LBM 2008 (CLA 2008)

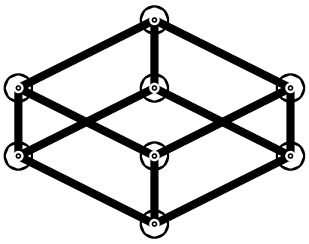
Proceedings of the Lattice-Based Modeling Workshop, in conjunction with
The Sixth International Conference on Concept Lattices and Their Applications

Palacký University, Olomouc, Czech Republic

ISBN 978–80–244–2112–4

Palacký University, Olomouc, Czech Republic

Lattice-Based Modeling Workshop



LBM 2008

Olomouc, Czech Republic
October 21–22, 2008

CLA 2008

The Sixth International Conference on
Concept Lattices and Their Applications

Edited by

Vassilis Kaburlasos
Uta Priss
Manuel Graña

LBM 2008 (CLA 2008)

Technical Editor:
Jan Outrata, jan.outrata@upol.cz

Page count: x+67
Impression: 70
Edition: 1^{st}
First published: 2008

Published and printed by:
Palacký University, Olomouc, Czech Republic

Organization

LBM 2008 was organized in conjunction with CLA 2008 by the Palacký University, Olomouc, and the State University of New York at Binghamton.

CLA 2008 General Chairs

Radim Belohlavek	State University of New York at Binghamton, USA
Sergei O. Kuznetsov	State University Higher School of Economics, Moscow, Russia

Program Chairs

Vassilis Kaburlasos	Technological Educational Institution of Kavala, Greece
Uta Priss	Napier University, Edinburgh, United Kingdom
Manuel Graña	Universidad del Pais Vasco, San Sebastian, Spain

Program Committee

Manuel Graña	Universidad del Pais Vasco, San Sebastian, Spain
Anestis Hatzimichailidis	Technological Educational Institution of Kavala, Greece
Vassilis Kaburlasos	Technological Educational Institution of Kavala, Greece
Etienne Kerre	Ghent University, Belgium
Jun Liu	University of Ulster, Northern Ireland, United Kingdom
Stelios Papadakis	Technological Educational Institution of Kavala, Greece
Uta Priss	Napier University, Edinburgh, United Kingdom
Gerhard Ritter	University of Florida, Gainesville, USA
Peter Sussner	State University of Campinas, Brasil
Gonzalo Urcid	National Institute of Astrophysics, Optics, and Electronics; Puebla, Mexico
Ivan Villaverde	Universidad del Pais Vasco, San Sebastian, Spain
Yang Xu	Southwest Jiaotong University, Chengdu, Sichuan, China

Organization Committee

Michal Krupka (Chair)	Palacký University, Olomouc, Czech Republic
Eduard Bartl	State University of New York at Binghamton, USA
Jan Konecny	State University of New York at Binghamton, USA
Tomas Kuhr	Palacký University, Olomouc, Czech Republic
Petr Osicka	Palacký University, Olomouc, Czech Republic
Jan Outrata	Palacký University, Olomouc, Czech Republic

Sponsoring Institutions

PIKE ELECTRONIC, Czech Republic – main sponsor

PIKE Electronic is one of the leading Central European IT companies providing complex service in the field of software application development. The Company delivers or participates on large-scale projects around the globe, cooperating with partners such as Siemens AG, Siemens E&A, SMS DEMAG, TIBCO, Hewlett–Packard, Accenture or Oracle. The main strategic target is specialization in "core business" areas, consolidation, effective distribution and use of long-term experiences and know-how in segments of Process Automation, SOA & Enterprise Application Integration , Information Systems for Health Insurance, Business Intelligence, Telecommunication, Banking, and Logistic.
PIKE ELECTRONIC and the Department of Computer Science of Palacký University have been cooperating for more than ten years. In the recent years, PIKE ELECTRONIC participated in a large research project on Relational Data Analysis which was supported by the Academy of Sciences of the Czech Republic during 2004–2008. The principal investigator was Professor Radim Belohlavek from the Department of Computer Science, the principal co-investigator was Dr. Stanislav Opichal from PIKE. The research results of this project were published in premier international journals and in proceedings of recognized international conferences.

SwissCentrum software, Czech Republic

Faculty of Science, Palacký University, Czech Republic

Table of Contents

Preface

With the proliferation of computers, a variety of domain-specific information processing paradigms have emerged. The corresponding analysis- and design-tools (of mathematical nature) are largely disparate due to the need to cope with disparate types of data including logic values, (fuzzy) numbers/sets, symbols, graphs, etc. A unification of the aforementioned tools is expected to result in a useful technology cross-fertilization. Nevertheless, an "enabling" mathematical framework is currently missing.

It turns out that popular types of data, including the aforementioned ones, are lattice-ordered. Hence, lattice theory (LT) emerges as a promising "enabling" mathematical framework; furthermore, *Lattice Computing (LC)* emerges as the corresponding information processing paradigm. In contrast to typical information processing, which carries out "number crunching" techniques in space R^N, an additional advantage of LC is its capacity to carry out *semantic computing*.

There is a number of isolated research communities, or Communities for short, which employ LT in various information processing domains including 1) Logic and Reasoning, for automated decision-making [5], 2) Mathematical Morphology, for signal- and image- processing [4], 3) Formal Concept Analysis, for knowledge-representation and information retrieval [1], 4) Computational Intelligence, for clustering, classification, and regression [2]. However, despite a creative interaction within a Community, different Communities typically work separately [3]. Hence, practitioners of LT typically develop their own tools/practices without being aware of valuable contributions by colleagues in other Communities. In conclusion, potentially useful work may be ignored or duplicated.

This workshop is an initiative towards a creative interaction/integration. Six papers are presented in this volume. The first one is a review paper, whereas the remaining ones present original (preliminary, though) research results in different domains of interest as explained in the following.

The paper by Priss and Old revisits ideas on lattices as underlying conceptual structures in information retrieval and machine translation applications.

The paper by Kaburlasos and Papadakis presents techniques for piecewise-linear approximation of nonlinear models based on Interval Numbers (INs).

The paper by Papadakis and Kaburlasos introduces a technique for input variable selection based on lattice-ordered Interval Numbers (INs).

The paper by Graña, García-Sebastián, Villaverde, and Fernandez introduces an approach to fMRI (image) analysis based on the lattice associative memory (LAM) endmember induction heuristic algorithm (EIHA).

The paper by Urcid, Ritter, and Nieves-Vázquez presents lattice-associative memories able to recall (image) patterns degraded by mixed or random noise.

Finally, the paper by Hatzimichailidis and Kaburlasos introduces a fuzzy implication stemming from a fuzzy lattice inclusion measure.

Vassilis Kaburlasos vgkabs@teikav.edu.gr
Uta Priss u.priss@napier.ac.uk
Manuel Graña ccpgrrom@si.ehu.es

Chairs, Lattice-Based Modeling (LBM 2008) Workshop
in the context of
The Sixth Intl. Conf. on Concept Lattices and Their Applications (CLA 2008)
21–23 October 2008, Olomouc, Czech Republic

References

1. B. Ganter, R. Wille, Formal Concept Analysis. Berlin, Germany: Springer, 1999.
2. V.G. Kaburlasos, Towards a Unified Modeling and Knowledge-Representation Based on Lattice Theory. Berlin, Germany: Springer, Studies in Computational Intelligence 27, 2006.
3. V.G. Kaburlasos, G.X. Ritter (eds.), Computational Intelligence Based on Lattice Theory. Berlin, Germany: Springer, Studies in Computational Intelligence 67, 2007.
4. G.X. Ritter, J.N. Wilson, Handbook of Computer Vision Algorithms in Image Algebra, 2nd ed. Boca Raton, FL: CRC Press, 2000.
5. Y. Xu, D. Ruan, K. Qin, J. Liu, Lattice-Valued Logic. Heidelberg, Germany: Springer, Studies in Fuzziness and Soft Computing 132, 2003.

Lattice-based Modelling of Thesauri

Uta Priss and L. John Old

Napier University, School of Computing
{u.priss,j.old}@napier.ac.uk

Abstract. This paper revisits ideas about the use of lattices as underlying conceptual structures in information retrieval and machine translation as suggested by researchers in the 1950s and 1960s. It describes how these ideas were originally presented, how they are related to each other and how they are represented in modern research, particularly with respect to Formal Concept Analysis.

Key words: Lattice-based modelling, thesauri, machine translation, information retrieval

1 Introduction

Over the past 20 years, Formal Concept Analysis[1] (FCA) has gained international recognition with respect to its applications of lattice theory to fields such as knowledge representation, information retrieval and linguistics. But there have also been other non-FCA applications of lattice theory in the same areas. Some of these are very similar to FCA applications in that they also emphasise a duality between two sets (what FCA calls "formal objects" and "formal attributes") which forms a Galois connection. Other non-FCA applications use only the lattice operations, but do not emphasise the Galois connection. For this paper we investigate two historic (1950-60s), non-FCA applications of lattice theory to the area of semantic structures, modelled using thesauri. In particular, we are interested in the impact that these developments had: is modern research in this area a continuation or just a repetition of ideas that were suggested 40-50 years ago? Do these old research papers still inspire modern work, or have the immense improvements in hardware and software made the old research obsolete?

The two research areas we are considering are the work by Margaret Masterman as the founder of the Cambridge Language Research Unit in the area of "mechanical translation" and the lattice-based retrieval model by Mooers and Salton that was proposed in Salton's (1968) influential textbook on information retrieval. Masterman et al. (1959) argue that both fields are part of a more general field of "semantic transformation" because mechanical translation uses a thesaurus as a retrieval tool in a similar manner to how thesauri are used as an interlingua in information retrieval. In modern terminology such a field might be

[1] This paper is not an introduction to FCA. Such an introduction can be found in Ganter and Wille (1999) or via the bibliography at http://www.fcahome.org.uk.

pp. 1–12, ISBN 978–80–244–2112–4, Palacký University, Olomouc, 2008.

called "conceptual structures" and would contain a range of formal structures (class hierarchies, ontologies, conceptual graphs), not just thesauri.

Both groups of researchers selected for this paper have had a tremendous impact on the development of their respective fields (natural language processing and information retrieval). But although the fields have grown, it seems to us that the use of lattice theory in these fields has not grown to the same degree and some of the original ideas appear lost in later work. While there is some use of lattices in modern information retrieval, most modern retrieval applications use the vector space model (which was also described by Salton (1968)) instead of the lattice model. Most modern natural language processing uses statistical and other non-lattice methods. Nevertheless there are a few modern lattice applications in these areas which are promising. Thus, it may be useful to revisit the old ideas.

The lattice-based applications we are interested in are not predominantly Boolean lattices. Many researchers have observed that Boolean lattices form a theoretical basis for information retrieval because the set of all possible subsets of documents or the set of all possible subsets of keywords form Boolean lattices. The elements of a Boolean logic or Boolean algebra also form a Boolean lattice. Thus, any computer program that uses 0's and 1's and AND, OR and NOT operates on Boolean lattices. Even the ancient Chinese book, the I Ching, with its 64 trigrams each consisting of 6 lines that can either be broken (i.e. corresponding to 0) or solid (i.e. corresponding to 1), describes a Boolean lattice with $2^6 = 64$ elements. Leibniz's representation of "primitive concepts by prime numbers and compound concepts by products of primes" (Sowa, 2006) in the 17th century is another Boolean lattice. Unfortunately, forming such Boolean lattices does not yield much information. In Leibniz's example, each possible combination of different prime numbers yields another element of the lattice. The I Ching contains every possible combination of a 6 character string of 0's and 1's. This is equivalent to forming every possible subset of a set with 6 elements. Thus, Boolean lattices occur in many situations but in each case they only list all possible combinations of a certain kind. In information retrieval applications, a Boolean lattice of query terms (or keywords) simply records the fact that every set of query terms can be formed. In Masterman's idea of using lattices as an interlingua for translating between languages, a Boolean lattice represents every possible combination of words.

In order to illustrate how such Boolean lattices can be visualised, an example using Docco [2] is shown in Fig. 1. Docco is an FCA-based tool that indexes files on a computer. In the example in Fig. 1, Docco was used to index a directory with email folders. The folders serve as the formal objects of the concept lattice. Their counts are displayed below the nodes representing the formal concepts. The formal attributes are terms entered into the search field. An attribute belongs to an object if the term occurs in any of the emails in that folder. The attribute names are displayed slightly above the nodes to which they belong. In this case a search for "meeting Friday project student" was submitted. A Boolean lattice

[2] `http://tockit.sourceforge.net/docco`

with four atoms is automatically drawn by Docco in response to the four search terms. Each concept in the lattice corresponds to a combination of the search terms. For example, the first node that can be reached by travelling down from "meeting" and "Friday" has three formal objects. Fig. 1 demonstrates how, after clicking on that concept, the result of a more narrow query, "meeting and Friday" within the broader query, is shown. The file hierarchy on the right is expanded to show the names of the three email folders which contain both the words "meeting" and "Friday", but not "project" or "student". In this case there is a folder with the name "studentproject", which is quite likely relevant to the query. Coincidentally, the emails in the studentproject folder do not themselves contain the words "project" and "student", but this can happen. The nodes below and above this concept are also highlighted in the diagram because quite often, if users do not find an exact match, slightly expanding or restricting a query will show relevant results. The idea for using a lattice representation instead of a listing of the results, is so that users obtain feedback on the structure of the result set.

Boolean lattices easily become too large to be represented graphically. Using Docco, it would be difficult to visualise searches with more than five or six terms, but most users probably only use two or three terms for these kinds of searches anyway. A Boolean lattice with n atoms contains 2^n elements. Thus, unless the application domain is very small, it is not practical to graphically represent the Boolean lattice and to plot actually occurring combinations on it. From an information theoretic viewpoint, lattices that are not Boolean are usually much more interesting because they contain information about which elements cannot be combined with which other elements, or which combinations of elements might imply other combinations of elements. Thus, while Boolean lattices are of theoretical value for describing, for example, query languages and interlinguas, for many applications, smaller, non-Boolean lattices or substructures of lattices are more interesting. Methods that extract such smaller lattices or substructures are of importance. This might be an explanation for why there was an initial enthusiasm about lattices in information retrieval and natural language processing: initially, it was discovered that Boolean lattices are of relevance in both domains. But until methods were developed that focused on extracting smaller substructures (as is achieved by FCA methods), the interest in lattices subsided.

The remainder of this paper is organised as follows: Section 2 describes the lattice model of information retrieval as described by Mooers and Salton in the 60s. Section 3 provides an overview of the application of lattice theory to the modelling of thesauri as proposed by the Cambridge Language Research Group under Margaret Masterman. Section 4 then revisits both ideas from a modern perspective and analyses how the ideas appear in modern implementations.

2 The Mooers-Salton Lattice-Based Retrieval Model

Salton's (1968) famous textbook on information retrieval contains a section on "retrieval models". It discusses several different mathematical models including

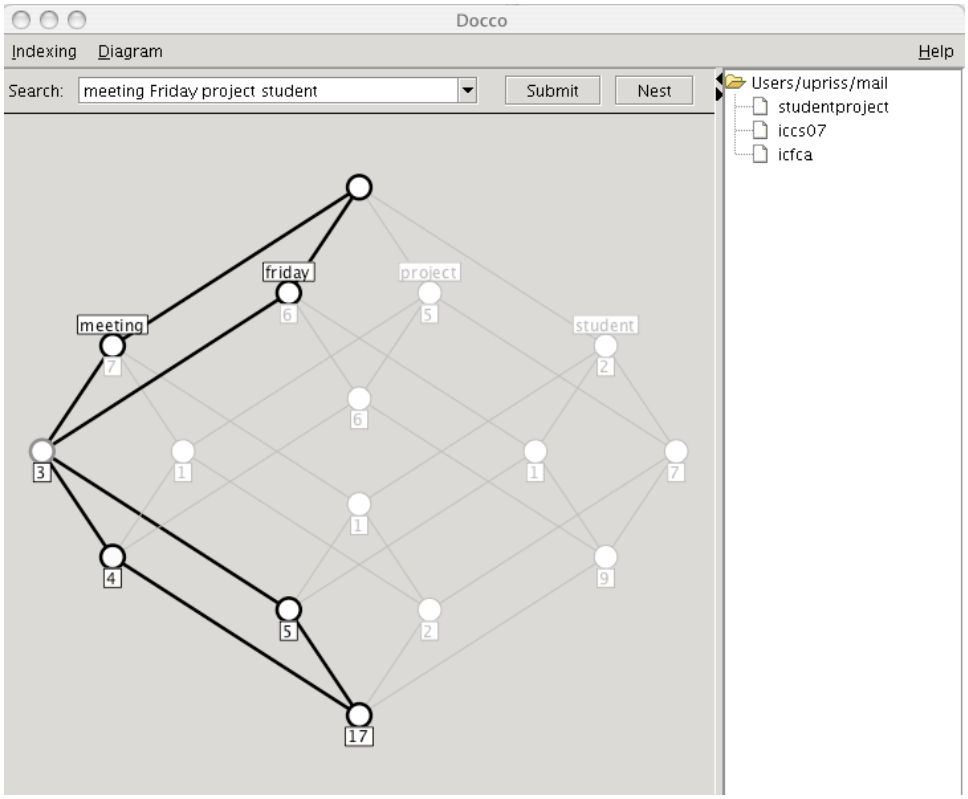


Fig. 1. A concept lattice in Docco[2] showing a query for email folders containing "meeting" and "Friday" within a search for "meeting Friday project student".

a lattice-based model which is in some ways similar to FCA. This lattice-based information retrieval model was described by Mooers (1958) in a semi-formal manner and elaborated with mathematical proofs by Woods (1964). Mooers credits Fairthorne (1947, 1956) with being the first person to suggest using lattices for information retrieval. Mooers (1958) sees Boolean lattices as the most important lattices. He describes different transformations from a space of "retrieval prescriptions" into the lattice of all possible document subsets. While his transformation T1 selects the set of documents that contain "exactly" the requested prescriptions, T2 selects the documents which contain "at least" the requested prescriptions. The transformation T2 represents "the well-known fact that as one adds more and more descriptors to a retrieval prescription, the set of retrieved documents becomes smaller and smaller, and that each of the smaller sets of documents is included within the larger set which is obtained with fewer descriptors in a prescription" (Mooers, 1958, p. 1342). Thus, T2 is a Galois connection between documents and prescriptions. In FCA terms, Mooers' discovery could be described much simpler by stating that prescriptions are the formal objects and the documents the formal attributes of a formal context (similar to the example in Fig. 1, although upside down).

In general, the information retrieval problem is described by Mooers as a problem of mappings between the space of retrieval prescriptions and the space of document descriptors. The mappings become more complicated if an additional hierarchy is defined on the descriptors (such as a library classification scheme), or if they can be combined using AND, OR and NOT. Woods (1964) and Soergel (1967) formalise and elaborate these mappings further. Woods's paper was written as a student paper and would probably have been forgotten if Salton had not included it in his book. Salton (1968) considers "inclusive retrieval functions" which are order-inverting maps between the retrieval space and the document space (because more prescriptions retrieve fewer documents and vice versa). The use of additional operators or of a classification system yields lattices which are even larger and more complicated than Boolean lattices. Salton shows a free distributive lattice resulting from three descriptors and a single operator, which has the meaning "having a topic in common" (p. 216). He also discusses the problems of negation in some detail (p. 223-227).

One of Salton's examples (p. 219) is shown in Fig. 2 (although with a slightly more FCA-like notation). This example is very similar to concept lattices in FCA. The request space is a Boolean lattice of three prescriptions. But the space of retrieved documents is not Boolean. In FCA terms, it is the concept lattice of a formal context of prescriptions and documents. The dashed arrows show the (order-inverting) retrieval mapping from each document to its set of prescriptions. Unfortunately, neither Salton, nor Woods, Mooers or Soergel saw the potential of these kinds of non-Boolean lattices. Salton's main interest was Boolean lattices because his concluding theorem shows under which circumstances the resulting lattice (as in Fig. 2) is Boolean.

As far as we know, Salton and Soergel eventually lost interest in lattices. Woods remains fascinated by lattices, but is less focused on their mathemat-

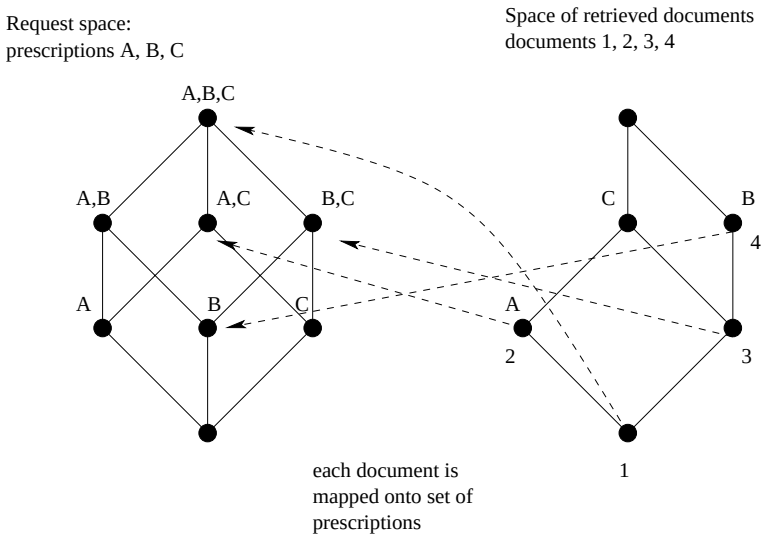


Fig. 2. Salton's example - similar to a concept lattice, but 15 years before FCA!

ical details. Towards developing intelligent computer assistants, Woods (1978) proposes situation lattices. These organise "things to be done and goals to be achieved" into a conceptual taxonomy of situations (p. 33). A situation description must be composite and structured, subparts of which will be instances of other concepts, and makes use of concepts of objects, (substances, times, events, places, individuals) represented as configurations of attributes standing in specified relationships to each other.

Woods credits Brachman (1978) for the generalisation, in his model, of the various notions of feature etc. to a single notion. "A concept node in Brachman's formulation consists of a set of "dattrs" [parts, constituents, features etc.] ... some of which are represented directly at a node, and others are inherited from other nodes" (p. 34). Situation descriptions may subsume other situation descriptions at lower levels of detail. "The space of possible situation descriptions forms a lattice under subsumption. At the top of the lattice is a single, most general situation we will call T ... anything that is universally true can be stored here" (p. 38). Conversely at the bottom of the lattice is a situation that is never satisfied.

A situation description can be made more general by (amongst other things) relaxing the constraints of a dattr, or made more specific by (amongst other things) tightening the condition on a dattr. Wood's description of the situation lattice, because it is meant to be a model of the working memory of an intelligent machine, is embedded in a complex description of situation recognition and classification, spreading activation (Quillian, 1967) and marker propagation, and other functions incorporated to make the lattice dynamic. Thus, Woods sees the

potential of lattices for describing conceptual structures, but he does not provide a precise mathematical description of how to implement these.

3 Lattices and Thesauri in Mechanical Translation

The second example of lattice-based modelling is in the field of what is nowadays called "natural language processing". As mentioned in the introduction, Masterman et al. (1959) argue that there is a connection between both fields because they both belong to the field of "semantic transformation". In 1956 at the International Conference on Mechanical Translation at MIT, four researchers from the Cambridge Language Research Group (Masterman (1956), Richens (1956), Parker-Rhodes (1956), Halliday (1956)) reported on their research of using a thesaurus as an interlingua in "mechanical translation" (MT), the term then used for "machine translation". The group's founder, Masterman, envisioned using mathematical lattice theory for building a thesaurus, i.e. a hierarchical structure with grouping of synonyms or near synonyms. She thought that a "multilingual MT dictionary is analogous in various respects, to a thesaurus" and that "the entries form, not trees, but algebraic lattices, with translation points at the meets of the sublattices" (Masterman, 1956). The advantage of this approach is that instead of having to consider different pairs of languages separately, each language needs to be translated only once (into the thesaurus). Adding a new language does not require any changes to the previously added languages. Masterman stated that "the complexity of the entries need not increase greatly with the number of languages, since translation points can, and do, fall on one another."

Of course, computational research in the 50s and 60s was influenced by the limitations of computers at that time. Considerations about computational speed and storage problems determined the algorithms. Parker-Rhodes (1956) extended Masterman's ideas by describing a mechanical translation program for such an interlingual thesaurus that uses Boolean operations which "can be performed with very great speed." The storage problem would be solved by storing "all the relevant information ... in the input and output dictionaries". Richens (1956) described the algebraic interlingua, NUDE, its code and an overview of its translation operations.

MT algorithms in that time often started with a chunk-by-chunk literal translation (Masterman et al, 1959). Every word stem and every grammatical indicator was translated from the input language to the output language using a dictionary and some rules. Masterman's use of lattices was novel because other linguists in that time (for example Lehmann (1978)) saw translation as a mapping between trees. A sentence from the input language was parsed into a tree structure. Each branch of the input language was mapped onto a branch of the output language. The branches in the output language formed another tree which had the output sentence as their root. Masterman argued that from a semantic viewpoint, lattices are a better model than trees. In a lattice, pairs of elements can have different numbers of parents and children, instead of having

only one parent each in a tree structure. Thus combinations of meanings can be represented more naturally.

In particular, Masterman (1957) was interested in Roget's Thesaurus (RT). Her idea was that each of the 1000 categories in RT could be used as a "head" which described the core meaning of a word. Because a word can occur more than once in RT, a word can have several heads. This leads naturally to a lattice, not tree structure. Of course, this implies that the meets and joins need to be calculated; without meets and joins, a thesaurus would be just a partially ordered set, not a lattice. Multiple occurrences of a word in the thesaurus might correspond to different meanings of the word or even homographs (such as "lead" the verb and "lead" the material). If one determines the heads of all the words of a sentence, the heads can provide an indication of what the sentence is about. Individual words can be disambiguated by comparing their heads to the other heads in the sentence. If a word has two different heads and only one of these also occurs for other words in the same sentence, then it is quite likely that that head corresponds to the meaning of the word in this sentence.

Masterman et al. (1959) saw a relationship between MT and information retrieval because in both cases a thesaurus could be used: either for retrieval or as an interlingua. Even syntax was dealt with by the thesaurus (Masterman, 1957) because grammatical indicators in the "intralinguistic context" relate to structures in the "extralinguistic context" that are shared across languages. For example, some languages have no genders (English), others have two (French), three (German) or six (Icelandic). But the distinction between "male" and "female" is extralinguistically motivated. Masterman et al, (1959) see an interlingua as consisting of a "logical system giving the structural principle on which all languages are based". In modern terminology, the thesaurus represents the "conceptual structures" that underly information retrieval and natural languages. In our opinion, this is quite similar to Woods' (1978) situation lattices. Because different languages share conceptual structures, they could share a thesaurus or conceptual structure. Only the lists of synonyms that were attached to every thesaurus head would be different in the different languages. Masterman was aware of Mooers' use of lattices and saw this as further evidence for the connection between the two fields.

4 Modern Descendants

The Mooers-Salton lattice-based retrieval model appears to have mostly been forgotten until it was rediscovered in the context of FCA (cf. Priss (2000) for an overview). Without being aware of the model in Salton's book, FCA researchers built formal contexts of documents and terms and studied their concept lattices (starting with Godin et al. (1989)). There are many FCA applications in this area. Just to name one example: Credo[3] provides an on-line interface for web search engines.

[3] `http://credo.fub.it/`

Masterman's research influenced many people, including Karen Spärck Jones who is considered to be one of the pioneers in information retrieval and natural language processing. Spärck Jones used Roget's Thesaurus, but as far as we know had not much interest in lattices. Similarly, Yarowsky (1992) described an implementation of the use of Roget's for word-sense disambiguation which was very similar to Masterman's ideas (although he does not cite her), but he uses statistical methods instead of lattices.

In 1960s in the US, Sally Yeates Sedelow obtained funding to convert the American edition of Roget's (1962) into a machine readable format with the purpose of aiding machine translation. The initial abstract models that she and her husband, Walter Sedelow, used did not rely on lattice theory (Dillon (1971), Bryan (1973), Bryan (1974), Talburt & Mooney (1989)). But Bryan's model describes a binary relation between words and senses, which is very similar to a formal context as used in FCA. Thus when the Sedelows met Rudolf Wille, the founder of FCA, in the early 1990s, they were enthusiastic about the possibilities that lattice theory had to offer for their research. Their paper about the concept "concept" (Sedelow & Sedelow, 1993) derives semantic neighbourhoods for words from the thesaurus which are then represented as "neighbourhood lattices". Our own research has used and elaborated this technique in a variety of papers (Priss & Old, 2004) and has recently led to the implementation of an on-line interface[4], which allows users to interactively generate such lattices. Thus, one can argue that this modern research is an implementation of Masterman's ideas, although the thesaurus research (of the Sedelow's) was initially separated from lattice research and was only recombined through FCA.

Another modern instantiation of Masterman's ideas is Helge Dyvik's (2004) research, although, as far as we know, he was not directly influenced by or aware of either FCA or Masterman. Dyvik's lattices are feature lattices in the sense of componential semantics. Dyvik's Semantic Mirrors Method extracts semantic information from bilingual corpora. His assumption is that if the same sentence is expressed in two different languages, then it should be possible to align words or phrases in one language with the corresponding words or phrases in the other language using statistical processes or semi-automated processes. Once the corpora are aligned the "translational images" of words in the other language are computed. This process can be repeated several times. Next, the translational images are algorithmically assigned to separate senses. The resulting structures can be represented either graphically as lattices, or as a thesaurus (using a WordNet-style representation). Both structures can be generated interactively through an on-line interface[5]. Priss & Old (2005) have shown that this procedure is similar to creating neighbourhood lattices in FCA, though Dyvik's research was developed independently of FCA.

It could be argued that Dyvik's Semantic Mirror's method is a proof of concept for Masterman's vision. Masterman's (1956) statement that a "multilingual MT dictionary is analogous in various respects, to a thesaurus" and that "the en-

[4] `http://www.roget.org`

[5] `http://ling.uib.no/helge/mirrwebguide.html`

tries form, not trees, but algebraic lattices, with translation points at the meets of the sublattices" prescribes exactly what Dyvik has implemented. Of course, it would not have been possible to implement a system like Dyvik's in the 1950s or 60s due to the limits of computers at that time. It seems to us, however, that maybe not all of Masterman's ideas have fully been explored using modern technology. For example, the "Twenty questions method of analysis" (Masterman et al 1959) that was used for extracting extralinguistic (or "semantic") information via an intralingual analysis appears to be similar to attribute exploration in FCA (Ganter & Wille, 1999). But this relationship has not yet been further investigated.

The modern descendants of Quillian (1967), Brachman (1978) and Woods (1978) are terminological or description logics, conceptual graphs and formal ontologies as used in the context of the Semantic Web. It appears to be generally accepted that the class or type hierarchies in these formalisms form lattices. But apart from the class or type hierarchies, these systems also contain a variety of other formal structures that do not form lattices. Thus Masterman's view of a thesaurus-lattice as the driving component in conceptual structures (or semantic transformations) was only partly correct. Lattices are important components, but not the only structures used in such systems. The connections between FCA and these fields have been established and are well documented (e.g. Rudolph (2006)).

5 Conclusion

In the introduction we questioned whether modern research in this area is a continuation or just a repetition of ideas that were suggested 40-50 years ago; whether this old research still inspires modern work; and whether improvements in hardware and software have made the old research obsolete. Returning to these questions, it can be stated that the 1950s and 60s research about lattice-based modelling of thesauri was visionary, but hindered by the limitations of the computer hardware and software of that time. In both fields, natural language processing and information retrieval, the theoretical relevance of lattice theory has been acknowledged since the 50s and 60s. But non-FCA researchers tend to use non-lattice operations for most of their algorithms. Only the FCA researchers in these fields focus on exploiting the lattice operations. Practical implementations of software using lattice theory have only been feasible since the 1990s. Some of these modern implementations (such as neighbourhood lattices of Roget's Thesaurus or Dyvik's Semantic Mirrors method) can be seen as "proof of concept" for ideas suggested in the 50s. But in modern research, thesauri and lattices are usually complemented with other structures under the general heading of "conceptual structures". Thus the older ideas have been validated and but also been extended in modern research. It ultimately remains to be seen what role lattices play with respect to the conceptual structures that underly these disciplines, whether lattices are a core, driving force in such systems

or whether they are merely one formal model amongst many other contributing models.

References

1. Brachman, R. J. (1978). *A Structural Paradigm for Representing Knowledge.* Technical Report No. 3605, Bolt, Beranek and Newman Inc., Cambridge, MA. (Available from the US Defense Technical Information Center (DTIC), http://www.dtic.mil/).
2. Bryan, R. (1973). *Abstract Thesauri and Graph Theory Applications to Thesaurus Research.* In: Sedelow, Sally Y. (ed.). Automated Language Analysis. Report on research 1972-73, University of Kansas, Lawrence.
3. Bryan, R. (1974). *Modeling in Thesaurus Research.* In: Sedelow, Sally Y. (ed.). Automated Language Analysis. Report on research 1973-74, University of Kansas, Lawrence.
4. Dillon, M.; Wagner, D. J. (1971). *Models of thesauri and their applications.* In: Sedelow, Sally Y. (ed.). Automated Analysis of Language Style and Structure in Technical and other Documents. Technical Report, University of Kansas, Lawrence.
5. Dyvik, H. (2004). *Translations as semantic mirrors: from parallel corpus to wordnet.* Language and Computers, 49, 1, Rodopi, p. 311-326.
6. Fairthorne, R. A. (1947). *The Mathematics of Classification.* Proc. British Society of International Bibliography, 9, 4.
7. Fairthorne, R. A. (1956). *The Patterns of Retrieval.* American Documentation, 7, p. 65-75.
8. Ganter, B.; Wille, R. (1999). *Formal Concept Analysis. Mathematical Foundations.* Berlin-Heidelberg-New York: Springer.
9. Godin, R.; Gecsei, J.; Pichet, C. (1989). *Design of browsing interface for information retrieval.* In: N. J. Belkin, & C. J. van Rijsbergen (Eds.), Proc. SIGIR '89, p. 32-39.
10. Halliday, M. A. K. (1956). *The Linguistic Basis of a Mechanical Thesaurus.* Mechanical Translation 3, 3, p. 81-88.
11. Lehmann, W. P.; Pflueger, S. M.; Hewitt, H.-J. J.; Amsler, R. A.; Smith, H. R. (1978). *Linguistic Documentation of Metal System.* Final technical report, Rome Air Development Center, RADC-TR-78-100.
12. Masterman, M. (1956). *Potentialities of a Mechanical Thesaurus.* MIT Conference on Mechanical Translation, CLRU Typescript. [Abstract]. In: Report on research: Cambridge Language Research Unit. Mechanical Translation 3, 2, p. 36.
13. Masterman, M. (1957). *The Thesaurus in Syntax and Semantics.* Mechanical Translation, 4, 1 and 2, p. 35-43.
14. Masterman, M.; Needham, R. M.; Sparck-Jones, K. (1959). *The Analogy between Mechanical Translation and Library Retrieval.* In: Proceedings of the International Conference on Scientific Information (1958). National Academy of Sciences - National Research Council, Washington, D.C., 1959, Vol. 2, p. 917-935.
15. Mooers, C. N. (1958). *A mathematical theory of language symbols in retrieval.* In: Proc. Int. Conf. Scientific Information, Washington D.C.
16. Parker-Rhodes, A. F. (1956). *Mechanical Translation Program Utilizing an Interlingual Thesaurus. [Abstract].* In: Report on research: Cambridge Language Research Unit. Mechanical Translation 3, 2, p. 36.

17. Priss, U. (2000). *Lattice-based Information Retrieval.* Knowledge Organization, 27, 3, p. 132-142.
18. Priss, U.; Old, L. J. (2004). *Modelling Lexical Databases with Formal Concept Analysis.* Journal of Universal Computer Science, 10, 8, p. 967-984.
19. Priss, U.; Old, L. J. (2005). *Conceptual Exploration of Semantic Mirrors.* In: Ganter; Godin (eds.), Formal Concept Analysis: Third International Conference, ICFCA 2005, Springer Verlag, LNCS 3403, p. 21-32.
20. Quillian, M. R. (1967). *Word Concepts: A Theory and Simulation of Some Basic Semantic Capabilities.* Behavioural Science 12, 5, p. 410-430.
21. Richens, R. H. (1956). *General Program for Mechanical Translation between any two Languages via an Algebraic Interlingua. [Abstract].* In: Report on research: Cambridge Language Research Unit. Mechanical Translation 3, 2, p.37.
22. Roget, P. M. 1962. *Roget's International Thesaurus.* 3rd Edition Thomas Crowell, New York.
23. Rudolph, S. (2006). *Relational Exploration - Combining Description Logics and Formal Concept Analysis for Knowledge Specification.* PhD Dissertation, Universitätsverlag Karlsruhe.
24. Salton, G. (1968). *Automatic Information Organization and Retrieval.* McGraw-Hill, New York.
25. Sedelow, S.; Sedelow, W. (1993). *The Concept concept.* Proceedings of the Fifth International Conference on Computing and Information, Sudbury, Ontario, Canada, p. 339-343.
26. Soergel, D. (1967). *Mathematical Analysis of Documentation Systems.* Information Storage and Retrieval, 3, p. 129-173.
27. Sowa, J. (2006). *Categorization in Cognitive Computer Science.* in: H. Cohen & C. Lefebvre (eds.), Handbook of Categorization in Cognitive Science, Elsevier, p. 141-163.
28. Talburt, J. R.; Mooney, D. M. (1989). *The Decomposition of Roget's International Thesaurus into Type-10 Semantically Strong Components.* Proceedings. 1989 ACM South Regional Conference, Tulsa, Oklahoma, p. 78-83.
29. Woods, W. A., Jr (1964). *Mathematical Theory of Retrieval Systems.* Harvard University. Applied Mathematics 221, Student Research Report.
30. Woods, W. A. (1978). *Taxonomic Lattice Structures for Situation Recognition.* In: Proceedings of the 1978 Workshop on Theoretical Issues in Natural Language Processing, p. 33-41.
31. Yarowsky, D. (1992). *Word-sense disambiguation using statistical models of Roget's categories trained on large corpora.* In: Proc. COLING92, Nantes, France.

Piecewise-Linear Approximation of Nonlinear Models Based on Interval Numbers (INs)

Vassilis G. Kaburlasos and S. E. Papadakis

Technological Educational Institution of Kavala
Department of Industrial Informatics
65404 Kavala, Greece
{vgkabs,spap}@teikav.edu.gr

Abstract. Linear models are ususally preferable due to their simplicity. However, nonlinear models often emerge in practice. A popular approach for dealing with nonlinearities is using a piecewise-linear approximation. In such context, inspired from both Fuzzy Inference Systems (FISs) of TSK type and Self-Organizing Maps (SOMs), this work introduces enhancements based on Interval Numbers and, ultimately, on lattice theory. Advantages include a capacity to deal with *granular* inputs, introduction of *tunable* nonlinearities, representation of *all-order* statistics, and induction of descriptive decision-making knowledge (*rules*) from the training data. Preliminary computational experiments here demonstrate a good capacity for generalization; furthermore, only a few rules are induced.

Key words: Fuzzy inference systems (FIS), Genetic optimization, Granular data, Interval number (IN), Lattice theory, Linear approximation, Rules, Self-organizing map (SOM), TSK model

1 Introduction

The need to employ a real *function* $y : \mathsf{R}^N \rightarrow \mathsf{R}^M$, i.e. a *model*, arises frequently in practice. In particular, linear models $y(\mathbf{x}) = c_0 + c_1x_1 + c_2x_2 + ... + c_Nx_N$ are preferable due to simplicity. However, most often, the dependence of output y on the input variables $x_1, ..., x_N$ is nonlinear.

A popular approach for dealing with nonlinearities is using a piecewise-linear approximation. For instance, in the context of fuzzy logic, the TSK (Tagaki-Sugeno-Kang) fuzzy model [13], [14], [15], [16] is popular. The computation of a TSK model, in the first place, involves the computation of clusters.

A popular scheme for clustering is the *self-organizing map* (SOM) devised for visualization of nonlinear relations of multidimensional data [10]. Lately, *granular* extensions of SOM were proposed in classification applications [8], [11], where a data cluster was represented by a *fuzzy interval number* (FIN).

This work proposes simpler acronym IN (Interval Number) for a FIN. In the sequel, it explains that a IN is a mathematical object, which may be interpreted as a probability/possibility distribution, an interval, and/or a real number. In conclusion, inspired from TSK modeling, this work proposes *lattice computing*

pp. 13–22, ISBN 978–80–244–2112–4, Palacký University, Olomouc, 2008.

techniques for an advantageous, piecewise-linear approximation based on a IN-extension of SOM.

We remark that the term *lattice computing*, or *LC* for short, was coined lately to denote an emerging Computational Intelligence paradigm based on lattice theory [3]. More accurately, LC is defined here as an evolving collection of tools and methodologies that can process disparate types of data including logic values, numbers, sets, symbols, and graphs based on mathematical lattice theory with emphasis on clustering, classification, regression, pattern analysis, and knowledge representation applications.

This paper is organized as follows. Section 2 describes the mathematical background. Section 3 outlines the proposed techniques. Section 4 presents preliminary experimental results. Section 5 concludes by summarizing the contribution. The Appendix summarizes the WRLS algorithm for incremental learning.

2 Mathematical Background

Here we summarize useful mathematical notions and tools regarding Interval Numbers (INs) [4], [6], [7], [8] using an improved mathematical notation [9].

2.1 The Vector Lattice $(\mathcal{\Delta},\leq)$ of Generalized Intervals

Assume the *complete latice* $(\mathsf{R},\leq)$ of real numbers with *least* and *greatest* elements denoted, respectively, by $O=-\infty$ and $I=+\infty$. A *generalized interval* is defined in the following.

Definition 1. *A* generalized interval *is an element of lattice* $(\mathsf{R},\leq^{\partial})\times(\mathsf{R},\leq)$.

We remark that $\leq^{\partial}$ in Definition 1 denotes the dual (i.e. converse) of order relation $\leq$, i.e. $\leq^{\partial}\equiv\geq$. Moreover, product lattice $(\mathsf{R},\leq^{\partial})\times(\mathsf{R},\leq)\equiv(\mathsf{R}\times\mathsf{R},\geq\times\leq)$ will be denoted by $(\mathcal{\Delta},\leq)$.

A generalized interval is denoted by $[x,y]$, where $x,y\in\mathsf{R}$. Apparently, the corresponding *meet* and *join* in lattice $(\mathcal{\Delta},\leq)$ are given, respectively, by $[a,b]\wedge[c,d]=[a\vee c,b\wedge d]$ and $[a,b]\vee[c,d]=[a\wedge c,b\vee d]$, where $a\wedge c$ $(a\vee c)$ denotes the *minimum* (*maximum*) of real numbers a and c.

The set of *positive* (*negative*) generalized intervals $[a,b]$, characterized by $a\leq b$ $(a>b)$, is denoted by $\mathcal{\Delta}_{+}$ $(\mathcal{\Delta}_{-})$. Apparently, lattice $(\mathcal{\Delta}_{+},\leq)$ of *positive* generalized intervals is *isomorphic*[1] to the lattice $(\tau(\mathsf{R}),\leq)$ of intervals (sets) in R, i.e. $(\tau(\mathsf{R}),\leq)\cong(\mathcal{\Delta}_{+},\leq)$. We have augmented lattice $(\tau(\mathsf{R}),\leq)$ by a *least* (empty) interval, denoted by $O=[+\infty,-\infty]$. Note that a *greatest* interval $I=[-\infty,+\infty]$

[1] A map $\psi:(\mathsf{P},\leq)\rightarrow(\mathsf{Q},\leq)$ is called *(order) isomorphism* if and only if both "$x\leq y\Leftrightarrow\psi(x)\leq\psi(y)$" and "$\psi$ is onto Q". Two lattices $(\mathsf{P},\leq)$ and $(\mathsf{Q},\leq)$ are called *isomorphic*, symbolically $(\mathsf{P},\leq)\cong(\mathsf{Q},\leq)$, if and only if there is an isomorphism between them.

already exists in $\tau(\mathsf{R})$. Hence, the complete lattice $(\tau_O(\mathsf{R}) = \tau(\mathsf{R}) \cup \{O\},\leq)$ emerges — For simplicity, we use symbols O and I to denote the least and greatest element, respectively, in any complete lattice.

A (strictly) decreasing *bijective*, the latter means "one-to-one", function $\theta_{\mathsf{R}} : \mathsf{R} \to \mathsf{R}$ implies an isomorphism $(\mathsf{R},\leq) \cong (\mathsf{R},\geq)$; i.e. $x < y \Leftrightarrow \theta_{\mathsf{R}}(x) > \theta_{\mathsf{R}}(y)$, $x, y \in \mathsf{R}$. Furthermore, a strictly increasing function $v_{\mathsf{R}} : \mathsf{R} \to \mathsf{R}$ is a *positive valuation*[2] in lattice $(\mathsf{R},\leq)$. Therefore, function $v_{\Delta} : \Delta \to \mathsf{R}$ given by $v_{\Delta}([a,b]) = v_{\mathsf{R}}(\theta_{\mathsf{R}}(a)) + v_{\mathsf{R}}(b)$ is a positive valuation in lattice $(\Delta,\leq)$ [5]. It follows a metric function $d_{\Delta} : \mathsf{R} \to \mathsf{R}^{\geq 0}$ given by $d_{\Delta}([a,b],[c,d]) = v_{\Delta}([a,b] \vee [c,d]) - v_{\Delta}([a,b] \wedge [c,d]) = [v_{\mathsf{R}}(\theta_{\mathsf{R}}(a \wedge c)) - v_{\mathsf{R}}(\theta_{\mathsf{R}}(a \vee c))] + [v_{\mathsf{R}}(b \vee d) - v_{\mathsf{R}}(b \wedge d)]$. In particular, metric d_{Δ} is valid in lattice $(\Delta_+ \cup \{O\},\leq) \cong (\tau_O(\mathsf{R}),\leq)$.

Functions $\theta_{\mathsf{R}}(.)$ and $v_{\mathsf{R}}(.)$ can be selected in many different ways. For instance, choosing both $\theta_{\mathsf{R}}(x) = -x$ and $v_{\mathsf{R}}(.)$ such that $v_{\mathsf{R}}(x) = -v_{\mathsf{R}}(-x)$ it follows positive valuation $v_{\Delta}([a,b]) = v_{\mathsf{R}}(b) - v_{\mathsf{R}}(a)$; hence, it follows metric $d_{\Delta}([a,b],[c,d]) = [v_{\mathsf{R}}(a \vee c) - v_{\mathsf{R}}(a \wedge c)] + [v_{\mathsf{R}}(b \vee d) - v_{\mathsf{R}}(b \wedge d)]$ [6]. In particular, for $\theta_{\mathsf{R}}(x) = -x$ and $v_{\mathsf{R}}(x) = x$ it follows metric $d_{\Delta}([a,b],[c,d]) = |a-c| + |b-d|$. In general, *parametric* functions $\theta_{\mathsf{R}}(.)$ and $v_{\mathsf{R}}(.)$ may imply tunable nonlinearities.

The space Δ of generalized intervals is a *real linear space* [4], [8] with

- *addition* defined as $[a,b] + [c,d] = [a+c, b+d]$.
- *multiplication* (by a scalar $k \in \mathsf{R}$) defined as $k[a,b] = [ka, kb]$.

A generalized interval in real linear space Δ is also called *vector*. A lattice-ordered vector space is called *vector lattice* [4].

A subset C of a linear space is called *cone* if and only if for $x_1, x_2 \in C$ and real numbers $\lambda_1, \lambda_2 \geq 0$ it follows $(\lambda_1 x_1 + \lambda_2 x_2) \in C$. It turns out that set Δ_+ is a cone. Likewise, set Δ_- is a cone.

2.2 The Cone Lattice $(\mathsf{F},\leq)$ of Interval Numbers (INs)

Generalized interval analysis in the previous section is useful for studying *interval numbers* (INs) in this section. A more general number type is defined first, in the following.

Definition 2. *A* generalized interval number*, or* GIN *for short, is a function* $G : (0,1] \to \Delta$.

Let G denote the set of GINs. It turns out that $(\mathsf{G},\leq)$ is a complete lattice since $(\mathsf{G},\leq)$ is the Cartesian product of complete lattices $(\Delta,\leq)$.

Addition and multiplication are extended from Δ to G as follows.

- The *sum* $G_1 + G_2$, $G_1, G_2 \in \mathsf{G}$ is defined as $G_s : G_s(h) = (G_1 + G_2)(h) = G_1(h) + G_2(h)$, $h \in (0,1]$.

[2] *Positive valuation* is a function $v : (\mathsf{L},\leq) \times (\mathsf{L},\leq) \to \mathsf{R}$, which satisfies both $v(x) + v(y) = v(x \wedge y) + v(x \vee y)$ and $x < y \Rightarrow v(x) < v(y)$.

– The *product* kG_1, $k \in \mathsf{R}$ and $G_1 \in \mathsf{G}$, is defined as $G_p : G_p(h) = kG_1(h)$, $h \in (0,1]$.

Our interest here focuses on the *sublattice*[3] of *interval numbers* defined next.

Definition 3. *An* Interval Number*, or* IN *for short, is a GIN F such that both $F(h) \in (\Delta_+ \cup \{O\})$ and $h_1 \leq h_2 \Rightarrow F(h_1) \geq F(h_2)$.*

Let F denote the set of INs. Conventionally, a IN will be denoted by a capital letter in italics, e.g. $F \in \mathsf{F}$. Moreover, a N-tuple IN will be denoted by a capital letter in bold, e.g. $\mathbf{F} = (F_1, ..., F_N) \in \mathsf{F}^N$.

From Definition 3 it follows that a general IN F is written as the set union of (conventional) intervals, e.g. $F = \underset{h\in(0,1]}{\cup}\{[a_h, b_h]\}$, where both interval-ends a_h and b_h are functions of $h \in (0,1]$ such that $a_h \leq b_h$.

We point out that a IN is a mathematical object, which may be interpreted as a probability/possibility distribution, an interval, and/or a real number. For instance, IN $F = \underset{h\in(0,1]}{\cup}\{[a,b]\}$ represents interval $[a,b]$ including real numbers for $a = b$. Moreover, a IN F may represent a probability distribution such that interval $F(h)$ includes $100(1-h)\%$ of the distribution, whereas the remaining $100h\%$ is split even both below and above interval $F(h)$ [4], [7], [8]. In addition, a IN may represent a fuzzy number as explained in subsection 2.3 below. In all cases, a IN can be interpreted as a *granule* (of information).

It has been shown that for $F, E \in \mathsf{F}$ there follow both $(F \wedge E) \in \mathsf{F}$ and $(F \vee E) \in \mathsf{F}$ [9]. Hence, $(\mathsf{F}, \leq)$ is a lattice with ordering $F_1 \leq F_2 \Leftrightarrow F_1(h) \leq F_2(h), \forall h \in (0,1]$.

The following proposition introduces a metric in lattice $(\mathsf{F}, \leq)$ based on a positive valuation function $v_{\mathsf{R}} : \mathsf{R} \to \mathsf{R}^{\geq 0}$ [9].

Proposition 1. *Let F_1 and F_2 be INs in the lattice $(\mathsf{F}, \leq)$ of INs. Assuming that the following integral exists, a metric function $d_{\mathsf{F}} : \mathsf{F} \times \mathsf{F} \to \mathsf{R}^{\geq 0}$ is given by*

$$d_{\mathsf{F}}(F_1, F_2) = \int_0^1 d_{\Delta}(F_1(h), F_2(h))dh \tag{1}$$

We remark that a Minkowski metric $d_p : \mathsf{F}^N \times \mathsf{F}^N \to \mathsf{R}^{\geq 0}$ can be defined between two N-tuple INs $\mathbf{F}_1 = [F_{1,1}, ..., F_{1,N}]^T$ and $\mathbf{F}_2 = [F_{2,1}, ..., F_{2,N}]^T$ as

$$d_p(\mathbf{F}_1, \mathbf{F}_2) = [d^p_{\mathsf{F}}(F_{1,1}, F_{2,1}) + ... + d^p_{\mathsf{F}}(F_{1,N}, F_{2,N})]^{1/p} \tag{2}$$

Minkowski metric $d_p(\mathbf{F}_1, \mathbf{F}_2)$ may involve a point $\mathbf{x} = [x_1, ..., x_N]^T \in \mathsf{R}^N$ such that an entry x_i is represented by *trivial* IN $x_i = \underset{h\in(0,1]}{\cup}\{[x_i, x_i]\}$, $i = 1, ..., N$.

Space F is a *cone* for $F_1, F_2 \in \mathsf{F}$ and real numbers $\lambda_1, \lambda_2 \geq 0$ it follows $(\lambda_1 F_1 + \lambda_2 F_2) \in \mathsf{F}$.

[3] A *sublattice* of a lattice $(\mathsf{L}, \leq)$ is another lattice $(\mathsf{S}, \leq)$ such that $\mathsf{S} \subseteq \mathsf{L}$.

2.3 Perspectives

A fundamental result in *fuzzy set theory* is the "resolution identity theorem", which states that a fuzzy set can, equivalently, be represented either by its membership function or by its α-cuts [19]. The aforementioned theorem was given little attention in practice, to-date. However, some authors have capitalized on it by designing fuzzy inference systems (FIS) based on α-cuts of fuzzy numbers, i.e. based on intervals in $\tau(\mathsf{R})$ [17], [18]. More specifically, advantages include faster (parallel) data processing "level-by-level" as well as "orders-of-magnitude" smaller computer memory requirements for representing, equivalently, fuzzy sets with arbitrary membership functions.

This work builds on the resolution identity theorem by, first, dropping the possibilistic interpretation for a (fuzzy) membership function and, second, by considering its equivalent α-cuts (interval) representation.

3 The Proposed Techniques

This section outlines computational techniques for achieving a piecewise-linear approximation of nonlinear models based on INs. Further details will be presented in a future publication.

3.1 Structure Identification

Structure identification is a term from "fuzzy TSK system modeling" [12],[15],[16] meaning a partition of a model's input space in *subspaces*, or *clusters*, such that the output to an "input point $\mathbf{x} = [x_1, ..., x_N]^T$, within a cluster" is a (usually) linear combination of the N inputs $x_1, ..., x_N$. It turns out that the task of *structure identification* is not trivial as illustrated in the following.

Consider the data points shown together with piecewise-linear approximations of two different single-input-single-output models in Fig. 1(a) and Fig. 1(b), respectively. Fig. 1(a) demonstrates an *effective* partition (of the input space) characterized by a small approximation error, whereas Fig. 1(b) demonstrates an *ineffective* partition characterized by a large approximation error.

A structure identification method is proposed next based (1) on a novel SOM extension, and (2) on an advantageous, novel structure identification algorithm.

3.2 A SOM Extension

Each cell $C_{i,j}$ in the SOM proposed here stores both a N-dimensional IN $\mathbf{F}_{i,j} = [F_{i,j,1}, ..., F_{i,j,N}]^T$ and a $(N+1)$-dimensional vector $\mathbf{c}_{i,j} = [c_{i,j,0}, c_{i,j,1}, ..., c_{i,j,N}]^T$, where $i = 1, ..., I$, and $j = 1, ..., J$. On one hand, IN $\mathbf{F}_{i,j} \in \mathsf{F}^N$ represents a population of data assigned to cell $C_{i,j}$. On the other hand, vector $\mathbf{c}_{i,j} \in \mathsf{R}^{N+1}$ stores the parameters of the following hyperplane

$$p_{i,j}(\mathbf{x}) = c_{i,j,0} + c_{i,j,1}x_1 + c_{i,j,2}x_2 + ... + c_{i,j,N}x_N \tag{3}$$

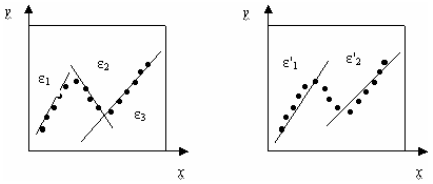


Fig. 1. Two different, piecewise-linear single-input-single-output models. (a) This model partitions the input space *effectively* with a small approximation error using three lines. (b) This model partitions the input space *ineffectively* with a large approximation error using two lines.

A cell is called *nonempty* if at least one datum is assigned to it. A nonempty cell represents a rule. In particular, the N INs in $\mathbf{F}_{i,j}$ of cell $C_{i,j}$ correspond to a (fuzzy) rule antecedent, whereas the $N+1$ hyperplane parameters in $\mathbf{c}_{i,j}$ constitute the corresponding rule's consequent.

3.3 INSOM: A Structure Identification Algorithm

Structure identification is carried out using the novel algorithm INSOM, below.

3.4 Parameter Identification

Algorithm INSOM above induces an "initial" (piecewise-linear) model from a series $(\mathbf{x}_k, y_k) \in \mathsf{R}^N \times \mathsf{R}$, $k = 1, 2, ..., n$ of training data. The objective in this section is to compute a globally optimum model.

The output of the aforementioned "initial" model is written analytically as

$$\widehat{y}(\mathbf{x}_k) = c_0 + \sum_{i=1}^{L} (c_{i,0}\sigma_i + \sum_{j=1}^{N} c_{i,j}\sigma_i x_{k,j}) \tag{4}$$

where $\mathbf{x}_k = [x_{k,1}, ..., x_{k,N}]^T$, furthermore the σ_is are functions of the (known) INs. In conclusion, a globally optimum set of hyperplanes is computed by algorithm WRLS in the Appendix.

Further improvement was sought by optimal parameter estimation techniques, which replaced a IN $F_{i,j}$ by IN $F'_{i,j} = a_{i,j}F_{i,j} + b_{i,j}$, where $a_{i,j} \in (0, 3]$ is a *scaling* parameter and $b_{i,j} \in [-1, 1]$ is a *translation* parameter, $i = 1, ..., L, j = 1, ..., N$. More specifically, the task was to compute optimal INs $F'_{i,j}$, in a mean square error sense, by optimal parameter $a_{i,j}$, $b_{i,j}$ estimation.

Optimization was pursued by genetic algorithms (GA) [1],[12], where the phenotype of an "individual" consisted of specific values of parameters $a_{i,j}, b_{i,j}$. There was a total number of $2 \times N \times L$ parameters binary-encoded to the chromosome of an "individual". We included 25 "individuals" per generation.

In conclusion, we point out that our "initial" model was computed by algorithm INSOM for structure identification without any employment of fuzzy

Algorithm 1 INSOM: A Structure Identification Algorithm

$I \leftarrow$ Number of rows in a SOM grid/map
$J \leftarrow$ Number of columns in a SOM grid/map
$d_\theta \leftarrow 0.8$, $\ell_\theta \leftarrow n/10$ *//user-defined parameters d_θ and ℓ_θ*
createANDinitializeMap(I, J)
for $r = 1$ to N_{epochs} **do** *//for each epoch*
  Calculate $B_{p,q}(r), a(r)$ *//$B_{p,q}(r)$ is a neighborhood; $a(r)$ is a weight coefficient*
  $w_k \leftarrow a(r)$
  for $k = 1$ to n **do** *//for each input datum $(\mathbf{x}_k, y_k)$*
    FindTheWinner$(\mathbf{x}_k, y_k)$
    $p \leftarrow$ winner row
    $q \leftarrow$ winner column
    Assign$(\mathbf{x}_k, y_k, p, q)$ *//assign input datum $(\mathbf{x}_k, y_k)$ to winner cell $C_{p,q}$*
    for $i = 1$ to I **do** *//for each row*
      for $j = 1$ to J **do** *//for each column*
        if $C_{i,j} \in B_{p,q}(r)$ **then**
          WRLS$(i, j, w_k, \mathbf{x}_k, y_k)$
        end if
      end for//*for j*
    end for//*for i*
  end for//*for k*
  ResetCellsConditionally(ℓ_θ)
  ComputeINs()
  MergeSimilarCells(d_θ)
end for//*for r*

logic. Whereas, thereafter, parameter identification was pursued based on standard fuzzy TSK modeling techniques.

4 Preliminary Experimental Results

The effectiveness of our proposed (piecewise-linear approximation) techniques is demonstrated in this preliminary work on a single-input-single-output nonlinear system. In the interest of simplicity positive valuation function $v_{\mathsf{R}}(x) = x$ was employed. Furthermore, both input- and output- data were normalized in the interval $[0, 1]$ by straightforward linear transformation. At the end of all computations, the output data were transformed back to their original domain for meaningful comparisons.

We considered the simple system described by the following equation.

$$y = sin(10x) \tag{5}$$

where $x \in [0, 1]$.

Forty input/output data pairs $(x_k, y_k) \in \mathsf{R} \times \mathsf{R}$, $k = 1, ..., 40$ were randomly (uniformly) generated. The scatter plot of the generated input/output data points is shown in Fig. 2(a). Following a popular practice, we employed the same data set for both training and testing. No validation set was employed.

A 4×4 SOM grid was used to compute a TSK model. The structure identification algorithm was applied for $N_{epochs} = 100$ epochs resulting in five nonempty cells — Recall that a nonempty cell represents a rule. The IN/antecedent and the hyperplane/consequent (the latter is a line here) in each cell are shown in Fig. 2(b) and Fig. 2(a), respectively. A visual inspection of Fig. 2 reveals that the proposed method partitions the input space well.

5 Conclusion

This work has proposed a new paradigm, inspired from both Fuzzy Inference Systems (FISs) of TSK type and Self-Organizing Maps (SOMs), for piecewise-linear approximation of nonlinear models based on Interval Numbers (INs).

A unique advantage of INs here is their effectiveness in computing colinear points within a cluster as it will be detailed in a future publication. Another advantage of our proposed techniques is the fast induction of an optimal number of rules. Note that the employment of SOM in fuzzy system modeling applications has been rather sporadic to-date. Nevertheless, different authors have confirmed the capacity of SOM for rapid data processing [2]. In our future work we have also planned additional experiments including alternative data sets.

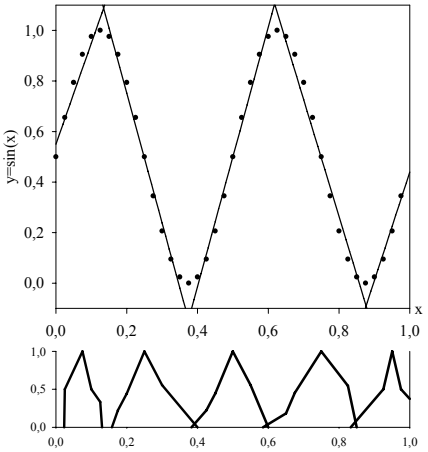


Fig. 2. (a) Scatter plot of function $y = sin(10x)$ including 40 input/output data points. The five lines correspond, respectively, to the consequents of five rules. (b) The five INs correspond, respectively, to the antecedents of five rules — Note that the corresponding consequent (line) for a IN is shown above the IN.

Appendix

Here we show the Weighted Recursive Least Squares (WRLS) algorithm for incremental learning.

Consider a series of data vectors $[x_{k,1}, ..., x_{k,M}, y_k]^T \in \mathsf{R}^M \times \mathsf{R}$, $k = 1, ..., n$. The WRLS algorithm computes incrementally the parameters of a hyperplane in R^{M+1}, *optimally fitted*, in a least square error sense, to the aforementioned data. The corresponding equations are shown next.

$$\begin{aligned} \mathbf{c}_{k+1} &= \mathbf{c}_k + \left(y_{k+1} - \mathbf{x}_{k+1}^T \cdot \mathbf{c}_k\right) \mathbf{k}_k \\ \mathbf{k}_k &= \frac{\mathbf{S}_k \mathbf{x}_{k+1}}{\frac{1}{w_k} + \mathbf{x}_{k+1}^T \mathbf{S}_k \mathbf{x}_{k+1}} \\ \mathbf{S}_{k+1} &= \left(\mathbf{I} - \mathbf{k}_k \mathbf{x}_{k+1}^T\right) \mathbf{S}_k \\ k &= 1, 2, ..., n. \end{aligned} \tag{6}$$

The equations above are initialized at $k = 0$ with $\mathbf{c}_0 = \mathbf{0}$ and $\mathbf{S}_0 = a\mathbf{I}$, where $a \in \mathsf{R}$ is typically large, e.g. $a = 1000$. Vector $\mathbf{c}_k = [c_{k,0}, c_{k,1}, ..., c_{k,M}]^T$ includes the *optimum* hyperplane parameters at a step.

References

1. Cordón, O., Gomide, F., Herrera, F., Hoffmann, F., Magdalena, L.: Ten years of genetic fuzzy systems: current framework and new trends. Fuzzy Sets and Systems **141(1)** (2004) 5–31

2. Er, M. J., Li, Z., Cai, H., Chen, Q.: Adaptive noise cancellation using enhanced dynamic fuzzy neural networks. IEEE Trans. Fuzzy Systems **13(3)** (2005) 331–342
3. Graña, M.: Lattice computing: lattice theory based computational intelligence. In: T. Matsuhisa, H. Koibuchi (eds) Proc. Kosen Workshop on Mathematics, Technology, and Education (MTE), Irabaki National College of Technology, Ibaraki, Japan, 15-18 February 2008, pp. 19–27
4. Kaburlasos, V. G.: Towards a Unified Modeling and Knowledge-Representation Based on Lattice Theory. Springer, Heidelberg, Germany, ser Studies in Computational Intelligence **27** (2006)
5. Kaburlasos, V. G., Athanasiadis, I. N., Mitkas, P. A.: Fuzzy lattice reasoning (FLR) classifier and its application for ambient ozone estimation. Intl. J. Approximate Reasoning **45(1)** (2007) 152–188
6. Kaburlasos, V. G., Kehagias, A.: Novel fuzzy inference system (FIS) analysis and design based on lattice theory. part I: working principles. Intl. J. General Systems **35(1)** (2006) 45–67
7. Kaburlasos, V. G., Kehagias, A.: Novel fuzzy inference system (FIS) analysis and design based on lattice theory. IEEE Trans. Fuzzy Systems **15(2)** (2007) 243–260
8. Kaburlasos, V. G., Papadakis, S. E.: Granular self-organizing map (grSOM) for structure identification. Neural Networks **19(5)** (2006) 623–643
9. Kaburlasos, V. G., Papadakis, S. E.: A granular extension of the fuzzy-ARTMAP (FAM) neural classifier based on fuzzy lattice reasoning (FLR). Neurocomputing (accepted).
10. Kohonen, T.: Self-Organizing Maps. Springer, Heidelberg, Germany, ser Information Sciences **30** (1995)
11. Papadakis, S. E., Kaburlasos, V. G.: Mass-grSOM: a flexible rule extraction for classification. In: Proc. 5th Workshop Self-Organizing Maps (WSOM 2005), Paris, France, 5-8 September 2005, pp. 553–560
12. Papadakis, S. E., Theocharis, J. B.: A GA-based fuzzy modeling approach for generating TSK models. Fuzzy Sets and Systems **131(2)** (2002) 121–152
13. Sugeno, M., Kang, G. T.: Fuzzy modelling and control of multilayer incinerator. Fuzzy Sets Systems **18(3)** (1986) 329–346
14. Sugeno, M., Tanaka, K.: Successive identification of a fuzzy model and its applications to prediction of a complex system. Fuzzy Sets Systems **42(3)** (1991) 315–334
15. Sugeno, M., Yasukawa, T.: A fuzzy-logic-based approach to qualitative modeling. IEEE Trans. Fuzzy Systems **1(1)** (1993) 7–31
16. Takagi, T., Sugeno, M.: Fuzzy identification of systems and its applications to modeling and control. IEEE Trans. Systems, Man & Cybernetics **15(1)** (1985) 116–132
17. Uehara, K., Fujise, M.: Fuzzy inference based on families of α-level sets. IEEE Trans. Fuzzy Systems **1(2)** (1993) 111–124
18. Uehara, K., Hirota, K.: Parallel and multistage fuzzy inference based on families of α-level sets. Information Sciences **106(1-2)** (1998) 159–195
19. Zadeh, L. A.: The concept of a linguistic variable and its application to approximate reasoning - III. Information Sciences **9(1)** (1975) 43–80

Computation of a Sufficient Condition for System Input Redundancy

S. E. Papadakis and V. G. Kaburlasos

Technological Institute of Kavala
Department of Industrial Informatics
65404 Kavala, Greece
{spap,vgkabs}@teikav.edu.gr

Abstract. The calculation of an optimal subset of inputs from a set of candidate ones is known in the bibliography of system modeling as *the input (or feature) selection problem*. In this work we introduce a remarkable attribute of the FLR classifier: it's capacity to identify redundant system inputs, from a set of input/output data. The proposed approach is applicable beyond R^N on any lattice ordered data set $\mathbb{L}^N$, which may include disparate types of data. Also, the proposed approach can deal with populations of data instead of crisp data vectors. Finally, it is highlighted that proposed approach can be employed for designing models with simple structure and significant performance. The method is successfully applied here on two well known *real world* classification problems, identifying redundant inputs and inducing FLR classifiers with simple structure and favorable classification performance.

Key words: Fuzzy Interval Number (FIN), Fuzzy Lattice Reasoning (FLR), classification, lattice theory

1 Introduction

One of the most important issues in computational intelligence is the structure identification of the model to be used for modeling effectively a physical system. According to established modeling methodology, physical systems are treated as *black boxes* where the only knowledge we have about them emanates from a finite set of samples, which are organized as ordered pairs of input (excitations) output (response) data. The next step in modeling, given a sufficient set of input/output data, is the definition of proper model's structure and a learning rule. Since samples are always finite, an effective modeling must produce models with sound generalization capacities. That is, models which have as close as possible behavior to physical system, especially on data lying outside the initial set of samples.

The related bibliography on effective model's definition including fuzzy modeling [1–5], neural networks [6–8] , self organizing maps [9], mathematical models [10–12], etc is vast. In general, structure's identification problem can be hierarchically divided into three principal sub-problems. The first one, namely *input*

pp. 23–31, ISBN 978–80–244–2112–4, Palacký University, Olomouc, 2008.

selection problem, is the choice from a set of (intuitively selected) candidate inputs of those ones that are necessary and sufficient to describe the specified target. Second, is the formation of inner model structure (i.e. number of rules, or neurons, input space partition etc.) in the space of selected inputs. Third, is the process of parameter identification (training) by applying a convenient learning rule.

The importance of input selection problem has been widely recognized by many researchers [13, 14, 4] . For instance, in [3, 4] the authors claim that using a scale of importance varying from one to one hundred, the input selection problem is rated as one hundred, the inner model's structure is rated as ten, while the task of parameters identification is rated as one.

A detailed review of several input selection approaches is referred to in [13, 15]. Although the proposed method encounter all aforementioned three types of structure identification problems, the discussion in this work focuses mainly on input selection issues.

The basic advantage of our methodology, presented below, is that it can be applied on non-homogeneous input/output data including real numbers, populations of data represented by probability/possibility distributions,etc. Disparate types of data can be represented by Fuzzy Interval Numbers (FINs) [16–18]. The set of FINs is lattice ordered, where a metric can be defined by establishing a parametric positive valuation function [16]. Parametric positive valuation functions introduce several tunable non-linearities, which are adjusted by a stochastic non linear optimization method toward maximization of performance.

The layout of this preliminary work is as follows: Section 2 presents the mathematical background. Section 3 presents the proposed method. Section 4 presents experimental results. Finally, section 5 concludes by summarizing the contribution of this work.

2 Mathematical Background

In order to make the proposed approach clear, some notions are shown next. A *generalized interval* is denoted by $[x, y],\ x, y \in R$. Let $(\Delta, \leq) = (R, \leq^{\partial}) \times (R, \leq)$ be the complete product lattice of generalized intervals. Note that the inverse $\geq$ of any order relation $\leq$ is itself an order relation. The order $\geq$ is called the *dual* of $\leq$, symbolically $\leq^{\partial}$, or $\leq^{-1}$, or $\geq$.

The corresponding *meet* and *join* in lattice $(\Delta, \leq)$ are given by $[a, b] \wedge [c, d] = [a \vee c, b \wedge d]$ and $[a, b] \vee [c, d] = [a \wedge c, b \vee d]$. Note that $(\Delta, \leq) = (\Delta_-, \leq) \cup (\Delta_+, \leq)$ where $(\Delta_-, \leq)$, $(\Delta_+, \leq)$ is the set of negative $(a > b)$ and positive $(a \leq b)$ generalized intervals, respectively. Note that lattice $(\Delta_+, \leq)$ is isomorphic to lattice $(\tau(R), \leq)$ of intervals in set R, That is $(\tau(R), \leq) \cong (\Delta_+, \leq)$.

A strictly decreasing function $\theta_R : R \to R$ implies an isomorphism $(R, \leq) \cong (R, \geq)$. Furthermore, a strictly increasing function $\nu_R : R \to R$ is a positive valuation function in lattice $(R, \leq)$. Hence, function $\nu_\Delta : \Delta \to R$, given by $\nu_\Delta([a, b]) = \nu_R(\theta_R(a)) + \nu_R(b)$ is a positive valuation function in lattice $(\Delta, \leq)$ [19]. It follows a metric $d_\Delta : \Delta \times \Delta \to R_0^+$ given by

$$d_{\Delta}([a,b],[c,d]) = [\nu_R(\theta_R(a \wedge c)) - \nu_R(\theta_R(a \vee c))] + [\nu_R(b \vee d) - \nu_R(b \wedge d)] \quad (1)$$

A **G**eneralized **I**nterval **N**umber is a function $f(0,1] \rightarrow \Delta$ where Δ denotes a complete product lattice $(\Delta, \leq) = (R, \leq^{\partial}) \times (R, \leq)$ of generalized intervals.

2.1 Fuzzy Interval Numbers

Fuzzy Interval Numbers or FINs have been extensively described in [16], [17], [18]. Let $\mathbb{L}$ to be the Lattice of FINs. Then a N-tuple FIN $\mathbf{F} \in \mathbb{L}^N$. A FIN F can be represented as the union set of generalized intervals $F = \underset{h \in (0,1]}{\cup} \{[a_h, b_h]\}$. IF $a_h = b_h \forall h \in (0,1]$ the FIN is called *trivial FIN*. Given a strictly increasing function: $\nu(.)$ and a strictly decreasing one: $\theta(.)$ an inclusion measure is defined as a function $\sigma_{\mathbb{L}} : \mathbb{L} \times \mathbb{L} \rightarrow [0,1]$, given by:

$$\sigma_{\mathbb{L}}(\mathbf{F}, \mathbf{E}) = \int_0^1 \frac{\sum_{i=1}^{N} [\nu_{R,i}(\theta_{R,i}(c_{i,h})) + \nu_{R,i}(d_{i,h})]}{\sum_{i=1}^{N} [\nu_{R,i}(\theta_{R,i}(a_{i,h} \wedge c_{i,h})) + \nu_{R,i}(b_{i,h} \vee d_{i,h})]} dh \quad (2)$$

Note that $\mathbf{F}, \mathbf{E}$ are N-tuple FINs. Functions $\nu : R \rightarrow R_0^+$ and $\theta : R \rightarrow R$ can be given by:

$$\begin{aligned} \nu_{R,i}(x) &= \tfrac{A_i}{1+e^{-\lambda_i(x-m_i)}} \\ \theta_{R,i}(x) &= 1 - 2m_i \end{aligned} \quad (3)$$

where $i = 1, 2, ..., N$. N denotes the number of inputs. $A_i, \lambda_i, m_i \in R$ tunable parameters. The size of a FIN is defined as a function $Z_F : \mathbb{F} \times \mathbb{F} \rightarrow R_0^+$, given by:

$$Z_F(\mathbf{F}) = \int_0^1 d_{\Delta}(a_h, b_h) dh \quad (4)$$

where $d_{\Delta}(a_h, b_h)$ is computed by Eq. (1)

3 The Proposed Method

The granular FLR is a set of labeled pairs $(\mathbf{E}_i, c_i)$ or *granules* , each represented by a N-tuple FIN $\mathbf{E}_i$ and a label c_i. Linguistically, a granule defines a rule of the form: *IF datum* $\boldsymbol{F}_j$ *is included in granule* $\boldsymbol{E}_i$ *then the class of datum* $\boldsymbol{F}_j$ *is* c_i. Hence a set $RB = \{E_i, c_i\}$ of granules defines a rule base for the FLR classifier.

The FLR algorithm is divided in two parts:

Algorithm 1 FLR for training

1: Initialize $RB = \{\mathbf{E}_\ell, c_\ell\} \mid \ell = 1, 2, ..., L$ of granules $\mathbf{E}_\ell$. $c_\ell \in C$ is the label of granule $\mathbf{E}_\ell$.
2: Do *set* all pairs in RB. Present the next input pair $(\mathbf{F}_i, c_i)$, i=1,...,n. Compute the degree of inclusion $\sigma_F(\mathbf{F}_i \leq \mathbf{E}_\ell)$ of input granule $\mathbf{F}_i$ to all granules $\mathbf{E}_\ell$, $\ell = 1, ..., L$.
3: IF no pairs are *set* in RB then store input pair $(\mathbf{F}_i, c_i)$ in RB, $L = L + 1$, goto 2.
4: The winner among *set* pairs in RB is $\mathbf{E}_j, c_j$ such that $j = \underset{\ell \in \{1,...L\}}{\arg\max} \{\sigma_F(\mathbf{F}_i \leq \mathbf{E}_\ell)\}$.
5: *The Assimilation Condition*: Both 1.) The size $Z(\mathbf{F}_i \vee \mathbf{E}_j)$ of granule $\mathbf{F}_i \vee \mathbf{E}_j$ is less than a user defined threshold size Z_{crit} and 2.) $c_i = c_j$.
6: if the *Assimilation Condition* is not satisfied then *reset* the winner and goto 3. Else, replace the winner with granule $\mathbf{F}_i \wedge \mathbf{E}_j$; and goto 2.

Algorithm 2 Stochastic optimization of tunable FLR parameters

1: Select a population of individuals and encode $Z_{crit}, A_i, \lambda_i, m_i$ where $i = 1, 2, ..., N$ into chromosome.
2: For each individual apply algorithm 1 and calculate its Fitness given by Eq. (5), (see Section 4).
3: Apply genetic operators to produce next generation.
4: if stopping criterion is not satisfied then goto 2.
5: Store the trained FLR, consisting of a RB of labeled granules and fine tuned parameters $Z_{crit}, A_i, \lambda_i, m_i$.

The results of algorithm 1 depend on both Z_{crit} and parameters $A_i, \lambda_i, m_i \,| i = 1, ..., N$ according to Eq. (2), (3). The tunable parameters Z_{crit} and parameters A_i, λ_i, m_i are optimized such that the rate of success classification is maximized.

In the case where $\nu_{R,i}(x) = const_i \; \forall \; x, i \mid const_i \in R$, it follows by equations 1, 2 that any calculated distances and similarity measures take constant values for every data $\mathbf{F}_j, \mathbf{F}_{k \neq j}$. Hence, there is no any discretization information and all data are equally distant (or equally similar) to each other. The classification process has to be entirely based only on the attributes (if any) with respective non flat sigmoid positive valuation functions. In other words attributes which have constant sigmoid positive valuation functions in the range of their interest, provides no discretization information and may be omitted. The aforementioned reasoning is experimentaly verified next.

4 Experimental Results

It has to be stressed that here we use input/output data, which are real numbers. However, the method has been developed using FINs data representation, which are members of a lattice. Hence, without loss of generality, a FIN represents here a real number.

4.1 Fisher's IRIS Classification Benchmark

Fisher Iris benchmark data set, downloaded from the UCI machine learning repository [20], is used to demonstrate the proposed feature selection technique. It includes measurements regarding four crinum flower attributes including x_1:*sepal-length*, x_2:*sepal-width*, x_3:*petal-length*, x_4:*petal-width*. The crinums are classified in three classes, namely *versicolor* (i.e. class 1), *setosa* (i.e. class 2), and *virginica* (i.e. class 3). In all, there are fifty 4-dimensional vectors per class available. After a random data permutation we employed the first 50 data vectors (33.33%) for training the next 50 (33.33%) for validation and the last 50 (33.33%) for testing. Each input datum is considered as a 4-tuple trivial FIN. Each sigmoid $\nu_i(x) \mid i = 1, 2, 3, 4$, is defined by three tunable parameters A_i, λ_i, m_i.

The genetic algorithm, presented in [21], is used for the optimization of FLR. The optimization of FLR lies in the calculation of sigmoid's parameter such that the rate of successful classification on both training and validation data set is as high as possible. Sigmoid parameters are binary encoded into the chromosome of every individual, using a word of 16 bits per parameter. Each individual represents a FLR model which is created using the encoded parameters' value, and applying the algorithm 2 on the training data set. Hence, the percent classification rate on both training and validation data set, namely (R_{trn}) and (R_{val}) respectively, is calculated. The fitness function is calculated by Eq. (5)

$$Q(\mathbf{p}_s) = w \cdot R_{trn} + (1 - w) \cdot R_{val} + \frac{0.1}{L} \tag{5}$$

The parameter $\mathbf{p}_s = [Z_{crit}, A_1, \lambda_1, m_1, ..., A_N, \lambda_N, m_N]$ denotes the vector of tunable parameter values, which are encoded into the chromoshome of individual $s, s = 1, 2, ...S \mid S$ denotes the population size. For N attributes the vector $\mathbf{p}_s$ has $3N + 1$ elements. Parameter L is a positive integer, which depends on the value of parameter Z_{crit} and denotes the number of granules in RB that constitute the FLR model. Term $0.1/L$ in Eq. (5) is used to lead the evolution into FLR models with RB having small number of granules. Finally, $w = 0.5$ is a relaxation factor, used to direct the evolution out of over trained solutions. The GA population includes 25 individuals and the evolution terminates when the quality function of the elite individual remains intact for 50 successive generations.

The sigmoid functions of the trained FLR are illustrated in Fig. 1. Stated by experimental results, we conclude that attribute *sepal width* is negligible, since all *sepal width* input values are mapped to a constant value approximately equal to 2. As a result attribute *sepal width* does not provides any class discretization information and should be removed. Our claim was experimentally verified by recalculating the classification rate without using *sepal width* values. We remarked that ignorance of *sepal width* does not affect training, validation and checking classification rate.

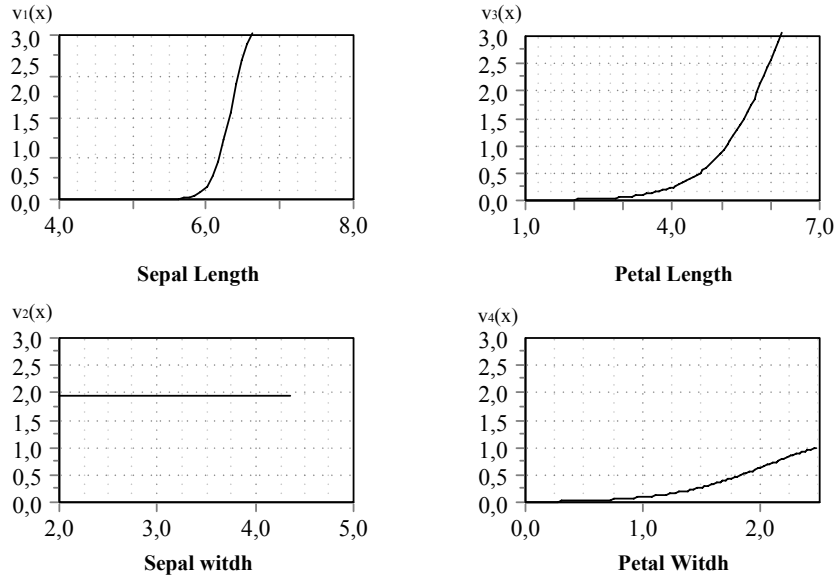


Fig. 1. The sigmoid positive valuation functions of the four attributes for the Fisher's IRIS classification benchmark.Since the sigmoid function for *Sepal width* is flat we conclude that the corresponding input is redundant.

4.2 The Wine Classification Benchmark

A more complex problem, the well known wine classification benchmark, is used in this example for further verification of proposed approach. In this problem is faced the classification of wines, in three categories according to 13 measured chemical attributes. Each input datum is created as a 13-tuple trivial FIN. Each datum's output takes an integer value 1,2, or 3 which represents the class of specific wine sample. The total of 178 input - output data were separated in three subsets, namely, training (60 data), validation (60 data) and testing (58 data) set. Algorithm FLR was applied using all 13 attributes. After training, the sigmoid functions of attributes are illustrated in Fig. 2. It is clearly observed that six attributes: Alcohol, Alkalinity of Ash, Magnesium, Total phenols, Nonflavonoid phenols and Proanthocyanins illustrate flat sigmoid in the range of interest. Thus, they are redundant. Ignoring redundant attributes a simplified FLR model classifier was built, which provides the same classification rate with FLR created by all thirteen attributes. This result confirms experimentally the statement that specific attributes are unnecessary.

5 Discussion and Conclusion

A novel and effective method for the determination of redundant attributes in classification problems was introduced. Our proposed method for input selection

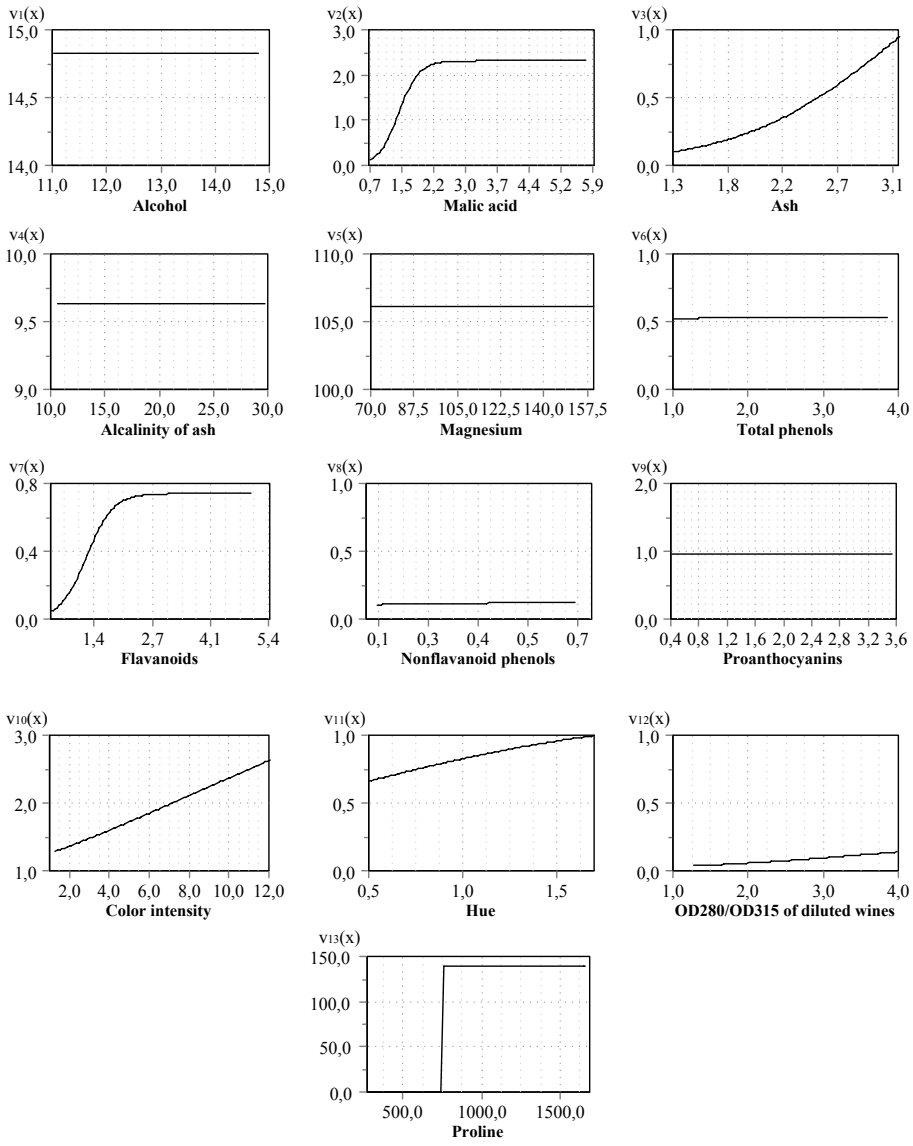


Fig. 2. The wine Classification Benchmark. Attributes with flat sigmoid positive valuation functions are redundunt ones since all their input values are mapped to a constant number.

using the FLR classifier computes only *sufficient* conditions. Hence when the sigmoid (positive valuation function) is constant in the range of interest of an input variable then we conclude that the latter input variable is redundant and can be omitted. In other words, for not-constant positive valuation functions we cannot tell whether the corresponding input variable is redundant or not. More work needs to be done in this direction in the future. Considering that we need only three tunable parameters per classifier's input, we conclude that the proposed methodology produces FLR classifiers with simple structure. Moreover, the proposed technique is applied on data sets which are Lattices. In a future work our proposed method will be applied on disparate types of data as symbols and populations of measurements represented by probability distributions.

References

1. Papadakis, S., Theocharis, J.: A GA-based fuzzy modeling approach for generating TSK models. Fuzzy Sets and Systems **131** (2002) 121–152
2. Simpson, P.: Fuzzy min-max neural networks -part i: Classification. IEEE Transactions on Neural Networks **3** (1992) 776–786
3. Sugeno, M., Tanaka, K.: Successive identification of a fuzzy model and its applications to prediction of a complex system. Fuzzy Sets and Systems **42** (1989) 315–334
4. Sugeno, M., Yasukawa, T.: A fuzzy-logic-based approach to qualitative modeling. IEEE Trans. Fuzzy Syst. **1** (1993) 7–31
5. Wang, X., Baets, B., Kerre, E.: A comparative study of similarity measures. Fuzzy Sets and Systems **73** (1995) 259–268
6. Chakraborty, D., Pal, N.: A neuro-fuzzy scheme for simultaneous feature selection and fuzzy rule-based classification. IEEE Transactions on Neural Networks **15** (2004) 110–123
7. Patra, J., Pal, R., Chatterji, B., Panda, G.: Identification of nonlinear dynamic systems using functional link artificial neural networks. IEEE Transactions on Systems, Man, and Cybernetics, Part B **29** (1999) 254–262
8. Rost, B., Sander, C.: Prediction of protein secondary structure at better than 70% accuracy. Journal of Molecular Biology **232** (1993) 584–599
9. Kohonen, T., Somervuo, P.: How to make large self-organizing maps for nonvectorial data. Neural Networks **15** (2002) 945–952
10. Lichstein, J., Simons, T., Shriner, S., Franzreb, K.: Spatial autocorrelation and autoregressive models in ecology. Ecological Monographs **72** (2002) 445–463
11. Rabiner, Lawrence, R., Juang, Biing-Hwang: Introduction to hidden markov models. IEEE ASSP magazine **3** (1986) 4–16
12. Zhang, B.: An EM algorithm for a semiparametric finite mixture model. Journal of Statistical Computation and Simulation **72** (2002) 791–802
13. Arauzo-Azofra, A., Benitez, J., Castro, J.: Consistency measures for feature selection. Journal of Intelligent Information Systems **30** (2008) 273–292
14. Ivakhnenko, A., Ivakhnenko, N.: Long-term prediction of random processes by gmdh algorithms using the unbiasedness criterion and balance-of-variables criterion. Soviet Automation and Control **7** (1974) 40–45
15. Papadakis, S., Tzionas, P., Kaburlasos, V., Theocharis, J.: A genetic based approach to the Type I structure identification problem. Informatica **16** (2005) 365–382

16. Kaburlasos, V.: FINs lattice theoretic tools for improving prediction of sugar production from population of measurements. IEEE Transactions on System Man and Cybernetics, Part-B **34** (2004) 1017–1030
17. Kaburlasos, V., Papadakis, S.: Granular self organizing map (grSOM) for structure identification. Neural Networks **19** (2006) 623–643
18. Papadakis, S., Marinagi, C., Kaburlasos, V., Theodorides, M.: Estimation of industrial production using the granular self-organizing map (grSOM). In: Proceedings of the 12th Mediterranean Conference on Control and Automation (MED'04), Kusadasi, Turkey (2004) session TuM2-D,proceedings in CD-ROM.
19. Kaburlasos, V., Athanasiadis, I., Mitkas, P.: Fuzzy lattice reasoning (FLR) classifier and its application for ambient ozone estimation. International Journal of Approximate Reasoning **45** (2007) 152–188
20. Newman, D., Hettich, S., Blake, C., Merz, C.: UCI repository of machine learning databases. (1998)
21. Kazarlis, S., Papadakis, S., Theocharis, J.B., Petridis, V.: Micro-genetic algorithms as generalized hill-climbing operators for GA optimazation. IEEE Transactions on Evolutionary Computation **5** (2001) 204–217

An Approach from Lattice Computing to fMRI Analysis

Manuel Graña, Maite García-Sebastián, Ivan Villaverde, Elsa Fernandez

Grupo de Inteligencia Computacional, UPV/EHU
www.ehu.es/ccwitnco

Abstract. We introduce an approach to fMRI analysis based on the Lattice Associative Memory (LAM) Endmember Induction Heuristic Algorithm (EIHA). Induced endmembers are used to compute the activation levels of voxels as result of an unmixing process. The endmembers correspond to diverse activation patterns, one of these activation patterns corresponds to the rest state of the neuronal tissue. We introduce a lattice normalization which consists in the mean centering for each voxel patttern independently. This lattice normalization is needed to remove scaling effects that rendered our approach inapplicable. Results on a case study are encouraging.

1 Introduction

Human brain mapping is a rapidly expanding discipline, and in recent years interest has grown in novel methods for imaging human brain functionality. Noninvasive techniques can measure cerebral physiologic responses during neural activation. One of the relevant techniques is functional magnetic resonance imaging (fMRI) [6], which uses the blood oxygenation level dependent (BOLD) contrast. Slight physiological alterations, such as neuronal activation resulting in changes of blood flow and blood oxygenation, are detected. These signal changes are related to changes in the concentration of deoxyhemoglobin, which acts as an intravascular contrast agent for fMRI. The most fMRI examinations are performed with BOLD-based methods using techniques sensitive to local distortions in the magnetic field (susceptibility sensitive techniques). These are T2 weighted spin echo pulse sequences or T2* weighted gradient echo pulse sequences. The various fMRI-methods have a good spatial and temporal resolution, limited only by the precision with which the autoregulatory mechanisms of the brain adjust blood flow in space to the metabolic demands of neuronal activity. Since these methods are completely noninvasive, using no contrast agent or ionizing radiation, repeated single-subject studies are becoming feasible [5].

To evaluate the resulting fMRI image series, sophisticated algorithms and great computational power are needed to separate the physiologically induced signals from noise or from artifacts resulting from patient movement or MRI detection techniques [13]. Appropriate postprocessing procedures for fMRI are currently being developed at a very rapid pace. Since many research groups are working in this area, no consensus has been reached about the analysis methods of the functional data up to now. A further reason for the large variety of different postprocessing procedures is the lack of a complete underlying theory of the BOLD effect.

pp. 33–44, ISBN 978–80–244–2112–4, Palacký University, Olomouc, 2008.

The fMRI experiment consists of a functional template or protocol (e.g., alternating activation and rest for a certain time) that induces a functional response in the brain.The aim of an fMRI experiment is to detect this stimulus response, resulting from the BOLD effect, in a defined volume element (voxel). The functional information of a voxel has to be extracted from its functional time course. Therefore, for each functional time point one fMRI volume is recorded. The complete four-dimensional dataset (three dimensions in space, one dimension in time) consists of subsequently recorded three-dimensional (3-D) volumes and thus for each voxel of a volume a functional time course exists. The acquisition of these functional volumes runs over periods lasting up to several minutes.

There are a number of sources of noise in the fMRI signal [14] that must be dealt with in appropriate preprocessing steps. The pulse sequence and the magnetic field stregth used can have an effect on the image quality. The long time duration of the experiments allow for head motions, even with strong restriction put into place. Experiment designs also affect the relative dynamics. We will assume that these noise sources have been dealt with appropriately.

The most extended analysis approach for fMRI signals is the Statistical Parametric Maps (SPM) [2] which has developed into a free open source software package. This method consists in the separate voxel based test of the generalized linear model (GLM) corresponding to the experimental design, followed by a segmentation of the spatial distribution of the individual voxel t-test values as a parametric map. There have been also approximations based on the Independent Component Analysis (ICA) [1] assuming that the time series observations are linear mixtures of independent sources which can not be observed. ICA assumes that the source signals are non-Gaussian and that the linear mixing process is unknown. The approaches to solve the ICA process obtain both the independent sources and the linear unmixing matrix. In previous works we have proposed an heuristic algorithm, which we have called Endmember Induction Heuristic Algorithm (EIHA) in [3] to solve a similar problem. There the assumption is that the data is generated from a set of endmembers which are the vertices of a convex polytope covering the data observations. Our approach is based on the relation between the Lattice Independence and Affine Independence [12], and the ability of Lattice Associative Memories to serve as detectors of Lattice Independent sets of vectors. The original works were devoted to unsupervised hyperspectral image segmentation, and here we try to apply it to fMRI analysis. The results are promising. This approach falls in the field of Lattice Computing algorithms, which have been introduced in [4] as the class of algorithms that either apply lattice operators inf and sup or use lattice theory to produce generalizations or fusions of previous approaches. In [4] an extensive and updated list of references, including previous works from authors contributing to this workshop, can be found.

The outline of the paper is as follows: Section 2 will present the undelying linear mixing model. Section 3 presents an sketch of the relation between Lattice Independence and Linear (Affine) Independence through the LAM theory. Section 4 recalls our heuristic algorithm. Section 5 presents results of the proposed approach on a case study. Section 6 provides some conclusions.

2 The Linear Mixing Model

The linear mixing model can be expressed as follows:

$$\mathbf{x} = \sum_{i=1}^{M} a_i \mathbf{s}_i + \mathbf{w} = \mathbf{S}\mathbf{a} + \mathbf{w}, \tag{1}$$

where $\mathbf{x}$ is the d-dimension pattern vector corresponding to the fMRI voxel time series vector, $\mathbf{S}$ is the $d \times M$ matrix whose columns are the d-dimension vertices of the convex region covering the data corresponding to the so called endmembers $\mathbf{s}_i, i = 1, .., M$, $\mathbf{a}$ is the M-dimension fractional abundance vector, and $\mathbf{w}$ is the d-dimension additive observation noise vector. The linear mixing model is subjected to two constraints on the abundance coefficients. First, to be physically meaningful, all abundance coefficients must be non-negative $a_i \geq 0, i = 1, .., M$. Second, to account for the entire composition, they must be fully additive $\sum_{i=1}^{M} a_i = 1$. That means that we expect the vectors in $\mathbf{S}$ to be affinely independent and that the convex region defined by them includes *all* the data points.

Once the convex region vertices have been determined the unmixing process is the computation of the matrix inversion that gives the coordinates of the point relative to the convex region vertices. The simplest approach is the unconstrained least squared error (LSE) estimation given by:

$$\widehat{\mathbf{a}} = \left(\mathbf{S}^T\mathbf{S}\right)^{-1}\mathbf{S}^T\mathbf{x}. \tag{2}$$

The coefficients that result from equation 2 do not necessarily fulfill the non-negativity and full additivity conditions. For simplicity, we will use the unconstrained estimation of equation 2 to compute the abundance coefficients. Moreover, the heuristic algorithm EIHA described in section 4 from [3] always produces convex regions that lie inside the data cloud, so that enforcing the non-negative and additivity to one conditions would be impossible for some data points. Negative values are considered as zero values and the additivity to one condition is not important as long as we are looking for the maximum abundances to assign meaning to the resulting spatial distribution of the coefficients. These coefficients are interpreted as fMRI voxel activation. That is, high positive values are interpreted as high voxel activation if the corresponding endmember does not correspond to the random pattern obtained as the resting state response.

3 Lattice Independence and Lattice Autoassociative Memories

The work on Lattice Associative Memories (LAM) stems from the consideration of the algebraic lattice structure $(\mathbb{R}, \vee, \wedge, +)$ as the alternative to the algebraic framework given by the mathematical field $(\mathbb{R}, +, \cdot)$ for the definition of Neural Networks computation. The LAM were first introduced in [8,9] as Morphological Associative Memories, but we follow the new convention introduced in [11,12] because it sets the works in the more general framework of Lattice Computing. The operators $\vee$ and $\wedge$ denote, respectively, the discrete $\max$ and $\min$ operators (resp. $\sup$ and $\inf$ in a continuous setting).

Given a set of input/output pairs of pattern $(X,Y) = \left\{ \left(\mathbf{x}^{\xi}, \mathbf{y}^{\xi} \right) ; \xi = 1, .., k \right\}$, a linear heteroassociative neural network based on the pattern's cross correlation is built up as $W = \sum_{\xi} \mathbf{y}^{\xi} \cdot \left(\mathbf{x}^{\xi} \right)'$. Mimicking this constructive procedure [8,9] propose the following constructions of Lattice Memories (LM):

$$W_{XY} = \bigwedge_{\xi=1}^{k} \left[\mathbf{y}^{\xi} \times \left(-\mathbf{x}^{\xi} \right)' \right] \text{ and } M_{XY} = \bigvee_{\xi=1}^{k} \left[\mathbf{y}^{\xi} \times \left(-\mathbf{x}^{\xi} \right)' \right], \tag{3}$$

where $\times$ is any of the $\boxtimes$ or $\boxtimes$ operators. Here $\boxtimes$ and $\boxtimes$ denote the $\max$ and $\min$ matrix product [8,9]. respectively defined as follows:

$$C = A \boxtimes B = [c_{ij}] \Leftrightarrow c_{ij} = \bigvee_{k=1,\ldots,n} \left\{ a_{ik} + b_{kj} \right\}, \tag{4}$$

$$C = A \boxtimes B = [c_{ij}] \Leftrightarrow c_{ij} = \bigwedge_{k=1,\ldots,n} \left\{ a_{ik} + b_{kj} \right\}. \tag{5}$$

If $X = Y$ then the LM memories are Lattice Autoassociative Memories (LAM). Conditions of perfect recall by the LM and LAM of the stored patterns proved in [8,9] encouraged the research on them, because in the continuous case, the LAM is able to store and recall any set of patterns: $W_{XX} \boxtimes X = X = M_{XX} \boxtimes X$, for any X. However, this result holds when we deal with noise-free patterns. Research on robust recall [7,9,10] based on the so-called kernel patterns lead to the notion of morphological independence, in the erosive and dilative sense, and finally to the definition of Lattice Independence (LI) and Strong Lattice Independence (SLI). We gather some results from [12] that set the theoretical background for the approach to endmember induction applied.

Definition 1. *Given a set of vectors* $\left\{ \mathbf{x}^1, ..., \mathbf{x}^k \right\} \subset \mathbb{R}^n$ *a* linear minimax combination *of vectors from this set is any vector* $\mathbf{x} \in \mathbb{R}^n_{\pm\infty}$ *which is a* linear minimax sum *of these vectors:*

$$x = \mathcal{L}\left(\mathbf{x}^1, ..., \mathbf{x}^k \right) = \bigvee_{j \in J} \bigwedge_{\xi=1}^{k} \left(a_{\xi j} + \mathbf{x}^{\xi} \right),$$

where J *is a finite set of indices and* $a_{\xi j} \in \mathbb{R}_{\pm\infty}$ $\forall j \in J$ *and* $\forall \xi = 1, ..., k$.

Definition 2. *The* linear minimax span *of vectors* $\left\{ \mathbf{x}^1, ..., \mathbf{x}^k \right\} = X \subset \mathbb{R}^n$ *is the set of all linear minimax sums of subsets of* X, *denoted* $LMS\left(\mathbf{x}^1, ..., \mathbf{x}^k \right)$.

Definition 3. *Given a set of vectors* $X = \left\{ \mathbf{x}^1, ..., \mathbf{x}^k \right\} \subset \mathbb{R}^n$, *a vector* $\mathbf{x} \in \mathbb{R}^n_{\pm\infty}$ *is* lattice dependent *if and only if* $x \in LMS\left(\mathbf{x}^1, ..., \mathbf{x}^k \right)$. *The vector* $\mathbf{x}$ *is* lattice independent *if and only if it is not lattice dependent on* X. *The set* X *is said to be* lattice independent *if and only if* $\forall \lambda \in \{1, ..., k\}$, $\mathbf{x}^{\lambda}$ *is lattice independent of* $X \backslash \left\{ \mathbf{x}^{\lambda} \right\} = \left\{ \mathbf{x}^{\xi} \in X : \xi \neq \lambda \right\}$.

The definition of lattice independence supersedes and improves the early definitions [10] of erosive and dilative morphological independence, which, however, have more intuitive appealing. Nevertheless, this definition has the additional advantage of establishing a formal parallelism with the definition of linear independence.

Definition 4. *A set of vectors* $X = \left\{\mathbf{x}^1, ..., \mathbf{x}^k\right\} \subset \mathbb{R}^n$ *is said to be* max dominant *if and only if for every* $\lambda \in \{1, ..., k\}$ *there exists and index* $j_\lambda \in \{1, ..., n\}$ *such that*

$$x^\lambda_{j_\lambda} - x^\lambda_i = \bigvee_{\xi=1}^{k} \left(x^\xi_{j_\lambda} - x^\xi_i\right) \forall i \in \{1, ..., n\}.$$

Similarly, X *is said to be* min dominant *if and only if for every* $\lambda \in \{1, ..., k\}$ *there exists and index* $j_\lambda \in \{1, ..., n\}$ *such that*

$$x^\lambda_{j_\lambda} - x^\lambda_i = \bigwedge_{\xi=1}^{k} \left(x^\xi_{j_\lambda} - x^\xi_i\right) \forall i \in \{1, ..., n\}.$$

The expressions that compound this definition appeared in the early theorems about perfect recall of Morphological Associative Memories [8,9]. Their value as an identifiable property of the data has been discovered in the context of the formalization of the relationship between strong lattice independence, defined below, and the classical affine independence.

Definition 5. *A set of lattice independent vectors* $\left\{\mathbf{x}^1, ..., \mathbf{x}^k\right\} \subset \mathbb{R}^n$ *is said to be* strongly lattice independent *(SLI) if and only if* X *is max dominant or min dominant or both.*

As said before, min and max dominance are the conditions for perfect recall. Per construction, the column vectors of Lattice Autoassociative Memories are min or max dominant, depending of their erosive or dilative nature, therefore they will be strongly lattice independent, *if* they are lattice independent.

Conjecture 1. [12] If $X = \left\{\mathbf{x}^1, ..., \mathbf{x}^k\right\} \subset \mathbb{R}^n$ is strongly lattice independent then X is affinely independent.

This conjecture (stated as theorem in [11]) is the key result whose proof would relate the linear convex analysis and the non-linear lattice analysis. If true, it means that the construction of the LAM provides the starting point for obtaining sets of affine independent vectors that could be used as endmembers for the unmixing algorithms described in section 2.

Theorem 1. *[12] Let* $X = \left\{\mathbf{x}^1, ..., \mathbf{x}^k\right\} \subset \mathbb{R}^n$ *and let* W *(* M*) be the set of vectors consisting of the columns of the matrix* W_{XX} *(*M_{XX}*). Let* $F(X)$ *denote the set of fixed points of the LAM constructed from set* X*. There exist* $V \subset W$ *and* $N \subset M$ *such that* V *and* N *are strongly lattice independent and* $F(X) = F(V) = F(N)$ *or, equivalently,* $W_{XX} = W_{VV}$ *and* $M_{XX} = M_{NN}$.

The key idea of this theorem is to test the lattice independence of the already known as min or max dominant sets of vectors. Removing lattice dependent vectors will not affect this min/max dominance property. The smart way to test lattice dependence lies in the fact that removing a lattice dependent vectors does not alter the set of fixed points of the remaining ones. This theorem is proved following a constructive reasoning, giving way to an algorithm for the construction of the set of affine independent sets of vectors from LAM dicussed in [3,12].

Algorithm 1 Endmember Induction Heuristic Algorithm (EIHA)

1. Shift the data sample to zero mean $\{\mathbf{f}^c(i) = \mathbf{f}(i) - \overrightarrow{\mu}; i = 1, .., n\}$.
2. Initialize the set of vertices $E = \{\mathbf{e}_1\}$ with a randomly picked sample. Initialize the set of lattice independent binary signatures $X = \{\mathbf{x}_1\} = \left\{\left(e_k^1 > 0; k = 1, .., d\right)\right\}$
3. Construct the LAM's based on the lattice independent binary signatures: M_{XX} and W_{XX}.
4. For each pixel $\mathbf{f}^c(i)$
   (a) compute the noise corrections sign vectors$\mathbf{f}^+(i) = (\mathbf{f}^c(i) + \alpha\overrightarrow{\sigma} > \mathbf{0})$ and $\mathbf{f}^-(i) = (\mathbf{f}^c(i) - \alpha\overrightarrow{\sigma} > \mathbf{0})$
   (b) compute $y^+ = M_{XX} \boxtimes \mathbf{f}^+(i)$
   (c) compute $y^- = W_{XX} \boxtimes \mathbf{f}^-(i)$
   (d) if $y^+ \notin X$ or $y^- \notin X$ then $\mathbf{f}^c(i)$ is a new vertex to be added to E, execute once 3 with the new E and resume the exploration of the data sample.
   (e) if $y^+ \in X$ and $\mathbf{f}^c(i) > \mathbf{e}_{y^+}$ the pixel spectral signature is more extreme than the stored vertex, then substitute $\mathbf{e}_{y^+}$ with $\mathbf{f}^c(i)$.
   (f) if $y^- \in X$ and $\mathbf{f}^c(i) < \mathbf{e}_{y^-}$ the new data point is more extreme than the stored vertex, then substitute $\mathbf{e}_{y^-}$ with $\mathbf{f}^c(i)$.
5. The final set of endmembers is the set of original data vectors $\mathbf{f}(i)$ corresponding to the sign vectors selected as members of E.

4 Endmember Induction Heuristic Algorithm (EIHA)

For the sake of completeness we recall here our Endmember Induction Heuristic Algorithm (EIHA) maintaining the notation used in the original references. Let us denote $\left\{\mathbf{f}(i) \in \mathbb{R}^d; i = 1, .., n\right\}$ the high dimensional data that may be the time series in a fMRI voxels, $\overrightarrow{\mu}$ and $\overrightarrow{\sigma}$ are, respectively, the mean vector and the vector of standard deviations computed over the data sample, α the noise correction factor and E the set of already discovered vertices. The noise amplitude of the additive noise in equation (1) is $\overrightarrow{\sigma}$, the patterns are corrected by the addition and subtraction of $\alpha\overrightarrow{\sigma}$, before being presented to the LAM's. The gain parameter α controls the amount of flexibility in the discovering of new endmembers. Let us denote by the expression $\mathbf{x} > \mathbf{0}$ the construction of the binary vector $(\{b_i = 1 \text{ if } x_i > 0; b_i = 0 \text{ if } x_i \leq 0\}; i = 1, .., n)$. The detailed description of the steps in the heuristic algorithm is presented as Algorithm 1. The starting endmember set consists of a randomly picked pixel. However, this selection is not definitive, because the algorithm may later change this endmember for another, more extreme, one. The noise correction parameter α has a great impact on the number of endmembers found. Low values imply large number of endmembers. It determines if a vector is interpreted as a random perturbation of an already selected endmember.

5 A Case Study

The experimental data corresponds to auditory stimulation test data of single person. It is freely available from ftp://ftp.fil.ion.ucI.ac.uk/spm/data, the file name is snrfM00223.zip. These data are the result of the preprocessing pipeline that removes many noise sources.

These whole brain BOLD/EPI images were acquired on a modified 2T Siemens MAGNETOM Vision system. Each acquisition consisted of 64 contiguous slices. Each slice being a 2D image of one head volume cut. There are 64x64x64 voxels of size 3mm x 3mm x 3mm. The data acquisition took 6.05s, with the scan-to-scan repeat time (RT) set arbitrarily to 7s. 96 acquisitions were made (RT=7s) in blocks of 6, i.e., 16 42s blocks. The condition for successive blocks alternated between rest and auditory stimulation, starting with rest. Auditory stimulation was bi-syllabic words presented binaurally at a rate of 60 per minute. The functional data starts at acquisition 4, image snrfMOO223-004. Due to T1 effects it is advisable to discard the first few scans (there were no "dummy" lead-in scans). We have discarded the first 10 scans. Figure 1 shows the plots of the time series corresponding to the slice #30 of the collected volume. It can be appreciated that there are an intensity displacement filling the whole range of intensities. There are few voxels showing an activation pattern on the top of the plots, and the vast majority of the voxels time series correspond to random non activation patterns at diverse intensities. The result of our algorithm applied to these raw data would be trivial and uninteresting, we would find the upper and lower patterns.

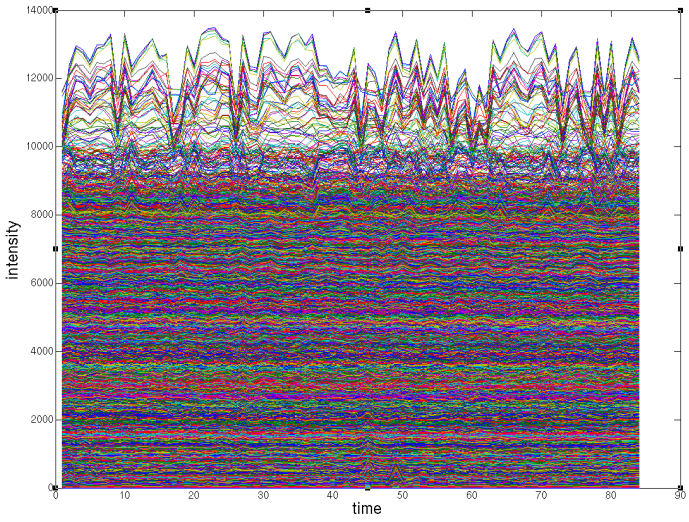


Fig. 1. Plot of the time series for the voxels of axial slice #30.

After susbtracting its mean to each voxel time series independently, the plots are collapsed as shown in figure 2 around the origin. It can be appreciated that most deactivated voxels are collapsed into a quite similar pattern, and that the diverse activation patterns stand out. This mean substraction corresponds to an scale normalization in the lattice computing sense. It removes scale effects that hinder the detection of meaningful lattice independent vectors.

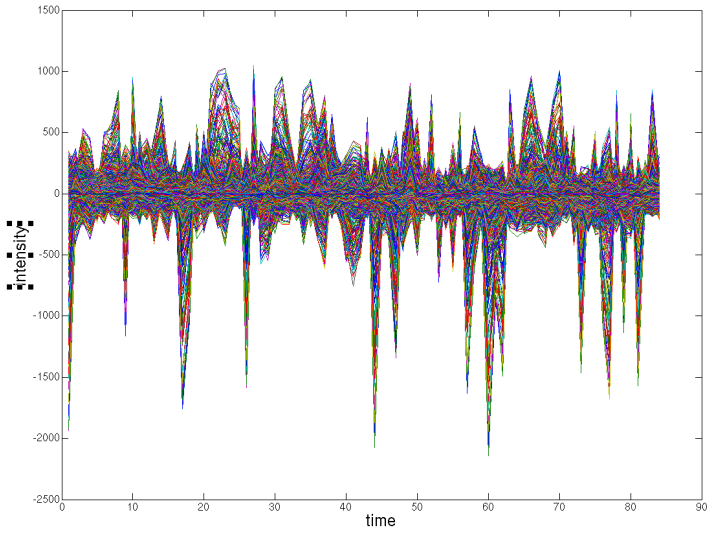


Fig. 2. Plots of time series of voxels in axial slice #30 after susbtracting their mean values from them. The time series are collapsed in the neighborhood of zero.

The application of the EIHA algorithm with $\alpha = 20$ to the lattice normalized time series of the whole 3D volume produces the collection of eleven endmembers shown in figure 3. Attending to the intensity scale it can be assumed that the first endmember (top left plot) corresponds to the non activation pattern, while the remaining endmembers correspond to some kind of activation pattern. These patterns correspond to individual voxels and do not reflect aggregated spatial behaviors like in other approaches.

The unmixing process applied to the whole volume voxels with the eleven endmembers of figure 3 produces the abundance images that we interpret as the activation levels of each pattern. To give an interpretation of these activation levels, we refer to the standard results obtained with the SPM software, presented in figure 4 as localized in the Talairach space, in sagital, coronal and axial cuts. There it can be observed that the activation appears around the axial slice #30. For this reason we present the abundances computed on this slice in figure 5. The figure presents the original slice where the voxels with abundance value above the 99% percentile of the distribution of this endmember abundance over the whole volume are set to white. It can be appreciated that the abundances for endmembers #8 and #11 have some activation regions overlapping the standard detections in figure 4, as well as showing some spurious activation regions. For a complete review of the activation detected by the endmember #11 abundances we show the 99% percentile detection on all the slices in the axial direction in figure 6. The figure shows that there are many spurious detections in slices corresponding to brain regions far away from the activations shown in figure 4.

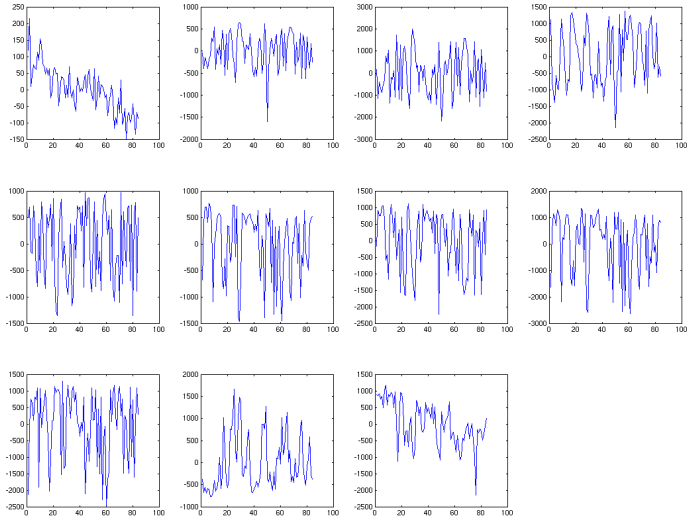


Fig. 3. Eleven endmbers detected by EIHA over the lattice normalized time series of the whole 3D volume.

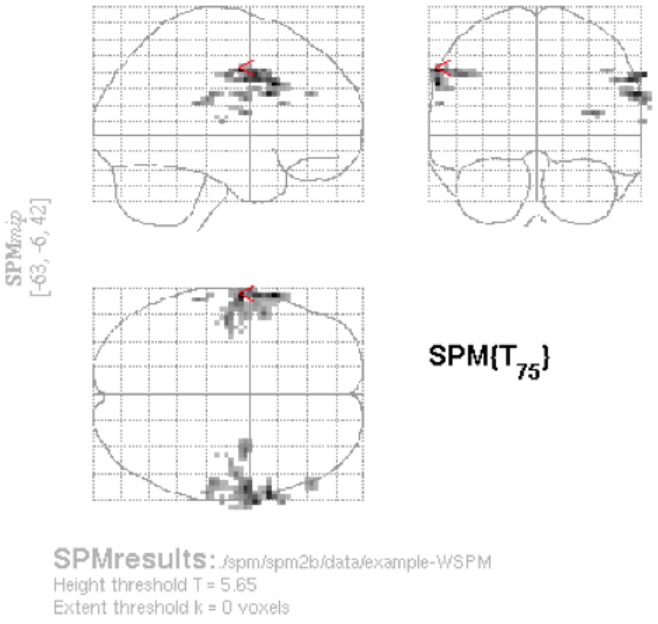


Fig. 4. Activation maps from SPM results over the experimental data

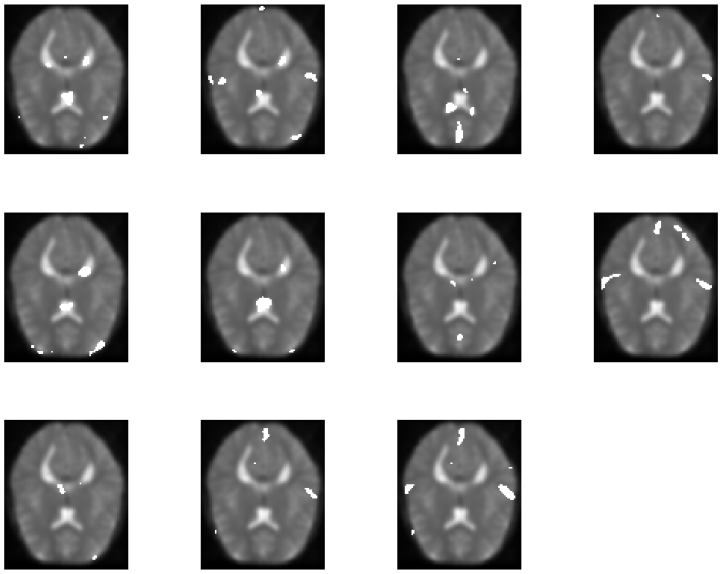


Fig. 5. Abundances for axial slice #30 for all eleven endmembers. White voxels correspond to abundance values above the 99% percentile of the distribution of the abundances for each endmember at this slice.

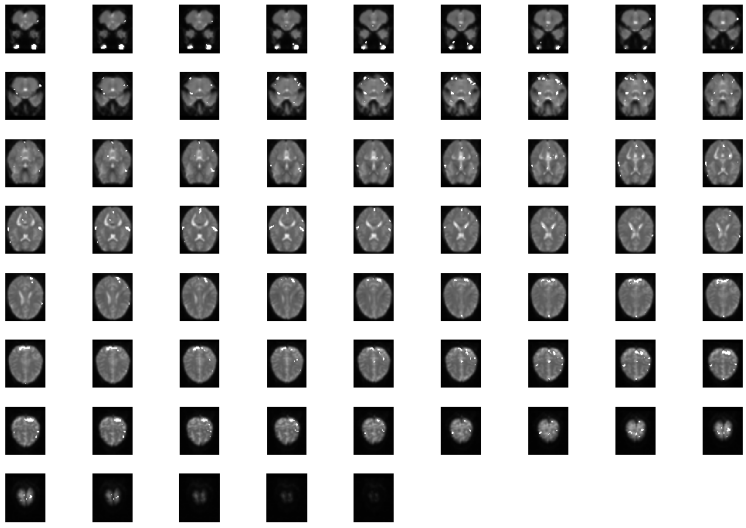


Fig. 6. Activations detected by the 99% percentile of the abundance images of endmember #11 of figure 5 in the axial direction.

6 Conclusions and Discussion

We have proposed and applied the endmember induction algorithm EIHA discussed in [3] to the task of brain region activation in fMRI. The idea is to try to mimic ICA application to fMRI activation detection [1,13], where the sources correspond to endmembers detected by the EIHA algorithm and the activation is computed as the abundance images obtained by unmixing the voxel time series on the basis of the found endmembers. The first obstacle that we find in this endeavor is that the distribution of the time series is not well aspected for the detection of Lattice Independence as a meaningful characteristic. In fact the voxel's fMRI time series show a dense distribution of intensity displacements from the origin, so almost all of them are lattice dependent and our proposed algorithm only recovers two endmembers. To overcome this problem we propose a normalization which corresponds to a scale normalization in the sense of Lattice Computing. We substract its mean to each voxel time series. The resulting lattice normalized data set shows a much more rich structure in terms of Lattice Independence. Our computational experiment with a well known fMRI data set, provided with the distribution of the SPM software, show some promising results in the sense that we are able to partially detect activations as the standard analysis with the SPM software. There are however some false detections that show that our approach is not consistent with the SPM analysis. We think that further research may lead to obtain consistent results. One important aspect of SPM is its process of the individual voxel t-test as a random field, this processing is lacking in our initial works. Finding ways to harmonize global random field analysis and our lattice computing approach may lead to such consistency.

References

1. Calhoun, V.D.,Adali, T.: Unmixing fMRI with independent component analysis, Engineering in Medicine and Biology Magazine, IEEE (2006) 25(2):79-90
2. Friston, K.J., Holmes, A.P., Worsley, K.J., Poline, J.P., Frith, C.D., Frackowiak, R.S.J.: Statistical parametric maps in functional imaging: A general linear approach, Hum. Brain Map. (1995) 2(4):189-210
3. Graña, M., Villaverde, I., Maldonado, J.O., Hernandez, C.: Two Lattice Computing approaches for the unsupervised segmentation of Hyperspectral Images, Neurocomputing (2008) in press.
4. Graña, M.: A Brief Review of Lattice Computing, Proc. WCCI 2008, pp. 1777-1781
5. Muller, H.-P., Kraft, E., Ludolph, A., Erne, S.N.: New methods in fMRI analysis, Engineering in Medicine and Biology Magazine, IEEE, (2002) 21(5):134-142
6. Pekar, J.J.: A brief introduction to functional MRI, Engineering in Medicine and Biology Magazine, IEEE (2006) 25(2):24-26
7. Raducanu, B., Graña, M., Albizuri, X.: Morphological scale spaces and associative morphological memories: results on robustness and practical applications, J. Math. Imaging and Vision (2003) 19(2):113–122
8. Ritter, G. X., Sussner, P., Diaz-de-Leon, J. L.: Morphological associative memories. IEEE Trans. on Neural Networks, (1998) 9(2):281–292,
9. Ritter, G. X., Diaz-de-Leon, J. L., Sussner, P.: Morphological bidirectional associative memories. Neural Networks, (1999) 12:851–867,

10. Ritter, G.X., Urcid, G., Iancu, L.: Reconstruction of patterns from noisy inputs using morphological associative memories, J. Math. Imaging and Vision (2003) 19(2):95–112
11. Ritter, G.X., Gader, P., Fixed points of lattice transforms and lattice associative memories In: Hawkes P (ed.) Advances in Imaging and Electron Physics (2006) 144:165–242, Elsevier, San Diego, CA
12. Ritter, G.X., Urcid, G., Schmalz, M.S.: Autonomous single-pass endmember approximation using lattice auto-associative memories Neurocomputing (2008) in press
13. Sarty, G.E.: Computing Brain Activation Maps from fMRI Time-Series Images, Cambridge University Press, 2007
14. Strother, S.C., Evaluating fMRI preprocessing pipelines, Engineering in Medicine and Biology Magazine, IEEE (2006) 25(2):27-41

Redundant Encoding of Patterns in Lattice Associative Memories

Gonzalo Urcid[1,⋆], Gerhard X. Ritter[2], and José-Angel Nieves-V.[1]

[1] Optics Department, INAOE, Tonantzintla, Pue. 72000, Mexico
{gurcid,janieves}@inaoep.mx
[2] CISE Department, University of Florida,
Gainesville, FL 32611–6120, USA
ritter@cise.ufl.edu

Abstract. Lattice matrix auto-associative memories also known as auto-associative morphological memories are artificial neural networks used to store and recall a finite set of binary or real valued patterns. They differ from other auto-associative memory models in the way exemplar patterns are encoded in the network as well as in the computation performed to recall a pattern. Both storage and recall mechanisms are based on minimax algebra operations that result in unique memory properties, such as, single step recall, perfect retrieval of all exemplar patterns, and infinite storage capacity. Two dual lattice matrix auto-associative memories have been developed so far. The min-memory is robust to erosive noise and the max-memory is robust to dilative noise; however, neither of these memories is able to recall patterns degraded by mixed or random noise. This paper introduces a redundant encoding of patterns based on the geometrical characterization of the set of fixed points common to both memories. Redundancy changes the size and shape of attraction basins of exemplar patterns and expands the set of fixed points, hence recall capability of patterns corrupted with random noise is possible using a simple scheme based on this type of memory networks.

Key words: Fixed point sets, Lattice associative memories, Minimax algebra, Morphological associative memories, Noisy patterns, Pattern recall, Redundant encoding

1 Introduction

Lattice matrix auto-associative memories approach the problem of pattern association from a minimax point of view. The Hebbian law of correlation encoding is still used to store a set of patterns but modified accordingly to be consistent with the mathematical framework of minimax algebra. Pattern retrieval is performed with minimax matrix products in a similar way as usual matrix multiplication of linear algebra is used in linear correlation associative memory models. Brief but complete summaries of the different developments in the field

⋆ Corresponding author. Fax: +52 (222) 247-2940; Tel.: +52 (222) 266-3100 Ext.8205

pp. 45–57, ISBN 978–80–244–2112–4, Palacký University, Olomouc, 2008.

of morphological associative memories can be consulted in [6, 10, 17, 18, 23]. More recently, the canonical *lattice auto-associative memories* (LAMs), also known as *auto-associative morphological memories*(AMMs), have been treated within the more general framework of lattice transforms. In this framework, the min-$\mathbf{W}_{XX}$ and max-$\mathbf{M}_{XX}$ auto-associative memories are examples of lattice transforms between n-dimensional real vectors. The set of fixed points of a lattice transform is a key concept since it provides a complete description of the transform behavior. Fixed points of AMMs where first presented in [15] and discussed further in [10, 18]; one of the main results is that $\mathbf{W}_{XX}$ and $\mathbf{M}_{XX}$, have the same set of fixed points denoted by $F(X)$. Recently, an algebraic as well as a geometrical characterization of the set of fixed points $F(X)$ has been established in [10, 12]. This work proposes to exploit the infinite storage capacity and to take advantage of the geometrical nature of $F(X)$, to enable recall capability of lattice matrix auto-associative memories for input patterns degraded by random noise.

Our work in the present paper is organized as follows: Section 2 provides the necessary matrix operations borrowed from minimax algebra for dealing with lattice matrix associative memories. Also, an abstract overview of the main theoretical results obtained from previous research on lattice auto-associative memories is given very briefly. Section 3 describes the redundant encoding technique and simple examples with small sized matrix associative memories are given to make explicit its potential. Section 4 presents some computational experiments with a set of high-dimensional patterns, consisting of gray-scale images, that demonstrate the recall capability of canonical LAMs or AMMs when presented with noisy inputs. Finally, conclusions of this research are given in Section 5.

2 Mathematical Background

Mathematical definitions and key results about lattice matrix associative memories are given here to support the geometrical encoding technique described in the next section. For further material on the mathematical basis the reader is invited to read previous works on the subject [4, 10, 18]. Computation in morphological neural networks is performed within a mathematical framework that is based on lattice algebra systems [1, 13]. More specifically, the extended real minimax algebra [2, 5], denoted by $(\mathbb{R}_{\pm\infty}, \wedge, \vee, +, +')$, is used for lattice associative memories where $\mathbb{R}_{\pm\infty} = \mathbb{R} \cup \{-\infty, +\infty\}$, the binary operations $\wedge, \vee$ denote, respectively, the min and max arithmetical operations between two numbers, and $+, +'$ correspond respectively, to addition and dual addition of arbitrary elements in $\mathbb{R}_{\pm\infty}$. For finite values $x, y \in \mathbb{R}$, we have that $x + y = x +' y$. Matrix computations in minimax algebra are defined elementwise in a similar way as matrix computations are defined in linear algebra. Two fundamental minimax matrix operations are *matrix conjugation* and *matrix max-multiplication*; given a matrix $\mathbf{A}$ of size $m \times p$ and a matrix $\mathbf{B}$ of size $p \times n$ over $\mathbb{R}_{\pm\infty}$, for each

$i = 1, \ldots, m$ and $j = 1, \ldots, n$, they are defined as (t denotes transposition)

$$a^*_{ij} = -a_{ji} \quad \text{or} \quad \mathbf{A}^* = -\mathbf{A}^t \tag{1}$$

$$c_{ij} = \bigvee_{k=1}^{p} (a_{ik} + b_{kj}) \quad \text{or} \quad \mathbf{C} = \mathbf{A} \boxtriangledown \mathbf{B}. \tag{2}$$

Dually, the *min-product* of matrices $\mathbf{A}$ and $\mathbf{B}$, denoted by $\mathbf{A} \boxtriangle \mathbf{B}$, is defined in terms of the generalized $\wedge$-min operation. There are two canonical lattice matrix associative memories known as the *min-memory*, denoted by $\mathbf{W}_{\mathrm{XY}}$, and the *max-memory*, denoted by $\mathbf{M}_{\mathrm{XY}}$. Following a minimax correlated weight rule, each memory stores a set of k associations $(\boldsymbol{x}^\xi, \boldsymbol{y}^\xi)$, represented by (X, Y), where $X = (\boldsymbol{x}^1, \ldots, \boldsymbol{x}^k) \subset \mathbb{R}^n$ and $Y = (\boldsymbol{y}^1, \ldots, \boldsymbol{y}^k) \subset \mathbb{R}^m$. Left equations in (3) and (4) are given in entry format for $i = 1, \ldots, m$ and $j = 1, \ldots, n$; right equations are in matrix notation.

$$w_{ij} = (\mathbf{W}_{\mathrm{XY}})_{ij} = \bigwedge_{\xi=1}^{k} (y_i^\xi - x_j^\xi) \;\; ; \;\; \mathbf{W}_{\mathrm{XY}} = Y \boxtriangledown X^* \tag{3}$$

$$m_{ij} = (\mathbf{M}_{\mathrm{XY}})_{ij} = \bigvee_{\xi=1}^{k} (y_i^\xi - x_j^\xi) \;\; ; \;\; \mathbf{M}_{\mathrm{XY}} = Y \boxtriangle X^* \tag{4}$$

If $Y \neq X$ then a memory is called *hetero-associative*, otherwise it is called *auto-associative*; this last case is where our attention is focused in this paper. We repeat two fundamental results on lattice auto-associative memories proved in [4]. First, $\mathbf{W}_{\mathrm{XX}}$ and $\mathbf{M}_{\mathrm{XX}}$ give *perfect recall* for *perfect input* in the sense that, $\mathbf{W}_{\mathrm{XX}} \boxtriangledown \boldsymbol{x}^\xi = \boldsymbol{x}^\xi$ (resp. $\mathbf{M}_{\mathrm{XX}} \boxtriangle \boldsymbol{x}^\xi = \boldsymbol{x}^\xi$) for $\xi = 1, \ldots, k$. Second, since the value of k is not restricted in anyway, $\mathbf{W}_{\mathrm{XX}}$ and $\mathbf{M}_{\mathrm{XX}}$ have *infinite* storage capacity.

Missing parts, occlusions or corruption of exemplar patterns can be considered as "noise" and we speak of *random noise*, when alterations in pattern entries follow a probability density function. Recall capabilities of LAMs for *non-perfect* inputs, requires noise to be classified in three basic types. Let $I = \{1, \ldots, n\}$ then, a distorted version $\widetilde{\boldsymbol{x}}$ of pattern $\boldsymbol{x}$ has undergone an *erosive change* whenever $\widetilde{\boldsymbol{x}} \leq \mathbf{x}$ or equivalently if $\forall i \in I$, $\widetilde{x}_i \leq x_i$. A *dilative change* occurs whenever $\widetilde{\boldsymbol{x}} \geq \mathbf{x}$ or equivalently if $\forall i \in I$, $\widetilde{x}_i \geq x_i$. Let $L, G \subset I$ be two non-empty disjoint sets of indexes. If $\forall i \in L$, $\widetilde{x}_i < x_i$ and $\forall i \in G$, $\widetilde{x}_i > x_i$, then the distorted pattern $\widetilde{\boldsymbol{x}}$ is said to contain *mixed* noise (random or structured). In practical situations, the "perfect recall" requirement is relaxed, and we say that $\mathbf{W}_{\mathrm{XX}}$ (resp. $\mathbf{M}_{\mathrm{XX}}$) is an *almost perfect recall* memory for a set X of patterns if and only if there exists a small rational number $\varepsilon > 0$ close to zero, such that $\mu(\mathbf{W}_{\mathrm{XX}} \boxtriangledown \widetilde{\boldsymbol{x}}, \boldsymbol{x}) \leq \varepsilon$ (resp. $\mu(\mathbf{M}_{\mathrm{XX}} \boxtriangle \widetilde{\boldsymbol{x}}, \boldsymbol{x}) \leq \varepsilon$) for all $\boldsymbol{x} \in X$ with respect to finite sets of noisy versions $\widetilde{\boldsymbol{x}}$ of $\boldsymbol{x}$, where $\mu(\cdot)$ is an adequate measure tailored to treat with binary or real valued patterns.

The lattice matrix auto-associative memories $\mathbf{W}_{\mathrm{XX}}$ and $\mathbf{M}_{\mathrm{XX}}$ can be viewed as lattice transforms of the real vector space $\mathbb{R}^n$ into itself [5, 10, 24]. Thus, for example, the min-memory is considered as a map $W_{\mathrm{X}} : \mathbb{R}^n \to \mathbb{R}^n$ for which $W_{\mathrm{X}}(\boldsymbol{x}) = \mathbf{W}_{\mathrm{XX}} \boxtriangledown \boldsymbol{x}$ for each $\boldsymbol{x} \in \mathbb{R}^n$. Sometimes, to simplify notation we write W for the min-memory lattice transform instead of W_{X}, if reference to the set X is understood in the context of discourse. The *natural factorization* of the map W, shown below in commutative diagram format, provides in a compact way the action that W performs on input vectors.

$$\begin{array}{ccc} \mathbb{R}^n & \xrightarrow{\quad W \quad} & \mathbb{R}^n \\ {\scriptstyle \pi}\downarrow & & \uparrow{\scriptstyle \iota} \\ \mathbb{R}^n/\mathcal{R}_{\mathrm{W}} & \xrightarrow{\quad \eta \quad} & \mathrm{Im}_{\mathrm{W}}(\mathbb{R}^n) \end{array} \tag{5}$$

In diagram (5), $\pi : \mathbb{R}^n \to \mathbb{R}^n/\mathcal{R}_{\mathrm{W}}$ is the projection map (surjective) from the lattice transform domain $\mathbb{R}^n$ to the quotient set $\mathbb{R}^n/\mathcal{R}_{\mathrm{W}}$ obtained from the induced equivalence relation $\mathcal{R}_{\mathrm{W}}$ between pairs of n-dimensional vectors, i.e., vector $\boldsymbol{x}$ is related to vector $\boldsymbol{y}$ if and only if $W(\boldsymbol{x}) = W(\boldsymbol{y})$. Let $W(\boldsymbol{x}) = \boldsymbol{x}$, then $W(\boldsymbol{y}) = \boldsymbol{x}$ means that $\boldsymbol{y}$ is attracted to the fixed point $\boldsymbol{x}$; otherwise, let $W(\boldsymbol{x}) = \boldsymbol{x}' \neq \boldsymbol{x}$, then $W(\boldsymbol{y}) = \boldsymbol{x}'$ and since $\boldsymbol{x}' = W(\boldsymbol{x}')$ (one-step output recall), $\boldsymbol{y}$ is attracted to another fixed point $\boldsymbol{x}'$. The new fixed point $\boldsymbol{x}'$ is simply a minimax combination of the exemplar patterns $\boldsymbol{x}^\xi \in X$, a fact established in [10, 18]. The natural map (bijective), $\eta : \mathbb{R}^n/\mathcal{R}_{\mathrm{W}} \to \mathrm{Im}_{\mathrm{W}}(\mathbb{R}^n)$ establishes a one to one and onto correspondence between each equivalence class $\Omega_{\mathrm{W}}(\boldsymbol{x})$ and the single image value $W(\boldsymbol{x}')$ computed by W on each $\boldsymbol{x}' \in \Omega_{\mathrm{W}}(\boldsymbol{x})$. The equivalence class $\Omega_{\mathrm{W}}(\boldsymbol{x})$ corresponds to the *orbit* or *attraction basin* of a given input pattern $\boldsymbol{x}$; note that the input pattern may be an exemplar pattern $\boldsymbol{x}^\xi \in X$ and that $\boldsymbol{x}'$ may be a corrupted version $\widetilde{\boldsymbol{x}}^\xi$ of $\boldsymbol{x}^\xi$. The fact that $\mathrm{Im}_{\mathrm{W}}(\mathbb{R}^n)$ equals the set of fixed points of the W transform was proved in [10], it is denoted by $F(X)$ or alternatively by $F(W) = F(W_{\mathrm{X}})$. It turns out that $F(X)$ coincide for both auto-associative memories, i.e., $F(X) = F(W) = F(M)$, and consists of the same elements that belong to the linear minimax span of X. Under the natural map η, the inverse image of any fixed point $\boldsymbol{x}$ gives its orbit, i.e., $\eta^{-1}(\boldsymbol{x}) = \Omega_{\mathrm{W}}(\boldsymbol{x})$. Finally, $\iota : F(X) \to \mathbb{R}^n$ is just an immersion map (injective) that assigns each fixed point to itself within the original range of the W map.

To illustrate the previous ideas, we describe a simple example in detail for a two-dimensional memory that stores just one pattern, i.e., let $X = \{\boldsymbol{x}^1\}$ where $\boldsymbol{x}^1 = (x_1^1, x_2^1) \in \mathbb{R}^2$; then, the min-memory matrix $\mathbf{W}_{\mathrm{XX}}$ is given by

$$\mathbf{W}_{\mathrm{XX}} = \begin{pmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{pmatrix} = \begin{pmatrix} 0 & w_{12} \\ w_{21} & 0 \end{pmatrix}. \tag{6}$$

Without loss of generality we take $x_1^1 > x_2^1$, thus $w_{12} = x_1^1 - x_2^1 > 0$, $w_{21} = x_2^1 - x_1^1 < 0$, and $w_{21} = -w_{12}$. There is no need to verify that $\mathbf{W}_{\mathrm{XX}} \boxtriangledown \boldsymbol{x}^1 = \boldsymbol{x}^1$, since this is a specific instance of the perfect recall property that lattice auto-associative memories have on X [4]. Another useful result is that each column

vector of the matrix $\mathbf{W}_{\mathrm{XX}}$ is also a fixed point (a proof based on graph theoretical concepts appears in [18], for an alternative shorter argument see the Appendix). The recall stage for an input vector is explained next. Let $\boldsymbol{x} = (x_1, x_2)$ be an input vector different to the stored exemplar pattern $\boldsymbol{x}^1$, i.e., $\boldsymbol{x} \neq \boldsymbol{x}^1$ as shown in Fig. 1.

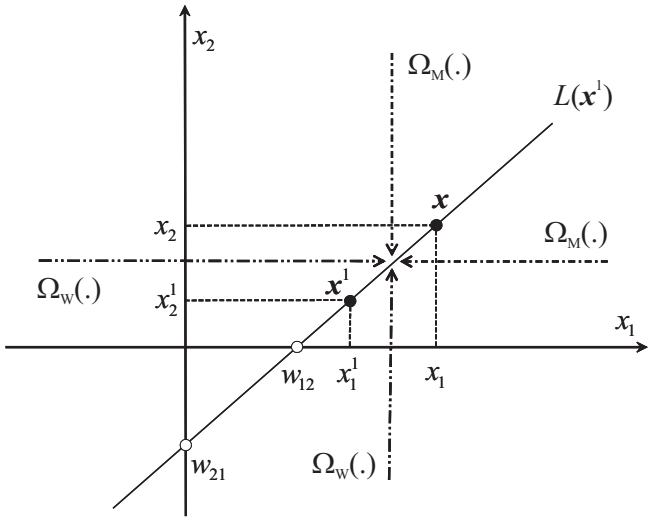


Fig. 1. The solid line $L(\boldsymbol{x}^1)$ of slope 1 equals $F(X)$, the set of fixed points of the min-W_{X} transform for $X = \{\boldsymbol{x}^1\}$. In the diagram, $\boldsymbol{x}^1, \boldsymbol{x} \in F(X)$ but $\boldsymbol{x} \neq \boldsymbol{x}^1$; the left and bottom half rays form a single orbit under W, the upper and right half rays form a single orbit under M, and the arrows show the sense of attraction towards $L(\boldsymbol{x}^1)$

The entries of the output vector $\boldsymbol{y}$ recalled with the min-memory are computed as

$$y_1 = (w_{11} + x_1) \vee (w_{12} + x_2) = x_1 \vee (x_1^1 - x_2^1 + x_2), \tag{7}$$

$$y_2 = (w_{21} + x_1) \vee (w_{22} + x_2) = x_2 \vee (x_2^1 - x_1^1 + x_1). \tag{8}$$

Consider the first coordinate y_1 of the recalled pattern (abscissa of the new point) and let $x_2 = x_2^1$ (same vertical position), then

$$y_1 = x_1^1 \Leftrightarrow x_1 < x_1^1 \tag{9}$$

$$\text{or} \quad y_1 = x_1 \Leftrightarrow x_1 > x_1^1, \tag{10}$$

similarly, for the second coordinate y_2 of the recalled pattern (ordinate of the new point), let $x_1 = x_1^1$ (same horizontal position), hence

$$y_2 = x_2^1 \Leftrightarrow x_2 < x_2^1 \tag{11}$$

$$\text{or} \quad y_2 = x_2 \Leftrightarrow x_2 > x_2^1. \tag{12}$$

From (9) and (11), the orbit of the fundamental memory $\boldsymbol{x}^1$, denoted by $\Omega_{\mathrm{W}}(\boldsymbol{x}^1)$, consists of all those points of $\mathbb{R}^2$ that lie in the *horizontal half-ray* determined by $x_1 \leq x_1^1$ (eroded 1st coordinate) and $x_2 = x_2^1$, together with the *vertical half-ray* determined by $x_1 = x_1^1$ and $x_2 \leq x_2^1$ (eroded 2nd coordinate). Note that, less than or equal signs are used since $\boldsymbol{x}^1$ is included in the orbit, therefore

$$\Omega_{\mathrm{W}}(\boldsymbol{x}^1) = \{\boldsymbol{x} \in \mathbb{R}^2 | x_1 \leq x_1^1, x_2 = x_2^1\} \cup \{\boldsymbol{x} \in \mathbb{R}^2 | x_1 = x_1^1, x_2 \leq x_2^1\}. \tag{13}$$

On the other hand, from (10) and (12), if the abscissa of the input pattern $\boldsymbol{x} \in \mathbb{R}^2$ is strictly to the right of x_1^1 with $x_2 = x_2^1$, or its ordinate is strictly above x_2^1 with $x_1 = x_1^1$, then the recalled pattern is a fixed point *different* to the stored exemplar. Hence, $\boldsymbol{x} \neq \boldsymbol{x}^1$ is such that $\mathbf{W}_{\mathrm{XX}} \boxbslash \boldsymbol{x} = \boldsymbol{x}$, or equivalently,

$$x_1 = x_1 \vee (w_{12} + x_2) \quad \text{and} \quad x_2 = x_2 \vee (w_{21} + x_1), \tag{14}$$

which are satisfied, respectively, if $x_2 = w_{21} + x_1$ and $x_1 = w_{12} + x_2$. However, each one of the last two expressions implies the other by the fact, stated earlier, that $w_{21} = -w_{12}$. Therefore, the coordinates of *any* new fixed point $\boldsymbol{x} \neq \boldsymbol{x}^1$ are related by a single linear equation given by

$$x_2 = x_1 + w_{21}, \tag{15}$$

which is a line of positive slope equal to 1 and intercept at the origin equal to w_{21}. The line passing through the points $(0, w_{21})$, $(x_1^1, x_2^1) = \boldsymbol{x}^1$, and $(x_1, x_2) = \boldsymbol{x}$, is denoted by $L(\boldsymbol{x}^1)$, to remind its relation to the exemplar pattern $\boldsymbol{x}^1 \in X$. Hence, the min-memory matrix transformation W_{X} has an infinite number of fixed points $\boldsymbol{x} \neq \boldsymbol{x}^1$, such that $\boldsymbol{x} \in L(\boldsymbol{x}^1)$, and the whole plane $\mathbb{R}^2$ is partitioned through their orbits, i.e.,

$$\mathbb{R}^2 = \bigcup \{\Omega_{\mathrm{W}}(\boldsymbol{x}) \mid \boldsymbol{x} \in L(\boldsymbol{x}^1)\}. \tag{16}$$

A similar analysis can be carried out for the max-memory transform M_{X} with $X = \{\boldsymbol{x}^1\}$. Although, the fixed points of M lie on the same line $L(\boldsymbol{x}^1)$, the orbit of the fundamental memory $\boldsymbol{x}^1$, $\Omega_{\mathrm{M}}(\boldsymbol{x}^1) \neq \Omega_{\mathrm{W}}(\boldsymbol{x}^1)$, since it is composed of points that lie in the horizontal half-ray determined by $x_1 \geq x_1^1$ (dilated 1st coordinate) and $x_2 = x_2^1$, together with the vertical half-ray determined by $x_1 = x_1^1$ and $x_2 \geq x_2^1$ (dilated 2nd coordinate), hence

$$\Omega_{\mathrm{M}}(\boldsymbol{x}^1) = \{\boldsymbol{x} \in \mathbb{R}^2 | x_1 \geq x_1^1, x_2 = x_2^1\} \cup \{\boldsymbol{x} \in \mathbb{R}^2 | x_1 = x_1^1, x_2 \geq x_2^1\}. \tag{17}$$

Also, $\Omega_{\mathrm{W}}(\boldsymbol{x}) \cap \Omega_{\mathrm{M}}(\boldsymbol{x}) = \{\boldsymbol{x}\}$ for any $\boldsymbol{x} \in L$, and the dual partition of the plane $\mathbb{R}^2$ is given by $\bigcup \{\Omega_{\mathrm{M}}(\boldsymbol{x}) \mid \boldsymbol{x} \in L(\boldsymbol{x}^1)\}$. We remark that, the line given in (15) can be obtained directly from the $\mathbf{W}_{\mathrm{XX}}$ matrix. Here, $\mathbf{W}_{\mathrm{XX}}^1 = (0, w_{21})^t$ is the intercept point on the x_2 axis and $\mathbf{W}_{\mathrm{XX}}^2 = (w_{12}, 0)^t$ is the intercept point on the x_1 axis. These two points, depicted as hollow dots on Fig. 1, define L. For this example, given a distorted version of $\boldsymbol{x}^1$, denoted by $\widetilde{\boldsymbol{x}}^1$, the min-memory $\mathbf{W}_{\mathrm{XX}}$ (resp. max-memory $\mathbf{M}_{\mathrm{XX}}$) is capable of perfect recall, i.e., $\mathbf{W}_{\mathrm{XX}} \boxbslash \widetilde{\boldsymbol{x}}^1 = \boldsymbol{x}^1$ (resp. $\mathbf{M}_{\mathrm{XX}} \boxbslash \widetilde{\boldsymbol{x}}^1 = \boldsymbol{x}^1$), if and only if $\widetilde{\boldsymbol{x}}^1 \in \Omega_{\mathrm{W}}(\boldsymbol{x}^1)$ (resp. $\widetilde{\boldsymbol{x}}^1 \in \Omega_{\mathrm{M}}(\boldsymbol{x}^1)$), meaning that $\widetilde{\boldsymbol{x}}^1$ must be eroded (resp. dilated) in one of its two coordinates but *not* both.

3 Redundant Encoding

Besides the algebraic characterization given for $F(X)$ in the previous section, a geometrical characterization of the set of fixed points has been recently developed in [10, 12]. The shape of $F(X)$ corresponds to an *m-dimensional convex prismatic beam* with $1 \leq m \leq n$, and the value of m depends on the nature of the underlying finite set X of exemplar pattern vectors. Schematic visualization of the set $F(X)$ is possible only for $n = 1, 2, 3$; for higher dimensions, used is made of concepts from n-dimensional geometry of which the necessary material together with a detailed treatment of the shape of $F(X)$ appears in [10, 12]. Simple numerical examples in lower dimensions (small sized memories) make evident the inherent limitations of any lattice matrix auto-associative memory, in the sense that, neither $\mathbf{W}_{\mathrm{XX}}$ nor $\mathbf{M}_{\mathrm{XX}}$ is able to recall correctly any exemplar pattern that has been corrupted with random noise. Also, the surprising robustness of the min-memory $\mathbf{W}_{\mathrm{XX}}$ with respect to erosive noise as well as the max-memory $\mathbf{M}_{\mathrm{XX}}$ against dilative noise is related to typical applications whose patterns are points in higher dimensional spaces, i.e., for $\xi = 1, \ldots, k$, $\boldsymbol{x}^{\xi} \in \mathbb{R}^n$ with $n \gg 1$. Complementary theoretical developments coupled with computational techniques such as the kernel methods [3, 4, 6, 14, 16, 21], fuzzy MAM models [19, 20, 25], enhanced memory schemes [22, 26], and dendrite based models [7–11], offer alternative solutions to robust pattern recall from noisy inputs. Recently, the technique of noise masking has been introduced to take advantage of the strong properties that single lattice matrix auto-associative memories have [23]. All the previous techniques share in common an increase in computational complexity or in the definition of the network topology; however, none of them exploits the infinite storage capacity property that $\mathbf{W}_{\mathrm{XX}}$ and $\mathbf{M}_{\mathrm{XX}}$ have. Therefore, we propose to use the aforementioned property together with the geometrical characterization of the fixed point set of LAMs, to introduce a simple procedure based on redundant encoding, to enable the recall of pattern approximations to exemplars from distorted inputs.

Given a set $X = \{\boldsymbol{x}^1, \boldsymbol{x}^2, \ldots, \boldsymbol{x}^k\}$ of k exemplar patterns $\boldsymbol{x}^{\xi}$, with finite real entries, *redundant encoding* consists in the storage of additional, carefully design patterns to set X. The new patterns must be spatially related to the original ones and for each $\boldsymbol{x}^{\xi} \in X$, with $\xi = 1, \ldots, k$, the same number p of related patterns, denoted by $\{\boldsymbol{x}^{\xi_1}, \boldsymbol{x}^{\xi_2}, \ldots, \boldsymbol{x}^{\xi_p}\}$, can be stored in the new min- and max-memories defined on the augmented set X^{G} given by

$$X^{\mathrm{G}} = X \cup \bigcup_{\xi=1}^{k} \bigcup_{q=1}^{p} \{\boldsymbol{x}^{\xi q}\} \tag{18}$$

If $|X| = k$ then $|X^{\mathrm{G}}| = k(p+1)$ which poses no problem, since W and M have infinite storage capacity. The purpose to add redundant patterns to the original exemplar pattern set X is to increase the number of fixed points by changing the geometrical shape of $F(X)$, and to expand it, in the sense that $F(X^{\mathrm{G}}) \supset F(X)$. It is *essential* for this scheme to work that the new encoded patterns include erosive and dilative, or mixed approximations of each exemplar pattern. In other

words, lattice matrix auto-associative memories should remember not only the exemplars but also some eroded and dilated, or mixed resemblances of them. This simple mechanism endows them with recall capability in the presence of arbitrary noisy inputs. In what follows, we consider the *minimum redundant* case, in which only a single approximate pattern is considered for each exemplar. Thus, $p = 1$ and only $2k$ patterns, $\{\boldsymbol{x}^1, \ldots, \boldsymbol{x}^k, \boldsymbol{x}^{11}, \ldots, \boldsymbol{x}^{k1}\}$, are stored in the augmented memories, $\mathbf{W}_{X^GX^G}$ and $\mathbf{M}_{X^GX^G}$.

To illustrate the basic idea behind the proposed technique, we elaborate further the example provided in Section 2. Thus, if set X has a single pattern $\boldsymbol{x}$ such that $x_1 = x_2$, then line $L(\boldsymbol{x})$ passes through the origin, and contains all the fixed points of the min- and max-transforms W_X and M_X, i.e., $F(X) = L(\boldsymbol{x})$ as shown by the dotted line at 45° with respect to the x_1 axis in Fig. 2. The points to the left, right, below, and above, respectively, denoted by, $\boldsymbol{x}^\ell, \boldsymbol{x}^r, \boldsymbol{x}^b$, and $\boldsymbol{x}^a$, represent close approximations to the exemplar pattern $\boldsymbol{x}$. For $\epsilon > 0$, let the coordinates of these points be defined by $\boldsymbol{x}^\ell = (x_1^\ell, x_2^\ell) = (x_1 - \epsilon, x_2)$, $\boldsymbol{x}^r = (x_1^r, x_2^r) = (x_1 + \epsilon, x_2)$, $\boldsymbol{x}^b = (x_1^b, x_2^b) = (x_1, x_2 - \epsilon)$, and $\boldsymbol{x}^a = (x_1^a, x_2^a) = (x_1, x_2 + \epsilon)$. Also, notice that, $\boldsymbol{x}^\ell$ and $\boldsymbol{x}^b$ are eroded, respectively, in its 1st and 2nd coordinates; similarly, $\boldsymbol{x}^r$ and $\boldsymbol{x}^a$ are dilated, respectively, in x_1 and x_2. Although we can adjoin all four points to X to build the augmented set X^G, the addition of *only* two points not on the same line, such as $\{\boldsymbol{x}^\ell, \boldsymbol{x}^r\}$, $\{\boldsymbol{x}^a, \boldsymbol{x}^b\}$, $\{\boldsymbol{x}^\ell, \boldsymbol{x}^b\}$, or $\{\boldsymbol{x}^a, \boldsymbol{x}^r\}$, give the same set of fixed points for X^G. Let $X^G = \{\boldsymbol{x}, \boldsymbol{x}^a, \boldsymbol{x}^r\}$, then the lattice auto-associative memories of X^G, computed from (3) and (4), are given by

$$\mathbf{W}_{X^GX^G} = \begin{pmatrix} 0 & -\epsilon \\ -\epsilon & 0 \end{pmatrix} \quad \text{and} \quad \mathbf{M}_{X^GX^G} = \begin{pmatrix} 0 & \epsilon \\ \epsilon & 0 \end{pmatrix}. \tag{19}$$

Alternatively, the max-memory matrix can be computed using the fact that $\mathbf{M} = \mathbf{W}^*$ together with (1). Next, to determine $F(X^G)$, we solve the equation $\mathbf{W}_{X^GX^G} \boxtimes \boldsymbol{x}' = \boldsymbol{x}'$ for $\boldsymbol{x}'$ not in X^G, equivalently, $x_i' = \bigvee_{j=1}^{2} (w_{ij} + x_j')$ for $i = 1, 2$. The resulting expressions are given by $x_1' = x_1' \vee (x_2' - \epsilon)$ and $x_2' = x_2' \vee (x_1' - \epsilon)$, from which $x_2' - \epsilon \leq x_1'$ and $x_1' - \epsilon \leq x_2'$, consequently, $x_1' - \epsilon \leq x_2' \leq x_1' + \epsilon$. The boundary values for x_2' in this last compound inequality provide us with the line equations of slope 1, $x_2' = x_1' + \epsilon$ and $x_2' = x_1' - \epsilon$, labeled in Fig. 2 as $L_a = L(\boldsymbol{x}^a)$ and $L_r = L(\boldsymbol{x}^r)$, respectively. Since $x_2' \in [x_1' - \epsilon, x_1' + \epsilon]$, the fixed point set of X^G consists of all points that belong to L_a and L_r, or lie in the infinite band between them. Mathematically, if $\bar{H}_a^-$ denotes the closed half plane on and *below* the line L_a and $\bar{H}_r^+$ denotes the closed half plane on and *above* the line L_r, then, $F(X^G) = \bar{H}_a^- \cap \bar{H}_r^+$. Notice that, by definition, $\bar{H}_a^+ \cap \bar{H}_a^- = L_a$ and $\bar{H}_r^+ \cap \bar{H}_r^- = L_r$, hence the boundary of the fixed point set of X^G is given by $\partial F(X^G) = L_a \cup L_r$. The geometric shape of $F(X)$ for n-dimensional patterns is fully developed in [12]. In this example, by encoding two redundant patterns that are spatially related to the single exemplar in X, an expanded fixed point set is obtained such that $F(X) \subset F(X^G)$. For pattern recall in the presence of noise, we may consider all fixed points $\boldsymbol{x}' \in F(X^G)$ that satisfy the inequality, $|x_1' - x_1| + |x_2' - x_2| \leq \epsilon$ (hatched rhombus shown in

Fig. 2), to be "good" approximations to a given exemplar pattern $\boldsymbol{x}$, in the sense described earlier for almost perfect recall using a single lattice auto-associative memory. The parameter ϵ gives us the possibility to increase or decrease the size of the rhombus shaped neighborhood and therefore to relax or constrain the admissible memory outputs.

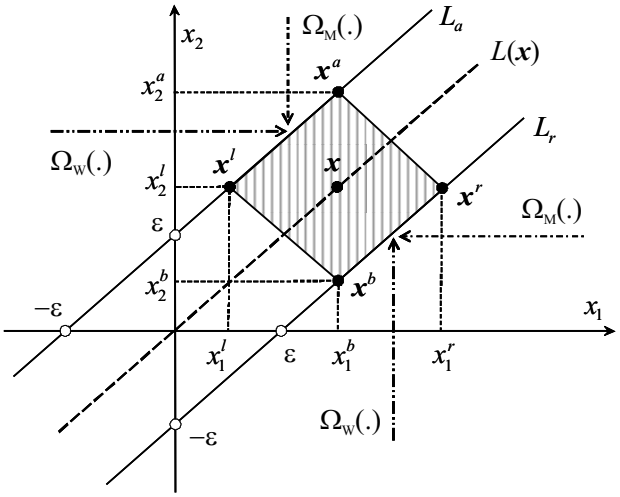


Fig. 2. The solid line $L(\boldsymbol{x})$ of slope 1 equals $F(X)$, the set of fixed points of the min-W_{X} transform for $X = \{\boldsymbol{x}\}$. The points, $\boldsymbol{x}^{\ell}, \boldsymbol{x}^{r}, \boldsymbol{x}^{a}, \boldsymbol{x}^{b} \in F(X^{\mathrm{G}})$ are eroded or dilated versions of $\boldsymbol{x}$ in a single coordinate. The left and bottom half rays are different orbits under $W_{\mathrm{X^G}}$, the upper and right half rays are distinct orbits under $M_{\mathrm{X^G}}$, and the arrows show the sense of attraction, respectively, towards L_a or L_r

It is reasonable to expect that for high dimensional patterns, redundant encoding can be achieved in various ways. The extension of the base examples, described previously, to higher dimensions ($n > 3$) is impossible to visualize. However, the basic procedure is still the same, since a subset of the coordinates of a given exemplar pattern in X can be eroded to a minimum prescribed value and another subset of coordinates, disjoint from the first, may be dilated to a maximum prescribed value. The rest of the coordinate values would remain the same as those in the original exemplar. In this manner, a *single* redundant pattern related spatially to each exemplar in X is encoded to build X^{G}. Adjoined patterns are then approximations to exemplars which also can be regarded as "noisy" versions of them, in the sense that, mixed *structured* noise is induced by eroding and dilating several coordinates. Clearly, the adequate selection of the coordinate subsets to be modified is determined by pattern dimensionality as well as their nature in a given application.

4 Application Example

To test our proposal, we conducted a computer experiment using 10 gray scale images of size 53×53 pixels as integer valued patterns of dimension 2809. Fig. 3 displays only four of these exemplar pattern images together with a single redundant encoded pattern obtained by assigning a minimum value (0 for extreme erosion) or a maximum value (255 for extreme dilation) to the gray values of a subset of the pixel coordinates, the same for each exemplar in X. Hence, using (3) and (4), the memory matrices $\mathbf{W}_{\mathrm{X^GX^G}}$ and $\mathbf{M}_{\mathrm{X^GX^G}}$, of size 2809×2809, store all associations between the 20 patterns in the augmented set X^{G}.

Fig. 3. 1st row: exemplar patterns $\boldsymbol{x}^1, \boldsymbol{x}^3, \boldsymbol{x}^6, \boldsymbol{x}^8 \in X$; 2nd row: redundant encoded patterns $\boldsymbol{x}^{11}, \boldsymbol{x}^{31}, \boldsymbol{x}^{61}, \boldsymbol{x}^{81}$ adjoined to X to form X^{G}; 3rd row: noisy versions of exemplars, respectively, with noise probabilities of $0.3, 0.4, 0.5$, and 0.6; 4th row: recalled pattern approximations.

Each pattern in X was contaminated 100 times by adding random noise with probabilities 0.1 to 0.9 and 128 as offset value. The measure $\mu(\cdot)$ used for almost perfect recall is the *normalized mean square error* (NMSE) defined by $\sigma(\boldsymbol{x}, \widetilde{\boldsymbol{x}}) = \sum_{i=1}^{n}(x_i - \widetilde{x}_i)^2 / \sum_{i=1}^{n} x_i^2$. Fig. 4 shows the performance curves obtained in the recall stage using as final output, the arithmetical mean of the min and max memory outputs, since $\mathbf{W}_{\mathrm{X^GX^G}}$ attracts points towards $F(X^{\mathrm{G}})$ by increasing coordinate values and $\mathbf{M}_{\mathrm{X^GX^G}}$ attracts points towards $F(X^{\mathrm{G}})$ by decreasing coordinate values. Also, if more and more pattern entries are affected by noise, these modified entries are less likely to be equal in the remaining patterns, thus it is no surprise that NMSE decreases as the probability of error increases. In spite of this abnormal behavior, adjoining redundant encoded patterns to the

original pattern set is a useful alternative that works best for highly corrupted inputs.

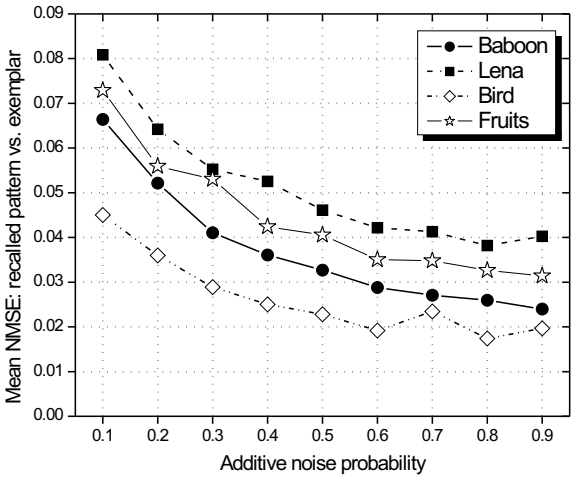


Fig. 4. Recall stage performance curves obtained over 100 trials of producing noise versions of four selected exemplars. NMSE is computed between recalled outputs and corresponding exemplars. In selected patterns, relative error range remains within the same order of magnitude (10^{-2}), for all noise levels

5 Conclusions

In this paper, we introduce a novel and interesting technique that encodes redundant patterns in order to endow lattice auto-associative memories with recall capability in the presence of noise. Basically, the set of fixed points generated by the min or max transforms on the original exemplar set is expanded to allow for the existence of local neighborhoods that might be considered as clusters of approximate versions to exemplars. A complete and simple description using two dimensional patterns has been provided to illustrate the encoding mechanism. Finally, a scenario for gray scale image storage and recall demonstrates the scope and usefulness of minimal redundant pattern encoding in LAMs. Future work will consider different geometrical configurations for pattern encoding, further study on fixed point neighborhood shapes, and more important, carefully designed tests – with or without additional encoding – to compare our proposed model against other associative memory models.

Appendix

Theorem 1. *Let $X = (\boldsymbol{x}^1, \ldots, \boldsymbol{x}^k) \subset \mathbb{R}^n$ be a set of k exemplar patterns and let $(\boldsymbol{w}^1, \ldots, \boldsymbol{w}^\lambda, \ldots, \boldsymbol{w}^n)$ denote the column vectors of the corresponding min-memory $\mathbf{W}_{XX}$ of size $n \times n$, then $\mathbf{W}_{XX} \boxtimes \boldsymbol{w}^\lambda = \boldsymbol{w}^\lambda$ for $\lambda = 1, \ldots, n$. Therefore, any column of the matrix W_{XX} is a fixed point for the min-memory transform.*

Proof. By definition, the entries of the column vector $\boldsymbol{w}^\lambda$ of $\mathbf{W}_{XX}$ for $j = 1, \ldots, n$ are given by $w_j^\lambda = \bigwedge_{\xi=1}^{k} (x_j^\xi - x_\lambda^\xi)$, and the entries of the recalled output pattern $\boldsymbol{y}$ are computed as follows, $y_i = (\mathbf{W}_{XX} \boxtimes \boldsymbol{w}^\lambda)_i = \bigvee_{j=1}^{n} (w_{ij} + w_j^\lambda) = w_i^\lambda \vee \bigvee_{j \neq i} (w_{ij} + w_j^\lambda)$, for $i = 1, \ldots, n$. Since, for all i, $w_{ij} + w_{j\lambda} \leq w_{i\lambda} = w_i^\lambda$, by Lemma 5.1 in [10], then $\bigvee_{j \neq i} (w_{ij} + w_j^\lambda) \leq w_i^\lambda \Rightarrow y_i = w_i^\lambda$.

A similar result holds for the max-memory $\mathbf{M}_{XX}$, i.e., any column of the $\mathbf{M}_{XX}$ matrix is a fixed point for the max-memory transform. Since $\mathbf{W}_{XX} \boxtimes \mathbf{W}_{XX} = \mathbf{W}_{XX}$ and $\mathbf{M}_{XX} \boxtimes \mathbf{M}_{XX} = \mathbf{M}_{XX}$, the previous result is equivalent to the fact that $\mathbf{W}_{XX}$ and $\mathbf{M}_{XX}$ are *idempotent*, respectively, under the max and min products.

Acknowledgment

Gonzalo Urcid thanks the National Research System (SNI) and the National Council of Science and Technology (CONACYT) in Mexico for the partial support provided for this work under grant 22036. José A. Nieves-Vázquez also thanks CONACYT for doctoral scholarship 181663.

References

1. Birkhoff, G.: Lattice Theory 3rd ed, Am. Math. Soc., Providence RI (1967).
2. Cuninghame-Green, R.: Minimax algebra and applications. In: Hawkes, P. (ed) Advances in Imaging and Electron Physics, **90**, Academic Press, New York (1995) 1–121.
3. Hattori, M., Fukui, A., Ito, H.: A fast method of constructing kernel patterns for morphological associative memories. Proc. 9th Int. Conf. on Neural Information Processing, Singapore China (2002) 1058–1063.
4. Ritter, G.X., Sussner, S., Diaz de Leon, J.L.: Morphological associative memories. IEEE Trans. on Neural Networks, **9**(2) (1998) 281–293.
5. Ritter, G.X.: Lattice algebra & Minimax algebra. In: Image Algebra, unpublished manuscript available via anonymous ftp, CISE Department, University of Florida, Gainesville FL (1999) 121–135.
6. Ritter, G.X., Urcid, G., Iancu, L.: Reconstruction of noisy patterns using morphological associative memories. J. of Math. Imaging and Vision, **19**(5) (2003) 95–111.
7. Ritter, G.X., Iancu, L.: A morphological auto-associative memory based on dendritic computing. Proc. IEEE (2004) 915–920.
8. Ritter, G.X., Iancu, L.: Lattice algebra approach to neural networks and pattern classification. Pattern Recognition and Image Analysis, **14**(2) (2004) 190–197.

9. Ritter, G.X., Urcid, G., Schmalz, M.S.: Lattice associative memories that are robust in the presence of noise. SPIE Proc., Mathematical Methods in Pattern and Image Analysis, **5916**, San Diego CA (2005) 59160Q1–Q6.
10. Ritter, G.X., Gader, P.: Fixed Points of Lattice Transforms and Lattice Associative Memories. In: Hawkes, P.(ed) Adv. in Imaging and Electron Physics, **144**. Elsevier, San Diego CA (2006) 165–242.
11. Ritter, G.X., Urcid, G.: Learning in Lattice Neural Networks that Employ Dendritic Computing. In: Kaburlasos, V.G. & Ritter, G.X.(eds) Computational Intelligence based on Lattice Theory, **67**. Springer, Heidelberg, Germany (2007) 25–44.
12. Ritter, G.X., Urcid, G.: Fast autonomous endmember determination using lattice algebra, *in preparation* for IEEE Trans. on PAMI.
13. Skornjakov, L.A.: Elements of Lattice Theory, Adam Hilger, Bristol UK (1977).
14. Sussner, P.: Observations on morphological associative memories and the kernel method. Neurocomputing, **31** (2000) 167–183.
15. Sussner, P.: Fixed points of autoassociative morphological memories. Proc. of the Int. Joint Conf. on Neural Networks, Como, Italy (2000) 611–616.
16. Sussner, P.: Associative morphological memories based on variations of the kernel and dual kernel methods. Neural Networks, **16**(5) (2003) 625–632.
17. Sussner, P.: Recall of patterns using binary and gray-scale autoassociative morphological memories. SPIE Proc., Mathematical Methods in Pattern and Image Analysis, **5916**, San Diego CA (2005) 59160M:1-10.
18. Sussner, P., Valle, M.E.: Gray-scale morphological associative memories. IEEE Trans. on Neural Networks, **17**(3) (2006) 559–570.
19. Sussner, P., Valle, M.E.: Implicative fuzzy associative memories. IEEE Trans. on Fuzzy Systems, **14**(6) (2006) 793–807.
20. Sussner, P., Valle, M.E.: Fuzzy Associative Memories and Their Relationship to Mathematical Morphology. In: Pedrycz, W., Skowron, A. & Kreinovich, V. (eds) Handbook of Granular Computing, Ch. 33, Wiley, (2008).
21. Urcid, G., Ritter, G.X.: Kernel computation in morphological associative memories for gray-scale image recollection. Proc. 5th IASTED Int. Conf. on Signal and Image Processing, Honolulu HI (2003) 450–455.
22. Urcid, G., Nieves-V., J.A. Reyes-G, C.A.: Enhanced fuzzy autoassociative morphological memory for binary pattern recall. Proc. 4th IASTED Int. Conf. on Signal Processing, Pattern Recognition, and Applications, Innsbruck, Austria (2007) 116–121.
23. Urcid, G., Ritter, G.X.: Noise Masking for Pattern Recall using a Single Lattice Matrix Associative Memory. In: Kaburlasos, V.G. & Ritter, G.X.(eds) Computational Intelligence based on Lattice Theory, **67**. Springer, Heidelberg, Germany (2007) 79–98.
24. Urcid, G., Valdiviezo-N., J.C., Generation of lattice independent vector sets for pattern recognition applications. SPIE Proc., Mathematics of Data/Image Pattern Recognition, Compression, Coding, and Encryption X with Applications,**6700**, San Diego CA (2007) 67000C :1–12.
25. Valle, M.E., Sussner, P.: A general framework for fuzzy morphological associative memories. Fuzzy Sets and Systems, **159**(7) (2008) 747–768.
26. Wang, M., Chen, S.: Enhanced fuzzy morphological auto-associative memory based on empirical kernel map. IEEE Trans. on Neural Networks, **16**(3) (2005) 557–564.

A Novel Fuzzy Implication Stemming from a Fuzzy Lattice Inclusion Measure

Anestis G. Hatzimichailidis and Vassilis G. Kaburlasos

Technological Educational Institution of Kavala
Department of Industrial Informatics
65404 Kavala, Greece
hatz_ane@hol.gr;vgkabs@teikav.edu.gr

Abstract. We introduce a fuzzy implication stemming from a fuzzy lattice inclusion measure. We study "reasonable axioms" and properties of the aforementioned fuzzy implication, which (properties) are typicaly required in the literature and could be important in certain applications.

Key words: Fuzzy lattice theory, Fuzzy implications, Inclusion measure

1 Introduction

A number of basic properties of the classical (logic) implication have been generalized in fuzzy implications; hence, a number of "reasonable axioms" have been proposed tentatively for fuzzy implications [10], [12].

Lately, an *inclusion measure* function was introduced in a mathematical lattice $(\mathsf{L}, \leq)$ for fuzzying the corresponding (crisp) partial order relation [6], [8]. In this paper we study a fuzzy implication stemming from the aforementioned inclusion measure. We show that the proposed fuzzy implication satisfies most of the "reasonable axioms" proposed in the literature. We study additional properties of our proposed fuzzy implication. The latter properties are typicaly required in the literature and could be important in certain applications.

The layout is as follows. Section 2 presents mathematical preliminaries. Section 3 introduces a novel fuzzy implication. Section 4 concludes by summarizing the contribution including also a description of potential future work.

2 Mathematical Preliminaries

This section presents basic notions of (Fuzzy) Lattice Theory and Fuzzy Sets Theory [2], [3], [6], [8], [10], [11].

2.1 Inclusion Measure

Consider the following definitions.

Definition 1. *Given a set P, a binary relation ($\leq$) on P is called* **partial order** *if and only if it satisfies the following conditions for all x, y, z ∈ P:*

pp. 59–66, ISBN 978–80–244–2112–4, Palacký University, Olomouc, 2008.

PO1. Reflexivity: $x \leq x$.
PO2. Antisymmetry: $x \leq y$ *and* $y \leq x \Rightarrow x{=}y$.
PO3. Transitivity: $x \leq y$ *and* $y \leq z \Rightarrow x \leq z$.

Condition $PO2$ can be replaced by the following equivalent condition:

$PO2'$. *Antisymmetry:* $x \leq y$ and $x \neq y \Rightarrow y \nleq x$.

Definition 2. *A* **partially ordered set**, *or* **poset** *for short, is a pair* $(P, \leq)$, *where* P *is a set and* $\leq$ *is a partial order relation on* P.

Definition 3. *A* **(crisp) lattice** *is a poset* $(\mathsf{L}, \leq)$ *any two of whose elements have both a greatest lower bound and a least upper bound. A lattice* $(\mathsf{L}, \leq)$ *is called* **complete** *when each of its subsets* X *has both a greatest lower bound and a least upper bound in* L.

Definition 4. *Let* $(\mathsf{L}, \leq)$ *be a complete lattice with least and greatest elements* O *and* I, *respectively. An* **inclusion measure** *in* $(\mathsf{L}, \leq)$ *is a map* $\sigma : \mathsf{L} \times \mathsf{L} \rightarrow [0, 1]$, *which satisfies the following conditions for* u, w, $x \in \mathsf{L}$*:*

IM0. $\sigma(x, O) = 0, \forall x \neq O$
IM1. $\sigma(x, x) = 1, \forall x \in \mathsf{L}$
IM2. $u \leq w \Rightarrow \sigma(x, u) \leq \sigma(x, w)$ *(Consistency Property)*
IM3. $x \wedge y < x \Rightarrow \sigma(x, y) < 1$

For a non-complete lattice condition $IM0$ is dropped.

Based on equivalence relation $x \wedge y < x \iff y < x \vee y$ ([1]) it follows that condition $IM3$ can, equivalently, be replaced by

$IM3'$. $y < x \vee y \Rightarrow \sigma(x, y) < 1$.

Conditions $IM1$ and $IM2$ imply $u \leq w \Rightarrow \sigma(u, u) \leq \sigma(u, w) \Rightarrow \sigma(u, w) = 1$, $u, w \in \mathsf{L}$. Hence, $\sigma(x, I) = 1$, $\forall x$ in a complete lattice $(\mathsf{L}, \leq)$.

2.2 Fuzzy Implications

Let X be a universe of discourse. A fuzzy set A in X ([14]) is defined as a set of ordered pairs $A = \{(x, \mu_A(x)) : x \in X\}$, where function $\mu_A : X \rightarrow [0, 1]$ indicates the degree of membership of an element $x \in X$ in fuzzy set A.

Definition 5. *A binary operation* $i : [0, 1] \times [0, 1] \rightarrow [0, 1]$, *is called* **fuzzy intersection** *if it is an extension of the classical Boolean intersection as follows*

BI1. $i(a, b) \in [0, 1]$, $\forall a, b \in [0, 1]$, *and*
BI2. $i(0, 0) = i(0, 1) = i(1, 0) = 0$*;* $i(1, 1) = 1$

A canonical model of fuzzy intersections is the family of *triangular norms*, or *t-norms* for short, defined rigorously next.

Definition 6. *A* **t-norm** *is a function* $i : [0,1] \times [0,1] \to [0,1]$*, which is commutative, associative, non-decreasing, and* $i(\alpha, 1) = \alpha$*,* $\forall \alpha \in [0,1]$*.*

A t-norm i is called *Archimedean* if it is both continuous and $a \in (0,1)$ implies $i(a,a) < a$; furthermore, a t-norm i is called *nilpotent* if it is both continuous and $\forall \alpha \in (0,1)$ there is a $\nu \in N$ such that $i(a, \ldots, a) = 0$. Archimedean norms are either *nilpotent* or *non-nilpotent*. The latter (norms) are also called *strict*.

Definition 7. *A function* $n : [0,1] \to [0,1]$ *is called* **negation** *if it is both non-increasing, i.e.* $n(a) \leq n(b)$ *for* $a \geq b$*, and* $n(0) = 1$*,* $n(1) = 0$*.*

A negation n is called *strict* if and only if n is both continuous and strictly decreasing, i.e. $n(a) < n(b)$ for $a > b$, $\forall a, b \in [0,1]$. A strict negation n is called *strong* if and only if it is *self-inverse*, i.e. $n(n(a)) = a$, $\forall a \in [0,1]$. The most popular strong negation is the *standard negation*: $n_S = 1 - a$.

A *triangular conorm*, or *t-conorm*, is a function $u : [0,1] \times [0,1] \to [0,1]$, which satisfies the following properties:

i) $u(a, 0) = a$, $\forall a \in [0,1]$,
ii) $u(a,b) \leq u(c,d)$ if both $a \leq c$ and $b \leq d$,
iii) $u(a,b) = u(b,a)$, $\forall a, b \in [0,1]$, and
iv) $u(u(a,b),c) = u(a,u(b,c))$, $\forall a, b, c \in [0,1]$.

A *fuzzy implication* is a function $g : [0,1] \times [0,1] \to [0,1]$, which for any truth values $a, b \in [0,1]$ of (fuzzy) propositions p, q, respectively, gives the truth value $g(a,b)$ of conditional proposition "if p then q". Function $g(.,.)$ should be an extension of the *classical implication* from domain $\{0,1\}$ to domain $[0,1]$.

The *implication operator* of classical logic is a map $m : \{0,1\} \times \{0,1\} \to \{0,1\}$ which satisfies the following conditions: $m(0,0) = m(0,1) = m(1,1) = 1$ and $m(1,0) = 0$. The latter conditions are typically the minimum requirements for a fuzzy implication operator. In other words, fuzzy implications are required to reduce to the classical implication when truth-values are restricted to 0 and 1; i.e. $g(0,0) = g(0,1) = g(1,1) = 1$ and $g(1,0) = 0$.

2.2.1 Properties of fuzzy implications

One way of defining an implication operator m in classical logic is using formula $m(a,b) = \overline{a} \vee b$, $a, b \in \{0,1\}$, where $\overline{a}$ denotes the negation of a. Another (equivalent) way of defining implication operator m in classical logic is using formula $m(a,b) = \max\{x \in \{0,1\} : a \wedge x \leq b\}$, $a, b \in \{0,1\}$.

Fuzzy logic extensions of the previous formulas, respectively, are

$$g(a,b) = u(n(a), b) \tag{1}$$

and

$$g(a,b) = \sup\{x \in [0,1] : i(a,x) \leq b\}, \tag{2}$$

$\forall a, b \in [0,1]$, where u, i and n denote a fuzzy union, a (continuous) fuzzy intersection, and a fuzzy negation, respectively. Note that functions u and i are *dual* (with respect to n) — Recall that a t-norm i and a t-conorm u are called *dual* (with respect to a fuzzy negation n) if and only if both $n(i(a,b)) = u(n(a), n(b))$ and $n(u(a,b)) = i(n(a), n(b))$ hold $\forall a, b \in [0,1]$.

Fuzzy implications obtained from (1) are usually referred to as *S-implications*, whereas fuzzy implications obtained from (2) are called *R-implications*.

Formula $m(a,b) = \overline{a} \vee b$ can also be rewritten, based on the law of *absorption of negation* in classical logic, as either $m(a,b) = \overline{a} \vee (a \wedge b)$ or $m(a,b) = (\overline{a} \wedge \overline{b}) \vee b$. Extensions of the latter equations in fuzzy logic are given, respectively, by

$$g(a,b) = u(n(a), i(a,b)) \tag{3}$$

and

$$g(a,b) = u(i(n(a), n(b)), b), \tag{4}$$

where u, i and n are required to satisfy the De Morgan laws. The fuzzy implications obtained from (3) are called *QL-implications* because they were originally introduced in quantum logic.

A number of basic properties of the classical (logic) implication have been generalized by fuzzy implications. Hence, a number of "reasonable axioms" emerged tentatively for fuzzy implications. Some of the aforementioned axioms are displayed next [10], [12].

A1. $a \leq b \Rightarrow g(a,x) \geq g(b,x)$ — *Monotonicity in first argument*
A2. $a \leq b \Rightarrow g(x,a) \leq g(x,b)$ — *Monotonicity in second argument*
A3. $g(a, g(b,x)) = g(b, g(a,x))$ — *Exchange Property*
This is a generalization of the equivalence between $a \Rightarrow (b \Rightarrow x)$ and $b \Rightarrow (a \Rightarrow x)$ in classical implication.
A4. $g(a,b) = g(n(b), n(a))$ — *Contraposition*
A5. $g(1,b) = b$ — *Neutrality of truth*
A6. $g(0,a) = 1$ — *Dominance of falsity*
A7. $g(a,a) = 1$ — *Identity*
A8. $g(a,b) = 1 \iff a \leq b$ — *Boundary Condition*
A9. g is a continuous function — *Continuity*

We remark that one can easily prove that a *S-implication* fulfills axioms A1, A2, A3, A5, A6 and, when negation n is strong, it also fulfills axiom A4. Furthermore, a *R-implication* fulfills axioms A1, A2, A5, A6, and A7.

3 A Novel Fuzzy Implication

An inclusion measure (σ) can be used to quantify partial (fuzzy) set inclusion. In this sense $\sigma(x,y)$ is similar to alternative definitions proposed in the literature for quantifying a degree of inclusion of a (fuzzy) set into another one [6]. However, the aforementioned "alternative" definitions typically involve only overlapping

(fuzzy) sets; otherwise, the corresponding inclusion index equals zero. Whereas, Definition 4 is more general and may give meaningful results in a general lattice — not only in the lattice of (fuzzy) sets. Indeed, $\sigma(x, y)$ can be interpreted as the fuzzy degree to which x is less than y; therefore notation $\sigma(x \leq y)$ is sometimes used in place of $\sigma(x, y)$.

Consider the following function in a lattice.

Definition 8. *A* **valuation** *in a lattice* $(\mathsf{L}, \leq)$ *is a real function* $v : \mathsf{L} \to \mathsf{R}$ *which satisfies* $v(x)+v(y) = v(x \wedge y)+v(x \vee y)$, $x, y \in \mathsf{L}$. *A valuation is called* **positive** *if and only if* $x < y$ *implies* $v(x) < v(y)$.

The following theorem shows an inclusion measure in a lattice based on a positive valuation function [6], [7].

Theorem 1. *If* $v : \mathsf{L} \to \mathsf{R}$ *is a positive valuation in a lattice* $(\mathsf{L}, \leq)$ *then function* $\sigma_\vee(x, y) = \frac{v(y)}{v(x \vee y)}$ *is an inclusion measure.*

In particular, for positive valuation function $v(x) = x$ inclusion measure $\sigma_\vee(x, y) = \frac{v(y)}{v(x \vee y)}$ equals $\sigma_\vee(x, y) = \frac{y}{x \vee y}$. The latter is a fuzzy implication because it reduces to the classical implication for truth values $x, y \in \{0, 1\}$; i.e. $\sigma_\vee(0, 0) = \sigma_\vee(0, 1) = \sigma_\vee(1, 1) = 1$ and $\sigma_\vee(1, 0) = 0$. Fig. 1 shows the graphical representation of fuzzy implication $\sigma_\vee$.

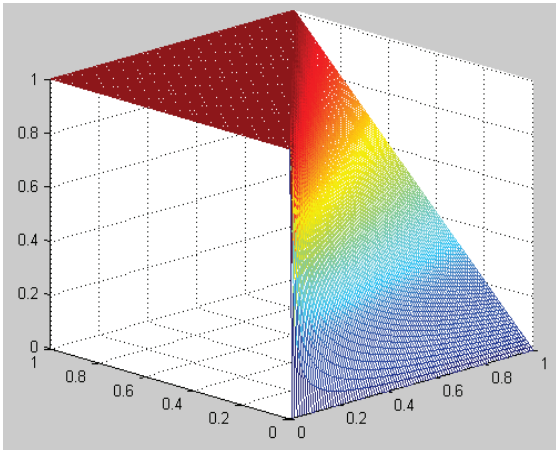


Fig. 1. Graphical representation of fuzzy implication $\sigma_\vee$

3.1 Properties of Fuzzy Implication $\sigma_\vee$

Fuzzy implication $\sigma_\vee$ satisfies the following "reasonable axioms" [10], [12] for fuzzy implications.

Proposition 1. *Consider fuzzy implication $\sigma_\vee$. Let $a, b, x \in [0,1]$. Then*

A1. $a \leq b \Rightarrow \sigma_\vee(a,x) \geq \sigma_\vee(b,x)$
A2. $a \leq b \Rightarrow \sigma_\vee(x,a) \leq \sigma_\vee(x,b)$
A3. $\sigma_\vee(a,\sigma_\vee(b,x)) = \sigma_\vee(b,\sigma_\vee(a,x))$
A4′. $\sigma_\vee(a,b) = \sigma_\vee(n(b),n(a))$ — *see in the following remark*
A5. $\sigma_\vee(1,b) = b$
A6. $\sigma_\vee(0,a) = 1$
A7. $\sigma_\vee(a,a) = 1$
A8. $\sigma_\vee(a,b) = 1 \iff a \leq b$
A9. $\sigma_\vee$ *is a continuous function*

We remark that axioms $(A1) - (A3)$ and $(A5) - (A9)$ in Proposition 1 can be proved immediately. For the standard fuzzy complement $n(a) = 1 - a$, axiom $(A4)$ holds only if $a \leq b$; whereas, for $a > b$, axiom $(A4)$ holds only if $a + b = 1$.

Additional "reasonable axioms" [13] include the following.

A10. $g(a, i(b,c)) = i(g(a,b), g(a,c))$
A11. $i(g(a,b), g(n(a),b)) = b$
A12. $i(g(0.5,b), g(0.5,b)) = b$
A13. $g(a, g(b,c)) = g(i(a,b),c)$

The next proposition shows how fuzzy implication $\sigma_\vee$ satisfies axioms *(A10)-(A13)* (with the "min" operator $(\wedge)$ employed as a fuzzy t-norm).

Proposition 2. *Consider fuzzy implication $\sigma_\vee$. Let $a, b, c \in [0,1]$. Then*

A10. $\sigma_\vee(a,(b \wedge c)) = \wedge(\sigma_\vee(a,b), \sigma_\vee(a,c))$
A11′. $\wedge(\sigma_\vee(a,b), \sigma_\vee(n(a),b)) > b$
A12′. $\wedge(\sigma_\vee(0.5,b), \sigma_\vee(0.5,b)) > b$
A13′. $\sigma_\vee(a,\sigma_\vee(b,c)) = \sigma_\vee(\wedge(a,b),c)$, *for* $c \geq a \wedge b$.

Proposition 2 can be proved immediately.

The following propositions describe some properties of fuzzy implication $\sigma_\vee$, which (properties) are often required in the literature because they could be important in certain applications [5], [10], [11].

Proposition 3. *Let us denote $\sigma_\vee(a,b)$ by $a \to b$. Then*

i) $(a \wedge b) \to c = (a \to c) \vee (b \to c)$
ii) $(a \vee b) \to c = (a \to c) \wedge (b \to c)$

Proposition 3 can be proved immediately.

Proposition 4. *Consider both fuzzy implication $\sigma_\vee$ and the standard negation $n_S = 1 - a$ (a' for short). Then*

i) $\sigma_\vee(a,1) = 1$, $\forall a \in [0,1]$
ii) $\sigma_\vee(\sigma_\vee(a,b),c) \leq \sigma_\vee(a,\sigma_\vee(b,c))$, $\forall a,b,c \in [0,1]$
iii) $\vee(a,\sigma_\vee(a,b)) = \sigma_\vee(a,\vee(b,c))$, *for* $a \leq b$

iv) $\wedge(a, \sigma_\vee(a,b)) \leq b$, *for* $a \leq b$
v) $\sigma_\vee(a, a') > a'$, $\forall a \in [0,1]$
vi) $\sigma_\vee(a', a) > a$, $\forall a \in [0,1]$
vii) $(\sigma_\vee(a,b))' \leq \sigma_\vee(a', b')$, $\forall a, b \in [0,1]$
viii) $\sigma_\vee(a,b) \geq b$, $\forall a, b \in [0,1]$
ix) $\sigma_\vee(a,b) \geq a'$, *for* $a \leq b$

Proposition 4 can be proved immediately.

Proposition 5. *Let us denote* $\sigma_\vee(a,b)$ *by* $a \to b$. *Then*

i) $(((a_1 \to a_2) \to a_3) \to \ldots) \to a_n = a_n$
when $a_1 \leq \ldots \leq a_n$ *and* n *is odd, and*
$(((a_1 \to a_2) \to a_3) \to \ldots) \to a_n = 1$
when $a_1 \leq \ldots \leq a_n$ *and* n *is even.*
ii) $(a_1 \to a_2) \to (a_2 \to a_3) \to \ldots \to (a_{n-1} \to a_n) = 1$
when $a_1 \leq \ldots \leq a_n$

Proposition 5 can be proved immediately.
Choosing the bounded sum, i.e. $a \oplus b = 1 \wedge (a+b)$, as a fuzzy union, it follows:

Proposition 6. *Let us denote* $\sigma_\vee(a,b)$ *by* $a \to b$. *Furthermore, let* $a, b \in [0,1]$. *Then*

i) $a \oplus (a \to b) = a \to (a \oplus b)$, *for* $a \leq b$
ii) $a \oplus (a \to b) \leq a \to (a \oplus b)$, *for* $a > b$

Proposition 6 can be proved immediately.

4 Discussion and Conclusion

This work has presented a novel fuzzy implication stemming from a fuzzy lattice inclusion measure function. It was shown that the proposed fuzzy implication satisfies a number of "reasonable axioms" and properties. Future work remains to study a number of interesting issues with a significant potential in practical applications as explained in the following.

A straightforward future extension includes consideration of intervals of truth-values in $[0,1]$ instead of the sole consideration of trivial intervals (single numbers) in $[0,1]$ — Note that the enabling technology for dealing with intervals was introduced recently [9]. An additional future extension regards consideration of L-fuzzy sets [4] towards a fuzzy implication involving granular (fuzzy) inputs [8].

Finally, note that using a different positive valuation function $v : \mathsf{L} \to \mathsf{R}$ than $v(x) = x$ is not expected to change any property because v is a strictly increasing function. We point out that a different inclusion measure function, namely $\sigma_\wedge(a,b) = \begin{cases} \frac{v(a \wedge b)}{v(a)}, & a \neq 0 \\ 1, & a = 0 \end{cases}$, [6], [7] is identical to inclusion measure $\sigma_\vee(a,b)$ under the assumptions of this work, i.e. $\sigma_\wedge(a,b) = \sigma_\vee(a,b)$.

References

1. Birkhoff, G.: Lattice Theory. American Mathematical Society, Providence, RI, ser Colloquium Publications **25** (1967)
2. Dubois, D., Prade, H.: Fuzzy Sets and Systems: Theory and Applications. Academic Press, New York, NY (1980)
3. Fodor, J. C., Roubens, M.: Fuzzy preference modelling and multicriteria decision support. In: Theory and Decision Library, Kluwer Academic Publishers, Dordrecht, The Netherlands (1994)
4. Goguen, J.: L-fuzzy sets. J. Math. Analysis and Applications **18(1)** (1967) 145–174
5. Hatzimichailidis, A. G, Kaburlasos, V. G., Papadopoulos, B. K.: An implication in fuzzy sets. In: Proc. World Congress on Computational Intelligence (WCCI), FUZZ-IEEE Program, Vancouver, BC, Canada, 16-21 July 2006, pp. 203-208
6. Kaburlasos, V. G.: Towards a Unified Modeling and Knowledge-Representation Based on Lattice Theory. Springer, Heidelberg, Germany, ser Studies in Computational Intelligence **27** (2006)
7. Kaburlasos, V. G., Athanasiadis, I. N., Mitkas, P. A.: Fuzzy lattice reasoning (FLR) classifier and its application for ambient ozone estimation. Intl. J. Approximate Reasoning **45(1)** (2007) 152–188
8. Kaburlasos, V. G., Kehagias, A.: Novel fuzzy inference system (FIS) analysis and design based on lattice theory. IEEE Trans. Fuzzy Systems **15(2)** (2007) 243–260
9. Kaburlasos, V. G., Papadakis, S. E.: A granular extension of the fuzzy-ARTMAP (FAM) neural classifier based on fuzzy lattice reasoning (FLR). Neurocomputing (accepted) (Special Issue: JCIS 2007)
10. Klir, G. J., Yuan, B.: Fuzzy Sets and Fuzzy Logic: Theory and Applications. Prentice Hall, Upper Saddle River, NJ (1995)
11. Nguyen, H. T., Walker, E. A.: A First Course in Fuzzy Logic. CRC Press (1997)
12. Smets, P., Magrez, P.: Implication in fuzzy logic. Intl. J. Approximate Reasoning **1(4)** (1987) 327–347
13. Türksen, B., Kreinovich, V., Yager, R.R.: A new class of fuzzy implications. Axioms of fuzzy implication revisited. Fuzzy Sets and Systems **100(1-3)** (1998) 267–272
14. Zadeh, L. A.: Fuzzy sets. Information and Control **8(3)** (1965) 338–353

Author Index

Editors:	Vassilis Kaburlasos, Uta Priss, Manuel Graña
Publisher & Print:	Palacký University, Olomouc Křížkovského 8, 771 47 Olomouc Czech Republic
Title:	LBM 2008 (CLA 2008), Proceedings of the Lattice-Based Modeling Workshop, in conjunction with The Sixth International Conference on Concept Lattices and Their Applications
Place, year, edition:	Olomouc, 2008, 1^{st}
Page count:	x+67
Impression:	70

Not for sale

ISBN 978–80–244–2112–4