot (\bar{a} \vee c \vee \bar{d}) \cdot (\bar{b} \vee \bar{c} \vee d) \cdot (\bar{b} \vee \bar{c} \vee d) \cdot$
a
0
1
b
c
d
0 0 0 0 1 1 0 0 0 0 1 1 0 0 0 0

Figure 1. An example of a satisfiable SAT instance showing its corresponding decision tree.

Recent studies have proposed the use of the *conflict analysis* procedure in the diagnosis engine [18]. The idea is whenever a conflict is detected, the procedure analyzes the variable assignments that cause one or more clauses to become unsatisfied. Such analysis can identify a small subset of variables whose current assignments can be blamed for the conflict. These assignments are turned into a *conflict-induced clause* and augmented with the clause database to avoid regenerating the same conflict in future parts of the search process. In essence, the procedure performs a form of learning from the encountered conflicts. Today, conflict analysis is implemented in almost all SAT solvers [16–18, 20, 21].

More Expressive Input

Restricting the input of SAT solvers to CNF formulas can restrict their usage in various domains. Therefore, researchers have focused on extending SAT solvers to handle stronger input representations. Specifically, SAT solvers

[14–16, 19, 22] have recently been extended to handle pseudo-Boolean (PB) constraints which are linear inequalities with integer coefficients that can be expressed in the normalized form [14] of

$$a_1x_1 + a_2x_2 + \cdot \cdot \cdot + a_nx_n \geq b, \tag{6}$$

where a_i, $b \in Z^+$ and x_i are literals of Boolean variables. Note that any CNF clause can be viewed as a PB constraint; for example, clause $(a \lor b \lor c)$ is equivalent to $(a + b + c \geq 1)$.

PB constraints can, in some cases, replace an exponential number of CNF constraints. They have been found to be very efficient in expressing "counting constraints" [14]. Furthermore, PB extends SAT solvers to handle *optimization* problems as opposed to only *decision* problems. Subject to a given set of CNF and PB constraints, one can request the minimization (or maximization) of an objective function which consists of a linear combination of the problem's variables:

$$\sum_{i=1}^{n} a_i x_i. \tag{7}$$

This feature has introduced many new applications to the SAT domain. Recent studies have also shown that SAT-based optimization solvers can in fact compete with the best generic integer linear programming (ILP) solvers [14, 15].

Hardware-Based SAT Solvers

Note that SAT solvers can be implemented in hardware. Several studies proposed the use of FPGA reconfigurable systems to solve SAT problems [26–29]. Hardware solvers could be a standalone or as an accelerator where the problem is partitioned between the hardware solver and the attached computer using software. Many different architectures were proposed to solve SAT problems in hardware. Linearly connected set of finite state machines, control unit, and deduction logic was proposed in [29]. The authors in [29] implemented their algorithm on Xilinx XC4028 FPGA. While in [26], the authors proposed a technique for modeling any Boolean expression. Their objective is to set the function output to 1. A backtrack algorithm is used to propagate the output back to the input and finding an assignment of the inputs to satisfy a logical 1 at the output.

The authors in [27] proposed an architecture for evaluating clauses in parallel. In their architecture, the clauses are separated into a number of groups and the deduction is performed in parallel. Then the results are merged together to allow the assignment to the variables.

A software/hardware solver for SAT was introduced in [28]. In their approach, they minimized the hardware compilation time which greatly reduced the total time to solve the problem. They also implemented their solver on an FPGA.

SAT MODEL FOR PN CODE ACQUISITION

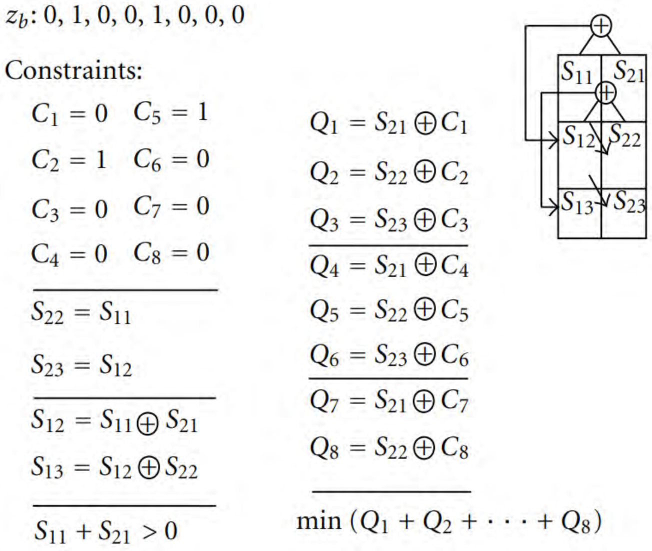
This section describes how to formulate the PN Code acquisition problem as a SAT instance to be able to process the received signal and find the delays of the L multipath components. As explained earlier, the received baseband signal is passed through a chip-matched filter to obtain the signal in (4). This signal contains delayed versions of the PN code (multipath components) plus a data part and noise. Since we are dealing with Boolean satisfiability (SAT), the first step is to convert the matched filter output to a binary sequence $z_b = \{z_b[0], z_b[1], \ldots, z_b[n-1]\}$ as follows:

$$z_b[i] = \begin{cases} 1, & z[i] \geq 0, \\ 0, & z[i] < 0. \end{cases} \qquad (8)$$

Although hard decisions are in general not sufficient statistics for estimating the delay, but in the context of the developed SAT model for PN acquisition it would be enough to provide an estimate of the received PN code and hence allows for the SAT search to be implemented as will be discussed later.

The basic idea of the proposed algorithm is to locally generate a block of size n of the PN code using the known shift register (SR) structure with different initial states. A state is basically the content of the shift register at any instant of time. The SAT solver is used to find the initial state that would result in a PN sequence that is very close (ideally the same) to the received sequence z_b. Since an m-stage SR is used, then we will have $2^m - 1$ possible initial states to be tested. However, the SAT solver uses intelligent algorithms to efficiently traverse the decision tree and quickly find a valid solution as described in Section 3. Once a solution is found, that is, finding an initial state of the SR that will result in the smallest difference (we also call it *distance*) between the locally generated PN code and the received sequence, the delay of the first multipath component is obtained from this initial state. The SAT solver then searches for the next initial state that would result in the next smallest distance to find the delay of the second multipath component. This process is repeated until all L multipath components are detected.

In order to illustrate how the state of the SR can be used to find the delay of a multipath component and without loss of generality, we assume a 2-stage SR used to generate a PN code of length $2^2 - 1 = 3$ chips as shown in Figure 2. Both stages are used to generate the feedback input to the SR through the XOR gate. Since we have two stages in the SR, there are 3 possible initial states, and once the SR is clocked at the chip rate, then the following states would be generated: {01, 10, 11}, {10, 11, 01}, or {11, 01, 10} depending on which initial state was used. Suppose that the transmitter uses a PN code with initial state of 01 and the channel causes a delay of one chip, then the initial state of the PN code to be used by the receiver to match the received signal would be 10. On the other hand, if the channel causes a two-chip delay, then the solution for the initial state would be 11. Hence, we can estimate the channel delay based on the initial state of the SR that would result in best match with the received signal.



z_b: 0, 1, 0, 0, 1, 0, 0, 0

Constraints:

$C_1 = 0 \quad C_5 = 1$

$C_2 = 1 \quad C_6 = 0$

$C_3 = 0 \quad C_7 = 0$

$C_4 = 0 \quad C_8 = 0$

$S_{22} = S_{11}$

$S_{23} = S_{12}$

$S_{12} = S_{11} \oplus S_{21}$

$S_{13} = S_{12} \oplus S_{22}$

$S_{11} + S_{21} > 0$

$Q_1 = S_{21} \oplus C_1$

$Q_2 = S_{22} \oplus C_2$

$Q_3 = S_{23} \oplus C_3$

$Q_4 = S_{21} \oplus C_4$

$Q_5 = S_{22} \oplus C_5$

$Q_6 = S_{23} \oplus C_6$

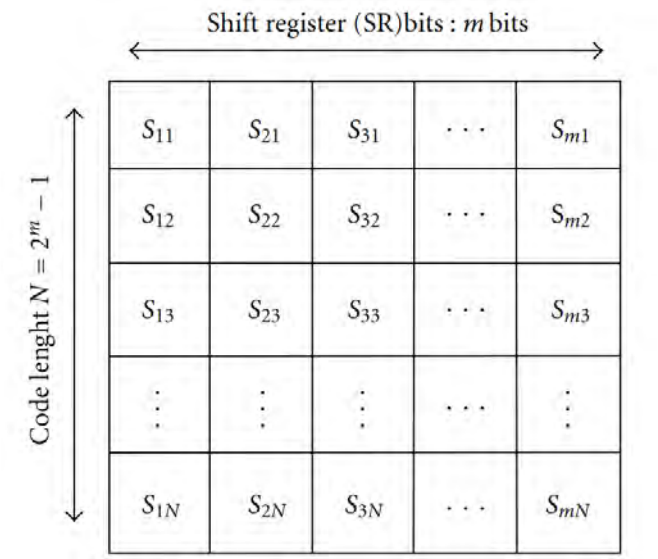
$Q_7 = S_{21} \oplus C_7$

$Q_8 = S_{22} \oplus C_8$

$\min (Q_1 + Q_2 + \cdots + Q_8)$

Figure 2. An example of a network with 8 data bits and 2 SR bits.

In order to use the advanced SAT solvers to find the L multipath delays in the received signal, the problem must be first expressed in the SAT solver input format as described in Section 3. To illustrate our approach, let us assume a system consisting of n received chips, and a Shift Register (SR) with m stages. The code length N is equal to (2^m − 1) levels as shown in Figure 3.



S_{11}	S_{21}	S_{31}	$\cdots$	S_{m1}
S_{12}	S_{22}	S_{32}	$\cdots$	S_{m2}
S_{13}	S_{23}	S_{33}	$\cdots$	S_{m3}
$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$
S_{1N}	S_{2N}	S_{3N}	$\cdots$	S_{mN}

Figure 3. Sample layout of Shift Register bits.

Three sets of *variables* are defined for the problem as follow.

- A Boolean variable C_i is defined for each chip at the matched filter output at sample time i, that is, a total of n variables. A value of 1 or 0 for each variable indicates that the corresponding chip is a 1 or 0, respectively. Note that this variable is the same as the sequence z_b that was introduced in (8).
- A Boolean variable Q_i is defined for each matched filter output as the difference between the C_i and the PN code chip, that is, a total of n variables.
- A Boolean variable S_{ij} is defined for each SR stage i at each level j, that is, a total of m × (2^m − 1) variables.

Thus, the total number of needed Boolean variables is equal to 2n + m × (2^m − 1).

The following set of CNF and PB constraints are generated.

- *Received Chips Constraints:* This constraint is used to set the input sequence utilized by the SAT solver to be compared with the locally generated PN code. The input sequence is obtained from (8). For each received chip i, its corresponding C_i bit is set to 0 or 1 depending on the feed data. This can be expressed using a single PB constraint per chip as follows:

$$C_i = z_b[i]; \quad i = 0, 1, \ldots, n-1, \tag{9}$$

that is, a total of *n* PB constraints.

- *Initial State Constraints:* This constraint is used to ensure that the initial SR state should have at least one bit assigned to 1 to avoid having an all-zero state for the SR. This can be expressed using a *single* PB constraint as follows:

$$\left(\sum_{i=1}^{m} S_{i1}^{k}\right) > 0; \quad k = 1, 2, \ldots, 2^m - 1. \tag{10}$$

- *Shifting Constraints:* This constraint implements the shifting operation as the shift register is clocked; for example, $S_{22} = S_{11}$, $S_{32} = S_{21}$, ..., is expressed using the following equality constraint per SR stage:

$$(S_{il} = S_{(i-1)(l-1)}); \quad l = 2, \ldots, N; \; i = 2, \ldots, m. \tag{11}$$

This results in a total of $(m - 1)(2^m - 2)$ equality constraints. Each equality constraint of format (x = y) can be expressed using two CNF constraints as shown in Table 1.

Table 1. Expressing logical constraints using CNF constraints.

Logical Constraint	CNF Constraint
$(x = y)$	$(\bar{x} \vee y) \cdot (x \vee \bar{y})$
$(x = y \oplus z)$	$(\bar{x} \vee y \vee z) \cdot (x \vee \bar{y} \vee z) \cdot (x \vee y \vee \bar{z}) \cdot (\bar{x} \vee \bar{y} \vee \bar{z})$

- *Feedback Constraints:* This constraint ensures that the correct SR stages as used in the feedback part of the PN code generator. The PN code feedback relation is expressed using the following XOR constraint per initial SR content:

$$\left[S_{1l} = S_{p(l-1)} \bigoplus \ldots \bigoplus S_{q(l-1)}\right]; \quad l = 2, \ldots, N, \tag{12}$$

where p, q ∈ {1, ... , n} is selected according to the feedback connection of the PN code generator. This results in a total of (2^m − 2) XOR constraints. Each XOR constraint of format (x = y ⊕ z) is expressed using four CNF constraints as shown in Table 1.

- *Difference Constraints:* The mismatch between the received chip sequence and locally generated PN code, taken from the mth stage of the SR and for a given initial state k, is calculated as follows:

$$\left[Q_i^k = S_{mi}^k \bigoplus C_i\right]; \quad i = 1, \ldots, n; \; k = 1, 2, \ldots, 2^m - 1. \tag{13}$$

This results in n XOR constraints. As mentioned earlier each XOR constraint can be expressed using four CNF constraints.

- *Optimization Function:* The objective of the SAT algorithm is to search through the possible initial SR states that results in minimizing the error (distance) between the received sequence and locally generated code. This is expressed using the following PB optimization objective:

$$D_k = \min\left(\sum_{i=1}^{n} Q_i\right); \quad k = 1, 2, \ldots, 2^m - 1. \tag{14}$$

The algorithm finds the smallest L values of the distance and the corresponding SR initial states. Then, the L multipath delays are estimated from the states as was explained earlier.

To further illustrate the formulation in SAT input, consider the example in Figure 2. The system consists of 8 data bits and 2 SR bits. Hence, the code length N is 3. The SAT problem generates a total of 2 × 8 + $2(2^2 - 1)$ = 22 Boolean variables. The figure displays the needed constraints.

SIMULATION RESULTS

In this paper, we simulated a direct-sequence spread spectrum system with a PN code of length 2047 (11-stage shift register) operating over a frequency-selective Rayleigh fading channel with uniform power delay profile and a normalized Doppler of 10^{-3}. The Doppler frequency is normalized by the PN code length. Every simulation was repeated for 2000 independent trials. Although a square pulse shape was used for each chip, other pulse shaping methods may be used with no impact on the use of the proposed scheme. The number of paths is assumed to be three. The performance is measured by the probability of detecting at least one, two, or three multipath components as a function of the signal-to-noise ratio per chip (SNRc). This was done by finding the three minimum distances according to (14) and then checking if the initial shift register state corresponds to the correct delay or not. If the state matches the delay, then detection is declared; otherwise a miss is declared; Note that this is possible because we are performing simulation analysis, but in practice we expect to use a threshold to decide if a path exists or not. The effect of the duration of the correlation period used in calculating the difference between the locally generated PN code and the received data on the detection probability is also investigated. The performance is compared to that of a conventional energy-detector algorithm that measures the correlation at every possible offset and selects the energy of the three strongest paths. In these simulations, the Boolean Satisfiability (SAT) algorithm finds the delays of the three initial states of the SR that results in minimum error. All experiments were performed on an Intel Xeon 3.2 GHz workstation with 4 GB of RAM. We used the PBS 0-1 SAT-based ILP solver [14] for all experiments. Note that the above parameters were chosen for illustration purposes and are not expected to cause any restriction in the application of the proposed algorithm. Figure 4 shows the detection probabilities for a relatively short correlation period of 128 chips. It is clear that the multipath detection performance is relatively poor for both the SAT-based and conventional algorithms, although the latter shows better performance.

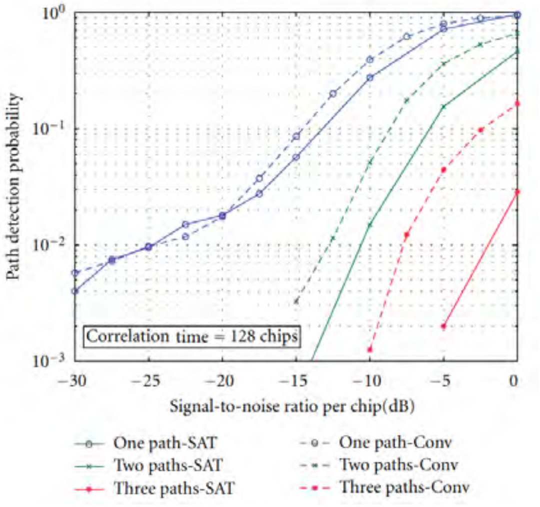


Figure 4. Probability of detection with 128 chips correlation.

The multipath detection performance is shown in Figures 5, 6, and 7 for a correlation period of 256, 512, and 1024 chips, respectively. The results show that the performance improved significantly to about 80% for detecting the three paths at an SNRc of zero dB. The detection of at least one or two paths is quite high indicating that the algorithm is successful in finding these delays. We also remark that as the correlation period increases, the SAT-based algorithm performance becomes closer to the conventional algorithm. Note that the SAT algorithm finds the correct delays by searching through the decision tree in an intelligent way and hence results in a reduced number of decisions compared to a brute force search strategy.

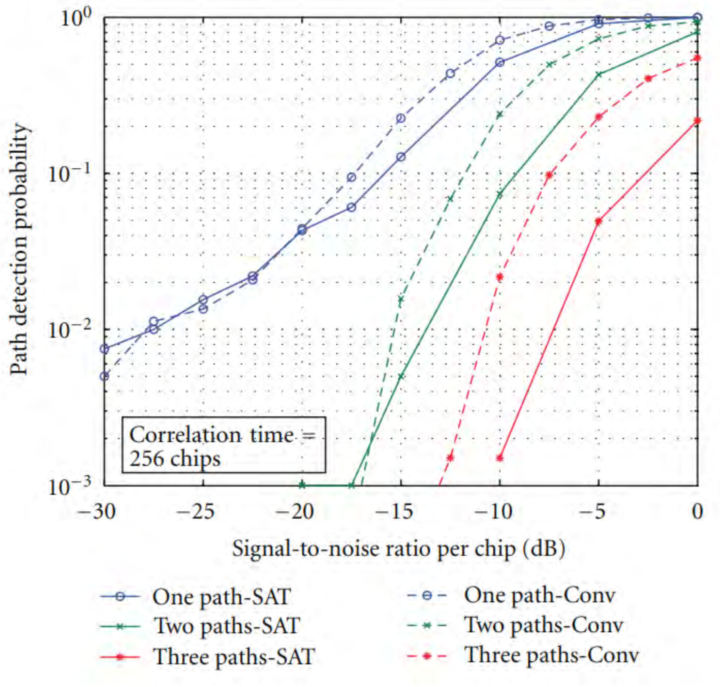


Figure 5. Probability of detection with 256 chips correlation.

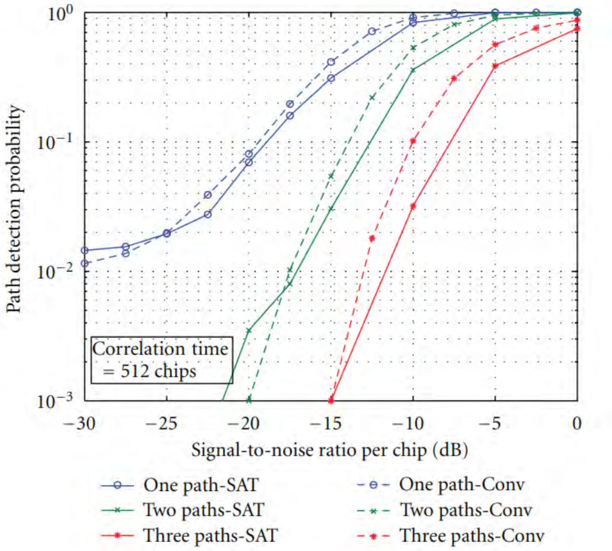


Figure 6. Probability of detection with 512 chips correlation.

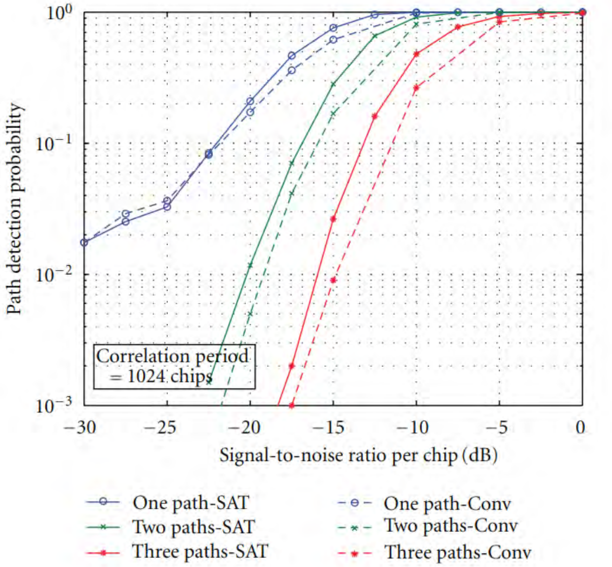


Figure 7. Probability of detection with 1024 chips correlation.

The SAT-based algorithm searched for the possible states that match the received signal with the PN code and the states that result in minimum difference, that is, the minimum distance between the received signal and locally generated sequence, is used to find the delay estimate. Figure 8 shows the minimum distance found at different values of the correlation period over an AWGN channel. It is observed that the difference tends to decrease as the SNRc increases because the SAT algorithm is supplied with more reliable data for the search.

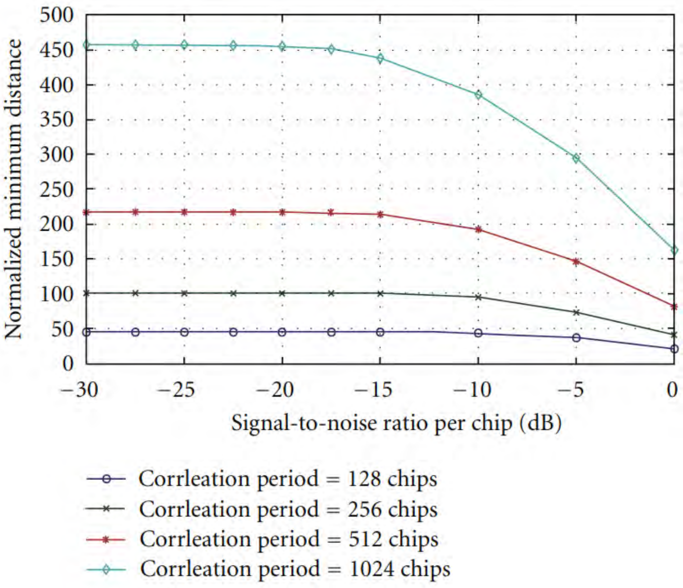


Figure 8. Minimum distance versus SNRc.

Finally, we notice that it is difficult to make a direct comparison of the computational cost between the proposed SAT-based algorithm and the conventional correlation based since the metrics used by the algorithms are different. In particular, the conventional scheme uses the number of multiplications and additions needed to search for the multipath components and this is typically quantified as N^2 multiply-and-add operations where N is the PN code length and assumed to be the search window. For the proposed SAT scheme, the complexity is measured by the number of decisions made while traversing the decision tree to look for a valid solution to the instance. The SAT solver uses advanced algorithms to intelligently traverse the decision tree and eliminate unsatisfiable paths. Depending on the instance's constraints, the SAT solver might be able to find a solution faster for some instances than others. In our simulations, most instances were solved after N decisions.

CONCLUSIONS

A new multipath detection algorithm using Boolean satisfiability (SAT) techniques has been presented. The SAT-based algorithm uses the deterministic structure of the PN spreading code to perform an intelligent search for the possible propagation delays. Simulation results showed that the proposed scheme was successful in providing correct delay estimates with high reliability over a multipath frequency-selective Rayleigh channel.

REFERENCES

1. M. Simon, J. Omura, R. Scholtz, and B. Levitt, *Spread Spectrum Communications*, McGraw-Hill, New York, NY, USA, 1994.
2. O. S. Shin and K. B. Lee, "Utilization of multipaths for spread-spectrum code acquisition in frequency-selective Rayleigh fading channels," *IEEE Transactions on Communications*, vol. 49, no. 4, pp. 734–743, 2001.
3. W. Suwansantisuk and Z. Win, "Multipath aided rapid acquisition: optimal search strategies," *IEEE Transactions on Information Theory*, vol. 53, no. 1, pp. 174–193, 2007.
4. W. Suwansantisuk, M. Z. Win, and L. A. Shepp, "On the performance of wide-bandwidth signal acquisition in dense multipath channels," *IEEE Transactions on Vehicular Technology*, vol. 54, no. 5, pp. 1584–1594, 2005.
5. L. L. Yang and L. Hanzo, "Serial acquisition performance of single-carrier and multicarrier DS-CDMA over nakagami-m fading channels," *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 692–702, 2002.
6. A. Biere, A. Cimatti, E. M. Clarke, M. Fujita, and Y. Zhu, "Symbolic model checking using SAT procedures instead of BDDs," in *Proceedings of the 36th Annual Design Automation Conference (DAC '99)*, pp. 317–320, June 1999.
7. G. J. Nam, F. Aloul, K. Sakallah, and R. Rutenbar, "A comparative study of two boolean formulations of FPGA detailed routing constraints," in *Proceedings of the International Symposium on Physical Design (ISPD '01)*, pp. 222–227, April 2001.
8. F. Aloul, S. Hassoun, K. Sakallah, and D. Blaauw, "Robust SAT-based search algorithm for leakage power reduction," in *Proceedings of the International Workshop on Power and Timing Modeling, Optimization, and Simulation (PATMOS '02)*, pp. 167–177, 2002.
9. A. Sagahyroon and F. A. Aloul, "Using SAT-based techniques in power estimation," *Microelectronics Journal*, vol. 38, no. 6-7, pp. 706–715, 2007.
10. F. A. Aloul and N. Kandasamy, "Sensor deployment for failure diagnosis in networked aerial robots: a satisfiability-based approach," in *Proceedings of the 10th International Conference on Theory and*

Applications of Satisfiability Testing (SAT '07), vol. 4501 of *Lecture Notes in Computer Science*, pp. 369–376, Springer, 2007.

11. M. Mneimneh, F. Aloul, C. Weaver, S. Chatterjee, K. Sakallah, and T. Austin, "Scalable hybrid verification of complex microprocessors," in *Proceedings of the 38th Design Automation Conference (DAC '01)*, pp. 41–46, June 2001.
12. N. Creignou, S. Kanna, and M. Sudan, *Complexity Classifications of Boolean Constraint Satisfaction Problems*, SIAM, 2001.
13. S. A. Cook, "The complexity of theorem proving procedures," in *Proceedings of the Symposium on the Theory of Computing*, pp. 151–158, 2004.
14. F. A. Aloul, A. Ramani, I. L. Markov, and K. A. Sakallah, "Generic ILP versus specialized 0-1 ILP: an update," in *Proceedings of the IEEE/ACM International Conference on Computer Aided Design (ICCAD '02)*, pp. 450–457, November 2002.
15. D. Chai and A. Kuehlmann, "A fast pseudo-boolean constraint solver," in *Proceedings of the 40th Design Automation Conference (DAC '03)*, pp. 830–835, June 2003.
16. N. Een and N. Sorensson, "An extensible SAT-solver," in *Proceedings of the International Conference on Theory and Applications of Satisfiability Testing (SAT '03)*, pp. 502–508, 2003.
17. K. Moskewicz, C. F. Madigan, Y. Zhao, L. Zhang, and S. Malik, "Chaff: engineering an efficient SAT solver," in *Proceedings of the 38th Design Automation Conference (DAC '01)*, pp. 530–535, June 2001.
18. J. P. Marques-Silva and K. A. Sakallah, "GRASP: a search algorithm for propositional satisfiability," *IEEE Transactions on Computers*, vol. 48, no. 5, pp. 506–521, 1999.
19. H. M. Sheini and K. A. Sakallah, "Pueblo: a modern Pseudo-Boolean SAT solver," in *Proceedings of the Design, Automation and Test in Europe Conference (DATE '05)*, pp. 684–685, March 2005.
20. E. Goldberg and Y. Novikov, "BerkMin: a fast and Robust SAT-solver," in *Proceedings of the Design Automation and Test in Europe Conference (DATE '02)*, pp. 142–149, 2002.
21. K. Pipatsrisawat and A. Darwiche, "A new clause learning scheme for efficient unsatisfiability proofs," in *Proceedings of the Conference on Artificial Intelligence*, pp. 1481–1484, July 2008.

22. H. E. Dixon and M. L. Ginsberg, "Inference methods for a pseudo-Boolean satisfiability solver," in *Proceedings of the 18th National Conference on Artificial Intelligence (AAAI '02)*, pp. 635–640, August 2002.
23. W. Suwansantisuk, M. Chiani, and M. Z. Win, "Frame synchronization for variable-length packets," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 1, pp. 52–69, 2008.
24. J. Hayes, *Introduction to Digital Logic Design*, Addison-Wesley, 1993.
25. M. Davis, G. Longman, and D. Loveland, "A machine program for theorem proving," *Journal of the ACM*, vol. 5, no. 7, pp. 394–397, 1962.
26. M. Abramovici and D. Saab, "Satisfiability on reconfigurable hardware," in *Proceedings of the International Workshop on Field Programmable Logic and Application (FPL '97)*, pp. 448–456, 1997.
27. A. Dandalis and V. K. Prasanna, "Run-time performance optimization of an FPGA-based deduction engine for SAT solvers," *ACM Transactions on Design Automation of Electronic Systems*, vol. 7, no. 4, pp. 547–562, 2002.
28. I. Skliarova and A. B. Ferrari, "A software/reconfigurable hardware SAT solver," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 12, no. 4, pp. 408–419, 2004.
29. P. Zhong, M. Martonosi, P. Ashar, and S. Malik, "Using configurable computing to accelerate Boolean satisfiability," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 18, no. 6, pp. 861–868, 1999.

INDEX

The Domain Theory in Computer Science

Domain theory is a branch of mathematics that studies special types of partially ordered sets, commonly known as domains. Consequently, domain theory can be considered as a branch of order theory. The field has major applications in computing, where it is used to determine denotational semantics, especially for functional programming languages. The domain theory very generally formalizes intuitive ideas about convergence, and convergence is closely related to topology. The domain theory as a field of mathematics deals with information and computation. It treats the idea of states that contain partial information, which are ordered by the quantity of information they contain. As a result of that, the developed mathematical theory has a large number of applications in many computer science topics, particularly in the semantics of programming languages. In this book, we will cover both the mathematical theory and the applications of domain theory. The main themes include: the topics of approximation and continuity defined in the domain theory and their important connections with topology; the bases for computation with infinite objects; the development of a rich theory of fix-points, as a foundation for recursive definitions; the development of a rich set of data type constructions, and recursive definitions of domains themselves; and power domains, to support ideas of non-deterministic and probabilistic computation.

This edition covers different topics from domain theory in computer science, including: partial orders and groups, power domains and metrics, recursive data types (binary trees) and algebraicity and Boolean algebras.

Section 1 focuses on partial orders and groups, describing characterizations and properties of a new partial order, natural partial orders on transformation semigroups with fixed sets, cyclic soft groups and their applications on groups, factorization of groups involving symmetric and alternating groups.

Section 2 focuses on power domains and metrics, describing FS+ domains, topology of GB-metric spaces, incoherency problems in a combination of description logics and rules, metrics for multiset-theoretic subgraphs.

Section 3 focuses on recursive data types (binary trees), describing binary tree's recursion traversal algorithm and its improvement, the design of the minimum spanning tree algorithms, generating tree-lists by fusing individual tree detection and nearest neighbor imputation using airborne LIDAR data, a recursive approach to the Kauffman bracket, a novel multiway splits decision tree for multiple types of data.

Section 4 focuses on algebraicity and Boolean algebras, describing the deformation theory of structure constants for associative algebras, the Boolean algebra and central Galois algebras, on addition of sets in Boolean space, algebra and geometry of sets in Boolean space, multipath detection using Boolean satisfiability techniques.

Jovan currently works as a presales Technology Consultant at Dell Technologies. He is a result-oriented technology leader with demonstrated subject matter expertise in planning, architecting and managing ICT solutions to reflect business objectives and achieve operational excellence. Jovan has broad deep technical knowledge in the fields of data center and big data technologies, combined with consultative selling approach and exceptional client-facing presentation skills. Before joining Dell Technologies in 2017, Jovan spent nearly a decade as a researcher, university professor and IT business consultant. In these capacities, he served as a trusted advisor to a multitude of customers in financial services, health care, retail, and academic sectors. He holds a PhD in Computer Science from RMIT University in Australia and worked as a post-doctoral visiting scientist at the renowned INRIA research institute in France. He is a proud father of two, an aspiring tennis player, and an avid Science Fiction/Fantasy book reader.

AP | ARCLER PRESS

ISBN 978-1-77469-677-4