Book Review

Witold Pedrycz *An Introduction to Computing with Fuzzy Sets: Analysis, Design, and Applications Springer, Cham, Switzerland, 2021*

Vladik Kreinovich*

Department of Computer Science, University of Texas at El Paso, El Paso, TX, USA

Why do we need yet another introductory book on fuzzy sets. There are quite a few introductory book on fuzzy sets, so why do we need yet another one? The title of this book provides a convincing answer, actually, two convincing answers.

- First, it is a book *not* about fuzzy sets per se, it is a book about *computing* with fuzzy sets.
- Second, it is not a book about foundations – although foundations are also mentioned: it is a book about *analysis*, *design*, and *applications* that use fuzzy sets.

What is this book's novelty. And hereby lies the book's novelty: while mathematics of fuzzy sets evolves while remains largely the same, applications do change a lot. So what this book describes is techniques underlying modern applications of fuzzy techniques.

How fuzzy techniques were used in the past. During the last fuzzy boom of 1980s-1990s, fuzzy techniques were mostly used for the same purpose as originally envisioned by Lotfi Zadeh: to translate the expert knowledge – formulated by using impre-

cise ("fuzzy") words from natural language – into a precise control strategy. We start with expert's rules about cooking rice – and we produce an efficient rice cooker. We start with expert's rules describing how an expert train engineers control a subway train – and we get an excellent fuzzy controlled subway train system.

In those days, we did not have a problem finding the rules: the rules cames "from the horse's mouth", i.e., from the experts themselves. Ingenuity was in how to translate these rules into a precise control strategy: how to elicit degrees, how to best interpolate these degrees to get membership functions, what "and"- and "or"-operations (t-norms and t-conorms) to use, what defuzzification procedure to use, later – whether to use type-1 or type-2 fuzzy sets. In all this, interesting and helping results and heuristics were developed, and this joint effect of theoretical development and practical testing led to the boom.

But what about now? But that boom is over. It is still possible to make further improvements, it is still possible to find some situations when the usual approach works – such situation still happen, as we can see from papers presented at major fuzzy conferences, but such situations are rare.

Fuzzy techniques still lead to many successful applications – but nowadays, it is a rare situation

*Corresponding author. Vladik Kreinovich, Department of Computer Science, University of Texas at El Paso, 500 W. University, El Paso, TX 79968, USA. E-mail: vladik@utep.edu.

when all we have is expert rules. Usually, we also have some models, we also have some recorded controls – all this needs to be taken into account to get a good decision or control system. In processing expert rules, fuzzy is still very efficient, but nowadays, it is rarely enough just to use fuzzy techniques – we usually need to combine them with other control and decision making techniques.

In effect, we moved from too little knowledge to too much knowledge. When Lotfi Zadeh invented fuzzy techniques in the mid-1960s, the problem he faced was that, in many situations, we had a very small amount of knowledge – e.g., we knew that if the input is small, the control should be small. The problem was to extract as much information as possible from this relatively small amount of knowledge. This is why, in early fuzzy papers, so much emphasis was on describing the expert's knowledge as accurately as possible – how to elicit the detailed membership function that best reflects the expert's opinion, how to find "and"- and "or"-operations that best describe expert reasoning.

Nowadays, the situation is opposite: we have an abundance of data. Sensors have become cheaper and cheaper, recording is easy, processing is easy – and, as a result, we have so much data that we do not know what to do with it.

So how do we handle all this data? How do we humans deal with the abundance of information? We get a visual picture of the world, with an equivalent of Megabytes of data, every second, but usually we do not get lost, we make fast and reasonable decisions. How do we do it?

The answer is straightforward: when we see a picture – e.g., when someone shows us a picture of an animal – we do not analyze it pixel-by-pixel as naive image processing algorithms do, we immediately divide this image into *granules*: tail, paws, face, etc. This granularity is the main way of dealing with large amounts of data.

Granular computing: the main focus of the book. Some granules are crisp: 1-D data can be divided into intervals, multi-D data into sets. However, in general, granules are not crisp, their borders are fuzzy – and such fuzzy granules is what lies behind most modern applications of fuzzy techniques.

Methods of processing these fuzzy granules are largely the same as in the past, but the main difference is that now granules do not come from simply translating expert's words, they come from the detailed analysis of all the available information, including expert rules and available numerical data.

This view of fuzzy sets as a particular case of granules is the main focus, the main leitmotif of this book.

How the book is structured: first part. Because of this focus, this book starts *not* with an explanation of what is a fuzzy set – as usual introductory books on fuzzy do – it starts with introducing the motion of an *information granule* and with the main ideas of granular computing (Chapter 1). Only after the reader understands the main ideas of granules and granular computing, fuzzy sets are introduced as an important class of granules (in Chapter 2).

Usual textbooks would jump from here to operations on fuzzy sets – but not this book, it first described operations on another important class of granules – a class of intervals. This is a pedagogically good way to introduce operations on fuzzy sets: indeed, intervals can be naturally viewed as a simple particular case of fuzzy sets, and it is always a good idea to first describe operations on a simple particular case and only then move to the general more complex one.

Once we learn operations on intervals, we can extend them to operations on the usual [0, 1]-based fuzzy sets – and this is also a reasonable starting point to extend the usual arithmetic and logical operations to more general aggregation operations (e.g., averaging), and to extend all these operations to more general types of granules – interval-valued and type-2 fuzzy sets, rough fuzzy sets, probabilistic granules, and hybrid granules – that combine fuzzy and probabilistic uncertainty.

We have learned how to deal with granules, but how should we form granules and how do we put all this together: the second part of the book. After learning how to process granules, the natural next step is how to form these granules.

The book starts explaining this, in Chapter 9, with fuzzy clustering – the most frequently used way to form fuzzy granules. In Chapter 10, the granular formation is explained on the general level of granular computing. A granule should cover a sufficient number of examples *and* at the same time be sufficiently specific – i.e., its elements should be different from others. If we, as frequently happens in fuzzy applications, interpret "and" as a product, the need to maximize the degree to which both objectives are satisfied leads to maximizing the product of appropriately defined degrees of coverage and specificity; this idea is known as the Principle of Justified Granular-

ity. This principle is then explained on the examples of fuzzy granules.

Once we have the conclusion in the form of a granule as well, how do we go from there to the actual recommendation? In fuzzy techniques, this is called defuzzification, for general granules, it is called degranulation. Such methods are described in Chapter 11, after which in Chapters 12 and 13, all this knowledge is combined together on the example of fuzzy models.

What we have described so far is a model based on all available knowledge. But knowledge does not remain unchanged: we get new knowledge all the time. How can we use this new knowledge to update our model? This model update is the domain of what is called machine learning, so a reasonable idea is to use machine learning techniques for the desired update. At present, the most effective machine learning techniques are neural networks.

How to combine fuzzy and neural techniques is the subject of Chapter 14. Finally, Chapter 15 provides advice on how to use all this in applications. In most applications, we want to optimize our models; different optimization techniques are recalled in the corresponding Appendix.

Who this book is for. This book is intended for graduate students or advanced undergraduate ones. It can also be highly recommended to practitioners and researchers interested in learning, using, and improving the state-of-the-art fuzzy- and granular-computing techniques. Enjoy!