KYBERNETIKA — VOLUME 38 (2002), NUMBER 3, PAGES 235–244

PROGRAM FOR GENERATING FUZZY LOGICAL OPERATIONS AND ITS USE IN MATHEMATICAL PROOFS

TOMÁŠ BARTUŠEK AND MIRKO NAVARA[1]

Fuzzy logic is one of the tools for management of uncertainty; it works with more than two values, usually with a continuous scale, the real interval $[0,1]$. Implementation restrictions in applications force us to use in fact a finite scale (finite chain) of truth degrees. In this paper, we study logical operations on finite chains, in particular conjunctions. We describe a computer program generating all finitely-valued fuzzy conjunctions (t-norms). It allows also to select these t-norms according to various criteria. Using this program, we formulated several conjectures which we verified by theoretical proofs, thus obtaining new mathematical theorems. We found out several properties of t-norms that are quite surprising. As a consequence, we give arguments why there is no "satisfactory" finitely-valued conjunction. Such an operation is desirable, e. g., for search in large databases. We present an example demonstrating both the motivation and the difficulties encountered in using many-valued conjunctions. As a by-product, we found some consequences showing that the characterization of diagonals of finitely-valued conjunctions differs substantially from that obtained for t-norms on $[0,1]$.

1. BASIC FACTS ABOUT MULTI-VALUED CONJUNCTIONS

We assume that we have a totally ordered set, P, of real numbers (truth values) containing a least element, $\perp$, and a greatest element, $\top$. A *fuzzy conjunction* (*triangular norm, t-norm*) is a binary operation $\underset{\bullet}{\wedge}\colon P^2 \to P$ which is commutative, associative, nondecreasing, and has $\top$ as the neutral element. If $P = \{0,1\}$, then the only t-norm on P is the Boolean conjunction. For $P = [0,1]$, the following three basic (continuous) t-norms are considered in most applications: *standard or Gödel t-norm* (minimum) $a \underset{\mathbf{S}}{\wedge} b = \min(a,b)$, *Łukasiewicz t-norm* $a \underset{\mathbf{L}}{\wedge} b = \max(a+b-1,0)$, and *product t-norm* $a \underset{\mathbf{P}}{\wedge} b = a \cdot b$. As all t-norms are commutative and associative, they admit a natural extension to more than two arguments.

[1]Work supported by the Czech Ministry of Education under Research Programme MSM 212300013 "Decision Making and Control in Manufacturing", grant 201/02/1540 of the Grant Agency of the Czech Republic, and CEEPUS net SK-042.

From the theoretical point of view, we may consider a continuum of truth values, usually the real interval $[0,1]$. Nevertheless, practical applications are limited to a finite number of truth values. First, the technical implementation allows us to work only with a finite (though very large) number of values. Second, when representing vagueness it is usually meaningless to distinguish a high number of truth values; only a small number suffices. We concentrate on the case when the set of truth values P is finite. To simplify the notation, we may restrict – without loss of generality – to the chains of truth values in the form $P_n = \{0, 1, \ldots, n\}$, $n \in N$, where n represents the truth value $\top$ and 0 represents $\perp$.

Remark 1.1. We have to distinguish carefully between t-norms on *continuous domains*, e. g., on $[0,1]$, and *continuous t-norms*, i. e., those which are continuous as real functions. Their continuity has to be considered relatively to P (with its topology inherited from the linear topology on the real line). In this sense, each t-norm on a discrete domain is continuous (with respect to the discrete topology).

We refer to [4, 7, 12] for discussion of which t-norms on continuous domains can be restricted to discrete domains, and conversely, which discrete t-norms may be extended to continuous domains. E. g., the restriction of the standard t-norm $\underset{\mathbf{S}}{\wedge} = \min$ to any subset having a greatest element is a t-norm. The Łukasiewicz t-norm can be restricted to a t-norm on $P = \{0, \frac{1}{n}, \ldots, \frac{n-1}{n}, 1\}$. In contrast to this, the product t-norm cannot be restricted to a discrete domain.

2. MOTIVATING EXAMPLE – SEARCH IN LARGE DATABASE

We present an example of use of many-valued conjunction. We search for the most relevant items in a large database. A query may be formulated as a logical formula composed from various requirements. This formula could be evaluated using the rules of many-valued logic. A conjunction must be implemented as one of the basic logical connectives because very often the query is understood as a conjunction of various criteria.

Let us demonstrate the properties of t-norms on an example of search of a flight. From numerous possibilities, none is completely satisfactory. We want to take into account many criteria, e. g.:

- the destination airport,
- the days of departure and return,
- time of flight,
- change between flights,
- price,
- comfort of services, etc.

A sample problem is demonstrated in Table 1. It is a highly simplified example inspired by a real-life situation. The criteria are reduced and we exhibit only the most attractive alternatives from thousands of possibilities. The degrees of satisfaction

Table 1. Sample problem of choosing a flight.

company	A	C	F	L	S
price	0.7	0.9	0.6	0.7	0.7
airport	1	0.5	1	1	1
day of departure	1	1	1	1	0.8
day of return	1	1	1	1	0.9
take off time there	0.7	0.7	1	0.8	0.6
arrival time there	1	1	0.8	0.7	1
take off time back	0.7	0.7	1	0.8	0.7
arrival time back	1	1	1	0.8	1
transit airport	0.8	1	0.6	0.9	1
transit time	0.8	1	0.8	0.8	1
product t-norm	*0.21952*	*0.2205*	*0.2304*	*0.18063*	*0.21168*
minimum t-norm	*0.7*	*0.5*	*0.6*	*0.7*	*0.6*
Łukasiewicz t-norm	*0*	*0*	*0*	*0*	*0*

of the criteria can be calculated according to the user's preferences and expressed relatively well by the numbers in the table. The aim is to formulate an algorithm which selects the best solution for the particular user. Moreover, we know that this task is in some sense solvable by a human expert – the travel agent, using even more rough data, has found a solution which seemed to be optimal (better than the others when compared mutually), namely company A. The problem is how to replace human reasoning by an algorithm.

The idea is to aggregate the criteria using a fuzzy conjunction, because they were supposed to be satisfied simultaneously. The three basic t-norms did not give useful results. The product t-norm resulted in very similar values. They can be distinguished, but it is not clear whether this preference really represents our priorities. Besides, it uses very many values, while the initial task was to keep only a limited finite number of truth values during the whole procedure. The standard t-norm (minimum) makes no distinction between two alternatives. What is more disappointing is that much information is ignored, e.g., the change between flights. Moreover, we would have got the same result even if we totally ignored the price! The Łukasiewicz t-norm is of little use in case of many entries; here it gives zero in all alternatives, making no distinction between them.

3. CRITERIA FOR THE CHOICE OF MULTI-VALUED CONJUNCTIONS

Numerous examples of discrete t-norms may be obtained, e.g., by the techniques of [5, 6, 7, 8]. Some constructions known for t-norms on $[0,1]$, in particular ordinal sums, can be used to compose new examples of discrete t-norms from simpler ones (see [7] for more details).

Definition 3.1. A continuous t-norm $\bigwedge\limits_{\bullet}$ on P is called

– *archimedean* iff

$$\forall a \in P \setminus \{\top\}\ \forall b \in P \setminus \{\bot\}\ \exists n \in N : \underbrace{a \underset{\cdot}{\wedge} a \underset{\cdot}{\wedge} \ldots \underset{\cdot}{\wedge} a}_{n-\text{times}} < b\,,$$

– *strict* iff

$$\forall a, c \in P\ \forall b \in P \setminus \{\bot\} : (a < c \Rightarrow a \underset{\cdot}{\wedge} b < c \underset{\cdot}{\wedge} b)\,.$$

These classes of t-norms can be recognized from their diagonals:

Definition 3.2. Let $\underset{\cdot}{\wedge}$ be a t-norm on P. The *diagonal* of $\underset{\cdot}{\wedge}$ is the unary function $d: P \to P$ defined by $d(a) = a \underset{\cdot}{\wedge} a$.

Proposition 3.3. Let P be a closed set of reals. A continuous t-norm $\underset{\cdot}{\wedge}$ on P with diagonal d is

– archimedean iff

$$\forall a \in P \setminus \{\bot, \top\} : d(a) < a\,,$$

– strict iff

$$\forall a \in P \setminus \{\bot, \top\} : \bot < d(a) < a\,.$$

The diagonals of continuous t-norms on $[0,1]$ were characterized in [10]:

Theorem 3.4. The conjunction of the following conditions is a necessary and sufficient condition for a function $d: [0,1] \to [0,1]$ to be a diagonal of a continuous t-norm on $[0,1]$:

– d is a continuous and nondecreasing, $d(0) = 0$, $d(1) = 1$,

– $\forall a \in [0,1] : d(a) \leq a$,

– d is strictly increasing on $[0,1] \setminus d^{-1}(I)$, where $I = \{a \in [0,1] : d(a) = a\}$.

4. WHY NOT? (WE HAVE PLENTY OF MANY–VALUED CONJUNCTIONS)

There is an abundance of discrete t-norms. We made a computer program which, for given $v \in N$, generates all v-valued t-norms (with values in P_n, where $n = v - 1$). A similar program was already written by De Baets [2], so we had a chance to verify some results. The number of t-norms grows with v faster than exponentially (see Table 2).

Further, we implemented procedures that filter the t-norms according to several criteria. They distinguish

Table 2. Numbers of v-valued t-norms.

v	no. of t-norms	archimedean	smooth	ordinal sums	none of these
2	1	1	1	0	0
3	2	1	2	1	0
4	6	2	4	3	1
5	22	6	8	11	5
6	94	22	16	45	27
7	451	95	32	205	151
8	2368	471	64	1021	894
9	13775	2670	128	5512	5593
10	86417	17387	256	32095	36935
11	590489	131753	512	201367	257369

- t-norms obtained as ordinal sums of other t-norms,
- archimedean t-norms,
- smooth t-norms (also *divisible*, [6]), i. e., such that $\forall a, c \in P : (a < c \Rightarrow \exists b \in P : a = c \wedge b)$.

The numbers of t-norms in these classes are in Table 2. (The second column gives the number of all v-valued t-norms, the last column contains the number of all t-norms which do not belong to any of the three classes.) It seems that we have a large choice even if we restrict, e. g., to archimedean t-norms. The user may become confused from such ambiguity and may have difficulties to choose one t-norm for his particular application.

To simplify our orientation, we may restrict attention to the diagonals. It is natural to ask whether an analogue of Theorem 3.4 holds for the discrete case, too. Surprisingly, the conditions of Theorem 3.4 are neither necessary nor sufficient for a function on a finite chain to be the diagonal of some discrete t-norm [1]:

Example 4.1. The following table determines a t-norm $\underset{1}{\wedge}$ on P_4 which has a diagonal violating the last condition of Theorem 3.4.

$\underset{1}{\wedge}$	0	1	2	3	4
0	0	0	0	0	0
1	0	0	0	0	1
2	0	0	1	1	2
3	0	0	1	1	3
4	0	1	2	3	4

Further examples will appear in Theorem 5.2. These discrete t-norms do not admit an extension to t-norms on continuous domains.

Example 4.2. The sequence $(0, 0, 1, 2, 4)$ is the set of values of a function d on P_4 which satisfies the conditions of Theorem 3.4, but it is not the diagonal of a t-norm on P_4. Similarly, the following sequences determine functions that do not appear as diagonals of t-norms on P_5: $(0,0,0,2,3,5)$, $(0,0,1,2,3,5)$. An example for P_6 is $(0,0,0,0,3,4,6)$. These examples were obtained by checking the diagonal of all t-norms on P_m, $v = 2, \dots, 7$, but they will also follow from Theorem 5.2.

5. WHY NOT! (NO MANY–VALUED CONJUNCTION SATISFIES COMPLETELY OUR GOAL)

Although there are numerous archimedean finitely-valued t-norms, none of them has satisfactory properties simulating those of the algebraic product. The drawbacks of the basic t-norms became apparent in Section 2. Idempotency of the standard t-norm is sometimes desirable, but it also reduces the information obtained from the arguments – only one of them determines the resulting value, the others become irrelevant. Further, a conjunction of many facts that are satisfied to the same extent should be evaluated by a lower value than any of them. Therefore it is often important to avoid idempotent elements at all, in other words, to restrict attention to *archimedean* t-norms. This is what we do in the sequel.

A natural candidate among archimedean t-norms could be the discrete Łukasiewicz t-norm. However, it vanishes at one half of its domain. This prevents the user from distinguishing low degrees of satisfaction of conjunctions of requirements. This feature becomes severe when we work with conjunctions of more arguments:

Proposition 5.1. For $z \in N$, let us consider the z-ary operation on P_n obtained by the natural extension of the discrete Łukasiewicz t-norm to z arguments. This operation vanishes except for less than $1/z!$ of its domain.

Thus the Łukasiewicz conjunction of many arguments results very often to zero and gives us very little information about its arguments, see Section 2.

We want to find archimedean many-valued conjunctions which are "nonvanishing" in the sense that nonzero inputs result in a zero output only "rarely". Although there is an abundance of discrete t-norms, none of them achieves values that resemble the product or another strict t-norm. Looking for the reason, we derived and proved the following quite surprising result:

Theorem 5.2. Let $n \geq 3$. Let $c \in P_n \setminus \{\perp, \top\}$. Suppose that $\dot{\wedge}$ is an archimedean t-norm on P_n such that

$$(c+1) \dot{\wedge} (c+1) = c\,.$$

Then

$$a \dot{\wedge} b = c \qquad \text{and} \qquad a \dot{\wedge} e < c$$

for all $a, b \in \{c+1, \dots, n-1\}$, $e \in \{0, \dots, c\}$.

Proof. We proceed by induction over $k = c+1, \dots, n-1$, proving that the statement holds for all $a, b \in \{c+1, \dots, k\}$. Notice that the condition $a \dot{\wedge} e < c$ is trivially satisfied for $e < c$, so it is enough to check the condition $a \dot{\wedge} c < c$ instead.

For $k = c+1$, we only have to check the condition $k \mathbin{\underset{\cdot}{\wedge}} c < c$, where the inequality $k \mathbin{\underset{\cdot}{\wedge}} c \leq c$ follows from monotonicity. It remains to exclude the case $k \mathbin{\underset{\cdot}{\wedge}} c = c$ which leads to a contradiction with associativity:

$$c = k \mathbin{\underset{\cdot}{\wedge}} c = k \mathbin{\underset{\cdot}{\wedge}} (k \mathbin{\underset{\cdot}{\wedge}} c) = (k \mathbin{\underset{\cdot}{\wedge}} k) \mathbin{\underset{\cdot}{\wedge}} c = c \mathbin{\underset{\cdot}{\wedge}} c < c\,.$$

So the statement of the theorem holds for $a = b = c + 1$.

In the inductive step, we assume that the statement holds for all $a, b \in \{c + 1, \ldots, k-1\}$, where $c + 1 < k < \top$, and we prove it for all $a, b \in \{c+1, \ldots, k\}$. Without loss of generality, we assume $a = k$.

Similarly as above, $k \mathbin{\underset{\cdot}{\wedge}} c = c$ leads to a contradiction with associativity:

$$c = k \mathbin{\underset{\cdot}{\wedge}} c = k \mathbin{\underset{\cdot}{\wedge}} (k \mathbin{\underset{\cdot}{\wedge}} c) = \underbrace{(k \mathbin{\underset{\cdot}{\wedge}} k)}_{<k} \mathbin{\underset{\cdot}{\wedge}} c < c\,,$$

where the latter inequality follows from the preceding paragraph and $k \mathbin{\underset{\cdot}{\wedge}} k < k$ from the archimedean property.

Let $b < k$. Due to monotonicity, $k \mathbin{\underset{\cdot}{\wedge}} b \geq b \mathbin{\underset{\cdot}{\wedge}} b = c$. If $k \mathbin{\underset{\cdot}{\wedge}} b > c$, then

$$c = (k \mathbin{\underset{\cdot}{\wedge}} b) \mathbin{\underset{\cdot}{\wedge}} b = k \mathbin{\underset{\cdot}{\wedge}} (b \mathbin{\underset{\cdot}{\wedge}} b) = k \mathbin{\underset{\cdot}{\wedge}} c < c\,,$$

a contradiction, hence $k \mathbin{\underset{\cdot}{\wedge}} b = c$.

In the remaining case, $b = k$, we suppose that $k \mathbin{\underset{\cdot}{\wedge}} k > c$ and we get a contradiction:

$$c = (c+1) \mathbin{\underset{\cdot}{\wedge}} (k \mathbin{\underset{\cdot}{\wedge}} k) = ((c+1) \mathbin{\underset{\cdot}{\wedge}} k) \mathbin{\underset{\cdot}{\wedge}} k = c \mathbin{\underset{\cdot}{\wedge}} k < c\,,$$

hence $k \mathbin{\underset{\cdot}{\wedge}} k = c$. This completes the inductive step and the whole proof. □

The latter theorem leads to examples extending Example 4.1 and proves the nonexistence of t-norms with diagonals presented in Example 4.2. It has the following consequence for $c = 1$:

Corollary 5.3. Let $n \geq 3$. There is *exactly one* archimedean t-norm $\mathbin{\underset{\cdot}{\wedge}}$ on P_n such that $2 \mathbin{\underset{\cdot}{\wedge}} 2 = 1$, namely the t-norm $\underset{\mathbf{d}}{\wedge}$ defined by

$$a \underset{\mathbf{d}}{\wedge} b = \begin{cases} a & \text{if} \quad b = n\,, \\ b & \text{if} \quad a = n\,, \\ 1 & \text{if} \quad a, b \in \{2, \ldots, n-1\}\,, \\ 0 & \text{if} \quad a = 0 \quad \text{or} \quad b = 0\,. \end{cases}$$

(For $n = 4$, this is the t-norm $\underset{\mathbf{1}}{\wedge}$ from Example 4.1.)

The latter corollary represents a very strict restriction. The values appearing in the assumption are not so unusual. In fact, the value $1 \underset{\cdot}{\wedge} 1 = 0$ is necessary for all archimedean t-norms on P_n. Trying to have the next diagonal entry nonzero, $2 \underset{\cdot}{\wedge} 2 = 1$ is the only possibility for archimedean t-norms. As we see, this single value leaves us only one archimedean t-norm, $\underset{\mathbf{d}}{\wedge}$. Moreover, the remaining values of $\underset{\mathbf{d}}{\wedge}$ (constant on a substantial part of its domain) are quite far from our intentions and do not exhibit a nice fuzzy conjunction. It is "nonvanishing", but still it does not give much information about the arguments. Moreover, if a, b, c are smaller than $\top = n$, then $a \underset{\mathbf{d}}{\wedge} b \underset{\mathbf{d}}{\wedge} c = \bot = 0$. Thus when we apply this t-norm to the example from Section 2, we get zeros everywhere, just as for the Łukasiewicz t-norm. Looking for discrete analogues of strict t-norms, we are forced to admit also $2 \underset{\cdot}{\wedge} 2 = 0$, thus coming closer to "vanishing" t-norms and the disadvantages of the Łukasiewicz t-norm formulated in Proposition 5.1.

Let us note that for $c > 1$ the t-norm in Theorem 5.2 is not unique.

Remark. The main results in this section were obtained using our software for investigation of finitely-valued t-norms (see the Appendix for its brief description). Computer programs were used for generation of examples, formulating and testing hypotheses. Then we looked for the proofs of the facts observed. This part is a purely mathematical job which cannot be algorithmized now.

6. CONCLUSION

We proved that there exists a single value in one point of the domain which is achieved by *just one* archimedean v-valued t-norm. Moreover, it is a value which can be naturally required. What is even worse, the only t-norm satisfying this condition is constant on most of the domain, giving rather poor information about the arguments. Therefore, despite the large number of all v-valued t-norms, our choice is very limited and unsatisfactory. This is a quite surprising conclusion.

Let us consider the choice of a many-valued conjunction for a database query. If we choose a t-norm which is not archimedean, we ignore arguments that are satisfied to a higher degree. Choosing an archimedean t-norm, we have to allow t-norms that vanish for many combinations of arguments, thus resulting in zero too often. Again, a substantial loss of information appears, especially if the query is formulated as a conjunction of many criteria.

To overcome these difficulties, one may relax some of the defining conditions of t-norms. Monotonicity seems to be an inevitable condition. Sometimes commutativity or associativity is not required. However, they are quite natural and express the fact that the order in which requirements are written is unimportant. Further, associativity plays a crucial role in the principles of fuzzy controllers [9, 11]. The existence of a neutral element, $\top$, need not be required, but again, it is a natural aspect of our reasoning. If some criteria have not yet been considered (e. g., because of having been completely satisfied in all previous items) and we decide to include them, the missing values can be replaced by $\top$ without any change of the evaluations already computed. Still the use of more general aggregation operators (without neutral elements) could be an alternative, although violating the above arguments. What remains a problem is the finite number of truth values which is rarely preserved by aggregation operators.

From the above discussion, it seems that the simplest solution would be to allow infinitely many truth values. Really, this seems to be the best way how to get useful and relatively satisfactory results and avoid a serious loss of information. We have already presented arguments (although maybe not very persuading) why we tried to avoid the use of a continuum of values. Looking for finitely-valued conjunctions, we encountered unexpected obstacles.

To choose an optimal many-valued conjunction from the point of view of information obtained, one has to develop other (quantitative) criteria. Our software tool possesses a basis for such investigation. This seems to be a perspective field for further research.

In this paper, we wanted to point out the problems and initiate a discussion about their possible solutions; this relates to the very basic definitions and we hope that it could serve as an inspiration for further nonstandard approaches to uncertainty management.

APPENDIX: SPECIALIZED SOFTWARE FOR MANY-VALUED CONJUNCTIONS AND ITS CONTRIBUTION TO MATHEMATICAL THEORY

The contribution of this paper was strongly dependent on a collection of programs for investigation of finitely-valued t-norms. Here we give an overview of its components and possibilities. More can be found in [1] and at URL

`http:\\cmp.felk.cvut.cz\~navara\d_tnorms\`

The cornerstone of our software is program `tnorms` which, for given $p \in N$, generates all v-valued t-norms (on P_{v-1}) for $v = 2, \dots, p$, and stores them in files. Output file tn$v$.dat contains all t-norms in v-valued logic. These files serve as an input for further procedures. Program `tnorms` also assigns to each t-norm a unique number which can be used for further reference (e. g., in examples). Each record of a t-norm contains the number of that t-norm.

Program `classify` filters t-norms from a given file according to the following three t-norm properties:

- archimedean property,
- smoothness,
- possibility of decomposition to a nontrivial ordinal sum.

Arbitrary conjunctions of these properties or their negations are allowed. Further criteria for classification can be easily added to the source code.

As the data files appear to be quite large for a higher number of values, we developed an internal format in which they are recorded. This format is not appropriate for humans; it should be converted to a readable, but longer format (text) by means of program `vizo`.

If the reader wants to restrict attention to the diagonals of t-norms only (which represents a significant reduction of data), a specialized program `diagonal` extracts the diagonals of t-norms from a file in the internal format and displays them.

All programs are written in C++ in order to allow their easy transfer to other systems. Our experiments were made on PC's with 32-bit Windows environment.

(Received January 30, 2002.)

REFERENCES

[1] T. Bartušek: Fuzzy Operations on Finite Sets of Truth Values (in Czech). Diploma Thesis, Czech Technical University, Praha 2001.

[2] B. De Baets and R. Mesiar: Triangular norms on product lattices. Fuzzy Sets and Systems *104* (1999), 61–75.

[3] B. De Baets and R. Mesiar: Discrete triangular norms. In: Topological and Algebraic Structures (E. P. Klement and S. Rodabaugh, eds.), Universität Linz 1999, pp. 6–10.

[4] B. De Baets and R. Mesiar: Discrete triangular norms. In: Topological and Algebraic Structures in Fuzzy Sets: Recent Developments in the Mathematics of Fuzzy Sets (S. Rodabaugh and E. P. Klement, eds.). Kluwer Academic Publishers, 2002, to appear.

[5] C. Drossos and M. Navara: Matrix composition of t-norms. In: Enriched Lattice Structures for Many-Valued and Fuzzy Logics (S. Gottwald and E. P. Klement, eds.), Univ. Linz 1997, pp. 95–100.

[6] L. Godo and C. Sierra: A new approach to connective generation in the framework of expert systems using fuzzy logic. In: Proc. 11th Internat. Symposium on Multiple-Valued Logic, IEEE Computer Society Press, Palma de Mallorca 1988, pp. 157–162.

[7] E. P. Klement, R. Mesiar, and E. Pap: Triangular Norms. Kluwer, Dordrecht – Boston – London 2000.

[8] G. Mayor and J. Torrens: On a class of operators for expert systems. Internat. J. Intell. Syst. *8* (1993), 771–778.

[9] E. H. Mamdani and S. Assilian: An experiment in linguistic synthesis with a fuzzy logic controller. J. Man-Machine Stud. *7* (1975), 1–13.

[10] R. Mesiar and M. Navara: Diagonals of continuous triangular norms. Fuzzy Sets and Systems *104* (1999), 34–41.

[11] B. Moser and M. Navara: Conditionally firing rules extend the possibilities of fuzzy controllers. In: Proc. Internat. Conf. Computational Intelligence for Modelling, Control and Automation (M. Mohammadian, ed.), IOS Press, Amsterdam 1999, pp. 242–245.

[12] P. Viceník: A note to a construction of t-norms based on pseudo-inverses of monotone functions. Fuzzy Sets and Systems *104* (1999), 15–18.

Ing. Tomáš Bartušek, Silicon & Software Systems, Philips Building, Šafránkova 1, Praha 5. Czech Republic.
e-mail: tomasbartusek@seznam.cz

Doc. Ing. Mirko Navara, DrSc., Center for Machine Perception, Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University, Technická 2, 166 27 Praha 6. Czech Republic.
e-mail: navara@cmp.felk.cvut.cz