ADVANCES IN INTELLIGENT SYSTEMS AND COMPUTING 226

Robert Burduk
Konrad Jackowski
Marek Kurzyński
Michał Woźniak
Andrzej Żołnierek (Eds.)

Proceedings of the 8th International Conference on Computer Recognition Systems CORES 2013

Springer

Advances in Intelligent Systems and Computing

Volume 226

Series Editor

J. Kacprzyk, Warsaw, Poland

For further volumes:
http://www.springer.com/series/11156

Robert Burduk · Konrad Jackowski
Marek Kurzyński · Michał Woźniak
Andrzej Żołnierek
Editors

Proceedings of the 8th International Conference on Computer Recognition Systems CORES 2013

Springer

Editors
Robert Burduk
Department of Systems and
Computer Networks
Wroclaw University of Technology
Wroclaw
Poland

Konrad Jackowski
Department of Systems and
Computer Networks
Wroclaw University of Technology
Wroclaw
Poland

Marek Kurzyński
Department of Systems and
Computer Networks
Wroclaw University of Technology
Wroclaw
Poland

Michał Woźniak
Department of Systems and
Computer Networks
Wroclaw University of Technology
Wroclaw
Poland

Andrzej Żołnierek
Department of Systems and
Computer Networks
Wroclaw University of Technology
Wroclaw
Poland

ISSN 2194-5357 ISSN 2194-5365 (electronic)
ISBN 978-3-319-00968-1 ISBN 978-3-319-00969-8 (eBook)
DOI 10.1007/978-3-319-00969-8
Springer Cham Heidelberg New York Dordrecht London

Library of Congress Control Number: 2013939970

© Springer International Publishing Switzerland 2013
This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed. Exempted from this legal reservation are brief excerpts in connection with reviews or scholarly analysis or material supplied specifically for the purpose of being entered and executed on a computer system, for exclusive use by the purchaser of the work. Duplication of this publication or parts thereof is permitted only under the provisions of the Copyright Law of the Publisher's location, in its current version, and permission for use must always be obtained from Springer. Permissions for use may be obtained through RightsLink at the Copyright Clearance Center. Violations are liable to prosecution under the respective Copyright Law.
The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.
While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

The goal of the CORES series of conferences is the development of theories, algorithms, and applications of pattern recognition methods. These conferences have always served as very useful forum where researchers, practitioners, and students working in different areas of pattern recognition can meet to come together and help each other keeping up with this active field of research. This book is collection of 87 carefully selected works which have been carefully reviewed by the experts from the domain and accepted for presentation during the 8^{th} International Conference on Computer Recognition Systems CORES 2013. We hope that the book can become the valuable source of information on contemporary research trends and most popular areas of application.

The papers are grouped into eight sections on the basis of the main topics they dealt with:

1. Features, learning, and classifiers consists of the works concerning new classification and machine learning methods;
2. Biometrics presents innovative theories, methodologies, and applications in the biometry;
3. Data Stream Classification and Big Data Analytics section concentrates on both data stream classification and massive data analytics issues;
4. Image processing and computer vision is devoted to the problems of image processing and analysis;
5. Medical applications presents chosen applications of intelligent methods into medical decision support software.
6. Miscellaneous applications describes several applications of the computer pattern recognition systems in the real decision problems.
7. Pattern recognition and image processing in robotics which presents pattern recognition and image processing algorithms aimed specifically at applications in robotics
8. Speech and word recognition, which consists of papers focused on speech recognition, automatic text processing and analysis.

Editors would like to express their deep thanks to authors for their valuable submissions and all reviewers for their hard work. Especially we would like to thank Prof. Piotr Porwik and his team from University of Silesia who organized special session devoted to Biometrics, Prof. Jerzy Stefanowski from Poznan University of Technology who helped us to organize special session on Data Stream Classification and Big Data Analytics, Prof. Andrzej Kasiński, Prof. Piotr Skrzypczyński and their team from Poznan University of Technology for their effort to organize special session on Pattern recognition and image processing in robotics. We believe that this book could be a reference tool for scientists who deal with the problems of designing computer pattern recognition systems.

CORES 2013 enjoyed outstanding keynote speeches by distinguished guest speakers:

Prof. Wodzisław Duch - Nicolaus Copernicus University,
Prof. Janusz Kacprzyk - Systems Research Institute, Polish Academy of Sciences,
Prof. Juliusz Lech Kulikowski - M. Nalecz Institute of Biocybernetics and Biomedical Engineering PAS,
Prof. Dariusz Plewczyński - University of Warsaw, Interdisciplinary Centre for Mathematical and Computational Modelling.
Although the last, not least we would like to give special thanks to local organizing team (Robert Burduk, Piotr Cal, Kondrad Jackowski, Bartosz Krawczyk, Maciej Krysmann, Bartosz Kurlej, Piotr Sobolewski, Marcin Zmyślony, Andrzej Żołnierek) who did a great job.

We would like to fully acknowledge support from the Wrocław University of Technology, especially from Dean of Faculty of Electronics, Chairs of Department of Systems and Computer Networks, and The Polish Association for Image Processing which has also supported this event.

Wroclaw,
May 2013

Robert Burduk
Konrad Jackowski
Marek Kurzyński
Michał Woźniak
Andrzej Żołnierek

Contents

Part I
Features, Learning, and Classifiers

Toward Computer-Aided Interpretation of Situations

Juliusz L. Kulikowski

Abstract. The problem of interpretation of situations as a widely extended and important component of living beings' behavior in real world is considered. A scheme of interpretation of situations in natural living beings is presented and a general scheme of inspired by the nature artificial situations interpreting system is proposed. Basic constraints imposed on computer-based situations interpreting systems are described. It is shown that the computer-based situations interpreting systems are an extension of pattern recognition systems and the differences between them are characterized. The role of domain ontologies and of ontological models in computer-based situations interpreting systems design is shown and it is illustrated by examples.

1 Introduction

70 years since the publication by W.S. Mc Culloch and W.H. Pitts of their concept of a *mathematical model of neuron* in 2013 will be passed. Their significant publication initiated an increasing interest of scientists and of engineers to construct mechanisms simulating functional abilities of a natural nervous system. A next important step in this direction was made by F. Rosenblatt in the late 50ths of the past century by publication of his concept of *perceptron* - a first *artificial neural network* recognizing simple graphical patterns. This was a starting point of *pattern recognition* as a new scientific discipline which for the following years resulted in numerous new concepts, original computer programs and many thousands of considerable papers. Since that time, pattern recognition evolves in two competing streams of works: the first one based on artificial neural networks (ANN)

Juliusz L. Kulikowski
Nalecz Institute of Biocybernetics and Biomedical Engineering, Polish Academy of Sciences, 4 ks. Trojdena Str., 02-109 Warsaw, Poland
e-mail: juliusz.kulikowski@ibib.waw.pl

R. Burduk et al. (Eds.): *CORES 2013*, AISC 226, pp. 3–18.
DOI: 10.1007/978-3-319-00969-8_1 © Springer International Publishing Switzerland 2013

and the second one using various algorithmic (statistical, functional, algebraic, formal linguistic, etc.) approaches to classification and recognition of patterns. In fact, the first approach, which originally tried to construct pattern recognition systems in hardware, finally realize them in the form of computer programs simulating artificial neural networks. Therefore, both approaches in some sense are "algorithmic", the main difference among them consisting in the sources inspiring with the concepts of algorithms: modeling of neural networks or modeling of formally defined classes of objects' similarity. Against original expectations, artificial neural networks rather few contributed to our understanding of the natural thinking mechanisms. This is rather the progress of neuropsychology and neurophysiology which for the last decades provided new concepts which may be useful in improving and extending the intelligent computer-aided data processing methods and algorithms. In particular, a lot of highly effective pattern recognition methods and computer programs in application to detection and recognition of simple printed or hand-written characters, voice signals, defects of machine details, diseases symptoms, biological cells, finger-print minutiae, etc. became available and are presently widely used. On the other hand, the problems of computer-aided image understanding on advanced semantic level, interpretation of observable situations connected with their pragmatic evaluation, early natural disasters prognoses, effective recognition of admissible ways to reach a goal in unclear circumstances, etc. still remain to be solved, despite the fact that our natural mind less or more effectively solves them everyday. Let us remark that the above-mentioned problems go beyond the limits of the conventional pattern recognition ones, however, they still concern recognition in a more general sense: assigning of some widely defined "objects" to fuzzy similarity classes in connection with their semantic and/or pragmatic assessment. Moreover, till now, no effective results of image, scene or situation analysis by artificial neural networks in the above-mentioned, wide sense have been achieved. This is clear because architecture of no artificial neural network (see, e.g. [1], [2], [3]), sophisticated as might it be, reminds the real functional structure of a natural brain (see, for a comparison [4], [5]). This does not mean that the state of ANN will not be evolving; nevertheless, it seems that in the nearest future more effective methods of composite objects recognition and interpretation will be provided by the approaches based on formal models. This expectation follows from the fact that a lot of computer-aided decision-making systems already exist; however, the role of computers in this type of systems is usually reduced to acquisition, primary processing, partial analysis, presentation and/or storage of data for final decision making by a human user. The user is faced with a deontological problem [6]:

If it is known that a situation A arose then undertake an action B in order to reach a more desirable situation C.

In this context, our problem consists in designing algorithms stating that a situation A really arose; this is a general objective of the *situation interpretation* theory and practice. At a first glance, it might seem that *pattern recognition* is a discipline solving the above-mentioned problem satisfactorily. E.g., if in a given histological

specimen cancer cells have been recognized then usually it follows from this that some therapeutic actions should be undertaken in order to cure the patient. However, the cancer cells are in this case no more but a component of a more general situation: they have been detected in an inner organ of a patient of a given gender, age, etc. whose less or more extended and accurate case record is known. All this information constitutes a situation a medical doctor has to deal with. Therefore, *pattern recognition* can be considered as a particular case of *situation interpretation*. Nevertheless, the notions of *situation* and *interpretation* remain to be more strictly defined. In this paper an attempt to the formulation of *situation interpretation* (SI) backgrounds as well as some suggestions concerning the computer-aided SI systems design are presented. We try to show that basic notions of SI can be founded on well known mathematical concepts on which ontological models are based while practical solution of SI can be inspired by the natural cognitive processes created by the evolutionary processes.

2 Basic Notions

2.1 Natural Interpretation of Situations

There are two basic approaches to definition of the notion of *situation*. In a widely used sense **situation is a passing state of a selected fragment of reality**. Reality can thus by a subject be considered as a stream of surrounding her/him or it, arising, evolving, affecting some other ones or being by some other ones affected and finally disappearing situations. Any living being deals with various situations as an observer, their active subject, object, commentator, etc. However, situations are not the components the reality consists of; they rather are the components of perception of the reality by living beings. The same objects or fragments of real processes can by different subjects be considered as elements of different situations. This follows from a general, in fact - subjective way a living being learns to perceive and to discriminate the situations. E.g., the same state considered by a subject in the context of some preceding and some following states can be assessed as "desired" or "undesired" and as such it will be differently interpreted. The way of situations recognition by simple living beings is illustrated in Fig. 1.

They reach their ability to perceive situations by perception of various temporary signals coming from the real world, primary assessment of their value as "more" or "less" agreeable, remembering, collecting according to some similarity features, discriminating their components, associating according to their time- and/or space-co-occurrence and assigning to them specific reactions in the form of signals or behaviors expressing subjects' emotions caused by the situations. According to our present knowledge about natural brain functions, most of the above-mentioned operations using memory takes place in an anatomic part of the brain, belonging to its temporal lobe, called a hippocampus. Emotional values to the impressions are assigned in adjacent to it amygdale body. The emotionally marked reactions to many

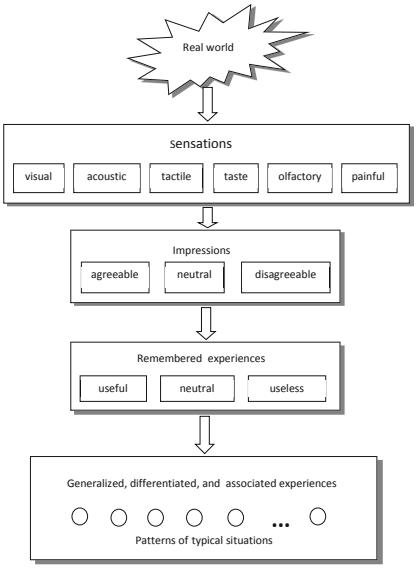


Fig. 1 A natural way to recognize situations in simple living beings

times in similar situations repeated impressions become established as germs of a code used to communicate about the situations by the members of a social group of living subjects. Therefore, recognition of situations even in simple living beings is closely connected with assessment of their importance for the recognizing subjects. Otherwise speaking, it contains some aspects of their "interpretation" even if not expressed in terms of any advanced language.

In more developed living beings the natural way of situations interpretation is in general similar, however, some differences should also be remarked:

- Instead of simple, sensations some formerly established sub-situations can be used as a basis to construct the concepts of higher-level situations;
- The scale of situations' practical importance assessment becomes more sophisticated, also in a multi-aspect sense;
- Linguistic terms are assigned to typical situations;
- Elements of classification of the situations may be used;
- Situations are assessed also from a point of view of their relations to other: past, existing, expected or possible situations.

In a large sense, **interpretation of situation** means a natural cognitive process consisting of:

- acquisition of observations;
- primary situation recognition;

- situation's position localization in a more general system of concepts;
- establishing of relations between the given and other (past, present, future possible, etc.) situations.

As a result, this leads to an ability to avoid emergency, to find food, to contact other social partners or to make another decision of vital importance.

A scheme of advanced natural situations interpretation is shown in Fig. 2. In fact, the scheme is rather hypothetical, because till now it is rather few known about the natural processes and mechanisms of situations interpretation in primates. However, a progress in modern brain imaging modalities (e.g. functional magnetic resonance, f-MRI [6], diffusion tensor imaging, DTI [7]) opens new possibilities in this research area.

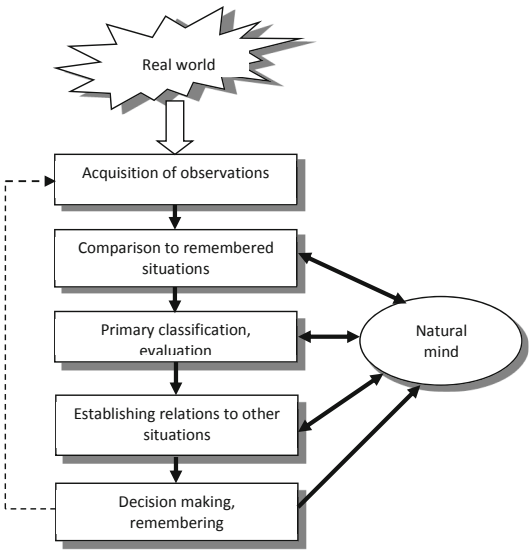


Fig. 2 Advanced natural interpretation of situations

2.2 Formal Model of Situation Interpretation

A general architecture of a computer-based situations interpretation (SI) systems can be inspired by the scheme shown in Fig. 2, excepting that the "natural mind" should be replaced with computer-based knowledge base and logical engines. However, a strong simulation of the natural way of learning the recognition and interpretation of situations for certain reasons seems to be unrealistic. First, because of a long, hundreds thousands years counting duration-time of the natural evolutionary processes. Second, because our knowledge about the natural mechanisms of everyday experiences acquisition, association, generalization, evaluation, storage and reminding is

still now incomplete. Therefore, the computer-based SI system cannot be modeled on the natural one; they rather should be based on an abstract idea of ontology which in the last years in information processing systems design became popular [8], [9].

A general ontology is usually defined as a part of metaphysics concerning the nature and theory of existence. In computer applications an idea of domain ontologies defined as "a common understanding of some domain" [8] or as "an abstract view of the world we are modeling, describing the concepts and their relationships" [9] became more useful. Any domain ontology defines a hierarchy of its concepts representing objects, attributes, states, actions etc. used to the description of the given part of reality. In [10] a domain ontology decomposition into an ensemble of ontological models was proposed:

$$O = \{OM_1, OM_2, \ldots, OM_I; Q, A\} \tag{1}$$

where any ontological model is defined as a quadruple:

$$OM_i = \{C_i, R_i, T_i, A_i\},\ i = 1, 2, \ldots, I. \tag{2}$$

Above, C_i denotes a non-empty family of concepts (objects, attributes, actions, etc.) the OM is based on; R_i is a family of relations described on selected subsets of C_i. Among the relations in R_i a multi-aspect taxonomy Ξ_i of the concepts is distinguished, a concept $T_i \in C_i$ being its top-level element, the highest concept of the OM_i.

Basic formal properties of the ontology as well as of its ontological models follow from the corresponding sets of axioms A and A_i-s. In particular, the axioms of the algebra of sets, of relations and of their extensions theory, of fuzzy sets, of probability theory, etc. if used as bases of the given OMs' definitions should be considered as components of their formal description.

A general structure of the ontology is described by a super-relation Q between the OMs. In particular, clustering of OMs into categories corresponding to various aspects of the domain reality suits to putting an ontology in a clearer order. Despite the fact that in practice, the axioms in the description of the ontological models are very often neglected, they should be strongly respected if an ontology is to be correctly designed.

Sequences of elements of C_i satisfying a given relation $\rho \in R_i$ are called *syndromes* of ρ . Similarity relations describe particular types of concepts called *patterns*. OMs consisting of similarity relations and of some based on them higher-order relations describe a particular type of syndromes called *scenes*. Otherwise speaking, a scene is a composition of objects representing some patterns. As such, it can be described in terms of concepts of a domain ontology.

Sets of syndromes commonly satisfying the relations of a given OM, more generally, will be called *situations* of the model. Observed scenes are thus

particular cases of observed situations; recognition of scenes or of patterns is a particular case of **situation recognition**. However, **interpretation of situation** consists of:

- situation recognition,
- its pragmatic aspects assessment,
- its logical extension.

The last term denotes detection within the given domain ontology other *OMs* whose syndromes may be logically consistent with the analyzed situation. Logical extension makes thus possible answering questions not only directly concerning the analyzed situation but also some questions concerning its possible connections with other situations.

Not all domain ontologies and ontological models can be used to situations interpretation. For this purpose, they should contain a *mechanism V* of the situations' *pragmatic aspects assessment*. It should provide a possibility to compare any syndromes from the point of view of their significance, utility, emergency level, etc. for the observer. In the simplest case it can take the form of a binary scale (e.g., "*important*", "*not important*"). Therefore, an *OM* admitting pragmatic evaluation of situations takes the form of a quintuple:

$$EOM_j = [C_j, R_j, T_j, V_j, A_j],\ j \in [1, 2, \ldots, I], \tag{3}$$

(C_j, R_j, T_j, A_j denoting the same objects as in (2)). This type of *OMs* can be called *evaluated ontological models*.

The above-presented elements of formalism lead to a hierarchy of real world representing models suitable to be used in a computer-based *SI* system, consisting of:

- *a general ontology*;
- *domain ontologies*;
- a *hierarchy of ontological models* describing the inner structure of the domain ontology as a composition of ontological models;
- *ontological models* (OM, EOM) describing selected aspects or states that in a given domain may occur and can be observed;
- *relations, super-relations or hyper-relations* of any type specifying the ontological models;
- *situations* described by the ontological models;
- *formal patterns* specifying the *similarity classes, structures* satisfying *ordering relations* and *syndromes* satisfying other types of relations, super-relations and/or hyper-relations;
- *sets of objects, values of attributes, alphabets* etc. on which the patterns, formal structures, symbolic expressions, etc. are defined.

The multi-level structure of models and their components is shown in Fig. 3.

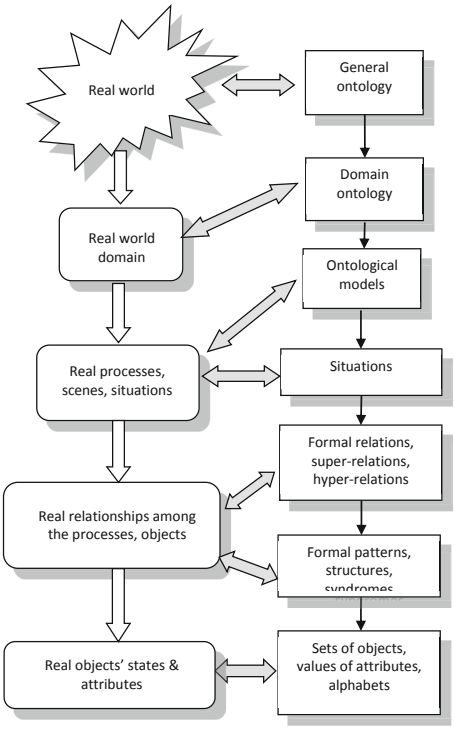


Fig. 3 Relationships between real world and its formal models

3 Evaluated Ontological Models

3.1 General Remarks and Constraints

From a practical point of view, *EOMs* are a form of acceptable by computers description of the situations that in a given application domain may arise. A repertoire of formal tools suitable to construction of *EOMs* is thus limited to those expressible by computer languages or data structures, Some sophisticated mathematical notions like: a *family of all subsets of a continuous set* or *a set of all transcendental numbers between 0 and 1*, etc. cannot in computer systems exactly be represented. Therefore, *OMs* or *EOMs* on this type of incalculable notions cannot be based. As a consequence, not all real situations can by ontological models be described.

More generally, it seems impossible to recognize and to interpret situations in the case of:

- real situations not expressible in correctly defined and calculable formal terms;
- inadequacy of the concepts of a domain ontology to real input (observation) data;

- lack of effective algorithms of processing on formal terms the ontological models are based on;
- lack of adequately chosen methods of situations similarity and pragmatic aspects assessment.

As examples of real situations that for the above-listed reasons (till now) cannot by a computer-based *SI* system be analyzed the following situations can be mentioned:

- connected with subjective psychical impressions or consciously non-controlled emotions (e.g. impression of fear evoked by a thriller emitted in tv, people's excitement caused by a street accident, etc.);
- described by logically incompact datasets (e.g. a total welfare level which in different countries is evaluated by different methods);
- concerning real life domains not described by strongly defined or measurable terms (e.g. ambience at a board meeting);

etc. Despite the above-mentioned limitations, an area of real *SI* problems possible to be solved by using available mathematical tools remains still large.

Design of an *EOM* for a given *SI* purpose can be facilitated by a preliminary answering the following questions:

- Concerning the domain:
  - What is the name of the area of interest;
  - Does it exist a domain ontology for it? (if not, start to construct it);
- Concerning a general structure of the domain:
  - What is the name of the specific sub-area of interest;
  - Does it exist a sub-ontology for it?
- Concerning the ontological model:
  - What is the name of the situation of interest within the ontology or sub-ontology;
  - Does it exist an ontological model for it?
- Concerning an "anatomy" of the situational ontological model:
  - What are the active and/or passive objects participating in the situation?
  - What are the attributes of the objects?
  - What are the types of substantial relations among the objects?
  - Are there any typical examples of instances of the relations?
  - What are the criteria for other objects' approval as instances of the relations?
  - What features of the relations' instances should be taken into account for their pragmatic value assessment?
  - What are the relations whose instances' time-evolution for pragmatic value assessment should be taken into consideration?
- Concerning the super-relations within the domain:

- Are there any other types of situations that in some way may be associated to the given one (e.g. coexisting in time, influencing one the other one, etc.)? Name them.
- Are the other types of situations described by ontological models?
- Is it formally possible to define a super-relation on the basis of the associated relations?
- What pragmatic values can be assigned to the instances of the super-relation as a function of pragmatic values of its components?

3.2 Categorization

Besides the above-mentioned super-relations described on some combinations of relations, the last can be categorized according to some criteria following from a general domain structure. E.g., within a domain *Health* service the following categories of *OMs* can be specified:

- topographical: *Warsaw City, South region, Country,* etc.;
- functional: *Diagnosis, Treatment, Education, Prevention,* etc.;
- medical: *Internal diseases, Psychiatry, Oncology*, etc.;
- organizational: *Hospitals, Outpatients clinics, Ambulance stations*, etc.

Evidently, the taxonomy of OMs may be horizontally or vertically-down extended, e.g.:

Warsaw Center Home visit rounds, Influenza, Private outpatient clinics.

This may associate by a super-relation an OM satisfying the above-given criteria: modeling all *Private outpatient clinics* in *Warsaw City* offering *home visits* in the case of *influenza*.

Categorization of *OMs* makes also possible recognition of "higher level" situations like: "*Abnormal number of patients registered on Monday in all public health service centers of Warsaw City with the symptoms of influenza*" and assigning to them pragmatic values like: *important* or *alarming*. This is possible if in the *OM* among the relations describing all types of *Health service centers* there are the ones representing an ontological concept *State of services* described by the following lower-level concepts (objects, attributes):

(Health service center), Home visit ♯, Date, Case (medical diagnosis)

On the basis of all relations of this type in all *OMs* describing *Health service centers* in *Warsaw City* it can be constructed a super-relation defined as an extended algebraic sum [12] of sub-relations obtained by fixing the data:

Date := Monday, Case := influenza.

Such a super-relation will be satisfied by all syndromes of the form:

$*, *$, *Monday, influenza*

where the unknown values of *Health service center* and *Home visit* ♯ have been denoted by $*$).

3.3 Pragmatic Value Assessment

From a formal point of view, a mechanism of situations' pragmatic value evaluation is a sort of a *weak semi-ordering relation V* imposed on the situations of a given *OM*. Before it to be defined, let us remind the notion of a *strong similarity* as a reflexive, symmetrical and transitive relation $\approx$ [13]. Then, if it is given a set of situations admitted by the *OM* and ω', ω'', ω''' are some particular situations then *V* should satisfy the following conditions:

- *V* is satisfied by any pair $[\omega', \omega']$ (reflexivity);
- If *V* is satisfied by an $[\omega', \omega'']$ then it is also satisfied by $[\omega'', \omega']$ if and only if $\omega' \approx \omega''$ holds (weak asymmetry);
- If *V* is satisfied by an $[\omega', \omega'']$ and by $[\omega'', \omega''']$ then it is also satisfied by $[\omega', \omega''']$ (transitivity).

We call *equivalence classes* the sets of objects mutually equivalent in the sense of the (mentioned in the definition) strong similarity ($\approx$) relation.

The mechanisms of pragmatic value evaluation into two groups can be divided:

- Single-aspect evaluation (*SAE*) mechanisms,
- Multi-aspect evaluation (*MAE*) mechanisms.

SAE mechanisms can directly be based on the above-given definition of weak semi-ordering relations. They may suit to evaluate such situations' properties as *importance, usefulness, level of interest, emergency level,* etc. However, they may take the forms of:

a. strong (linear) ordering;
b. semi-ordering;
c. partial semi-ordering

Strong linear ordering means that for any different situations ω', ω'' the relation *V* is satisfied either by $[\omega', \omega'']$ or by $[\omega'', \omega']$; however, the equivalence classes may contain no more but one element. In "standard" (denoted b)) and partial (denoted c))semi-ordering multi-element equivalence classes are admissible. However, partial semi-ordering admits also that for certain pairs of situations the relation *V* is neither

by [ω', ω''] nor by [ω'', ω'] satisfied. This is illustrated in Fig.4 a, b, c, where circles represent situations, arrows - increasing pragmatic values and multi-element equivalence classes are by dotted contours denoted.

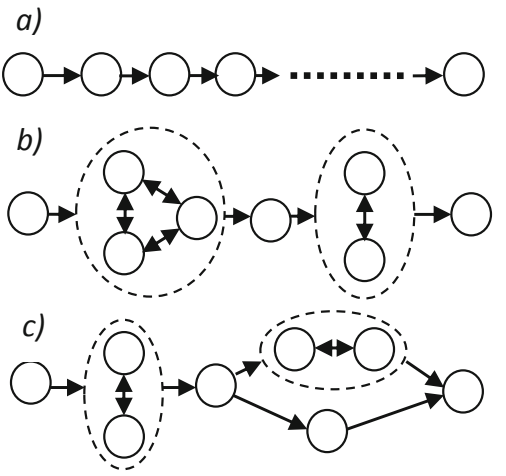


Fig. 4 Types of pragmatic values ordering: a) strong (linear), b) semi- ordering, c) partial semi-ordering

In the cases a) and b) numerical or nominal scales of pragmatic values can be used while in the case c) pragmatic values by their relative comparison only can be assessed.

MAE mechanisms on the basis of particular *SAE* mechanisms can be defined. One of the simplest ways to reach this is introducing a higher-level lexicographic order between the *SAEs*. For this purpose, let the set of particular *SAEs* be linearly ordered as $[SAE^{(1)}, SAE^{(2)}, \ldots, SAE^{(k)}]$. Then, for a given pair of situations $\{\omega', \omega''\}$ to be compared their order is first established in $SAE^{(1)}$. If they belong to different equivalence classes then the corresponding order is to them assigned and the procedure is finished. Otherwise, if they belong to the same equivalence class, their order is established according to the $SAE^{(2)}$ rule, etc., up to reaching $SAE^{(k)}$.

4 Interpretation of Situations

4.1 Recognition of Situation

Let ***u*** denote a sequence (ordered set) of results of observations in a certain application area. The elements of ***u*** may be of **various physical and/or formal nature**.

Recognition of a represented by $\boldsymbol{u}$ situation consists in **finding in a corresponding domain ontology a model** OM_i whose relations are by $\boldsymbol{u}$ (or by its subsets) fulfilled. Apparently, it reminds a typical pattern recognition problem in which the role of *OMs* is played by some predefined patterns [14]. However, the differences between situations and pattern recognition are substantial:

- the universe $\boldsymbol{U}$ of all possible observations usually does not constitute any homogenous, well-defined mathematical space;
- the observations $\boldsymbol{u}$, $\boldsymbol{u}\in\boldsymbol{U}$, should fulfill several relations of various nature;
- relations can be characterized analytically, by examples or by logical tests [15];
- the number of relations in different *OMs* may be different;
- situation recognition is a multi-step decision process controlled by pragmatic assessment of the results reached at the preceding steps.

Recognition of a situation needs a preliminary analysis of $\boldsymbol{U}$ from a point of view of its **formal and semantic consistency** with the concepts defined in the domain ontology. The mentioned in Sec. 3 basic constraints imposed on *OMs* should also be obeyed. Moreover, in the case of *OMs* based on random or fuzzy relations the fact that they are by $\boldsymbol{u}$ fulfilled can be expressed not only in a binary ("yes", "no") logical scale but sometimes also in a continuous numerical scale of probability or of a membership measure. It thus arises a problem, **how to define a total measure of the fact that a given *OM* is by $\boldsymbol{u}$ fulfilled as a function of various particular measures of the corresponding *OMs* relations being by $\boldsymbol{u}$ fulfilled**.

The problem can be solved by imposing some constraints on the form of a fulfilling measure $\varphi_\rho(\boldsymbol{u})$ of a relation being fulfilled by the syndrome $\boldsymbol{u}$. It will be assumed that in general $\varphi_\rho(\boldsymbol{u})$ is a function satisfying the conditions:

1. $0 \le \varphi_\rho(\boldsymbol{u}) \le 1$
2. $\varphi_\rho(\boldsymbol{u}) = 0$ if $\boldsymbol{u}$ does not satisfy ρ (also, if ρ is an empty relation);
3. for any $\boldsymbol{u}$ if $\rho' \subseteq \rho''$ then $\varphi_{\rho'}(\boldsymbol{u}) \le \varphi_{\rho''}(\boldsymbol{u})$

The following types of measures can be taken into consideration:

1. based on discrete scales 0 - does not satisfy, $1,2,\ldots,k$ - certainly satisfies; $k=1$ (binary), $k=2$ (ternary), etc.;
2. based on a continuous scale $[0,1]$.

Then, if $\varphi_0(\boldsymbol{u})$ denotes a total measure and $\varphi_n(\boldsymbol{u})$, $n=1,2,\ldots,N$, the measures of particular relations being satisfied by $\boldsymbol{u}$, it can be assumed that:

$$\varphi_0(\boldsymbol{u}) = \varphi_1(\boldsymbol{u}) \cdot \varphi_2(\boldsymbol{u}) \cdot \ldots \cdot \varphi_N(\boldsymbol{u}). \tag{4}$$

Example 1 It is given an observation:

Date / Hour / Name / Address / Age / Sex / Diagnosis / State
21.03 / 18.00 / N.N. / aaa / 67 / male / apoplexy / suspected

A doctor called to the urgent case needs a remote medical consultation of this case with a neurological specialist. For this purpose, he should introduce his observation data to a *Medical service OM* containing the relations:

Medical center, Center's address, Center's e-mail;
Center's departments, Center's departments' phones;
Center's department's specialists;
Center's department's duty schedule.

This makes possible to establish a relation:

Date / Hour / Diagnosis / State / Medical center / Center neurological department / Center neurological department's phone / Neurological specialist on duty.

Finally, this leads to recognition of a situation like:

21.03 / 18.00 / N.N. / aaa / 67 / male / apoplexy / suspected / Medical center XX / Neurological department / Phone number zzz / Neurological specialist dr YY /.

4.2 Interpretation of Situation

The above-presented example might suggest that *SI* is a simple one-step decision process. In fact, it is an iterative process consisting of several steps, as it has been mentioned in Sec. 2.2. This can be illustrated by a next example.

Example 2
Let us take into consideration a computer-aided street-traffic monitoring system. Its tv camera has caught a series of shot pictures illustrated schematically in Fig.5.

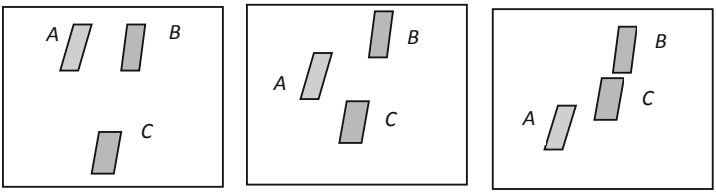


Fig. 5 Three consecutive shot pictures of a street-traffic film

Let us assume that three objects (denoted *A*, *B*, *C*) have been detected in the pictures. A preliminary comparison of the pictures leads to finding that:

A is approaching, B is steady, C is going away

The aim of the *SI* system is to detect and to record traffic accidents. The interpretation process may run as shown in Fig6.

In the above-described process a following qualitative linear scale of values has been used:

Not important → *Possibly important* → *Suspected* → *Important* → *Highly important*

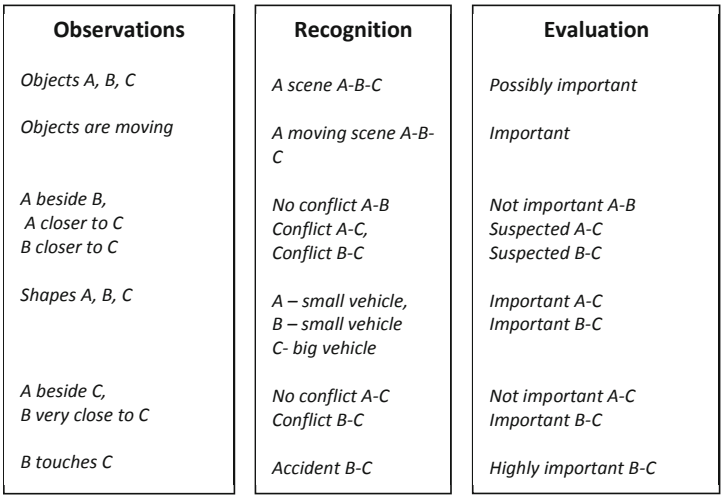


Fig. 6 Street traffic monitoring process

The *SI* process contains several pattern recognition acts: recognition of a *scene* (general), *moving scene*, *conflict*, *vehicle*, *accident*, etc. The mentioned here types of recognized objects as some ontological concepts are described by corresponding *OMs*. For example, the concept of *accident* can be defined by a hyper-relation H_Q described on a set Q consisting of at least two *objects* localized in a metric space and constructed as follows:

1. It is taken into account a Cartesian product $Q \times Q$;
2. It is defined a relation $r \subset Q \times Q$ as a set of all pairs of different elements of Q;
3. It is defined a table D_t of *metric distances* between the pairs of elements of r;
4. There are taken into consideration the tables D_{t1}, D_{t2}, D_{t3} for three consecutive time-instants $t_1 < t_2 < t_3$;
5. H_Q is fulfilled if for at least one pair of objects $q', q'' \in Q$ their metric distances satisfy the relation:

$$0 < dt_1(q', q'') > dt_2(q', q'') > dt_3(q', q'') = 0. \tag{5}$$

5 Conclusion

Interpretation of situations is a substantial component of living beings' behavior in real world. It consists of recognition of arising situations, their pragmatic assessment and establishing their relations to other situations. Similar type of decision making processes can be modeled in computer systems. Computer-based *SI* systems cannot

directly simulate the natural processes but they can be by them inspired. For this purpose domain ontologies and ontological models can be used. In the paper a general approach to the *SI* systems design has been proposed. There also have been indicated some basic constraints imposed on the artificial *SI* systems. In general, they are connected with incalculability of some formal (mathematical) concepts. Nevertheless, the ideas of extended algebra of relations and of super- and hyper-relations theory, including their fuzzy extensions, as well as suitable domain ontologies and ontological models are powerful tools for realization of effective computer-based *SI* systems for various important applications.

References

1. Hertz, J., Krogh, A., Palmer, R.G.: Introduction to the theory of neural computation. Addison-Wesley Publishing Co., Redwood City (1991)
2. Levy, B.C., Adams, M.B.: Global optimization with stochastic neural networks. In: IEEE Conf. on Neural Networks, San Diego, USA, pp. 681–689 (1987)
3. Hopfield, J.J.: Artificial neural networks. IEEE Circuits and Devices Magazine, 3–10 (1988)
4. Rilling, J.K., Insel, T.R.: The primate neocortex in comparative perspective using magnetic resonance imaging. Journal of Human Evolution 37, 191–223 (1999)
5. Allen, J.S.: The lives of the brain. The Harvard College (2009)
6. Rademacher, J., Galaburda, A.M., Kennedy, D.N., et al.: Human cerebral cortex: Localization, parcellation and morphometry with magnetic resonance imaging. Journal of Cognitive Neuroscience 4, 352–374 (1992)
7. Ramnani, N., Behrens, T.E.J., Johansen-Berg, H., et al.: The evolution of prefrontal inputs to the corticopontine system: Diffusion imaging evidence from macaque monkeys and humans. Cerebral Cortex 16, 811–818 (2006)
8. Fernandez-Lopez, M., Gomez-Perez, A.: Overview and analysis of methodologies for building ontologies. The Knowledge Eng. Rev. 17, 129–156 (2002)
9. Gruber, T.R.: A translation approach to portable ontologies. Knowledge Acquisition 5, 199–220 (1993)
10. Kulikowski, J.L.: The Role of ontological models in pattern recognition. In: Kurzynski, M., et al. (eds.) Computer Recognition Systems. Advances in Soft Computing, pp. 43–52. Springer, Berlin (2005)
11. Kulikowski, J.L.: Relational approach to structural analysis of images. Machine Graphics and Vision 1, 299–309 (1992)
12. Kulikowski, J.L.: Pattern recognition based on ambiguous indications of experts. In: Kurzynski, M. (ed.) Komputerowe Systemy Rozpoznawania, KOSYR 2001, Wroclaw, pp. 15–22 (2001)
13. Owsinski, J.W., Brüggemann, R. (eds.): Multicriteria ordering and ranking: partial orders, ambiguities and applied issues. System Research Institute, Warsaw (2008)
14. Duda, R., Hart, O., Stork, D.: Pattern classification. John Wiley, New York (2000)
15. Przytulska, M., Kulikowski, J.L., et al.: Computer methods of radiological images analysis for pathomorphological lesions assessment in selected inner organs. Report on the project N N518 4211 33 IBBE PAS Warsaw (2010)

Classification of Multi-dimensional Distributions Using Order Statistics Criteria

A. Thomas and B. John Oommen*

Abstract. This paper proposes a novel classification paradigm in which the properties of the *Order Statistics* (OS) have been used to perform an optimal/near-optimal solution for multi-dimensional problems. In our initial works in [5] and [6], we proposed the foundational theory of CMOS, Classification by the Moments of Order Statistics, for some uni-dimensional symmetric and asymmetric distributions of the exponential family. In this paper, we generalize those results for various multi-dimensional distributions. The strategy is analogous to a Naïve-Bayes' approach, although it, really, is of an *anti*-Naïve-Bayes' paradigm. We provide here the analytical and experimental results for the two-dimensional Uniform, Doubly-exponential and Gaussian and Rayleigh distributions, and also clearly specify the way by which one should extend the results for higher dimensions.

Keywords: Classification using Order Statistics (OS), Moments of OS.

1 Introduction

Pattern Recognition (PR) is the process by which unknown feature vectors are categorized into groups or classes based on their features [2]. The basic concept of traditional *parametric* PR is to model the classes based on the assumptions related to the underlying class *distributions*, and this has been achieved by performing a learning phase in which the moments of the respective classes are evaluated. However, there are some families of indicators (or distinguishing quantifiers) that have noticeably been *uninvestigated* in the Pattern Recognition (PR) literature. In particular, we refer to the use of phenomena that have utilized the properties of the *Order Statistics*

A. Thomas · B. John Oommen
School of Computer Science, Carleton University, Ottawa, Canada : K1S 5B6

* *Chancellor's Professor* ; *Fellow: IEEE* and *Fellow: IAPR*. This author is also an *Adjunct Professor* with the University of Agder in Grimstad, Norway. The work of this author was partially supported by NSERC, the Natural Sciences and Engineering Research Council of Canada.

R. Burduk et al. (Eds.): *CORES 2013*, AISC 226, pp. 19–29.
DOI: 10.1007/978-3-319-00969-8_2 © Springer International Publishing Switzerland 2013

(OS), which have not reported earlier. In [5] and [6], we proposed the paradigm named CMOS, i.e., Classification by the Moments of Order Statistics, in which the concept of OS was applied in a pioneering manner to the field of PR so as to achieve optimal (or near-optimal) accuracies for various classification problems. In this paper, we generalize those results for multi-dimensional distributions by proposing a strategy that is analogous to a Naïve-Bayes' approach, although it, really, is of an *anti*-Naïve-Bayes' paradigm. Using such a Naïve-Bayes' approach, we demonstrate how a CMOS classifier can be both designed and implemented. In order to prove our claims, we provide analytical and experimental results for the two-dimensional and multi-dimensional Uniform, Doubly-exponential, Gaussian and Rayleigh distributions, whence we show that the results are clearly conclusive.

Contributions of this Paper: The novel contributions of this paper are:

- We extend the OS-based PR for multi-dimensional distributions;
- We prove that the new approach attains the optimal bound for symmetric distributions, and a near-optimal accuracy for asymmetric distributions;
- To justify these claims, we also submit a formal analysis and the experimental results for a few distributions within the exponential family.

2 Relevant Background Areas Regarding OS

The entire theory of OS-based PR is based on the stochastic properties of the *distribution* of the OS of any random variable. Since the distribution of the OS is, in general, not easily computable, the literature rather concentrates on its moments – which, in actuality, is quite distinct from the moments of the random variable itself. The explicit form of these moments is crucial to our study and so it is presented here, in all brevity. Let $\mathbf{x}_1$, $\mathbf{x}_2$, ..., $\mathbf{x}_n$ be a univariate random sample of size n that follows a continuous distribution function Φ, where the probability density function (pdf) is $\varphi(\cdot)$. Let $\mathbf{x}_{1,n}$, $\mathbf{x}_{2,n}$, ..., $\mathbf{x}_{n,n}$ be the corresponding OS. The r^{th} OS, $\mathbf{x}_{r,n}$, of the set is the r^{th} smallest value among the given random variables. The pdf of $\mathbf{y} = \mathbf{x}_{r,n}$ is:

$$f_{\mathbf{y}}(y) = \frac{n!}{(r-1)!(n-r)!} \{\Phi(y)\}^{r-1} \{1-\Phi(y)\}^{n-r} \varphi(y),$$

where $r = 1, 2, ..., n$. The fundamental theorem concerning the OS that we invoke is found in many papers [3, 4, 8], and is summarized below.

Let $n \geq r \geq k+1 \geq 2$ be integers. Then, since Φ is a nondecreasing and right-continuous function from $\mathbb{R} \rightarrow \mathbb{R}$, $\Phi(\mathbf{x}_{r,n})$ is uniform in [0,1]. If we now take the k^{th} moment of $\Phi(\mathbf{x}_{r,n})$, it has the form [4]:

$$E[\Phi^k(\mathbf{x}_{r,n})] = \frac{B(r+k, n-r+1)}{B(r, n-r+1)} = \frac{n!\,(r+k-1)!}{(n+k)!\,(r-1)!}, \tag{1}$$

where $B(a,b)$ denotes the *Beta* function, and $B(a,b) = \frac{(a-1)!(b-1)!}{(a+b-1)!}$ since its parameters are integers.

The above fundamental result can also be used for characterization purposes as explained in [4]. If $n = 2$, implying that only *two* samples are drawn from $\mathbf{x}$, we can deduce from Eq. (1) that $E[\Phi^1(\mathbf{x}_{1,2})] = \frac{1}{3}, \Rightarrow E[\mathbf{x}_{1,2}] = \Phi^{-1}\left(\frac{1}{3}\right)$ and $E[\Phi^1(\mathbf{x}_{2,2})] = \frac{2}{3}, \Rightarrow E[\mathbf{x}_{2,2}] = \Phi^{-1}\left(\frac{2}{3}\right)$. Thus, from a computational perspective, the first moment of the first and second 2-order OS would be the values where the cumulative distribution Φ equal $\frac{1}{3}$ and $\frac{2}{3}$ respectively.

3 Uni-dimensional OS-Based Classification: The Generic Classifier

The multi-dimensional OS-based classifier is based on its uni-dimensional counterpart developed in [5] and [6]. Since the understanding of the latter is crucial to this paper, this is briefly explained here.

Consider a 2-class problem with classes ω_1 and ω_2, where their class-conditional densities are $f_1(x)$ and $f_2(x)$ respectively. If we perform a classification based on v_1 and v_2, the *medians* of the distributions, this is equivalent to the strategy in which the task is performed based on a *single* OS. For all symmetric distributions, this classification attains the Bayes' accuracy – which is not too astonishing because the median is identical to the mean. But the intriguing aspect emerges when we use higher order OS that are not located centrally, but rather *distant* from the means. Indeed, for uni-dimensional OS-based PR, our methodology is based on considering the n-order OSs, and comparing the testing sample with the $n - k^{th}$ OS of the first distribution and the k^{th} OS of the second. By considering the entire spectrum of the possible values of k, the results in [5] and [6] showed that the specific value of k is usually not so crucial. Further, if these symmetric pairs of the OS are used in PR, the classification based on *these* attains the optimal Bayes' bound for a large number of symmetric distributions of the exponential family [5] and a near-optimal bound for the asymmetric distributions [6].

Our task is to now generalize these results for multi-dimensional distributions. As mentioned earlier, the generalization of CMOS for multi-dimensional classification problems is not so trivial. We have opted to do this by invoking a Naïve-Bayes' approach, which essentially implies that that the first moments of the OS in each of the dimensions are uncorrelated[2].

4 Uniform Distribution

We first extend the uni-dimensional results of [5] to show that CMOS can also attain a similar optimal bound for their multi-dimensional counterparts. To prove this claim, we first consider two-dimensional distributions, and thereafter, extend the result for higher dimensions.

[2] Although the uncorrelation is *sufficient*, we are not certain whether the *independence* of the features is *necessary*. As far as we are concerned, this is still an open issue.

Order Statistics: For a *prima facie* case, we consider two 2-dimensional uniform distributions U_1 and U_2 in which both the features are in $[0,1]^2$ and $[h,1+h]^2$ respectively. Consequently, we see that the overlapping region of the distributions forms a square In this case, it is easy to verify that the Bayesian classifier is the diagonal that passes through the intersection points of the distributions. For the classification based on the moments of the 2-OS, because the features are independent for both dimensions, we can show that this is equivalent to utilizing the OS at position $\frac{2}{3}$ of the first distribution for both dimensions, and the OS at the position $h+\frac{1}{3}$ of the second distribution for both dimensions.

Theoretical Analysis - 2-OS: Consider the points $\mathbf{u}_1 = \left(\frac{2}{3},\frac{2}{3}\right)$ of the first distribution and $\mathbf{u}_2 = \left(h+\frac{1}{3},h+\frac{1}{3}\right)$ of the second distribution in the two-dimensional space. We now show that if we compare the testing point $\mathbf{x} = (x_1,x_2)$ with these points, the PR attains the optimal classification, i.e., that which is the result of comparing it with the corresponding means.

Theorem 1. *For the 2-class problem in which the two 2-dimensional class conditional distributions are Uniform and identical, CMOS attains the optimal Bayes' bound.*

Proof. The proof is omitted here in the interest of brevity, but is in [7]. □

Experimental Results - 2-OS: The CMOS method for 2-dimensional uniform distributions U_1 (in $[0,1]$ in both dimensions) and U_2 (in $[h,1+h]$ in both dimensions) has been tested, and the results are given in Table 1. For each of the experiments, we generated 1,000 points for the classes ω_1 and ω_2.

Table 1 Classification of Uniformly distributed 2-dimensional classes by the CMOS 2-OS method for different values of h

h	0.95	0.90	0.85	0.80	0.75	0.70	0.65	0.60
Bayesian	99.845	99.505	98.875	98.045	97.15	95.555	94.14	91.82
CMOS	99.845	99.505	98.875	98.045	97.15	95.555	94.14	91.82

Theoretical Analysis - k-OS: We shall now discuss the efficiency of the k-OS CMOS. As in the case of the 2-OS CMOS, by enforcing the fact that the features are independent, we can, indeed, prove that the k-OS CMOS also attains the optimal Bayesian bound.

Theorem 2. *For the 2-class problem in which the two class conditional distributions are Uniform and identical as $U(0,1)^2$ and $U(h,1+h)^2$, optimal Bayesian classification can be achieved by using symmetric pairs of the n-OS, i.e., the $n-k$ OS for ω_1 (represented by u_1) and the k OS for ω_2 (represented by u_2) if and only if $k > \frac{(n+1)(1-h)}{2}$ for both the features. If this condition is violated, the CMOS classifier uses the* Dual *condition, i.e., the k OS for ω_1 and the $n-k$ OS for ω_2 for both the features.*

Proof. The proof can be found in [7] and is omitted here. □

Experimental Results - k-OS: The k-OS CMOS method for 2-dimensional Uniform distributions U_1 (in $[0,1]$ in both dimensions) and U_2 (in $[h,1+h]$ in both dimensions) has been tested, and the results are given in Table 2.

Table 2 Classification of Uniformly distributed 2-dimensional classes by the k-OS CMOS method for different values of h. The scenarios when we have invoked the *Dual* condition are specified by the notation "(D)"

$h \longrightarrow$	**0.95**	**0.90**	**0.85**	**0.80**	**0.75**	**0.70**	**0.65**
$\langle \frac{2}{3}, \frac{1}{3} \rangle$	99.92	99.58	98.86	97.94	96.78	95.69	93.73
$\langle \frac{4}{5}, \frac{1}{5} \rangle$	99.92	99.58	98.86	97.94	96.78	95.69	93.73
$\langle \frac{6}{7}, \frac{1}{7} \rangle$	99.92	99.58	98.86	97.94	96.78	95.69 (D)	93.73 (D)
$\langle \frac{4}{7}, \frac{3}{7} \rangle$	99.92	99.58	98.86	97.94	96.78	95.69	93.73
$\langle \frac{8}{9}, \frac{1}{9} \rangle$	99.92	99.58	98.86	97.94	96.78 (D)	95.69 (D)	93.73 (D)
$\langle \frac{7}{9}, \frac{2}{9} \rangle$	99.92	99.58	98.86	97.94	96.78	95.69	93.73

If we examine Table 2, we can see that k-OS CMOS attained the optimal Bayes' bound for all the cases where the condition is strictly enforced. But, for the cases where the condition failed, the dual condition holds and so the CMOS positions should be reversed so as to attain the optimal accuracy. For example, consider the classification with the CMOS positions, $\langle \frac{8}{9}, \frac{1}{9} \rangle$ for $h = 0.75$. As stated earlier, for any symmetric pair, the condition which is to be enforced is that the $\frac{n-1+k}{n+1}^{th}$ percentile should be less than the $\frac{k}{n+1}^{th}$ percentile for *every* dimension. But, for the pairs $\langle \frac{8}{9}, \frac{1}{9} \rangle$, this is not true, and hence, the dual CMOS has to be invoked to obtain the optimal bound.

Multi-dimensional Extension: As the multi-dimensional distribution naturally imposes the independence for the Uniform scenario, we can extend the result of Theorems 1 and 2 to obtain a classifier for such a problem.

Theorem 3. *For the* 2*-class problem in which the two class conditional distributions are Uniform and identical as* $U(0,1)^d$ *and* $U(h,1+h)^d$*, the classifier* $x_1 + x_2 + \ldots + x_d \underset{\omega_2}{\overset{\omega_1}{\lessgtr}} \frac{d}{2}(h+1)$ *obtained by using symmetric pairs of the n-OS, i.e., the* $n-k$ *OS for* ω_1 *and k OS for* ω_2*, leads to an optimal Bayesian classification if and only if* $k > \frac{(n+1)(1-h)}{2}$ *for all the features. If this condition is violated, the CMOS classifier uses the* Dual *condition, i.e., the k OS for* ω_1 *and the* $n-k$ *OS for* ω_2 *for both the features.*

Proof. The proofs for higher order multi-dimensional distributions follow due to the independence and due to the arguments analogous to those used in Theorems 1 and 2. The details are omitted here in the interest of brevity. □

5 Doubly Exponential Distribution

We earlier worked with the 2-class problem in which the class conditional distributions are uni-dimensional Doubly Exponential and identical, and demonstrated that the optimal Bayesian classification can be achieved by using symmetric pairs of the n-OS, i.e., the $n-k$ OS for the first distribution and the k-OS for the second distribution if and only if $ln\left(\frac{2k}{n+1}\right) > \frac{c_1-c_2}{2}$ where c_1 and c_2 are the respective means of the distributions. Now, we shall extend this result for the multi-dimensional Doubly Exponential distributions.

Order Statistics: Let ω_1 and ω_2 be the two classes where the features follow two-dimensional Doubly Exponential distributions. Then, since the random vectors have independent components[3], the pdfs can be represented as:

$$f_1(\mathbf{x}) = \frac{\lambda_{11}}{2}e^{-\lambda_{11}|x_1-\mathbf{c_{11}}|} \cdot \frac{\lambda_{12}}{2}e^{-\lambda_{12}|x_2-\mathbf{c_{12}}|}, \quad -\infty < x_1 < \infty, -\infty < x_2 < \infty,$$

$$f_2(\mathbf{x}) = \frac{\lambda_{21}}{2}e^{-\lambda_{21}|x_1-\mathbf{c_{21}}|} \cdot \frac{\lambda_{22}}{2}e^{-\lambda_{22}|x_2-\mathbf{c_{22}}|}, \quad -\infty < x_1 < \infty, -\infty < x_2 < \infty,$$

where $\mathbf{c}_1 = (c_{11}, c_{12})$ and $\mathbf{c}_2 = (c_{21}, c_{22})$ are the respective means of the distributions, and the values λ_{11}, λ_{12}, λ_{21} and λ_{22} are the corresponding parameters of the distributions in the respective dimensions.

In [5], we had derived the k-OS CMOS positions for the uni-dimensional Doubly Exponential distribution as $u_1 = c_1 - \frac{1}{\lambda_1}\, ln\left(\frac{2k}{n+1}\right)$ and $u_2 = c_2 + \frac{1}{\lambda_2}\, ln\left(\frac{2k}{n+1}\right)$. As the individual features of the Doubly Exponential distribution are independent, the CMOS positions are computed directly using *these* independent univariate distributions, and thus have the corresponding forms as those of the positions obtained for the uni-dimensional distributions. Consequently, for the two dimensional distributions for classes ω_1 and ω_2, the CMOS positions $\mathbf{u}_1$ and $\mathbf{u}_2$ are $\left[c_{11} - \frac{1}{\lambda_{11}}\, ln\left(\frac{2k}{n+1}\right),\, c_{12} - \frac{1}{\lambda_{12}}\, ln\left(\frac{2k}{n+1}\right)\right]^T$ and $\left[c_{21} + \frac{1}{\lambda_{21}}\, ln\left(\frac{2k}{n+1}\right),\, c_{22} + \frac{1}{\lambda_{22}}\, ln\left(\frac{2k}{n+1}\right)\right]^T$ respectively.

Theoretical Analysis: We prove the optimal properties of CMOS for two-dimensional identical and symmetrically placed Doubly Exponential distributions, with means $(0,0)$ and (c,c) respectively, and with identical λ.

Theorem 4. *For the* 2*-class problem in which the two class conditional distributions are two-dimensional Doubly Exponential, identical and symmetric, optimal Bayes' classification can be achieved by using symmetric pairs of the n-OS, i.e., the* $n-k$ *OS for* ω_1 *and the k OS for* ω_2 *if and only if* $ln\left(\frac{2k}{n+1}\right) > \frac{c_1-c_2}{2}$ *for both the features. If this condition is violated, the CMOS classifier uses the* Dual *condition, i.e., the k OS for* ω_1 *and the* $n-k$ *OS for* ω_2 *for both the features.*

Proof. The proof is not omitted here, but is included in [7]. □

[3] This independence is a consequence of the fact that the exponential terms can be factored so that each factor only possesses a *single* variable.

Experimental Results: The CMOS classifier has been rigorously tested for a number of experiments with various distributions with means $\mathbf{c}_1 = \langle 0,0 \rangle$ and $\mathbf{c}_2 = \langle c,c \rangle$ respectively. The test results are depicted in Table 3.

Table 3 Classification of Doubly Exponentially distributed 2-dimensional classes by the CMOS k-OS method for different means

No.	c	w.r.t Mean	$\langle \frac{2}{3}, \frac{1}{3} \rangle$	$\langle \frac{4}{5}, \frac{1}{5} \rangle$	$\langle \frac{5}{7}, \frac{2}{7} \rangle$	$\langle \frac{8}{9}, \frac{1}{9} \rangle$
1	3	96.55	96.55	96.55	96.55	96.55
2	2.5	95.5	95.5	95.5	95.5	95.5
3	2	92	92	92	92	92
4	1.5	89.3	89.3	89.3 (D)	89.3	89.3 (D)

Now, consider the results presented in the row denoted by Trial No. 4. In this case, the testing attained the Bayes' accuracy for the symmetric OS pairs $\langle \frac{2}{3}, \frac{1}{3} \rangle$ and $\langle \frac{5}{7}, \frac{2}{7} \rangle$, but the dual pairs had to be used for the pairs $\langle \frac{4}{5}, \frac{1}{5} \rangle$ and $\langle \frac{8}{9}, \frac{1}{9} \rangle$, since these values violated the condition imposed by Theorem 4.

Multi-Dimensional Extension: As the features are again independent because of the explicit factorizability, we can extend the result of Theorem 4 to perform a classification with respect to the $\langle \frac{n+1-k}{n+1}, \frac{k}{n+1} \rangle$ positions of each of the features, for identical and symmetrical distributions.

Theorem 5. *For the 2-class problem in which the two class conditional distributions are d-dimensional Doubly Exponential, identical and symmetric, the optimal Bayesian classifier is* $x_1 + x_2 + ... + x_d = \frac{d}{2} \cdot c$ *and is exactly the CMOS classifier obtained by using symmetric pairs of the n-OS, i.e., the* $n-k$ *OS for* ω_1 *and the* k *OS for* ω_2*, if and only if* $ln\left(\frac{2k}{n+1}\right) > \frac{c_1 - c_2}{2}$ *for all the features. If this condition is violated, the CMOS classifier uses the* Dual *condition, i.e., the* k *OS for* ω_1 *and the* $n-k$ *OS for* ω_2 *for all the features.*

Proof. The details of the proof are omitted here to avoid repetition. □

6 Gaussian Distribution

In this section, we intend to work with multi-dimensional Gaussian distribution. Earlier in [5], we showed that CMOS can attain optimal classification for uni-dimensional Gaussian distribution. The expected values of the first moment of the 2-OS can be determined as $E[\mathbf{x}_{1,2}] = \mu - \frac{\sigma}{\sqrt{2\pi}}$ and $E[\mathbf{x}_{2,2}] = \mu + \frac{\sigma}{\sqrt{2\pi}}$ as shown in [1]. We initially deal with the two-dimensional Gaussian distribution, and then extend the result for higher dimensions.

Order Statistics: Let ω_1 and ω_2 be the two classes where the features follow two-dimensional Gaussian distributions. As the pdfs of the Gaussian distributions are

not factorizable, we need to assume the independence (i.e., uncorrelation) of the features. Then, the pdfs can be represented as:

$$f_1(\mathbf{x}) = \frac{1}{\sqrt{2\pi}\sigma_{11}} e^{\frac{-(x_1-\mu_{11})^2}{2\sigma_{11}^2}} \cdot \frac{1}{\sqrt{2\pi}\sigma_{12}} e^{\frac{-(x_2-\mu_{12})^2}{2\sigma_{12}^2}}, \quad -\infty < x_1, x_2 < \infty, \text{and}$$

$$f_2(\mathbf{x}) = \frac{1}{\sqrt{2\pi}\sigma_{21}} e^{\frac{-(x_1-\mu_{21})^2}{2\sigma_{21}^2}} \cdot \frac{1}{\sqrt{2\pi}\sigma_{22}} e^{\frac{-(x_2-\mu_{22})^2}{2\sigma_{22}^2}}, \quad -\infty < x_1, x_2 < \infty,$$

where $\mu_1 = (\mu_{11}, \mu_{12})$ and $\mu_2 = (\mu_{21}, \mu_{22})$ are the respective means and $\sigma_1 = (\sigma_{11}, \sigma_{12})$ and $\sigma_2 = (\sigma_{21}, \sigma_{22})$ are the corresponding standard deviations of the distributions.

Then, the 2-OS CMOS positions for the uni-dimensional Gaussian distribution are [5] $u_1 = \mu_1 - \frac{\sigma}{\sqrt{2\pi}}$ and $u_2 = \mu_2 + \frac{\sigma}{\sqrt{2\pi}}$. As the individual features of the Gaussian distribution are independent, the CMOS positions are computed directly using *these* independent univariate distributions, and thus have the same forms as those of the positions obtained for the uni-dimensional distributions. Thus, for the two dimensional features, the CMOS positions $\mathbf{u}_1$ and $\mathbf{u}_2$ are $\left[\mu_{11} - \frac{\sigma_{11}}{\sqrt{2\pi}},\ \mu_{12} - \frac{\sigma_{12}}{\sqrt{2\pi}}\right]^T$ and $\left[\mu_{21} - \frac{\sigma_{21}}{\sqrt{2\pi}},\ \mu_{22} - \frac{\sigma_{22}}{\sqrt{2\pi}}\right]^T$ respectively.

Theoretical Analysis: Without loss of generality, we consider the distributions to have the means $(0,0)$ and (μ,μ) respectively, and with identical standard deviations, σ ($\sigma_{11} = \sigma_{12} = \sigma_{21} = \sigma_{22} = \sigma$).

Theorem 6. *For the 2-class problem in which the two 2-dimensional class conditional distributions are Gaussian, identical and symmetric, the 2-OS CMOS attains the optimal Bayes' bound.*

Proof. The proof is included in [7] and is omitted here for brevity. □

Experimental Results: The CMOS has been rigorously tested for 2-dimensional Gaussian distributions, and the results are given in Table 4.

Table 4 Classification of 2-dimensional Gaussian distributions by the CMOS 2-OS method for different means $(0,0)$ and (μ,μ)

μ	1	1.5	2	2.5	3	3.5	4	4.5
Bayesian	75.985	85.485	91.93	96.13	98.335	99.34	99.81	99.95
CMOS	75.985	85.485	91.93	96.13	98.335	99.34	99.81	99.95

Multi-dimensional Extension: The result of Theorem 4 can be generalized for higher dimensions. As the features are assumed to be independent, the classification can be done with regard to the 2-OS CMOS positions of each of the features for the identical and symmetrical distributions.

Theorem 7. *For the* 2*-class problem in which the two class conditional distributions are d-dimensional Gaussian, identical and symmetric, the optimal Bayesian classifier has the form* $x_1 + x_2 + ... + x_d = \frac{d}{2} \cdot \mu$*, and this is again the classifier obtained by using symmetric 2-OS CMOS positions.*

Proof. The proof is straightforward and can be found in [7]. □

7 Rayleigh Distribution

In [6], we had earlier worked with uni-dimensional Rayleigh distributions in which CMOS attained near-optimal classification with regard to the classifiers based on the medians of the distribution. We also showed that the error difference created by the CMOS classifier when compared to the Bayesian classifier is negligible by considering the differences of the error probabilities quantified by the differences between the areas under the curves of the resulting errors. We now investigate the two-dimensional scenario.

Order Statistics: Let ω_1 and ω_2 be the two classes where the features follow two-dimensional Rayleigh distributions. Earlier, in [5], we derived the 2-OS CMOS positions for the uni-dimensional Rayleigh distribution as $u_1 = \sigma\sqrt{2\ ln(3)}$ and $u_2 = \theta + \sigma\sqrt{2ln\left(\frac{3}{2}\right)}$. In order to extend this result for a two-dimensional case, as before, we assume a Naïve-Bayes' approach, in which the first moments of the OS in each of the dimensions are uncorrelated. Consequently, for the two dimensional distributions for the classes the CMOS positions $\mathbf{u}_1$ and $\mathbf{u}_2$ are respectively $\mathbf{u}_1 = \left[\sigma_{11}\sqrt{2\ ln(3)},\ \sigma_{12}\sqrt{2\ ln(3)}\right]^T$ and $\mathbf{u}_2 = \left[\theta_1 + \sigma_{21}\sqrt{2ln\left(\frac{3}{2}\right)},\ \theta_2 + \sigma_{22}\sqrt{2ln\left(\frac{3}{2}\right)}\right]^T$.

Theoretical Analysis: As in the uni-dimensional case in [6], we can compute the differences in the corresponding analogous volumes created by the classifiers for the respective distributions. The "ceiling" of the volume is rather complex because it involves the difference between the corresponding three-dimensional surfaces. However, we can easily obtain an upper bound for this volume by considering the smallest bounding rectanguloid.

The maximum bound of the error can be numerically evaluated and can be found to be nearly zero. Thus, we conclude that the maximum error of the CMOS classifier is negligible and attains a near-optimal bound.

Experimental Results: The CMOS method has been rigorously tested with different possibilities of the k-OS and for various values of n, and the test results are given in Table 5.

Table 5 Classification of Rayleigh distributed 2-dimensional classes by the CMOS k-OS method for different values of θ

No.	**Order(n)**	**Moments**	$\theta = 2$	$\theta = 1.5$	$\theta = 1.3$	$\theta = 1.2$
1	Median	$\left[\left(\frac{1}{2},\frac{1}{2}\right),\left(\frac{1}{2},\frac{1}{2}\right)\right]$	98.25	95.3	93.15	89.75
2	Two	$\left[\left(\frac{2}{3},\frac{1}{3}\right),\left(\frac{2}{3},\frac{1}{3}\right)\right]$	98.35	95.3	92.95	89.6
3	Four	$\left[\left(\frac{4}{5},\frac{1}{5}\right),\left(\frac{4}{5},\frac{1}{5}\right)\right]$	98.65	95.3	92.7	90.6
4	Six	$\left[\left(\frac{6}{7},\frac{1}{7}\right),\left(\frac{6}{7},\frac{1}{7}\right)\right]$	98.75	95.15	92.2 (D)	90.35 (D)
5	Six	$\left[\left(\frac{4}{7},\frac{3}{7}\right),\left(\frac{4}{7},\frac{3}{7}\right)\right]$	98.25	95.2	93.15	89.75
6	Eight	$\left[\left(\frac{8}{9},\frac{1}{9}\right),\left(\frac{8}{9},\frac{1}{9}\right)\right]$	98.8	95.2 (D)	92.25 (D)	89.9 (D)
7	Eight	$\left[\left(\frac{7}{9},\frac{2}{9}\right),\left(\frac{7}{9},\frac{2}{9}\right)\right]$	98.65	95.3	92.85	90.45

8 Other Multi-dimensional Distributions

In Sections 4 - 7, we dealt with some of the multi-dimensional distributions of the exponential family and proved that the CMOS can attain the optimal Bayes' bound for all the symmetric distributions and a near-optimal bound for the asymmetric distribution. One can see that the strategy is analogous to a Naïve-Bayes' approach, although it, really, is of an *anti*-Naïve-Bayes' paradigm. For the distributions of which the class-conditional densities are not factorizable, it is important to assume that the features are uncorrelated. The same argument can be used for any higher-order distributions.

9 Conclusions

In this paper, we generalized the results of CMOS for multi-dimensional distributions. We provided the analytical and experimental results for the two-dimensional Uniform, Doubly-exponential, Gaussian and Rayleigh distributions, and also generalized the approach for higher dimensions. We have showed that CMOS can attain the optimal Bayes' bound for symmetric distributions and near-optimal results for asymmetric distributions. The analogous results for the other distributions in the exponential family, which were discussed in [5, 6] are also available, but omitted here to avoid repetition.

References

1. Ahsanullah, M., Nevzorov, V.B.: Order Statistics: Examples and Exercises. Nova Science Publishers, Inc. (2005)
2. Duda, R.O., Hart, P.: Pattern Classification and Scene Analysis. A Wiley Interscience Publication (2000)

3. Morris, K.W., Szynal, D.: A goodness-of-fit for the Uniform Distribution based on a Characterization. Journal of Mathematical Science 106, 2719–2724 (2001)
4. Lin, G.D.: Characterizations of Continuous Distributions via Expected values of two functions of Order Statistics. Sankhya: The Indian Journal of Statistics 52, 84–90 (1990)
5. Thomas, A., Oommen, B.J.: The Fundamental Theory of Optimal "Anti-Bayesian" Parametric Pattern Classification Using Order Statistics Criteria. Pattern Recognition 46, 376–388 (2013)
6. Thomas, A., Oommen, B.J.: Optimal Order Statistics-based "Anti-Bayesian" Parametric Pattern Classification for the exponential family. Pattern Recognition (2012)
7. Thomas, A., Oommen, B.J.: Order Statistics-based Parametric Classification for Multi-dimensional Distributions (2012) (to be submitted)
8. Too, Y., Lin, G.D.: Characterizations of Uniform and Exponential Distributions. Academia Sinica 7(5), 357–359 (1989)

Knowledge Extraction from Graph-Based Structures in Conceptual Design

Grażyna Ślusarczyk

Abstract. This paper deals with using knowledge in order to support the conceptual stage of the design process. Both a visual language composed of design drawings and the internal graph-based structures representing these drawings are specified on the basis of the specified conceptualization of the design domain. The defined ontological commitment enables to transform knowledge about drawings encoded in their internal representations into logic formulas. The extracted facts concerning drawings together with axioms describing general domain-oriented design knowledge allow the design support system to reason about validity of created design solutions. The approach is illustrated on examples of designing floor layouts.

1 Introduction

Although there are many CAD tools for describing, editing, analyzing, and evaluating design projects, the conceptual design phase is the least supported one [7]. This paper deals with extracting and processing knowledge which allows the design system to support the conceptual stage of the design process. The presented method is an extension of the approach proposed in [5, 6]. It describes early design drawings with internal graph-based representations created automatically, and proposes a logic-based method of reasoning about validity of projects. A clear distinction between axioms describing design constraints which must be obeyed and formulas describing requirements specified by the user is made.

In this paper a conceptualization of a visual design domain, which specifies concepts that are assumed to exist in this domain and relationships that hold among them [8] is outlined. This conceptualization constitutes the basis for defining a

Grażyna Ślusarczyk
The Faculty of Physics, Astronomy and Applied Computer Science
Jagiellonian University, Reymonta 4, 30-059 Kraków, Poland
e-mail: gslusarc@uj.edu.pl

R. Burduk et al. (Eds.): *CORES 2013*, AISC 226, pp. 31–40.
DOI: 10.1007/978-3-319-00969-8_3 © Springer International Publishing Switzerland 2013

problem-oriented visual language composed of design drawings, which contain general ideas about design objects, and is related to the internal representation of drawings in the form of hypergraphs, where hypergraph atoms represent concepts and relations of the domain.

Hypergraphs encode the syntactic and semantic information about early design solutions [8]. Their hyperedges represent drawing components and multi-argument relations among fragments of components. Due to the specified ontological commitment between elements of the vocabulary of the first-order logic language and entities of the conceptualization the knowledge stored in hypergraphs is extracted in the form of atomic formulas describing designs.

The reasoning module of the proposed system tests the consistency of drawings with axioms describing general domain-specific design knowledge and requirements/constraints expressed as first-order logic formulas by means of inference on the basis of atomic formulas (facts) describing designs. The inference reasoning, where the validity of logic formulas is checked using the software package ANTLR (ANother Tool for Language Recognition), is discussed. The resulting assessment of drawings given by the system supports decision-making process throughout the whole design process.

The approach is illustrated on examples of designing floor layouts created by means of our prototype HSSDR system [6].

2 Visual Design System

A schema of the proposed visual design system (called HSSDR) is shown in Fig.1. The design interface allows the designer to create design drawings by means of a problem-oriented visual language. A rule editor being a part of the design interface enables to define design constraints and requirements which should be obeyed during the whole design process.

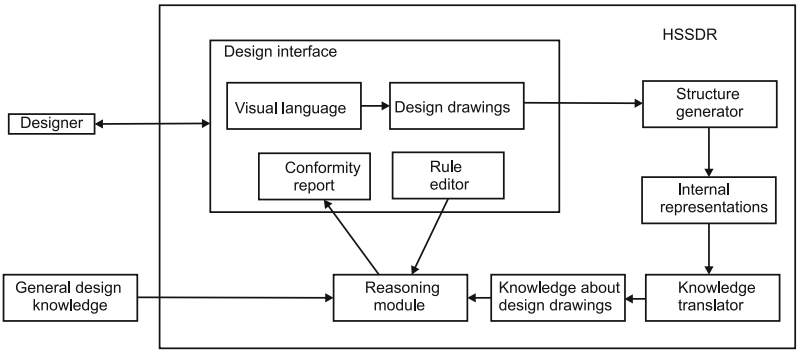


Fig. 1 A schema of the proposed computer-aided visual design system

The design drawings are automatically transformed into their internal representations in the form of hypergraphs by the structure generator module. The design knowledge stored in these structures is transformed into atomic formulas (facts) describing design drawings by the knowledge translator module. Then, the reasoning module checks the validity of design solutions by comparing logic formulas expressing both general design knowledge specific to a particular design task and design requirements and/or constraints with facts concerning created designs. The module makes the conformity report, which is treated as a feedback provided to the designer. The system supports rapid and intelligent decision-making throughout the whole conceptual design process.

3 Domain-Specific Visual Design Language

During the initial phase of design the conceptualization of the design domain being a classification and categorization of the knowledge [1,3] concerning this domain is defined. It specifies domain concepts, their attributes, taxonomy and relations which can hold among concepts. Each specialized design domain has its own drawing components related to concepts used in this domain and a convention which allows to express connections among drawing components according to relations of the conceptualization. The first level of the ontological commitment in the man-machine interaction is specified between mental images of design concepts and relations, and their externalization in the form of drawing components and their arrangements. Admissible layouts of drawing components in the specified domain form a specialized visual language.

The designer communicates with the design system using a visual editor which enables him to create design drawings with the use of a domain-specific visual language. This language is characterized by a vocabulary being a finite set of basic drawing components and a finite set of rules specifying possible configurations of these components.

Let us assume that a design task is to create a layout of one floor of an office building. On the basis of general requirements concerning this task the designer generates a design drawing visualizing an early solution (Fig. 2). A vocabulary of the visual language used in this case is composed of shapes corresponding to design concepts being rooms, walls, doors and motion sensors, i.e., polygons, line segments, small rectangles and circles, respectively. Thus the design drawing is composed of polygons which are placed in an orthogonal grid and represent rooms of a floor layout. The adjacency and accessibility between rooms, and fastening of sensors on walls constitute the considered relations between design concepts. The adjacency relation between rooms is expressed by line segments shared by polygons, while the accessibility relation is represented by line segments with small rectangles located on them. Black dots located on line segments represent sensors, while grey circle sectors correspond to spatial ranges of sensors. It should be noted that range spaces of sensors (the region of space that lies within their scope) do not correspond

to any physical entities nevertheless they are treated as drawing elements [2]. The sectors are not drawn by the designer, but they are automatically generated by the system on the basis of the sensor attributes specified by the designer while placing sensors in the drawing. Thus, the designer has the possibility to model constraints involving spatial functionality of objects.

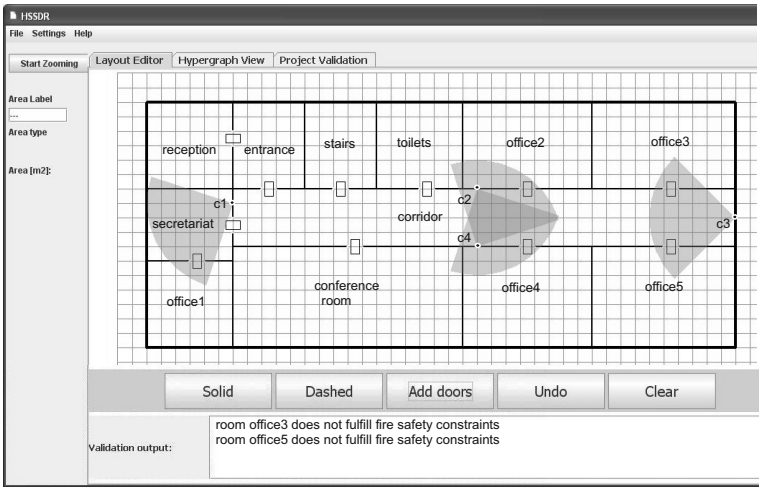


Fig. 2 A design of a floor layout of an office building with four sensors

4 Graph-Based Data Structures

In the considered system, the design drawings have internal representations in the form of attributed hypergraphs [5]. Hypergraph atoms represent concepts and relations corresponding to the elements of the conceptualization determined by the designer. Hypergraphs enable the system to store knowledge about syntactic and semantic aspects of created drawings.

The proposed hypergraphs have two types of hyperedges, called *object hyperedges* and *relational hyperedges*. Hyperedges of the first type correspond to design components and are labelled by component names. Hyperedges of the second type represent relations among fragments of components and can be either directed or non-directed. They are labelled by names of relations. Object hyperedges are connected with relational hyperedges by means of hypergraph nodes corresponding to fragments of design components. Attributes assigned to hyperedges and nodes represent properties of the corresponding design components and relations between them.

A hypergraph corresponding to the floor layout presented in Fig.2 is shown in Fig.3. It contains 16 object hyperedges, $e_1, ..., e_{16}$, (denoted by rectangles), where 12 of them represent rooms and 4 represent sensors. They are connected with the

relational ones (denoted by ovals) by means of hypergraph nodes which represent walls of rooms or fragments of sensors. The numbers assigned to hypergraph nodes representing walls correspond to the numbers of polygon sides, which are ordered clock-wise. There are 11 relational hyperedges (labelled *acc*) representing the accessibility relation between rooms, 8 relational hyperedges (labelled *adj*) representing the adjacency relation, and 4 directed relational hyperedges (labelled *on*) representing sensor fastening.

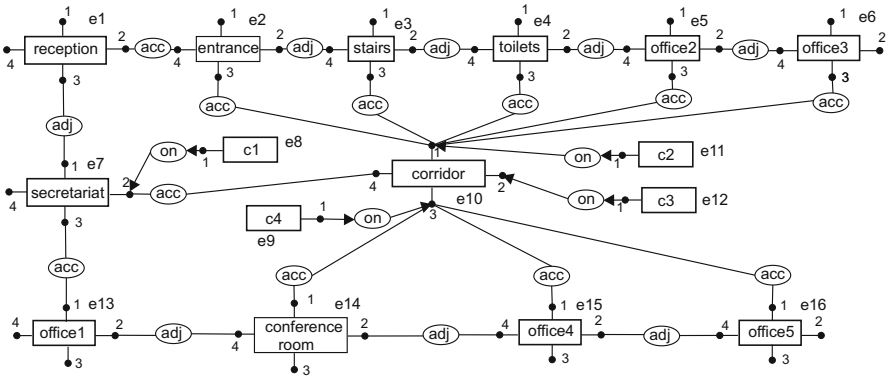


Fig. 3 A hypergraph corresponding to the floor layout presented in Fig.2

The values of such attributes as *area*, which are assigned to object hyperedges representing rooms, are automatically set by the system at the time of creating rooms on the basis of the occupied part of the grid on which designs are drawn. The values of other attributes, like *material* characterizing walls or *range* characterizing sensors, are specified by the designer at the time of establishing the relations between components. Attributes *length*, *door_number* and *loc_door* assigned to nodes representing walls of rooms specify the length of a wall, the number of doors in a wall and the location of doors, respectively.

5 Knowledge Extraction and Reasoning Mechanism

On the basis of the set of initial design requirements the designer externalizes his/her mental model of a design solution in the form of a design drawing. While analyzing successive drawings the designer infers useful information what results in adding the devised requirements or changing the existing ones. This process can be effectively supported by the interaction between the designer and computer during the whole design process. The visualization of solutions together with the feedback obtained from the system, which assesses the solutions, make the designer search for different possibilities of developing the solution further. He/she can either

consider new requirements or apply actions which lead to the fulfillment of still unsatisfied ones.

In the proposed approach information stored in hypergraphs corresponding to design drawings is translated to atomic formulas of the first-order logic, called *facts*. The reasoning module checks validity of design drawings by comparing these facts with first-order logic formulas expressing both general design knowledge and design requirements.

In order to define logic formulas their vocabulary $T = \{C, F, P\}$, where C is a set of constant symbols, F is a set of multi-argument function symbols, and P is a set of multi-argument predicates, is specified. Then the mapping called *ontological commitment* between elements of the vocabulary and entities of the conceptualization is defined. The drawing components are assigned to the constant symbols, their attributes are assigned to function symbols, while relations between the components correspond to the predicate symbols. This commitment allows the system to transform semantic and syntactic information encoded in the internal representations of drawings into logic formulas.

While designing floor layouts elements of a set of conceptualization concepts (rooms, walls, sensors) and relations (adjacency, accessibility and fastening) are associated with symbols of the vocabulary of the logic language. Walls of rooms, rooms and sensors are associated with constant symbols, while relations between them are assigned to predicate symbols. Attributes determined for walls, rooms and sensors correspond to function symbols.

We assume that we have a set of variables. The set of *terms* is formed starting from constant symbols and variables and closing off under function application, i.e., if $t_1, ..., t_n$, $n \geq 1$, are terms and $f \in F$ is a n-ary function symbol, then $f(t_1, ..., t_n)$ is also a term. An *atomic formula* is either of the form $p(t_1, ..., t_k)$, where $p \in P$ is a k-ary relation symbol and $t_1, ..., t_k$ are terms, or of the form $t_1 = t_2$, where t_1 and t_2 are terms. The set of general logical formulas is built over atomic ones using logical connectives and quantifiers.

The semantics of first-order formulas uses a *relational structure* consisting of a domain of individuals and a way of associating with elements of the vocabulary corresponding entities over the domain [4]. A relational T-structure L consists of a domain denoted $dom(L)$, an assignment of a k-ary relation $p^L \subseteq dom(L)^k$ to each k-ary relation symbol $p \in P$, an assignment of a n-ary function $f^L : dom(L)^n \rightarrow dom(L)$ to a n-ary function symbol $f \in F$, and an assignment of a $c^L \in dom(L)$ to each constant symbol c.

In the proposed approach, where structures of drawings have hypergraph-based representations, the domain of the relational structure contains hypergraphs. The relational structure assigns hypergraph nodes, object hyperedges and their labels to constants and terms, their attributes to functions, and relational hyperedges to predicates of formulas. The interpretation of each predicate is a relational hyperedge coming from nodes of at least one object hyperedge and coming into nodes of other object hyperedges.

In case of designing floor layouts, the relational structure assigns nodes representing walls, object hyperedges representing rooms and sensors to terms, and their

attributes to functions. The relations considered in this case, acc, adj and on, are binary, the first two are undirected ones, while the third one is directed. Each of these relations holds among design components if in the hypergraph there exists a relational hyperedge e_r labelled by the name of relation p^L ($p^L = lb(e_r)$) which connects nodes assigned to two different object hyperedges e_1 and e_2 corresponding to the design components and labelled lab_1, lab_2, where $lab_1 = lb(e_1)$, $lab_2 = lb(e_2)$ and lb is a hyperedge labelling function. In this case the formula $p(lab_1, lab_2)$ is created for the considered drawing. For each attribute f^L assigned to a hypergraph node v or hyperedge e a formula $f(lab) = n$ is created, where $lab = lb(v)$ or $lab = lb(e)$, and n is an admissible value of the attribute f^L. The above two types of formulas form a set Δ of atomic formulas describing the design drawing.

The computer system evaluates design drawings on the basis of a set Δ of atomic first-order logic formulas extracted from hypergraph representations of these drawings. These atomic formulas (facts) are related to topological properties of drawings encoded in hypergraph structures, and both geometrical and non-geometrical ones, which are stored in hypergraph attributes. The reasoning module automatically checks if the facts concerning the generated drawing are consistent with the axioms being first-order logic formulas expressing general design knowledge specific to a particular design task. The set of axioms Φ describes design standards like architectural norms, fire regulations, etc. In the next step facts describing drawings are compared with design constraints and requirements, which are also in the form of logic formulas of a set Ψ. Moreover, there exists the possibility to specify designer's own requirements and restrictions, which are added to the set Ψ, using a rule editor being a part of the design interface.

In order to check if a given formula is true in a structure L a specification of an interpretation of variables is needed. A valuation ω on a structure L is a function from variables to elements of $dom(L)$. Given a structure L, a valuation ω on L is inductively extended to functions that maps terms to elements of $dom(L)$.

Let us assume that for a given design drawing d, $dom(L)$ contains the hypergraph representing d. A design drawing d satisfies a formula $\varphi \in (\Phi \cup \Psi)$ if there exists a valuation ω from terms of φ to elements of $dom(L)$ such that φ is true under this valuation in L ($(L, \omega) \models \varphi$). It means that φ corresponds to a formula δ build over the set Δ of atomic formulas describing drawing d.

The reasoning module of the system in order to check if the design drawing satisfies a given formula φ decomposes this formula for each valuation of its terms into atomic parts which are compared with facts of Δ extracted from the hypergraph representation of this drawing. After checking all formulas of $\Phi \cup \Psi$ the system presents the report through the design interface.

Let us consider designing of the layout of a floor of an office building. Some of the design requirements can be as follows:

- $\psi_1 \equiv \exists t_1, t_2, t_3, t_4 : t_i = office_i$, i.e., there should be at least four offices on a floor,
- $\psi_2 \equiv \exists t_1, t_2 : t_1 = secretariat, t_2 = office_i \wedge acc(t_1, t_2)$, i.e., at least one office should be accessible from a secretariat,
- $\psi_3 \equiv \forall t_i = office_i \; \exists t_j \in sensor : observed(t_i, t_j)$, i.e., doors to all offices should be monitored by sensors.

Automatically obtained atomic formulas describing relations which hold among layout elements of a designed floor (Fig. 2) concern adjacency and accessibility between rooms, and fastening sensors on room walls. For example the relations concerning the room labelled *secretariat* presented in Fig. 2 are described by the following atomic formulas:

- $\delta_1 \equiv acc(secretariat, corridor)$ - accessibility from the corridor,
- $\delta_2 \equiv acc(secretariat, office_1)$ - accessibility from the office number 1,
- $\delta_3 \equiv adj(secretariat, reception)$ - adjacency to the reception, and
- $\delta_4 \equiv on(c_1, secretariat)$ - fastening the sensor number 1 on a wall of the secretariat.

The design solution presented in Fig. 2 satisfies requirements ψ_1, ψ_2, ψ_3. Formula ψ_1 is satisfied as the reasoning module finds a valuation $\omega(t_1) = e_{13}$, $\omega(t_2) = e_5$, $\omega(t_3) = e_6$, $\omega(t_4) = e_{15}$, $\omega(t_5) = e_{16}$, with $lb(e_{13}) = office1$, $lb(e_5) = office2$, $lb(e_6) = office3$, $lb(e_{15}) = office4$, $lb(e_{16}) = office5$, where e_5, e_6, e_{13}, e_{15} and e_{16} are object hypereges and lb is a hyperedge labelling function. Formula ψ_2 is satisfied when $\omega(t_1) = e_7$, $\omega(t_2) = e_{13}$, as $lb(e_7) = secretariat$, $lb(e_{13}) = office1$ and we have the formula of $\Delta\ \delta_2 \equiv acc(secretariat, office_1)$.

The predicate $observed(t_i, t_j)$, is satisfied if the doors of the room corresponding to term t_i are in the spatial range of the sensor corresponding to term t_j. In order to test this condition the system first represents it as a formula composed of atomic parts expressing the location of the sensor labelled t_j on the wall of one of the rooms ($on(t_j, t_k)$), the value of the attribute *range* defined for the sensor t_j ($range(t_j)$) and the value of the attribute *loc_door* assigned to the wall number w of the room t_i ($loc\_door(t_i.w)$). In other words $observed(t_i, t_j) \Leftrightarrow on(t_j, t_k) \wedge inrange(range(t_j), loc\_door(t_i.w))$, where the predicate *inrange*, is satisfied if the door located in the wall $t_i.w$ are in the spatial range of sensor t_j. Then the system checks if the rectangle representing the door is inside the fragment of the circle representing the range of the sensor. In Fig. 2 the doors of rooms $office1$, $office2$ and $office4$ are observed by sensors c_1, c_2 and c_4, respectively, while the doors of rooms $office3$ and $office5$ are observed by sensor c_3, and thus the formula ψ_3 is satisfied.

The conformity of the floor layout from Fig. 2 with the Polish Fire Code regulations is also checked. The regulations require that each evacuation route leading to a staircase should be not longer than 30 meters. The axiom corresponding to this condition for all corridor doors has the following form: $\phi \equiv \forall x \in wall, blg(x) = corridor, door\_number(x) \geq 1\ \exists x' \in wall, blg(x') = staircase,\ door\_number(x') \geq 1 :\ \forall i \in \{1, ..., door\_number(x)\} \exists j \in \{1, ..., door\_number(x')\} : dist(loc\_door_i(x), loc\_door_j(x')) \leq 30$, where blg is the function specifying the room to which a given wall belongs (i.e., it assigns an object hyperedge to one of its nodes), *dist* is a function computing the distance between two points, and the wall attribute *loc_door* specifies the coordinates of the door located on the wall. The condition is not satisfied for the doors of the $office3$ and $office5$, thus the message about it is shown in the bottom panel in Fig. 2. The situation can be easily corrected by moving the doors or changing the location of the staircase.

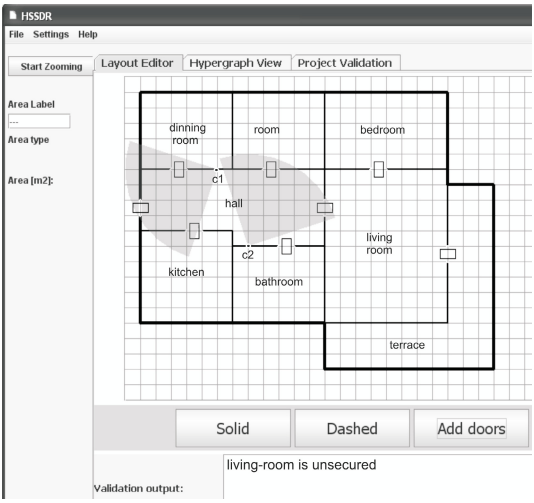


Fig. 4 A house floor layout with the unsecured living-room

After each design step the system checks which design constraints are not satisfied and shows the appropriate messages on the monitor screen. Modifications of drawings, automatically impose changes both in the hypergraph structures and facts concerning drawings which are extracted from these structures. Thus the set of atomic formulas describing designs forms dynamic knowledge about design solutions.

Let the design task be creating a floor layout of a one-storey house with monitoring which ensures the security. It means that all doors leading outside should

Fig. 5 A house floor layout with different placement of sensors

be observed by motion sensors. For the floor layout presented in Fig. 4 the system shows the message saying that there exist rooms which are not secured properly. The designer can find many alternative places for adding new sensors in order to satisfy design constraints (Fig. 5a and Fig. 5b).

6 Conclusions

This paper deals with using design knowledge to support the conceptual phase of design. The conceptualization of the design domain and ontological commitment are used to specify the method of automatic reasoning based on information stored in hypergraph structures corresponding to design drawings. The proposed logic model of reasoning uses atomic formulas extracted from attributed hypergraphs. Based on these formulas the verification of logic formulas representing design requirements specific to particular design tasks and defined by the designer is preformed. The consistence of design solutions with axioms expressing general design knowledge is also tested. The proposed inference method has been described on examples of designing floor layouts with the use of the prototype visual design system. Its reasoning module uses ANTLR software to verify logic formulas by decomposing them into atomic parts and comparing with formulas obtained from hypergraphs.

In the future the system will be extended by 3D visual languages allowing form-oriented design. The externalization of design ideas could be also supported by means of shape grammars. Generation methods used to create architectural forms will require using 3-D shape grammars.

References

1. Coyne, R.D., Rosenman, M.A., Radeford, A.D., Balachandran, M., Gero, J.S.: Knowledge based design systems. Addison-Wesley Publishing Company (1990)
2. Bhatt, M., Freksa, C.: Spatial computing for design: An artificial intelligence perspective. In: Visual and Spatial Reasoning for Design Creativity (SDC 2010) (2011)
3. Bhatt, M., Hois, J., Kutz, O.: Ontological modelling of form and function for architectural design. Journal of Applied Ontology 7, 233–267 (2012)
4. Fagin, R., Halpern, J.Y., Moses, Y., Vardi, M.Y.: Reasoning About Knowledge. MIT Press, Cambridge (1995)
5. Grabska, E., Lachwa, A., Ślusarczyk, G., Grzesiak-Kopeć, K., Lembas, J.: Hierarchical Layout Hypergraph Operations and Diagrammatic Reasoning. Machine GRAPHICS and VISION 16, 23–38 (2007)
6. Grabska, E., Borkowski, A., Palacz, W., Gajek, S.: Hypergraph system supporting design and reasoning. In: EG-ICE International Workshop, Berlin (2009)
7. Grabska, E., Ślusarczyk, G.: Knowledge and reasoning in design systems. Automation in Construction 22, 927–934 (2011)
8. Ślusarczyk, G.: Visual language and graph-based structures in conceptual design. Advanced Engineering Informatics 26, 267–279 (2012)

Estimation of the Relations of: Equivalence, Tolerance and Preference on the Basis of Pairwise Comparisons

Leszek Klukowski

Abstract. The paper presents the estimators of three relations: equivalence, tolerance and preference in a finite set on the basis of multiple pairwise comparisons, disturbed by random errors; they have been developed by the author. The estimators can rest on: binary (qualitative), multivalent (quantitative) and combined comparisons. The estimates are obtained on the basis of discrete programming tasks. The estimators require weak assumptions about distributions of comparisons errors, especially allow non-zero expected values. The estimators have good statistical properties, in particular consistency. The paper summarizes the results obtained by the author; the broader view is presented in Klukowski 2011a.

1 Introduction

Estimation of the relations of equivalence, tolerance, or preference, on the basis of multiple pairwise comparisons with random errors, is aimed at determination of an actual structure of data. It also provides the properties of estimates: consistency, distributions of errors, efficiency, etc.

The approach applied in the work rests on a statistical and optimisation paradigms: to determine the relation form, which minimizes the inconsistencies (differences) with a sample - in the form of multiple pairwise comparisons: statistical tests, experts' opinions or other procedures, prone to generating random errors. The approach presented here is an original contribution of the author to the subject. The comparisons are assumed in two forms: binary - expressing qualitative features, e.g. the direction of preference, and multivalent - expressing quantitative features of a pair, e.g. the difference of ranks of elements. The assumptions about distributions of errors of comparisons are weaker than those commonly used in the literature (David,

Leszek Klukowski
Systems Research Institute Polish Academy of Sciences, Newelska 6, 01-447 Warsaw, Poland
e-mail: Leszek.Klukowski@ibspan.waw.pl

R. Burduk et al. (Eds.): *CORES 2013*, AISC 226, pp. 41–52.
DOI: 10.1007/978-3-319-00969-8_4 © Springer International Publishing Switzerland 2013

1988; Slater 1961). The estimators can be applied also in the case of unknown distributions of comparison errors.

The estimators have good statistical properties, obtained on the basis of: properties of random variables expressing differences between the relation form and comparisons, the probabilistic inequalities (Hoeffding 1963, Chebyshev), properties of order statistics (David, 1970). The properties guarantee consistency of estimates for the number of independent comparisons of each pair approaching infinity.

The literature on pairwise comparisons with random errors concerns mainly ranking problems – classical results are presented in: David (1988), Bradley (1976, 1984), Davidson (1976) (bibliography), Brunk (1960). The authors mentioned present and discuss a complete range of existing methods: assumptions, estimators and their properties, tests for validation of results.

The literature concerning classification methods, based on pairs of elements is extremely extensive (see e.g. Gordon 1999, Hand 1986, Kaufman, Rousseeuv 1990, Hastie, Tibshirani, Friedman 2002, Kohonen 1995, Hartigan 1975). However, it should be emphasized that existing approaches do not cover entirely the problems presented in the work.

The paper consists with 5 sections. The second section presents main ideas of estimation, in particular the form of estimators. The next section – presents properties of estimators obtained by the author. The forth section discusses briefly optimization algorithms, which can be applied for determining of estimates. Last section summarizes results of the author in the area under consideration and shows problems for further researches.

2 Estimation of the Relations – Main Ideas

2.1 Definitions, Notations and Formulation of the Estimation Problems

The problem of estimation of relation on the basis of pairwise comparisons can be stated as follows.

We are given a finite set of elements $X = \{x_1, ..., x_m\} (3 \leqslant m < \infty)$. There exists in the set $\mathbf{X}$: the equivalence relation $\mathbf{R}^{(e)}$ (reflexive, transitive, symmetric), or the tolerance relation $\mathbf{R}^{(\tau)}$ (reflexive, symmetric), or the preference relation $\mathbf{R}^{(p)}$ (alternative of the equivalence relation and strict preference relation). Each relation generates some family of subsets $\chi_1^{(\ell)*}, ..., \chi_n^{(\ell)*}$ $(\ell \in \{p, e, \tau\}; n \geqslant 2)$.

The equivalence relation generates the family $\chi_1^{(e)*}, ..., \chi_n^{(e)*}$ having the following properties:

$$\bigcup_{q=1}^{n} \chi_q^{(e)*} = \mathbf{X}, \tag{1}$$

$$\chi_r^{(e)*} \cap \chi_s^{(e)*} = \{\mathbf{0}\} \tag{2}$$

where:

0 – the empty set,

$$x_i, x_j \in \chi_r^{(e)*} \equiv x_i, x_j - \text{equivalent elements}, \tag{3}$$

$$(x_i \in \chi_r^{(e)*}) \cap (x_j \in \chi_s^{(e)*}) \equiv x_i,\ x_j \text{ - non-equivalent elements for } i \neq j,\ r \neq s. \tag{4}$$

The tolerance relation generates the family $\chi_1^{(\tau)*}$, ..., $\chi_n^{(\tau)*}$ with the property (1), i.e. $\bigcup_{q=1}^{n} \chi_q^{(\tau)*} = \mathbf{X}$, and the properties:

$\exists r,s\ (r \neq s)$ *such that* $\chi_r^{(\tau)*} \cap \chi_s^{(\tau)*} \neq \{\mathbf{0}\}$,

$$x_i, x_j \in \chi_r^{(\tau)*} \equiv x_i, x_j \text{ - equivalent elements}, \tag{5}$$

$$(x_i \in \chi_r^{(\tau)*}) \cap (x_j \in \chi_s^{(\tau)*}) \equiv x_i, x_j \text{ - non-equivalent elements for } i \neq j \text{ and } (x_i, x_j) \notin \chi_r^{(\tau)*} \cap \chi_s^{(\tau)*}, \tag{6}$$

$$\text{each subset } \chi_r^{(\tau)*}\ (1 \leqslant r \leqslant n) \text{ includes an element } x_i \text{ such that } x_i \notin \chi_s^{(\tau)*}\ (s \neq r). \tag{7}$$

The preference relation generates the family $\chi_1^{(p)*}$, ..., $\chi_n^{(p)*}$ with the properties (1), (2) and the property:

$$(x_i \in \chi_r^{(p)*}) \cap (x_j \in \chi_s^{(p)*}) \equiv x_i \text{ is preferred to } x_j \text{ for } r < s. \tag{8}$$

The relations defined by the conditions (1) - (8) can be expressed, alternatively, by the values (functions) $T_{\upsilon}^{(\ell)}(x_i, x_j), ((x_i, x_j) \in \mathbf{X} \times \mathbf{X}; \quad \ell \in \{p,\ e,\ \tau\}, \quad \upsilon \in \{b, \mu\}$; symbols b, μ denote – respectively – the binary and multivalent comparisons), defined as follows:

$$T_b^{(e)}(x_i, x_j) = \begin{cases} 0\ \textit{if exists } r \textit{ such that } (x_i, x_j) \in \chi_r^{(e)*}, \\ 1\ \textit{otherwise}; \end{cases} \tag{9}$$

• the function $T_b^{(e)}(x_i, x_j)$, describing the equivalence relation, assuming binary values, expresses the fact if a pair (x_i, x_j) belongs to a common subset or not;

$$T_b^{(\tau)}(x_i,x_j)=\begin{cases}0\ if\ exists\ \ r,\ s\ (r=s\ not\ excluded)\ such\ \ that\\ \quad (x_i,x_j)\in \chi_r^{(\tau)*}\cap\chi_s^{(\tau)*},\\ 1\ otherwise;\end{cases} \tag{10}$$

• the function $T_b^{(\tau)}(x_i,x_j)$, describing the tolerance relation, assuming binary values, expresses the fact if a pair (x_i,x_j) belongs to any conjunction of subsets (also to the same subset) or not; the condition (7) guarantees uniqueness of the description;

$$T_\mu^{(\tau)}(x_i,x_j)=\#(\Omega_i^*\cap\Omega_j^*), \tag{11}$$

where:

Ω_l^* – the set of the form $\Omega_l^*=\{s\mid x_l\in\chi_s^{(\tau)*}\}$,

$\#(\Xi)$ – the number of elements of the set Ξ;

• the function $T_\mu^{(\tau)}(x_i,x_j)$, describing the tolerance relation, assuming multivalent values, expresses the number of subsets of conjunction including both elements; condition (7) guarantees the uniqueness of the description;

$$T_b^{(p)}(x_i,x_j)=\begin{cases}0\ if\ there\ exists\ r\ such\ that\ (x_i,x_j)\in\chi_r^{(p)*},\\ -1\ if\ x_i\in\chi_r^{(p)*},\ x_j\in\chi_s^{(p)*}\ and\ r<s;\\ 1\ \ if\ \ x_i\in\chi_r^{(p)*},\ x_j\in\chi_s^{(p)*}\ and\ r>s;\end{cases} \tag{12}$$

• the function $T_b^{(p)}(x_i,x_j)$, describing the preference relation, assuming binary values, expresses the direction of preference in a pair or the equivalence of its elements;

$$T_\mu^{(p)}(x_i,x_j)=d_{ij}\ \Leftrightarrow\ x_i\in\chi_r^{(p)*},\ \ x_j\in\chi_s^{(p)*},\ \ d_{ij}=r-s; \tag{13}$$

• the function $T_\mu^{(p)}(x_i,x_j)$, describing the preference relation, assuming multivalent values, expresses the difference of ranks of elements x_i and x_j.

2.2 Assumptions about Pairwise Comparisons

The relation $\chi_1^{(\ell)*}$, ..., $\chi_n^{(\ell)*}$ is to be estimated on the basis of N ($N\geqslant 1$) comparisons of each pair $(x_i,x_j)\in\mathbf{X}\times\mathbf{X}$; any comparison $g_{\upsilon k}^{(\ell)}(x_i,x_j)$ evaluates the actual value of $T_\upsilon^{(\ell)}(x_i,x_j)$ and can be disturbed by a random error. The following assumptions concerning the comparison errors are made:

A1. The relation type (equivalence or tolerance or preference) is known, the number of subsets n - unknown.

A2. Any comparison $g_{\upsilon k}^{(\ell)}(x_i,x_j)\quad(\ell\in\{e,\ \tau,\ p\};\ \upsilon\in\{b,\ \mu\};\quad k=1,\ ...,\ N)$, is the evaluation of the value $T_\upsilon^{(\ell)}(x_i,x_j)$, disturbed by a random error. The probabilities of errors $g_{\upsilon k}^{(\ell)}(x_i,x_j)-T_\upsilon^{(\ell)}(x_i,x_j)$ have to satisfy the following assumptions:

$$P(g_{bk}^{(\ell)}(x_i,x_j) - T_b^{(\ell)}(x_i,x_j) = 0 \mid T_b^{(\ell)}(x_i,x_j) = \kappa_{bij}^{(\ell)}) \geqslant 1-\delta \quad (\kappa_{bij}^{(\ell)} \in \{-1,\ 0,\ 1\},\quad \delta \in (0,\ 1/2)), \tag{14}$$

$$\sum_{r\leqslant 0} P(g_{\mu k}^{(\ell)}(x_i,x_j) - T_\mu^{(\ell)}(x_i,x_j) = r \mid T_\mu^{(\ell)}(x_i,x_j) = \kappa_{\mu ij}^{(\ell)}) > 1/2 \quad (\kappa_{\mu ij}^{(\ell)} \in \{0,\ ...,\ \pm m\},\ \ r\text{– zero or an integer number}), \tag{15}$$

$$\sum_{r\geqslant 0} P(g_{\mu k}^{(\ell)}(x_i,x_j) - T_\mu^{(\ell)}(x_i,x_j) = -r \mid T_\mu^{(\ell)}(x_i,x_j) = \kappa_{\mu ij}^{(\ell)}) > 1/2 \quad (\kappa_{\mu ij}^{(\ell)} \in \{0,\ ...,\ \pm m\},\ \ r\text{– zero or an integer number}), \tag{16}$$

$$P(g_{\mu k}^{(\ell)}(x_i,x_j) - T_\mu^{(\ell)}(x_i,x_j) = r) \geqslant P(g_{\mu k}^{(\ell)}(x_i,x_j) - T_\mu^{(\ell)}(x_i,x_j) = r+1 \mid T_\mu^{(\ell)}(x_i,x_j) = \kappa_{\mu ij}^{(\ell)}) \quad (\kappa_{\mu ij}^{(\ell)} \in \{0,\ ...,\ m\},\ \ r>0), \tag{17}$$

$$P(g_{\mu k}^{(\ell)}(x_i,x_j) - T_\mu^{(\ell)}(x_i,x_j) = r) \geqslant P(g_{\mu k}^{(\ell)}(x_i,x_j) - T_\mu^{(\ell)}(x_i,x_j) = r-1 \mid T_\mu^{(\ell)}(x_i,x_j) = \kappa_{\mu ij}^{(\ell)}) \quad (\kappa_{\mu ij}^{(\ell)} \in \{0,\ ...,\ m\},\ \ r<0), \tag{18}$$

A3. The comparisons $g_{\upsilon k}^{(\ell)}(x_i,x_j)$ $(\ell \in \{e,\ \tau,\ p\};\ \upsilon \in \{b,\ \mu\};\quad (x_i,x_j) \in \mathbf{X} \times \mathbf{X};$ $k = 1,...,\ N)$ are independent random variables.

The assumption A3 makes it possible to determine the distributions of estimation errors of estimators. However, determination of the exact distributions of the (multidimensional) errors, in an analytic way, is complicated and, in practice, unrealizable. The main properties of the estimators, especially their consistency, are valid without the assumption.

The assumption A3 can be relaxed in the following way: the comparisons $g_{\upsilon k}^{(\ell)}(x_i,x_j)$ and $g_{\upsilon l}^{(\ell)}(x_r,x_s)$ $(l \neq k;\ r \neq i,j;\ s \neq i,j)$, have to be independent.

In the case of the preference relation including equivalent elements, the condition (14) can be relaxed to the form (15) – (16).

The assumptions A2 – A3 reflect the following properties of distributions of comparisons errors: the probability of correct comparison is greater than of the incorrect one - in the case of binary comparisons (inequality (14)); zero is the median of each distribution of comparison error (inequalities (14) – (16)); zero is the mode of each distribution of comparison error (inequalities (14) – (18)); the set of all comparisons comprises the realizations of independent random variables; the expected value of any comparison error can differ from zero.

2.3 The Form of Estimators

Two forms of estimators are examined. Their properties have been proven by the author (see References) on the basis of the well-known probabilistic inequalities (Hoeffding, 1963, Chebyshev), properties of order statistics (David, 1970), and convergence of variances of relevant variables to zero.

The estimate based on the total sum of differences, denoted $\hat{\chi}_1^{(\ell)}$, ..., $\hat{\chi}_{\hat{n}}^{(\ell)}$ (or $\hat{T}_\upsilon^{(\ell)}(x_i,x_j) \quad < i,j > \in R_m$), results from the minimization problem:

$$\min_{\chi_1^{(\ell)},\ldots,\chi_r^{(\ell)} \in F_{\mathbf{X}}^{(\ell)}} \left\{ \sum_{<i,j>\in R_m} \sum_{k=1}^{N} \left| g_{\upsilon k}^{(\ell)}(x_i,x_j) - t_\upsilon^{(\ell)}(x_i,x_j) \right| \right\} \tag{19}$$

where:

$F_{\mathbf{X}}^{(\ell)}$ – the feasible set, i.e. the family of all relations $\chi_1^{(\ell)}$, ..., $\chi_r^{(\ell)}$ of ℓ- th type in the set $\mathbf{X}$,

$t_\upsilon^{(\ell)}(x_i,x_j)$ - the function describing any relation $\{\chi_1^{(\ell)}, \ldots, \chi_r^{(\ell)}\}$ of ℓ-th type,

R_m – the set of the form $R_m = \{< i,j > \mid 1 \leqslant i,j \leqslant m;\ j > i\}$

(symbol $g_{\upsilon k}^{(\ell)}(x_i,x_j)$ is used for both random variables and realizations).

In the case of the preference relation and binary comparisons the following transformation is also applied:

$$\theta(g_{\upsilon k}^{(\ell)}(x_i,x_j) - t_\upsilon^{(\ell)}(x_i,x_j)) = \begin{cases} 0 \ \textit{if} \ g_{\upsilon k}^{(\ell)}(x_i,x_j) = t_\upsilon^{(\ell)}(x_i,x_j); \\ 1 \ \textit{if} \ g_{\upsilon k}^{(\ell)}(x_i,x_j) \neq t_\upsilon^{(\ell)}(x_i,x_j). \end{cases} \tag{20}$$

The criterion function with the use of the transformation (20) is simpler from the computational point of view. The properties of both approaches are similar (Klukowski, 1990b).

The estimate based on medians, denoted $\widehat{\chi}_1^{(\ell)}$, ..., $\widehat{\chi}_r^{(\ell)}$ (or $\widehat{T}_\upsilon^{(\ell)}(x_i,x_j)$), is obtained on the basis of the following minimization problem:

$$\min_{\chi_1^{(\ell)},\ldots,\chi_r^{(\ell)} \in F_X} \left\{ \sum_{<i,j>\in R_m} \left| g_\upsilon^{(\ell,me)}(x_i,x_j) - t_\upsilon^{(\ell)}(x_i,x_j) \right| \right\}, \tag{21}$$

where:

$g_\upsilon^{(\ell,me)}(x_i,x_j)$ - the sample median in the set $\{g_{\upsilon,1}^{(\ell)}(x_i,x_j),$
..., $g_{\upsilon N}^{(\ell)}(x_i,x_j)\}$.

The estimate, resulting from the criterion (19) or (20) will be denoted with symbols $\hat{\chi}_1^{(\ell)}$, ..., $\hat{\chi}_r^{(\ell)}$ or (equivalently) $\hat{T}_\upsilon^{(\ell)}(x_i,x_j)$, while the estimate resulting from the criterion (21) - with symbols $\widehat{\chi}_1^{(\ell)}$, ..., $\widehat{\chi}_r^{(\ell)}$ or $\widehat{T}_\upsilon^{(\ell)}(x_i,x_j)$.

In the case of the preference relation and medians from comparisons, the same transformation can be also applied.

The number of estimates, resulting from the criterion functions (19), (21) can exceed one; the unique estimate can be determined in a random way or as a result of validation. Multiple estimates can appear also in other methods (see David 1988, Ch. 2).

The assumptions A1 – A3 allow for inference about distributions of errors of estimates. Let us discuss first the estimator based on of the criterion (19). For each relation type one can determine a finite set including all possible realizations of comparisons

$$g_{\upsilon k}^{(\ell)}(x_i,x_j),(\ell \in \{e,\tau,p\},\ \upsilon \in \{b,\mu\},\ k=1,\ ...,\ N;<i,j>\in R_m)$$

and the probability of each realization. The use of the criterion (19) determines: the estimate, its probability and estimation error. The error has the form: $\{\hat{T}_{\upsilon}^{(\ell)}(x_i,x_j) - T_{\upsilon}^{(\ell)}(x_i,x_j);\ <i,j>\in R_m\}$, i.e. it is a multidimensional random variable. The analysis of such error is, in fact, unrealizable and it is suggested to replace it with one-dimension error:

$$\hat{\Delta}_{\upsilon}^{(\ell)} = \sum_{<i,j>\in R_m} |\hat{T}_{\upsilon}^{(\ell)}(x_i,x_j) - T_{\upsilon}^{(\ell)}(x_i,x_j)|. \tag{22}$$

The estimate with the error $\hat{\Delta}_{\upsilon}^{(\ell)} = 0$ is the errorless estimate. The probability of such error can be determined in the analytic way – as a sum of probabilities of all realizations of comparisons indicating the errorless estimate. It is clear that its value (probability) depends on the number of comparisons N and the variance of comparison errors; increase of N decreases the probability of such error and decreases the variance of the estimator. The probabilities of errors different from zero can be determined in a similar way; all possible errors and their probabilities determine the distribution function of the estimation error. Determination of the probability function in the analytic manner is complicated and involves huge computational cost - even for moderate m. Therefore, simulation approach has to be used for this purpose.

Similar considerations apply for the criteria (20), (21).

3 Properties of Estimators

The analytical properties of the estimators, established by the author, are based on properties of random variables expressing differences between pairwise comparisons and the relation form (expressed by $T_{\upsilon}^{(\ell)}(x_i,x_j)$). It has been proved by the author that the variables corresponding to the actual relation form have different properties than the variables corresponding to any other relation. The following results have been obtained: (i) the expected values of the variables expressing differences between comparisons and the relation form (see e.g. (26) below), corresponding to actual relation form are lower than the expected values of variables corresponding to any other relation (see e.g. (30); (ii) the variances of the variables expressing differences between comparisons and the relation form, both - actual and different than actual, divided by the number of comparisons N in the case of sum of differences, converge to zero for $N \to \infty$; (iii) the probability of the event that the variable corresponding to actual relation assumes a value lower than the variable corresponding to a relation other than actual converges to one for $N \to \infty$; the speed of convergence guarantees good efficiency of the estimates.

Properties (i) - (iii) provide the basis for construction of the estimators; these properties have been complemented with some additional features (Klukowski 1994) and a simulation experiment. An important result of the experiment consists

in the fact that efficiency of the estimator based on the sum of inconsistencies is higher than of the median estimator; the latter estimator is, though, simpler from computational point of view and more robust.

Let us illustrate these considerations by the simplest case, i.e. equivalence relation and the estimator resulting from the criterion (19). The differences between any comparison $g_{bk}^{(e)}(x_i,x_j)$ and $T_b^{(e)}(x_i,x_j)$ assume the form:

$$U_{bk}^{(e)*}(x_i,x_j) = \begin{cases} 0 \;\; if \;\; g_{bk}^{(e)}(x_i,x_j) = T_b^{(e)}(x_i,x_j); \;\; T_b^{(e)}(x_i,x_j) = 0; \\ 1 \;\; if \;\; g_{bk}^{(e)}(x_i,x_j) \neq T_b^{(e)}(x_i,x_j); \;\; T_b^{(e)}(x_i,x_j) = 0, \end{cases} \tag{23}$$

$$V_{bk}^{(e)*}(x_i,x_j) = \begin{cases} 0 \;\; if \;\; g_{bk}^{(e)}(x_i,x_j) = T_b^{(e)}(x_i,x_j); \;\; T_b^{(e)}(x_i,x_j) = 1; \\ 1 \;\; if \;\; g_{bk}^{(e)}(x_i,x_j) \neq T_b^{(e)}(x_i,x_j); \;\; T_b^{(e)}(x_i,x_j) = 1. \end{cases} \tag{24}$$

The sum of differences assumes, for any $k \quad (1 \leqslant k \leqslant N)$, the form:

$$\sum_{<i,j>\in I^{(e)*}} U_{bk}^{(e)*}(x_i,x_j) + \sum_{<i,j>\in J^{(e)*}} V_{bk}^{(e)*}(x_i,x_j), \tag{25}$$

where:

$I^{(e)*}$ – the set of pairs $\{<i,j> \mid T_b^{(e)*}(x_i,x_j) = 0\}$,

$J^{(e)*}$ – the set of pairs $\{<i,j> \mid T_b^{(e)*}(x_i,x_j) = 1\}$.

The total sum of the differences between the relation form and the comparisons is equal:

$$W_{bN}^{(e)*} = \sum_{k=1}^{N} (\sum_{<i.j>\in I^{(e)*}} U_{bk}^{(e)*}(x_i,x_j) + \sum_{<i,j>\in J^{(e)*}} V_{bk}^{(e)*}(x_i,x_j)). \tag{26}$$

Under the assumptions A1, A2, A3, the expected values of the variables $U_{bk}^{(e)*}(x_i,x_j)$, $V_{bk}^{(e)*}(x_i,x_j)$ satisfy the inequalities: $E(U_{bk}^{(e)*}(x_i,x_j)) \leqslant \delta, E(V_{bk}^{(e)*}(x_i,x_j)) \leqslant \delta$. Therefore, the expected value of the variable $W_{bN}^{(e)*}$ satisfies the inequality $E(W_{bN}^{(e)*}) \leqslant \frac{Nm(m-1)}{2}\delta$. Assumptions A1 – A3 allow for determining the variance $Var(W_{bN}^{(e)*})$; its value is finite and satisfies the inequality $Var(W_{bN}^{(e)*}) \leqslant \frac{Nm(m-1)}{2}\delta(1-\delta)$. Obviously:

$$E(\frac{1}{N}W_{bN}^{(e)*}) \leqslant \frac{m(m-1)}{2}\delta, \tag{27}$$

$$\lim_{N\to\infty} Var(\frac{1}{N}W_{bN}^{(e)*}) = 0. \tag{28}$$

Let us consider any relation $\tilde{\chi}_1^{(e)}$, ..., $\tilde{\chi}_{\tilde{n}}^{(e)}$ different than $\chi_1^{(e)*}$, ..., $\chi_n^{(e)*}$; this means that there exist pairs (x_i,x_j), such that $\tilde{T}_b^{(e)}(x_i,x_j) \neq T_b^{(e)}(x_i,x_j)$. Define the

random variables $\tilde{U}_{bk}^{(e)}(x_i,x_j)$, $\tilde{V}_{bk}^{(e)}(x_i,x_j)$, $\tilde{W}_{b}^{(e)}(x_i,x_j)$ corresponding to the such values $\tilde{T}_{b}^{(e)}(x_i,x_j)$:

$$\tilde{U}_{bk}^{(e)}(x_i,x_j) = \begin{cases} 0 \;\; if \;\; g_{bk}^{(e)}(x_i,x_j) = \tilde{T}_{b}^{(e)}(x_i,x_j); \;\; \tilde{T}_{b}^{(e)}(x_i,x_j) = 0; \\ 1 \;\; if \;\; g_{bk}^{(e)}(x_i,x_j) \neq \tilde{T}_{b}^{(e)}(x_i,x_j); \;\; \tilde{T}_{b}^{(e)}(x_i,x_j) = 0, \end{cases}$$

$$\tilde{V}_{bk}^{(e)}(x_i,x_j) = \begin{cases} 0 \;\; if \;\; g_{bk}^{(e)}(x_i,x_j) = \tilde{T}_{b}^{(e)}(x_i,x_j); \;\; \tilde{T}_{b}^{(e)}(x_i,x_j) = 1; \\ 1 \;\; if \;\; g_{bk}^{(e)}(x_i,x_j) \neq \tilde{T}_{b}^{(e)}(x_i,x_j); \;\; \tilde{T}_{b}^{(e)}(x_i,x_j) = 1, \end{cases} \tag{29}$$

$$\tilde{W}_{bN}^{(e)} = \sum_{k=1}^{N} (\sum_{\tilde{I}^{(e)}} \tilde{U}_{bk}^{(e)}(x_i,x_j) + \sum_{\tilde{J}^{(e)}} \tilde{V}_{bk}^{(e)}(x_i,x_j)). \tag{30}$$

where:

$\tilde{I}^{(e)}$ – the set of pairs $\{< i,j > \mid \tilde{T}_{b}^{(e)}(x_i,x_j) = 0\}$,

$\tilde{J}^{(e)}$ – the set of pairs $\{< i,j > \mid \tilde{T}_{b}^{(e)}(x_i,x_j) = 1\}$.

The expected values $E(\tilde{U}_{bk}^{(e)}(x_i,x_j))$, $E(\tilde{V}_{bk}^{(e)}(x_i,x_j))$ assume the form:

$$\begin{aligned} E(\tilde{U}_{bk}^{(e)}(x_i,x_j)) = 0 * P(g_{bk}^{(e)}(x_i,x_j) = 0 \mid T_{b}^{(e)}(x_i,x_j) = 1) + \\ 1 * P(g_{bk}^{(e)}(x_i,x_j) = 1 \mid T_{b}^{(e)}(x_i,x_j) = 1) \geqslant 1 - \delta, \end{aligned} \tag{31}$$

$$\begin{aligned} E(\tilde{V}_{bk}^{(e)}(x_i,x_j)) = 0 \times P(g_{bk}^{(e)}(x_i,x_j) = 0 \mid T_{b}^{(e)}(x_i,x_j) = 0) + \\ 1 \times P(g_{bk}^{(e)}(x_i,x_j) = 1 \mid T_{b}^{(e)}(x_i,x_j) = 0) \geqslant 1 - \delta, \end{aligned} \tag{32}$$

and:

$$E(\tilde{W}_{bN}^{(e)}) = \sum_{k=1}^{N} (\sum_{\tilde{I}^{(e)}}^{E} (\tilde{U}_{bk}^{(e)}(x_i,x_j)) + \sum_{\tilde{J}^{(e)}}^{E} (\tilde{V}_{bk}^{(e)}(x_i,x_j))) > \frac{m(m-1)}{2}\delta. \tag{33}$$

The formulae (27) – (33) indicate that the expected value $E(\frac{1}{N} W_{bN}^{(e)*})$, corresponding to the actual relation $\chi_1^{(e)*}$, ..., $\chi_n^{(e)*}$, is lower than the expected value $E(\frac{1}{N}\tilde{W}_{bN}^{(e)})$, corresponding to any other relation $\tilde{\chi}_1^{(e)}$, ..., $\tilde{\chi}_{\tilde{n}}^{(e)}$. The variances of both variables converge to zero for $N \to \infty$. The variables $U_{bk}^{(e)*}(x_i,x_j)$, $V_{bk}^{(e)*}(x_i,x_j)$ assume values equal to $\left| g_{bk}^{(e)}(x_i,x_j) - T_{b}^{(e)}(x_i,x_j) \right|$, used in the criterion function (19). Moreover, it can be also shown (see Klukowski, 1994), that:

$$P(W_{bN}^{(e)*} < \tilde{W}_{bN}^{(e)}) \geqslant 1 - \exp\{-2N(\frac{1}{2} - \delta)^2\}. \tag{34}$$

The above facts indicate that the estimator $\hat{\chi}_1^{(e)}$, ..., $\hat{\chi}_{\hat{n}}^{(e)}$, minimizing the number of inconsistencies with comparisons, guarantees the errorless estimate for $N \to \infty$. The inequality (34) shows that the errorless estimate can be obtained with the probability close to one for finite N and indicates the influence of δ and N on the precision of the estimator.

The properties of the median estimator are similar, especially (see Klukowski, 1994):

$$P(W_{bN}^{(p,me)*} < \tilde{W}_{bN}^{(me,p)}) \geqslant 1 - 2\exp\{-2N(\frac{1}{2} - \delta)^2\}.$$

The case of multivalent comparisons, can be analyzed in a similar way the details are presented in Klukowski 2011a, Chap. 6 and 8.

4 Solving of Optimization Problems

Minimization of the functions (19), (21) is, in general, not an easy problem, because of the dimensions of the feasible set. Currently, the algorithms are available only for ranking problems based on binary single comparisons (see David, 1988, Chapt. 2, Hansen P., et al 1994); they refer to the dynamic programming or branch-and-bound algorithms, some of them can be used for known n . The algorithms are efficient for the moderate number of elements m. In the case of large m, the problems can be also solved with the use of heuristic algorithms: genetic (Falkenauer, 1998), artificial neural networks, random search (Ripley, 2006), etc.

In the case of multivalent comparisons the exact algorithms are not available now. The problems with moderate number of elements m, i.e. 3 – 12, can be solved with the use of complete enumeration. Problems with higher number of elements can be solved using heuristic algorithms, mentioned above.

It is obvious that the estimators based on multivalent comparisons require more computations than those based on binary comparisons.

5 Summary – Achievements of the Work and Further Researches

The work presents the synthesis of main results, of the author (Klukowski 2011a), concerning estimation of three relations – equivalence, tolerance, and preference – on the basis of pairwise comparisons with random errors (see Klukowski in References). The problems of that type occur often in applications and have been investigated in literature.

The following new results, presented here, should be emphasized.

1^0. Two types of data have been taken into account: binary and multivalent.

2^0. The assumptions concerning the comparison errors are weaker than those commonly used in the literature.

3^0. Two estimators have been examined; the first one is based on the sum of differences between the relation form and the comparison data, the second is based on differences between the relation form and the median from comparisons of each pair. The estimators have analytical properties guaranteeing good efficiency.

4^0. The analytical properties of the estimators have been complemented with the results of simulation study (Klukowski 2011a). This allows for determining of

parameters, especially the number of comparisons N, guaranteeing the required precision of estimates; a definite value of N provides for the frequency of errorless result close to one or equal one.

5^0. The properties of estimates can be thoroughly validated (Klukowski 2011a, Chapt. 10); validation comprises the fact of existence of the relation and the assumptions as to the comparison errors.

6^0. The approach proposed allows for combining of comparisons obtained from different sources, e.g. statistical tests, experts, neural networks. It is also possible to combine binary and multivalent data and to apply two-stage estimators, based, in the first stage, on binary comparisons, and in the second stage – on multivalent comparisons, obtained in the first stage.

7^0. The approach presented will be developed in the following directions: statistical learning, estimation of more complex structures of data, e.g. hierarchical, multiple criteria comparisons, etc. An important field is also constituted by application of the estimators and tests developed.

References

1. Bradley, R.A.: Science, statistics and paired comparisons. Biometrics 32, 213–232 (1976)
2. Bradley, R.A.: Paired comparisons: some basic procedures and examples. In: Krishnaiah, P.R., Se, P.K. (eds.) Handbook of Statistics, vol. 4, pp. 299–326. North-Holland, Amsterdam (1984)
3. Brunk, H.D.: Mathematical models for ranking from paired comparisons. JASA 55, 503–520 (1960)
4. David, H.A.: Order Statistics. J.Wiley (1970)
5. David, H.A.: The Method of Paired Comparisons, 2nd edn. Ch. Griffin, London (1988)
6. Davidson, R.R.: A bibliography on the Method of Paired Comparisons. Biometrics 32, 241–251 (1976)
7. Falkenauer, E.: Genetics Algorithms and Grouping Data. J. Wiley (1998)
8. Gordon, A.D.: Classification, 2nd edn. Chapman&Hall/CRC (1999)
9. Hand, D.J.: Discrimination and Classification. J. Wiley (1986)
10. Hartigan, J.J.: Clustering Algorithms. J. Wiley (1975)
11. Hansen, P., Jaumard, B., Sanlaville, E.: Partitioning Problems in Cluster Analysis: A Review of Mathematical Programming Approaches. In: Studies in Classification, Data Analysis, And Knowledge Organization. Springer (1994)
12. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning. Data Mining, Inference and Prediction. Springer (2002)
13. Hoeffding, W.: Probability inequalities for sums of bounded random variables. JASA 58, 13–30 (1963)
14. Kaufman, L., Rousseeuw, P.J.: Findings Groups In Data: An Introduction to Cluster Analysis. J. Wiley (1990)
15. Klukowski, L.: Algorithm for classification of samples in the case of unknown number variables generating them. Przeglad Statystyczny XXXVII, 167–177 (1990a) (in Polish)
16. Klukowski, L.: Ranking of items on the basis of pairwise comparisons in the case of random errors. In: Kulikowski, R., Kacprzyk, J. (eds.) SYSTEMS RESEARCH, vol. 3, pp. 212–268. Omnitech Press, Warsaw (1990b) (in Polish)

17. Klukowski, L.: Some probabilistic properties of the nearest adjoining order method and its extensions. Annals of Operational Research 51, 241–261 (1994)
18. Klukowski, L.: The nearest adjoining order method for pairwise comparisons in the form of difference of ranks. Annals of Operations Research 97, 357–378 (2000)
19. Klukowski, L.: Estimation of the Preference Relation on the Basis of Medians from Pairwise Comparisons in the Form of Difference of Ranks. In: Kurzynski, et al. (eds.) Computer Recognition Systems 2. AISC, vol. 45, pp. 232–241. Springer, Heidelberg (2007a)
20. Klukowski, L.: Estimation of tolerance relation the basis of multiple pairwise comparisons with random errors. Control and Cybernetics 36, 443–466 (2007b)
21. Klukowski, L.: Estimation of the preference relation the basis of multiple pairwise comparisons in the form of differences of ranks. Control and Cybernetics 37, 711–729 (2008)
22. Klukowski, L.: Methods of Estimation of Relations of: Equivalence, Tolerance, and Preference in a Finite Set, Warsaw. IBS PAN, Series: Systems Research, vol. 69 (2011a)
23. Klukowski, L.: Estimation of Tolerance Relation on the Basis of Pairwise Comparisons. In: Burduk, R., et al. (eds.) Computer Recognition Systems 4. AISC, vol. 95, pp. 232–241. Springer, Heidelberg (2011b)
24. Kohonen, T.: Self-Organizing Maps. Springer (1995)
25. Ripley, B.D.: Stochastic Simulation. J. Wiley (2006)
26. Slater, P.: Inconsistencies in a schedule of paired comparisons. Biometrika 48, 303–312 (1961)

Generalized Constraint Design of Linear-Phase FIR Digital Filters

Norbert Henzel and Jacek M. Leski

Abstract. We consider the design of digital finite-impulse response (FIR) filters satisfying constraints on the amplitude response. Constrained FIR filter design with frequency-domain linear constraints on the amplitude response usually uses the least-squares or the Chebyshev error criterion, which can be generally reformulated as quadratic programming (QP) problem. This paper presents an novel algorithm for the design of constrained low-pass FIR filters according to a variously defined error. This approach does not require the transition bands specification, is characterized by rapid convergence and is suitable for high order filter design.

Keywords: Biomedical Signal Processing, Digital Filter Design, FIR Filters.

1 Introduction

Linear-phase finite impulse response (FIR) digital filters play a crucial role in a large number of signal processing problems, for example, biomedical signal processing, image processing, telecommunication application, etc. Therefore, linear-phase filters design methods have been widely explored [1].

The FIR filter design, in majority of cases, can be regarded as an optimization problem, where a desired, ideal frequency response is approximated.

The FIR filter design process typically require several steps. First step consists in defining a desired, ideal, frequency response. Second step demands selecting the

Norbert Henzel
Institute of Electronics, Silesian University of Technology, 44-101 Gliwice,
Akademicka 16, Poland, Institute of Medical Technology and Equipment,
41-800 Zabrze, Roosevelta 118, Poland
e-mail: nhenzel@polsl.pl

Jacek M. Leski
Institute of Electronics, Silesian University of Technology, 44-101 Gliwice,
Akademicka 16, Poland
e-mail: jleski@polsl.pl

R. Burduk et al. (Eds.): *CORES 2013*, AISC 226, pp. 53–62.
DOI: 10.1007/978-3-319-00969-8_5 © Springer International Publishing Switzerland 2013

desired filter length. Third, establishing a measure of error between desired and obtained filter frequency responses (different optimality criteria result in different filter behavior). Fourth step consist in applying (or developing) an optimization method to find the filter coefficients. In scientific literature a vast number of different techniques for designing digital FIR filters have been proposed and the majority of them uses one (or a combination) of the following error criteria [2], [4]: least-squares ([5], [15], [17]), Chebyshev (minimax) ([10], [12]) and maximally flat ([16]).

The constrained design of FIR filters using the least squares error function for the first time was presented in [2]. This approach to FIR filter design has also been next considered and further developed in a vast number of papers, e.g. [3], [6], [13], [14].

The goal of this paper is to present a new method of constrained FIR filter design and to investigate its performances for different design problems. Although presented in the context of low-pass filter design it can be easily extended to other FIR filter design problems.

2 Preliminaries

A digital filter is a linear time-invariant system, operating on an input sequence $x(n)$ to produce an output sequence $y(n)$, where n denotes discrete time. This system can be completely described by the impulse response sequence $h(n)$. The input-output relation for digital filter is given by [11], [8]

$$y(k) = \sum_{m=-\infty}^{\infty} x(m)h(k-m) = \sum_{m=-\infty}^{\infty} x(k-m)h(m). \tag{1}$$

Typically, $h(m) = 0$ for $0 > m > N-1$, so we obtain

$$y(k) = \sum_{m=0}^{N-1} x(k-m)h(m). \tag{2}$$

The number of impulse response coefficients, N, is said to be the length of the filter, and the quantity $N-1$ is called the order of the filter [1].

The frequency response $H\left(e^{j\omega}\right)$ of an FIR filter is given by the discrete-time Fourier transform of its impulse response $h(n)$ [1]:

$$H\left(e^{j\omega}\right) = \sum_{n=0}^{N-1} h\left(n\right) e^{-j\omega n} \tag{3}$$

where the frequency $\omega \in [0, \pi]$.

If the impulse response $h(n)$ of the FIR filter has even symmetry, $h(n) = h(N-1-n)$, or odd symmetry, $h(n) = -h(N-1-n)$, the phase response of the designed filter is linear and the obtained design problem is real-valued. In this case, the frequency response function $H\left(e^{j\omega}\right)$ can be written as [1]

$$H\left(e^{j\omega}\right) = e^{-j(N-1)/2\omega} e^{-j\beta} H_0(\omega) \tag{4}$$

where $H_0(\omega)$ is a real-valued function, called amplitude response and the constant β satisfies $\beta = 0$ or $\beta = \pi/2$. In the first case, $\beta = 0$, the filter amplitude response is given by [1]

$$H_0(\omega) = \begin{cases} \sum_{n=0}^{(N-1)/2} b_n \cos(\omega n) & \text{for } N-1 \text{ even}, \\ \sum_{n=0}^{N/2} b_n \cos\left(\omega\left(n-\frac{1}{2}\right)\right) & \text{for } N-1 \text{ odd}. \end{cases} \tag{5}$$

where the coefficients b_n are related to $h(n)$ in as follows:

$$b_n = \begin{cases} h\left(\frac{N-1}{2}\right) & \text{for } N-1 \text{ even}, n = 0, \\ 2h\left(\frac{N-1}{2} - n\right) & \text{for } N-1 \text{ even}, n \neq 0, \\ 2h\left(\frac{N}{2} - n\right) & \text{for } N-1 \text{ odd}. \end{cases} \tag{6}$$

Similar expressions can be developed for $\beta = \pi/2$ [1].

The linear phase response of a FIR filter is a very desirable property in many applications, e.g. processing of an electrocardiographic (ECG) signals, acoustics signals, etc.

Low-pass FIR digital filters are characterized by: the length of the impulse response N, the passband edge frequency f_p, the stopband edge frequency f_s, the maximum passband ripple (maximum passband gain) δ_p and minimum stopband attenuation (maximum stopband gain) δ_s. The last two values are often expressed in decibels:

$$D_p = 20\log_{10}(\delta_p)\ [dB], \quad D_s = 20\log_{10}(\delta_s)\ [dB]. \tag{7}$$

Figure 1 depicts the filter parameters and performance measures discussed so far.

The relation between the linear-phase FIR filter amplitude response $H_0(\omega)$ for $\beta = 0$ and $N-1$ even, and the coefficients b_n for a given set of frequency points ω_i, $i = 1, \cdots, L$, distributed over the frequency domain can be compactly represented in matrix form. For example, the first case in (5) can be written as

$$\mathbf{H}_0 \triangleq [H_0(\omega_1), H_0(\omega_2), \cdots, H_0(\omega_L)]^\top = \mathbf{T}\mathbf{b}, \tag{8}$$

where

$$\mathbf{b} = [b_0,\ b_1,\ \cdots, b_M]^\top; \quad M = (N-1)/2, \tag{9}$$

$$H_0(\omega_i) = \mathbf{b}^\top \mathbf{t}(\omega_i), \tag{10}$$

$$\mathbf{t}(\omega_i) = [\cos(0\omega_i), \cos(1\omega_i), \cdots, \cos(M\omega_i)]^\top, \tag{11}$$

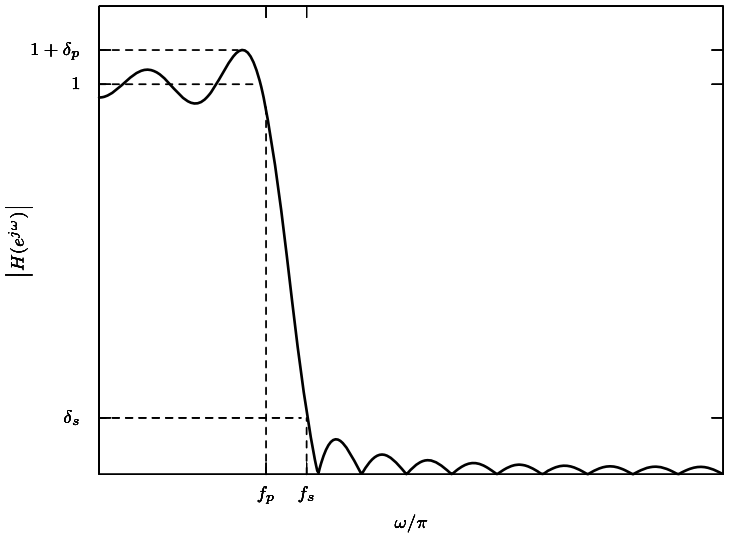


Fig. 1 Filter parameters and performance measures

$$J(\mathbf{b}) = \sum_{i=1}^{L} \left[\mathbf{b}^\top \mathbf{t}(\omega_i) - H_D(\omega_i)\right]^2 \tag{12}$$

and

$$\mathbf{T} = \begin{bmatrix} \cos(0\cdot\omega_1) & \cos(\omega_1) & \cdots & \cos(M\cdot\omega_1) \\ \cos(0\cdot\omega_2) & \cos(\omega_2) & \cdots & \cos(M\cdot\omega_2) \\ \vdots & \vdots & \ddots & \vdots \\ \cos(0\cdot\omega_L) & \cos(\omega_L) & \cdots & \cos(M\cdot\omega_L) \end{bmatrix} = \begin{bmatrix} \mathbf{t}(\omega_1)^\top \\ \mathbf{t}(\omega_2)^\top \\ \vdots \\ \mathbf{t}(\omega_L)^\top \end{bmatrix}. \tag{13}$$

Lets us define an error as

$$E(\omega_i) = H_0(\omega_i) - H_D(\omega_i) \tag{14}$$

where $H_0(\omega_i)$ and $H_D(\omega_i)$, $i = 1, \cdots, L$, are the actual and the desired frequency response of the filter, respectively.

Now, the weighted square error criterion can be written as:

$$J(\mathbf{b}) = \|\mathbf{E}\|_2^2 = \mathbf{E}^\top \mathbf{E} = (\mathbf{T}\mathbf{b} - \mathbf{H}_D)^\top (\mathbf{T}\mathbf{b} - \mathbf{H}_D), \tag{15}$$

where

$$\mathbf{H}_D = [H_D(\omega_1),\ H_D(\omega_2),\ \cdots,\ H_D(\omega_L)]^\top \tag{16}$$

is a real-valued desired amplitude response vector.

3 New Method for Low-Pass FIR Filter Design

In the proposed method of FIR filter design various loss functions are used. We seek vector $\mathbf{b}$ by the following minimization

$$\min_{\mathbf{b}\in\mathbb{R}^{M+1}} J(\mathbf{b}) \triangleq \sum_{i=1}^{L} g_i \mathscr{L}\left(\mathbf{b}^\top \mathbf{t}(\omega_i) - H_D(\omega_i)\right), \tag{17}$$

where $\mathscr{L}(\cdot)$ stands for a loss function used, and g_i is a weight corresponding to the ith frequency. If we choose the quadratic loss function then in matrix notation (17) takes the form

$$\min_{\mathbf{b}\in\mathbb{R}^{M+1}} J(\mathbf{b}) \triangleq (\mathbf{T}\mathbf{b} - \mathbf{H}_D)^\top \mathbf{G}(\mathbf{T}\mathbf{b} - \mathbf{H}_D), \tag{18}$$

where $\mathbf{G} = \operatorname{diag}(g_1, g_2, \cdots, g_L)$. The role of g_is parameters may be twofold: (i) they may correspond to our weight of the ith frequency (${}^c g_i \in [0,1]$), (ii) through the proper selection of the parameters values we may change various error functions to the quadratic loss (${}^l g_i \in \mathbb{R}^+ \cup \{0\}$). In the last case, the values of the parameters depend on the obtained residuals. In turn, the residuals depend on $\mathbf{b}$. Thus, criterion function (18) should only be minimized by iteratively reweighting scenario. Let us denote $\mathbf{b}$, $\mathbf{G}$ and $\mathbf{E}$ in the kth iteration as $\mathbf{b}^{(k)}$, $\mathbf{G}^{(k)}$ and $\mathbf{E}^{(k)}$, respectively. Criterion function (18) for the kth iteration takes the form

$$J^{(k)}\left(\mathbf{b}^{(k)}\right) \triangleq \left(\mathbf{T}\mathbf{b}^{(k)} - \mathbf{H}_D\right)^\top \mathbf{G}^{(k)} \left(\mathbf{T}\mathbf{b}^{(k)} - \mathbf{H}_D\right), \tag{19}$$

where the elements on the main diagonal of $\mathbf{G}^{(k)} = \operatorname{diag}\left(g_1^{(k)}, g_2^{(k)}, \cdots, g_N^{(k)}\right)$ depend on the residuals from the previous iteration

$$\mathbf{E}^{(k-1)} = \mathbf{T}\mathbf{b}^{(k-1)} - \mathbf{H}_D. \tag{20}$$

and take the form

$$g_i^{(k)} = {}^c g_i \cdot {}^l g_i^{(k)}. \tag{21}$$

Parameter ${}^c g_i$, representing *a priori* confidence to the ith frequency does not depend on the iteration index k. In contrast, parameter ${}^l g_i^{(k)}$ depends on the ith residual from the previous iteration, $(k-1)$th. The following form of ${}^l g_i^{(k)}$ is proposed

$${}^l g_i^{(k)} = \begin{cases} 0, & E(\omega_i)^{(k-1)} = 0, \\ \mathscr{L}\left(E(\omega_i)^{(k-1)}\right) \Big/ \left(E(\omega_i)^{(k-1)}\right)^2, & E(\omega_i)^{(k-1)} \neq 0. \end{cases} \tag{22}$$

Indeed, for the quadratic loss function, we obtain ${}^l g_i^{(k)} = 1$, for all $i = 1, 2, \cdots, L$; $k = 1, 2, 3, \cdots$. The absolute error function is easy to obtain by taking [7]

$$ {}^{l}g_i^{(k)} = \begin{cases} 0, & E(\omega_i)^{(k-1)} = 0, \\ 1 \Big/ \left| E(\omega_i)^{(k-1)} \right|, & E(\omega_i)^{(k-1)} \neq 0. \end{cases} \tag{23} $$

Many other loss functions easily may be obtained (see [7]).

To start this sequential optimization, we set the weights in the 0th iteration as ${}^{l}g_i^{(0)} = 1$ for all i.

The ε-insensitive loss function disregards errors below some $\varepsilon > 0$, chosen a priori:

$$ \mathscr{L}(\zeta) = \begin{cases} 0, & |\zeta| \leq \varepsilon, \\ |\zeta| - \varepsilon, & |\zeta| > \varepsilon. \end{cases} \tag{24} $$

Various ε-insensitive loss functions may be considered, including ε-insensitive quadratic, ε-insensitive Huber, and so on. Let us start our consideration from the ε-insensitive quadratic loss

$$ \mathscr{L}(\zeta) = \begin{cases} 0, & |\zeta| - \varepsilon \leq 0, \\ (\varepsilon - \zeta)^2, & \varepsilon - \zeta < 0, \\ (\varepsilon + \zeta)^2, & \varepsilon + \zeta < 0. \end{cases} \tag{25} $$

Taking into account the above equation (17), assuming $g_i = 1$ for all $i = 1, 2, \cdots, L$, may be written as

$$ \sum_{i=1}^{L} \mathscr{L}\left(\mathbf{b}^\top \mathbf{t}(\omega_i) - H_D(\omega_i) \right) = \sum_{i=1}^{L} g_i^+ \left(-\mathbf{b}^\top \mathbf{t}(\omega_i) + H_D(\omega_i) + \varepsilon^+(\omega_i) \right)^2 + \sum_{i=1}^{L} g_i^- \left(\mathbf{b}^\top \mathbf{t}(\omega_i) - H_D(\omega_i) + \varepsilon^-(\omega_i) \right)^2, \tag{26} $$

where g_i^+ (g_i^-) are equal to zero for $-\mathbf{b}^\top \mathbf{t}(\omega_i) + H_D(\omega_i) + \varepsilon^+(\omega_i) \geq 0$ ($\mathbf{b}^\top \mathbf{t}(\omega_i) - H_D(\omega_i) + \varepsilon^-(\omega_i) \geq 0$) and 1 otherwise. Thus, the ε-insensitive quadratic loss function may be decomposed into two asymmetric quadratic loss functions. Let $\mathbf{T}_e$ be the $2L \times (M+1)$ matrix

$$ \mathbf{T}_e^\top \triangleq \left[\mathbf{T}^\top, -\mathbf{T}^\top \right] \tag{27} $$

and $\mathbf{H}_{De}$ be the $2L$-dimensional vector $\mathbf{H}_{De}^\top = \left[\mathbf{H}_D^\top - \boldsymbol{\varepsilon}^+, -\mathbf{H}_D^\top - \boldsymbol{\varepsilon}^- \right]$ and $\boldsymbol{\varepsilon}^+ = [\varepsilon^+(\omega_1), \varepsilon^+(\omega_2), \cdots, \varepsilon^+(\omega_L),]^\top$, $\boldsymbol{\varepsilon}^- = [\varepsilon^-(\omega_1), \varepsilon^-(\omega_2), \cdots, \varepsilon^-(\omega_L),]^\top$. Using the above mentioned notation, criterion function (19) for kth iteration takes the form

$$ J^{(k)}\left(\mathbf{b}^{(k)} \right) \triangleq \left(\mathbf{T}_e \mathbf{b}^{(k)} - \mathbf{H}_{De} \right)^\top \mathbf{G}^{(k)} \left(\mathbf{T}_e \mathbf{b}^{(k)} - \mathbf{H}_{De} \right), \tag{28} $$

where the elements on the main diagonal of $\mathbf{G}^{(k)}$ (now, $(2L) \times (2L)$ matrix) depend on residuals from the previous iteration

$$ \mathbf{E}^{(k-1)} = \mathbf{T}_e \mathbf{b}^{(k-1)} - \mathbf{H}_{De}. \tag{29} $$

The fitting of the ith frequency is represented by the ith and the $(i+L)$th element of $\mathbf{E}$. If both $E(\omega_i)^{(k)}$ and $E(\omega_{i+L})^{(k)}$ are greater than or equal to zero, then characteristics for the ith frequency falls, for the kth iteration, into the insensibility zone, i.e., $\mathbf{H}_D - \boldsymbol{\varepsilon}^- \preceq \mathbf{H}_0 \preceq \mathbf{H}_D + \boldsymbol{\varepsilon}^+$, where the symbol $\preceq$ stands for componentwise inequality. If $E(\omega_i)^{(k)}$ ($E(\omega_{i+L})^{(k)}$) is less than zero, then the ith frequency is below (above) the insensibility zone in the kth iteration and should be penalized. For the ε-insensitive quadratic (εSQ) loss we have

$$ {}^l g_i^{(k)} = \begin{cases} 0, E(\omega_i)^{(k-1)} \geq 0, \\ 1, E(\omega_i)^{(k-1)} < 0. \end{cases} \tag{30} $$

Other ε-insensitive loss functions easily may be obtained, for example [7]:

- HUBer (εHUB) with parameter $\delta > 0$

$$ {}^l g_i^{(k)} = \begin{cases} 0, & e_i^{(k-1)} \geq 0, \\ 1/\delta^2, & 0 > e_i^{(k-1)} \geq -\delta, \\ -1 \Big/ \left(\delta \left| e_i^{(k-1)} \right|\right), & e_i^{(k-1)} < -\delta. \end{cases} \tag{31} $$

- SIGmoidal (εSIG) with parameters $\alpha, \beta > 0$

$$ {}^l g_i^{(k)} = \begin{cases} 0, & e_i^{(k-1)} \geq 0, \\ 1 \Big/ \left(\left(e_i^{(k-1)}\right)^2 \left(1 + \exp\left(\alpha \left(e_i^{(k-1)} + \beta\right)\right)\right)\right), & e_i^{(k-1)} < 0. \end{cases} \tag{32} $$

Our *a priori* confidence to the ith datum (${}^c g_i \in [0,1]$) should be 'doubled', i.e., ${}^c g_{i+L} = {}^c g_i$, for $i = 1, 2, \cdots, L$, because every frequency in criterion function (28) is also doubled.

The optimality condition for the kth iteration is obtained by differentiating (19) with respect to $\mathbf{b}$ and setting the result equals to zero

$$ \left(\mathbf{T}_e^\top \mathbf{G}^{(k)} \mathbf{T}_e\right) \mathbf{b}^{(k)} = \mathbf{T}_e^\top \mathbf{G}^{(k)} \mathbf{H}_{De}. \tag{33} $$

The procedure of Iteratively Reweighted least square error minimization for Constrained Filter Design (IRCFD) can be summarized in the following steps [7]:

4 Design Examples

The proposed method has been used to solve many filter design problems. The obtained results show that the proposed method is competitive when compared with results obtained by other methods. In this section, we only present the results obtained for three examples. The computation was performed in MatLab® computing environment. Two other methods were used to compare the results of the proposed method: the method proposed by McClellan, Parks and Rabiner in [9] (MPR) and

the method presented by Selesnick, Lang and Burrus in [13] (SLB). It should be noted that the MPR method is not a constraint design method and the resulting pass-band ripples and stopband attenuation depends on the inclination of the transition band (part of the filter frequency characteristics between f_p and f_s, see Figure 1).

1: Fix $\tau > 0$ and $\mathbf{G}^{(0)} = \mathbf{I}$. Set the iteration index $k = 0$.
2: Obtain $\mathbf{b}^{(k)}$ by using conjugate gradient method to (33).
3: $\mathbf{E}^{(k)} = \mathbf{T}_e \mathbf{b}^{(k)} - \mathbf{H}_{De}$.
4: $\mathbf{G}^{(k+1)} = \text{diag}\left(g_1^{(k+1)}, g_2^{(k+1)}, \cdots, g_N^{(k+1)}\right)$, where each $g_i^{(k+1)}$, for $i = 1, 2, \cdots, L$ is obtained by (21) and depends on the selected loss function.
5: **if** $k > 1$ and $\left\|\mathbf{b}^{(k)} - \mathbf{b}^{(k-1)}\right\|_2 < 10^{-3}$ **then**
6: stop
7: **else**
8: $k \leftarrow k + 1$
9: goto 2.
10: **end if**

Example 1: We first consider the low-pass filter described in [13]: $\omega_p = 0.2728\pi$, $\omega_s = 0.3270\pi$, $\delta_p = \delta_s = 0.02$ and $N = 61$. Figure 2(a) shows the frequency responses of the obtained filter using the εSQ error function. A comparison with the two selected methods is given in Table 1(a).

Table 1 Parameters of the frequency response for *Example 1* (a) and *Example 2* (b) low-pass filters

	(a)			(b)		
	MPR	SLB	IRCFD(εSQ)	MPR	SLB	IRCFD(εSQ)
E_p	0.0198	0.0039	0.0041	0.1142	0.1079	0.0136
δ_p^{max}	0.0168	0.0143	0.0134	0.0770	0.0574	0.0287
D_s^{min} [dB]	−35.4	−28.0	−28.3	−22.2	−15.7	−14.5

The proposed method has the smallest maximum ripple (δ_p^{max}) and the second smallest error in the pass band (E_p). The minimum attenuation in stopband (D_s^{min}) is comparable with the error of the SLB method; the best D_s^{min} is obtained with the MPR method.

Example 2: Next we consider the narrow low-pass filter described in [2]: $\omega_p = 0.0625\pi$, $\omega_s = 0.0804\pi$, $\delta_p = 0.0575$, $\delta_s = 0.006$ dB and $N = 95$. Figure 2(b) shows the frequency responses of the obtained filter using the εSQ error function. A comparison with the two selected methods is given in Table 1(b).

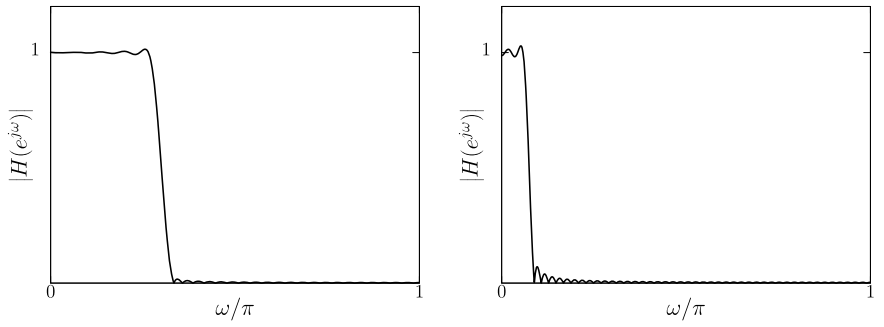


Fig. 2 Frequency responses for *Example 1* (a) and *Example 2* (b) low-pass filters

In this case the proposed method has the smallest maximum ripple (δ_p^{max}) and the smallest error in the passband (E_p). The minimum attenuation in stopband (D_s^{min}) is comparable with this of the SLB method; the best D_s^{min} is given once again by the MPR method.

Example 3: In the last example we consider another narrow low-pass filter described in [2]: $\omega_p = 0.021\pi$, $\omega_s = 0.03125\pi$, $\delta_p = 0.0575$, $\delta_s = 0.0334$ and $N = 128$. A comparison with the two selected methods is given in Table 2.

Table 2 Parameters of the frequency response for *Example 3* low-pass filter

	MPR	SLB	IRCFD(εSIG)	IRCFD(εHUB)
E_p	0.0751	0.0238	0.0091	0.0001
δ_p^{max}	0.1172	0.0421	0.0460	0.0074
D_s^{min} [dB]	-18.6	-12.7	-7.8	-3.8

The smallest maximum ripple (δ_p^{max}) and the smallest error in the passband (E_p) are obtained with the IRCFD(εHUB) method at the expense of (D_s^{min}). The (δ_p^{max}) of SLB and IRCFD(εSIG) methods are comparable but the E_p of IRCFD(εSIG) is much better. The best D_s^{min} is obtained for the MPR method. Once again it is observed that the smaller E_p or (δ_p^{max}), the smaller D_s^{min}.

5 Conclusion

In this paper we developed a FIR filter design method based on a discrete desired filter frequency response and the procedure of Iteratively Reweighted least square

error minimization for Constrained Filter Design (IRCFD). From our included examples it could be concluded that the proposed method is efficient and competitive with respect to other well-known from scientific literature methods. Additionally, the proposed method offers the possibility to select different error measures, appropriate for the design problem.

References

1. Parks, T.W., Burrus, C.S.: Digital Filter Design. Wiley, New York (1987)
2. Adams, J.W.: FIR Digital Filters with Least-Squares Stopbands Subject to Peak-Gain Constraints. IEEE Transactions on Circuits and Systems 39, 376–388 (1991)
3. Adams, J.W., Sullivan, J.L.: Peak-constrained least squares optimization. IEEE Transactions on Signal Processing 46, 306–321 (1998)
4. Burrus, C.S., Barreto, A.: Least p-power error design of filters. In: Proc. IEEE Int. Symp. Circuits. Syst (ISCAS), vol. 1, pp. 545–548 (1992)
5. Burrus, C.S., Soewito, A.W., Gopinath, R.A.: Least squared error FIR filter design with transition bands. IEEE Trans. Signal Process. 40, 1327–1340 (1992)
6. Lang, M., Selesnick, I.W., Burrus, C.S.: Constrained LS design of 2D FIR filters. IEEE Transactions on Signal Processing 44, 1234–1241 (1996)
7. Leski, J., Henzel, N.: Generalized Ordered Linear Regression with Regularization. Bull. Pol. Ac.: Tech. (in print)
8. McClellan, J., Parks, T.: A united approach to the design of optimum FIR linear-phase digital filters. IEEE Transactions on Circuits and Systems 20, 679–701 (1973)
9. McClellan, J., Parks, T., Rabiner, L.: A computer program for designing optimum FIR linear phase digital filters. IEEE Transactions on Audio and Electroacoustics 21, 506–526 (1973)
10. McClellan, J., Parks, T.: A unified approach to the design of optimum FIR linear-phase digital filter. IEEE Trans. Circuit Theory 20, 697–701 (1973)
11. Parks, T., McClellan, J.: A program for the design of linear phase finite impulse response digital filters. IEEE Transactions on Audio and Electroacoustics 20, 195–199 (1972)
12. Rabiner, L.R., McClellan, J.H., Parks, T.W.: FIR digital filter design techniques using weighted Chebyshev approximations. Proc. IEEE 63, 595–610 (1975)
13. Selesnick, I.W., Lang, M., Burrus, C.S.: Constrained least square design of FIR filters without specified transition bands. IEEE Transactions on Signal Processing 44, 1879–1892 (1996)
14. Selesnick, I.W., Lang, M., Burrus, C.S.: A modified algorithm for constrained least square design of multiband FIR filters without specified transition bands. IEEE Transactions on Signal Processing 46, 497–501 (1998)
15. Tufts, D.W., Francis, J.T.: Designing digital low-pass filters- comparison of some methods and criteria. IEEE Trans. Audio Electroacoust. 18, 487–494
16. Vaidyanathan, P.P.: Efficient and multipliers design of FIR filters with very sharp cutoff via maximally flat building blocks. IEEE Trans. Circuits Syst. 32, 236–244
17. Vaidyanathan, P.P., Nguyen, T.Q.: Eigenfilters: A new approach to least squares FIR filter design and applications including Nyquist filters. IEEE Trans. Circuits Syst. 34, 11–23 (1987)

Reduced Kernel Extreme Learning Machine

Wanyu Deng, Qinghua Zheng, and Kai Zhang

Abstract. We present a fast and accurate algorithm–reduced kernel extreme learning machine (Reduced-KELM). It randomly selects a subset from given dataset, and uses $\mathscr{K}(X,\tilde{X})$ in place of $\mathscr{K}(X,X)$. The large scale kernel matrix with size of $n \times n$ is reduced to $n \times \tilde{n}$, and the time-consuming computation for inversion of kernel matrix is reduced to $O(\tilde{n}^3)$ from $O(n^3)$ where $\tilde{n} \ll n$. The experimental results show that Reduced-KELM can perform at a similar level of accuracy as KELM and at the same time being significantly faster than KELM.

1 Introduction

Kernel extreme learning machine (KELM) [1] generalize extreme learning machine [2] from explicit activation function to implicit mapping function. KELM can produce better generalization than ELM in most applications. But like other kernel machines such as support vector machine (SVM) [3], least square support vector machine(LS-SVM) [4], kernel principle component analysis(KPCA) [5]. The entire samples have to be stored in the memory. When the samples are large, the computer may run out of memory. Additionally, it is very time-consuming for kernel matrix operations such as inversion, multiplication and eigenvalue decomposition. In order to accelerate the computation efficiency, many methods have been proposed such as sequential minimal optimization (SMO) [6], Kronecker product decomposition [7], random samples [8] [9], etc. SMO is an iterative algorithm for solving the optimization problem described above. SMO breaks this problem into a series of smallest possible sub-problems, which are then solved analytically. Kronecker decomposition decomposes the large kernel matrix with size into two small matrices with

Wanyu Deng · Kai Zhang
Xi'an University of Posts & Telecommunications, China
e-mail: wanyu.deng@gmail.com, zhangkaial@163.com

Qinghua Zheng
Xi'an Jiaotong University, China
e-mail: qhzheng@mail.xjtu.edu.cn

R. Burduk et al. (Eds.): *CORES 2013*, AISC 226, pp. 63–69.
DOI: 10.1007/978-3-319-00969-8_6 © Springer International Publishing Switzerland 2013

small size, and then run eigenvalue decomposition or other corresponding operation on these two small matrix. The main idea of random sample method is selecting small subset from the entire dataset and finding one hyperspace in this subset. The method was introduced into LS-SVM and produced good performance. Motivated by this work, we will apply this method in KELM and desire to get one fast algorithm Reduced-KELM. The paper is organized as follows. Section 2 gives a brief review of the KELM. Section 3 presents the derivation of Reduced-KELM. Performance evaluation of Reduced-KELM is shown in Section 4 based on the benchmark problems from UCI repository. Conclusions and future works are highlighted in Section 5.

2 Kernel Extreme Learning Machine

Extreme learning machine (ELM) was originally proposed for the single hidden layer feedforward neural networks (SLFNs) where the hidden nodes can be any piecewise nonlinear function including additive/RBF hidden nodes, multiplicative nodes, and non-neural alike nodes [10] [11]. The output function of ELM is

$$f_L(x) = \sum_{i=1}^{L} \beta_i h_i(x) = h(x)\beta \tag{1}$$

where $\beta = [\beta_1, ..., \beta_L]^T$ is the vector of the output weights between the hidden layer and the output layer, and $h(x) = [h_1(x), ..., h_L(x)]$ is the outputs of the hidden nodes with respect to the input x. $h(x)$ actually maps the data from the -dimensional input space to the L-dimensional feature space. The minimal norm least square method was used in the original implementation of ELM[2]:

$$\beta = H^{\dagger}T \tag{2}$$

where $H^{\dagger}$ is the Moore-Penrose generalized inverse of matrix H while H is the hidden layer output matrix:

$$H = \begin{pmatrix} h(x_1) \\ \vdots \\ h(x_n) \end{pmatrix} \tag{3}$$

one of the methods to calculate Moore-Penrose generalized inverse of a matrix is the orthogonal projection method [12]:

$$H^{\dagger} = H^T(HH^T)^{\dagger} \tag{4}$$

According to the ridge regression theory [13], one can add a positive value C to the diagonal of HH^T such that the solution is more stable and tends to produce better generalization performance:

$$f(x) = h\beta = h(x)H^T(\frac{I}{C} + HH^T)^{-1}T \tag{5}$$

Different from SVM, feature mapping $h(x)$ is usually known to users in ELM. However, if a feature mapping $h(x)$ is unknown a kernel matrix for ELM can be defined:

$$\mathscr{K}(X,X) = HH^T \tag{6}$$

Thus, the output function of kernel extreme learning machine (KELM) can be written as

$$f(x) = h(x)H^T(\frac{I}{C} + HH^T)^{-1}T = \begin{pmatrix} \mathscr{K}(x,x_1) \\ \vdots \\ \mathscr{K}(x,x_n) \end{pmatrix}^T \left(\frac{I}{C} + \mathscr{K}(X,X)\right)^{-1} T \tag{7}$$

Interestingly, KELM solution (7) is consistent to LS-SVM solution when the bias b is not used in the constraint conditions. Different from other learning algorithms [14], ELM is to minimize the training error as well as the norm of the output weights. Minimize:$||H\beta - T||^2$and $||\beta||$

3 Reduced Kernel Extreme Learning Machine

The motivation for reduced kernel extreme learning machine (Reduced-KELM) comes from the practical objective of generating a non-linear separating surface for a large dataset which requires a small portion of the dataset for its characterization. The reasons lead to the difficulty in using nonlinear kernels on large datasets lies in twofold: 1) the kernel matrix $\mathscr{K}(X,X)$ of the entire data may be prohibitively expensive in storage and typically leads to the computer running out of memory; 2) the computing time increase dramatically with the size of samples so that it is impractical in real applications. To tackle all these difficulties, we adapt the method of random sample [2, 13] to select a small random subset $\tilde{X} = \{x_i\}_{i=1}^{\tilde{n}}$ from the original $n$data points $X = \{x_i\}_{i=1}^{n}$ with $\tilde{n} \ll n$, and use $\mathscr{K}(X,\tilde{X})$ in place of $\mathscr{K}(X,X)$ to cut problem size and computing time. This method has been applied in LS-SVM and produce good performance. Thus it is desired to be able to be applied in KELM.

Removing the regulator from the formula (7), we can get the kernel representation without regulator:

$$f(x) = \begin{pmatrix} \mathscr{K}(x,x_1) \\ \vdots \\ \mathscr{K}(x,x_N) \end{pmatrix}^T (\mathscr{K}(X,X))^{-1} T \tag{8}$$

Modifying this formulation for the reduced dataset $\tilde{X} \in R^{\tilde{n}\times m}$ with corresponding rectangular kernel matrix ,

$$f(x) = \begin{pmatrix} \mathscr{K}(x,x_1) \\ \vdots \\ \mathscr{K}(x,x_{\tilde{n}}) \end{pmatrix}^T \left(\mathscr{K}(\tilde{X},X)\right)^{-1} T \tag{9}$$

Since $\mathscr{K}(X,\tilde{X}) \in R^{n\times\tilde{n}}$ is a thin matrix, we can calculate Moore-Penrose generalized inverse by the formula $\mathscr{K}(X,\tilde{X})^{-1} = \left(\mathscr{K}(X,\tilde{X})^T \mathscr{K}(X,\tilde{X})\right)^{-1} \mathscr{K}(X,\tilde{X})^T$ to accelerate efficiency:

$$f(x) = \begin{pmatrix} \mathscr{K}(x,x_1) \\ \vdots \\ \mathscr{K}(x,x_{\tilde{n}}) \end{pmatrix}^T \left(\mathscr{K}(X,\tilde{X})^T \mathscr{K}(X,\tilde{X})\right)^{-1} \mathscr{K}(X,\tilde{X})^T T \tag{10}$$

According to the ridge regression theory [15], one can add a positive value to the diagonal of $\mathscr{K}(X,\tilde{X})^T \mathscr{K}(X,\tilde{X})$ such that the solution is more stable and tends to have better generalization performance:

$$f(x) = \begin{pmatrix} \mathscr{K}(x,x_1) \\ \vdots \\ \mathscr{K}(x,x_{\tilde{n}}) \end{pmatrix}^T \left(\frac{I}{C} + \mathscr{K}(X,\tilde{X})^T \mathscr{K}(X,\tilde{X})\right)^{-1} \mathscr{K}(X,\tilde{X})^T T \tag{11}$$

The Reduced-KELM algorithm can be summarized as following:

Algorithm Reduced-KELM Algorithm
Given a training set $\aleph = \{(x_k,t_k)|x_k \in R^m, t_k \in R^\ell\}_{k=1}^n$, an kernel function,
(1) Choose a random subset matrix $\tilde{X} \in R^{\tilde{n}\times m}$; Typically $\tilde{n} \ll n$.
(2) Construct rectangular kernel matrix $\mathscr{K}(X,\tilde{X}) \in R^{n\times\tilde{n}}$;
(3) Get the prediction for one new data x

$$f(x) = \begin{pmatrix} \mathscr{K}(x,x_1) \\ \vdots \\ \mathscr{K}(x,x_{\tilde{n}}) \end{pmatrix}^T \left(\frac{I}{C} + \mathscr{K}(X,\tilde{X})^T \mathscr{K}(X,\tilde{X})\right)^{-1} \mathscr{K}(X,\tilde{X})^T T$$

4 Experimental Results

We evaluate and compare the performance of the proposed Reduced-KELM with ELM and KELM on six classification problems described in Table 1. For each problem, the results are averaged over 50 trials. The average training time, testing time, training accuracy and testing accuracy are reported.

All the simulations have been conducted in MATLAB 12 environment running on an ordinary PC with 2.6 GHZ CPU. The Gaussian RBF activation function has

Table 1 Specification of Classification Benchmark Datasets

Dataset	#Training samples	#Testing samples	#Attributes	#Classes
Segment	1500	810	18	7
Satimage	4435	2000	36	7
Shuttle	43500	14500	9	7
SkinSeg	145057	100000	4	2
Waveform	3000	2000	40	3
Madelon	1000	1600	500	2

Table 2 Parameters of Reduced-KELM, ELM, and KELM

Dataset	Reduced-KELM$(\tilde{n}, C, \sigma)$	ELM(#Nodes)	KELM(C, σ)
Segment	$(400, 2^5, 2^{-1})$	200	$(2^6, 2^{-1})$
Satimage	$(700, 2^9, 2^1)$	500	$(2^4, 2^{-2})$
Shuttle	$(50, 2^10, 2^{-1})$	50	–
SkinSeg	$(300, 2^10, 2^{-3})$	300	–
Waveform	$(300, 2^0, 2^0)$	400	$(2^6, 2^{-1})$
Madelon	$(400, 2^2, 2^1)$	400	$(2^2, 2^1)$

Table 3 The Training Time and Testing Time of Reduced-KELM, ELM, and KELM

Dataset	Reduced-KELM (Training,Testing)		ELM (Training, Testing)		KELM (Training, Testing)	
Segment	0.065	0.011	0.2390	0.012	0.4673	0.043
Satimage	0.386	0.287	3.1044	0.115	17.4850	0.046
Shuttle	0.096	0.020	0.4867	0.0338	—	—
SkinSeg	2.963	0.825	27.9034	1.674	—	—
Waveform	0.082	0.023	4.676	0.0988	1.289	0.199
Madelon	0.079	0.053	0.665	0.2080	0.189	0.130

been used in the simulations of Reduced-KELM and KELM, while sigmoid function is used in ELM. The input attributes are normalized into the range [-1, 1]. Table 2 summarizes the parameter settings, while Table 3 and 4 report the results for the classification problems for each algorithm. As observed from Table 3 and 4, the training time taken by Reduced-KELM is much less than ELM and KELM while testing accuracy is better than ELM and slightly less than KELM. Take Satimage dataset as example, the training time for Reduced-KELM is 0.3867s, ELM is 3.1044s while KELM is 17.485s. Reduced-KELM is 9 times faster than ELM while 50 times faster than KELM. For the dataset Shuttle and Skin Segmentation, since the training samples are too large, the KELM will run out of memory. This shows

Table 4 The Training Accuracy and Testing Accuracy of Reduced-KELM, ELM, and KELM for Classification Problems

Dataset	Reduced-KELM (Training,Testing)		ELM (Training, Testing)		KELM (Training, Testing)	
Segment	0.974	0.959	0.975	0.939	0.998	
Satimage	0.935	0.911	0.925	0.896	1.0000	
Shuttle	0.988	0.988	0.979	0.979	—	—
SkinSeg	0.998	0.9981	0.997	0.997	—	—
Waveform	0.867	0.85	0.895	0.838	1.00	0.85
Madelon	0.742	0.57	0.806	0.531	1.00	0.596

that Although KELM can produce the best generalization ability, it is unsuitable for large size problems.

5 Conclusion and Future Work

In this paper, a fast and accurate reduced kernel extreme learning machine (Reduced-KELM) has been developed. It uses a randomly selected subset of the data (typically much less than the original dataset) to obtain a nonlinear separating surface. Although Reduced-KELMąŕs testing accuracy is slightly less than KELM that uses the entire data, it is tens of times faster than KELM. This is very important for massive datasets such as those in the millions scale. Reduced-KELM appears to be a very promising method for handling large problems. Additionally, Reduced-KELM is suitable to implement online sequential learning for data stream. There are some open problems for the current work such as 1) does random selection affect the prediction performance, 2) how to select the optimized subset and 3) what size the subset should be. These are our next work in the future.

References

1. Huang, G.-B., Zhou, H., Ding, X., Zhang, R.: Extreme Learning Machine for Regression and Multiclass Classification. IEEE Transactions on Systems, Man, and Cybernetics, Part B 42(2), 513–529 (2012)
2. Huang, G.-B., Zhu, Q.-Y., Siew, C.-K.: Extreme learning machine: Theory and applications. Neurocomputing 70(1-3), 489–501 (2006)
3. Vapnik, V.: The Nature of Statistical Learning Theory. Springer, Heidelberg (1999)
4. Suykens, J.A.K., Vandewalle, J.: Least squares support vector machine classifiers. Neural Processing Letters 9(3), 293–300 (1999)

5. Scholkopf, B., Smola, A., Mller, K.R.: Kernel principal component analysis. In: Gerstner, W., Hasler, M., Germond, A., Nicoud, J.-D. (eds.) ICANN 1997. LNCS, vol. 1327, pp. 583–588. Springer, Heidelberg (1997)
6. Keerthi, S.S., Shevade, S.K., Bhattacharyya, C., Murthy, K.R.K.: Improvements to Platt's SMO algorithm for SVM classifier design. Neural Computation 13, 637–649 (2001)
7. Wu, G., Zhang, Z., Chang, E.Y.: Kronecker factorization for speeding up kernel machines. In: SIAM Int. Conference on Data Mining (SDM), pp. 611–615 (2005)
8. Lin, K.M., Lin, C.J.: A study on reduced support vector machines. IEEE Transactions on Neural Networks 14(6), 1449–1459 (2003)
9. Liu, Q., He, Q., Shi, Z.-Z.: Extreme support vector machine classifier. In: Washio, T., Suzuki, E., Ting, K.M., Inokuchi, A. (eds.) PAKDD 2008. LNCS (LNAI), vol. 5012, pp. 222–233. Springer, Heidelberg (2008)
10. Huang, G.B., Chen, L.: Enhanced random search based incremental extreme learning machine. Neurocomputing 71(16), 3460–3468 (2008)
11. Huang, G.B., Chen, L.: incremental extreme learning machine. Neurocomputing 70(16), 3056–3062 (2007)
12. Courrieu, P.: Fast computation of Moore-Penrose inverse matrices. Neural Information Processing - Letters and Reviews 8(2), 25–29 (2005)
13. Hoerl, A.E., Kennard, R.W.: Ridge regression: Biased estimation for nonorthogonal problems. Technometrics 12(1), 55–67 (1970)
14. Rumelhart, D.E., Hintont, G.E., Williams, R.J.: representations by back-propagating errors. Nature. 323(6088), 533–536 (1986)
15. Hoerl, A.E., Kennard, R.W.: Ridge regression: biased estimation for nonorthogonal problems 42(1), 80–86 (2000)

Using Positional Information in Modeling Inflorescence Discs

Malgorzata Prolejko

Abstract. The aim of this paper is the introduction to the concept of positioners. It is the new way to use the positional information while modeling compound models of plants or their parts. A specific way of compound fruit growth requires to search new solutions and further develop techniques already known. The result was obtained from decomposition of the module and applying positional information to its featured parts.

First, there is presented a mathematical description of compound objects, called geometrical method. Next, positioners are described as an extension of geometrical method. Furthermore some examples are demonstrated.

Keywords: positioner, L-system, plant growth, sunflower.

1 Introduction

Most plants have a modular structure, which allows distinction of fragments of the plant that develops in a similar way. With this observation, methods characteristic for modular objects can be used. The most popular method is L-systems [4]. There are many papers describing, expanding or modifying algorithms based on L-systems [2] [3] [5]. Most of them assume, that modules are identical and similarly develop in time. The modules in one compound can be distinguishes only by their positional information.

The aim of this article is to present a new method of using positional information during modeling modular objects. The new concept called *positioners* is introduced. Its main task is to manage multiple semi-similar modules in continuous time, which is difficult in L-system methods.

Malgorzata Prolejko
Faculty of Mathematics and Computer Science, University of Warmia and Mazury, Olsztyn
e-mail: m.prolejko@matman.uwm.edu.pl

R. Burduk et al. (Eds.): *CORES 2013*, AISC 226, pp. 71–80.
DOI: 10.1007/978-3-319-00969-8_7 © Springer International Publishing Switzerland 2013

2 Geometrical Method of Modeling Modular Objects

A mathematical method of modular modeling based on L-systems is presented in [2]. The main change is that it operates in continuous time domain rather than discrete steps. Furthermore, graphic representation of modules is combined with hierarchy structure in one equation (3). Its main advantage for plant modelers is that it allows to avoid searching differential growth functions of plant parts [3]. Nevertheless, the method is limited. It uses only one kind of module, which is equivalent of L-system with one letter.

Mathematical background

Geometrical method is a mathematical notation of evolution model in time and space. It is used to describe three-dimensional shape of modules depending on time as well as their hierarchical structure. The general assumption is that the object consists of modules of the same kind, developing identically in time and space, only with delay dependent on a hierarchy level.

The set $V(t)$ consists of points belonging to a single module. It describes a space-time development in its local coordinate system. The hierarchical structure is build based on *characteristic vertices*. These are points to which child modules are assembled (Fig. 1). Characteristic vertices can move freely and their movement is described by movement matrix $\mathbf{D}(t)$ [2]. There is exactly one child module associated with each characteristic vertex. Child modules are placed according to the coordinate system of their characteristic vertices and move along with them. The number of characteristic vertices of one module is denoted by q and called *branching degree*.

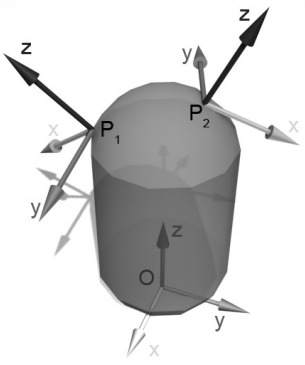


Fig. 1 Local coordinate systems of the characteristic vertices P_1, P_2 and the module O

Let $\mathbf{p}(t)$ be the position of a vertex P in local coordinate system of the module. Position of vertex P in a parental coordinate system $\mathbf{p}'(t)$ can be calculated with the equation $\mathbf{p}'(t) = \mathbf{D}(t)\mathbf{p}(t)$. Generalizing it on all vertices, a description of the child module is obtained [2]:

$$V'(t) = \mathbf{D}(t)V(t). \tag{1}$$

Not all modules develop simultaneously. For a given plant species a constant delay T (*modulus*) can be defined. It determines the time delay after which child modules start to develop. Then the child module can be described as follows (2):

$$V'(t) = \mathbf{D}(t)V(t-T) \tag{2}$$

Putting all modules in hierarchy together description of the whole object $Z(t)$ is obtained:

$$Z(t) = V(t) \cup \bigcup_{i=1}^{k} \bigcup_{\hat{g}^i \in \mathbf{G}_i} \mathbf{D}_{\hat{g}^i}(t)V(t-iT). \tag{3}$$

The equation (3) is the union of root module $V(t)$ and unions of the all child modules in hierarchy structure. i stands for level of the hierarchy and $\mathbf{G}_i$ is the set of all possible combination of ancestors indexes in level i.

In the basic approach (3), there is no time limit, which means that the plant develops infinitely. The maximum level number k can be given, so that none module with that hierarchy level produce children. The development of a single module is finished in time T_d.

$$t_s = k * T + T_d \tag{4}$$

Limiting the number of levels to a preset value determines static state after t_s, when all modules do not develop (4) [2].

Possible extensions

The primary assumption of the identity of the modules not always suits for modeling process. We can distinguish a group of cases in which the development of the older modules differs from the others. Geometric method allows to use positional information [5] to formulate a set $V_i(t)$. Modification of the coordinate systems can also take place.

Assumptions and definitions necessary to be declared before start to produce compound objects are shown below.

- $V(t)$ – single module space-time development,
- q – branching degree,
- $\mathbf{D}_p(t)$ – set of movement matrices for every characteristic vertex,
- T – time delay,
- k – hierarchy levels number,
- T_d – maturation time of the module $V(T)$.

3 The Concept of Positioners

The concept of *positioners* is an extension of the geometrical method, which was introduced as a mean of constructing flower models with spiral phyllotaxis. It allows to differentiate parts of module (e.g. single flower) based on its hierarchy index number [2].

Modeling a sunflower inflorescence disc with basic graphical method provides the unrealistic results. The reason for that is the identical module development being one of the bases of the aforementioned method. It strictly associates the position of flower module on a disc ray with its development stage. As a result, all flower-modules are mature only on a disc edge.

One can exclude the movement on the disc from the development process. Then flower-module only ripens and inflorescence arrangement must be done separately. Moreover, information about hierarchy is contained in the arrangement structure, so its impossible to delay growth of the flowers in the center of the disc. Positioners allow to differentiate rather than exclude the movement from the development process in order to keep positional information for entire module.

There are two examples to use positioners described in this paper. The first one shows how positioner determines position of flower on inflorescence disc. The second one shows how multiple plant spices can be modeled with one basic module.

3.1 The Positioner Characteristic

Separation of movement from module development motivates creation of separator fundamental to decomposition. In basic geometrical method, positional information was applied with function F_i [2] (5).

$$V_i(t) = F_i(V(t)) \tag{5}$$

This function is applied to all points from the set $V(t)$. When the decomposition (6) is conducted, new local coordinate systems must be selected. The ones that will be modified, are separated into category of *positioner*.

$$V(t) = \bigcup_{j=1}^{n} v_j(t) \tag{6}$$

Decomposition of the set $V(t)$ into n sub-modules $v_j(t)$ $(j = 1,\ldots,n)$ is shown in Fig. 2. A single module consists of a set of points, dedicated local coordinate system and characteristic vertices with their own coordinate systems (Fig. 2a). After decomposition module is divided into new sets of points (Fig. 2b). The coordinate system of the *Sphere SXYZ* is positioner. Consequently, element *Sphere* has positioner status.

Inner hierarchy of a single module introduced in Fig. 2b. Rectangle indicates coordinate system of a single element. The element which is highlighted (*Sphere*) is a positioner with its coordinate system singled out.

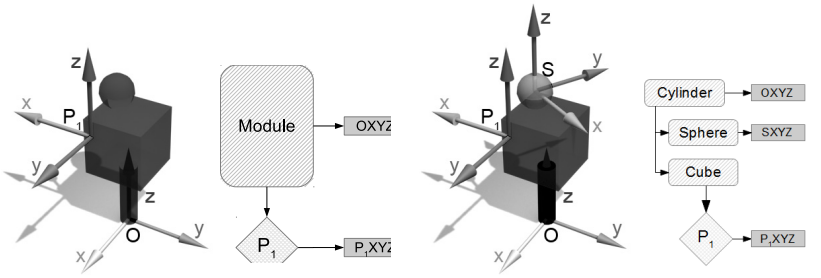


Fig. 2 Objects in module: a) basic method, b) with positioner

Part of the module which gains positioner status is denoted by $\tilde{v}_j(t)$ and its index $j \in \{1,...,n\}$ is added to the positioner index set P. Fragments with indexes in a complementary set $P' = \{1,...,n\} - P$ are not modified by the function F_i. The whole module is described by equation (7).

$$V_i(t) = \bigcup_{j \in P'} v_j(t) \;\cup\; \bigcup_{j \in P} F_i(\tilde{v}_j(t)) \tag{7}$$

Positioner movement in coordinate system of the module is described by a displacement matrix $\mathbf{H}(t)$ (8).

$$\mathbf{H}(t) = \begin{bmatrix} r_{1x}(t) & r_{2x}(t) & r_{3x}(t) & d_x(t) \\ r_{1y}(t) & r_{2y}(t) & r_{3y}(t) & d_y(t) \\ r_{1z}(t) & r_{2z}(t) & r_{3z}(t) & d_z(t) \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{8}$$

Each positioner has its own matrix $\mathbf{H}(t)$. Therefore for the purpose of the whole module a set of matrices is designed $\{\mathbf{H}_j(t)\}$, where $j \in P$. These matrices must be taken into account while defining sets $\tilde{v}_j(t)$ (9).

$$\tilde{v}_j(t) = \mathbf{H}_j(t)\tilde{v}'_j(t) \tag{9}$$

Function F_i refers to whichever transforming function. For the purposes of this paper, a composition of two scaling functions are taken into account. The first one (Ψ) scales the size and second (Φ) scales position of positioner and its children. Both are index dependent and time independent. Scaling matrices $\mathbf{S}(i)$ (10) [1] are used to define functions Ψ and Φ.

$$\mathbf{S}(i) = \begin{bmatrix} s_x(i) & 0 & 0 & 0 \\ 0 & s_y(i) & 0 & 0 \\ 0 & 0 & s_z(i) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{10}$$

Function Φ, which scales the size of the object is typical (11).

$$\Psi(i) = \mathbf{S}_{\Psi}(i) \tag{11}$$

Function $\Phi(i)$ is defined by the scaling matrix $\mathbf{S}_{\Phi}(i)$, which is applied to the displacement matrix $\mathbf{H}_j(t)$ (12).

$$\Phi(i) = \mathbf{S}_{\Phi}(i)\mathbf{H}_j(t)\mathbf{S}_{\Phi}^{-1}(i) \tag{12}$$

Function F_i assemblies scalings (12) and (11).

$$F_i(\tilde{v}_j(t)) = \mathbf{S}_{\Phi}(i)\mathbf{H}(t)\mathbf{S}_{\Phi}^{-1}(i)\mathbf{S}_{\Psi}(i)\tilde{v}'_j(t) \tag{13}$$

After decomposition of the module (7) with inclusion of the modifying function F_i, the entire object $Z(t)$ (3) can be described by equation (14).

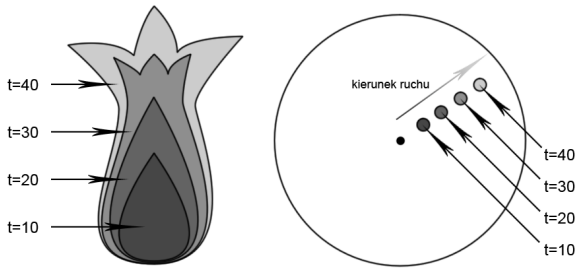
$$Z(t) = V(t) \cup \bigcup_{i=1}^{k} \bigcup_{\hat{g}^i \in \mathbf{G}_i} \mathbf{D}_{\hat{g}^i}(t) \Big(\bigcup_{j \in P'} v_j(t - iT) \cup \bigcup_{j \in P} F_i(\tilde{v}_j(t - iT)) \Big) \tag{14}$$

Overall, the space-time development of the entire object can be easly described with the notation obtained from geometrical method. Every element of the equation (14) has simple representation in any 3d design program, thus implementation of the objects is intuitive.

3.2 The Use of Positioners to Model a Inflorescence Disc

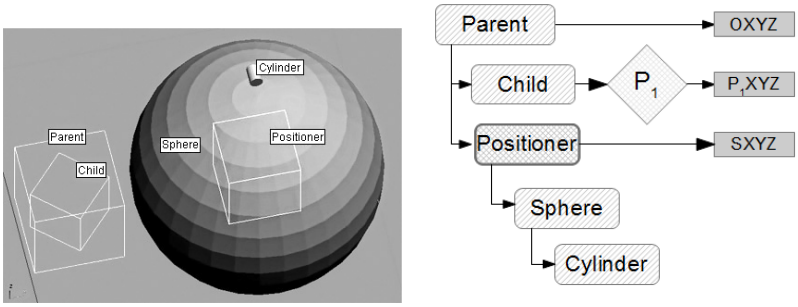
Various stages of the single disc flower development are position independent. Defining positioners helps to separate a single ray flower development from inflorescence modeling (Fig. 3) during modeling process. The use of the positioners to create inflorescence disc are described in this subsection.

Various configurations of positioner scaling based on one module structure is shown in Fig. 4. A module is composed with animated sphere *Sphere*, cylinder *Cylinder* and three Dummy objects: *Parent*, *Child* and *Positioner*. Maturation time of the module is $T_d = 500$ frames. Animated texture allows to determine the state of development in current frame - from light gray in early stages to dark gray when module is mature. The *Cylinder* has variable length and is positioned along *OY* axis in the coordinate system of the module. The *Positioner* is strictly connected with the center of the sphere, in order to be observed the movement of the whole module.



(a) (b)

Fig. 3 Separation animation of the module from its movement on the disc: a) module development; b) movement of the module along disc ray



(a) (b)

Fig. 4 Module prepared to simulation: a) elemets, b) hierarchy

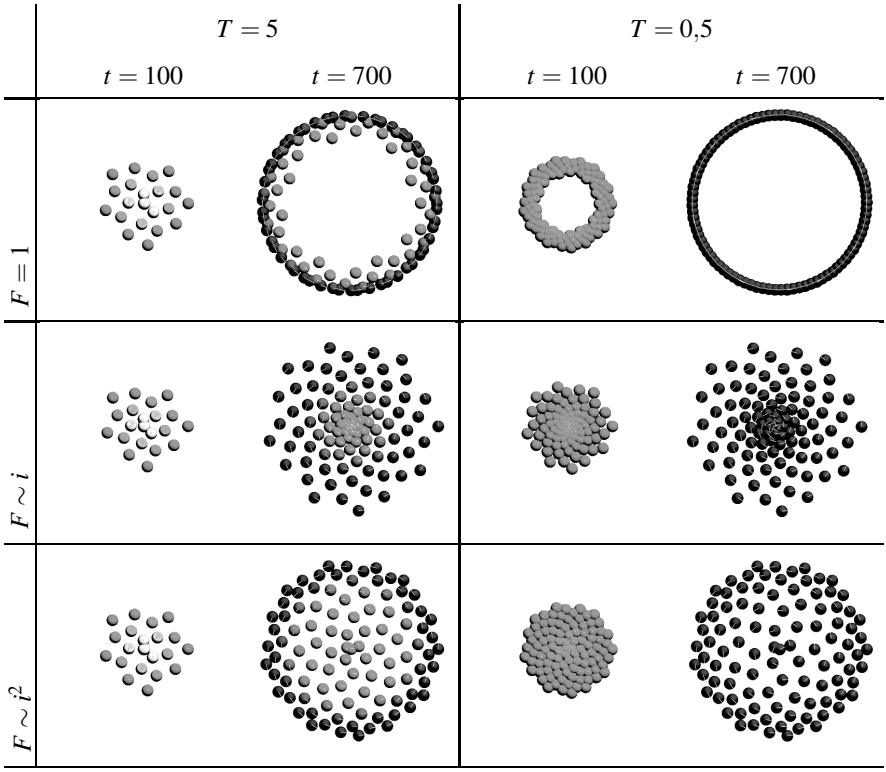
There were six simulations performed on the module (Fig. 4). All of them had the same hierarchy levels number $k = 100$. Simulation parameters with visualizations of 100^{th} and 700^{th} frame are presented in Table 1. All three scaling functions Φ depend only on the index of the module. There were no Ψ functions included.

Two different modulus T apportion columns of the Table 1. In rows, different scaling functions Φ are applied. In every table cell, there are two figures. The left ones are obtained from frame 100^{th} and the right ones from frames 700^{th}.

$$n = \min(\lfloor frame/T \rfloor + 1, k) \tag{15}$$

Different modulus T affects the time the object maturates and the number of produced modules n (15). Discs in the left column have 21 modules in 100^{th} frame and in all other figures there are all 100 modules. The state is constant when $t > t_s$ (4) and all the modules in right column are mature (frame $t = 700$). Color of a single module determines its development stage. The bigger the modulus, the more contrast of the modules color are in the object.

Table 1 Table of disc views with different simulation parameters. Left images are acquired for frame 100 and the right ones are for 700.



The scaling function has an effect in arrangement of modules in the disc. While the divergence angle is constant and equal 34/89 (137°31') [6], scaling applies only to the distance from center of the disc. In the case when scaling does not take place (row 1 of Table 1), similarly to the basic geometrical method, development stage of a single module depends on its position on the disc ray. When all modules are mature, they all are placed on the disc edge. After linear scaling, dense group of modules appears in the center of the disc (row 2 of Table 1). Quadratic scaling causes the arrangement of modules to be quite uniform. This is a more satisfying and awaited effect.

3.3 The Use of Positioners to Differentiate Parts of the Module

Any object in the module hierarchy can be picked as a positioner during modeling process. This positioner, with its children, will be differentiated with scale factor from other elements in the module. This can be observed on the example of the ear.