A Natural Extension of F-Transform to Triangular and Triangulated Domains Necessitates the Use of Triangular Membership Functions

Hana Zámečníková[1], Irina Perfilieva[1], Olga Kosheleva[2][0000−3333−2587−4209], and Vladik Kreinovich[3][0000−0002−1244−1650]

[1] IRAFM, University of Ostrava, 30. dubna 22, 701 03 Ostrava, Czech Republic, {hana.zamecnikova,irina.perfilieva}@osu.cz
[2] Department of Teacher Education, University of Texas at El Paso, 500 W. University, El Paso TX 79968, USA, olgak@utep.edu
[3] Department of Computer Science, University of Texas at El Paso, 500 W. University, El Paso TX 79968, USA, vladik@utep.edu

Abstract. In many practical situations when we process 1-D data, the method of F-transform turned out to be very useful. In this method, we can use either triangular membership functions or more complex ones. Because this method has been so successful in 1-D applications, a natural idea is to extend it to functions defined on 2-D and higher-dimensional domains – e.g., to images. This method allows natural generalization to rectangular domains, where it indeed turned out to be very effective. A recent paper showed that it can extended to more general domains – e.g., to triangular domains and to more general domains that are divided into triangular domains by triangulation. Interestingly, while all 1-D membership functions can be extended to the rectangular domains, the current extension to triangular and more general domains was produced only for triangular membership functions. In this paper, we show that this restriction is not accidental: a natural extension of F-transform to triangular domains is only possible for triangular membership functions. This may explain why such membership functions are often very effective.

Keywords: F-transform· Triangular and triangulated domains · Triangular membership functions.

1 Introduction

1-D and multi-D F-transform: outline. In many practical situations when we process 1-D data, the method of F-transform turned out to be very useful; see, e.g., [3, 4].

In this method, we can use both piecewise-linear (e.g., triangular) membership functions, as well as more complex ones. Because this method has been so successful in 1-D applications, a natural idea is to extend it to functions defined on 2-D and higher-dimensional domains – e.g., to images.

This method allows natural generalization to rectangular domains, where it indeed turned out to be very effective; see, e.g., [2, 6]. A recent paper [5] showed that it can extended to more general domains – e.g., to triangular domains and to more general domains that are divided into triangular domains by triangulation.

F-transform for triangular domains: challenge. Interestingly:

- while all 1-D membership functions can be extended to the rectangular domains,
- the current extension to triangular and more general domains was produced only for triangular membership functions.

What we do in this paper. In this paper, we show that this restriction is not accidental: a natural extension of f-transform to triangular domains is only possible for triangular membership functions.

This may explain why such membership functions are often very effective.

How this paper is structured. In Section 2, we briefly remind the readers about 1-D F-transform. In Section 3, we explain the main ideas behind extending 1-D F-transform to triangular domains. Finally, in Section 4, we explain our main result: that extension to triangular domains is only possible for triangular membership functions.

2 1-D F-transform: a brief reminder

In many practical situations when we process 1-D data $x(t)$, $t \in [0, T]$, the method of F-transforms turned out to be very useful.

In this method, we divide the interval $[0, T]$ into several subintervals $[t_0, t_1]$, $\ldots$, $[t_{n-1}, t_n]$ of equal length, and select continuous membership functions $A_0(t)$, $A_1(t)$, $\ldots$, $A_{n-1}(t), A_n(t)$ each of which $A_i(i)$ is equal to 0 outside the interval $[t_{i-1}, t_{i+1}]$ and whose sum is equal to 1. These functions may be triangular, or they may be more complex. Usually, with the exception of the first and the last of these functions $A_0(t)$ and $A_n(t)$, all these functions can be obtained from each other by shift: $A_i(t - t_i) = A_j(t - t_j)$.

Then, we replace the original signal $x(t)$ with the values

$$F_i = \frac{\int A_i(t) \cdot x(t)\, dt}{\int A_i(t)\, dt}.$$

These values $F_0, \ldots, F_n$ form what is known as *F-transform* of the original signal. To form these values, we take the weighted average of the original signal – and thus, drastically decrease the random noise component of the measured signal.

Based on the F-transform, we can reasonably accurately reconstruct the original signal by applying the inverse transform:

$$\overline{x}(t) = \sum_{i=0}^{n} F_i \cdot A_i(t).$$

3 Main Idea Behind Extending 1-D F-Transform to Triangular Domains

To describe the desired extension, let us first reformulate 1-D transform in a more general form. On the local level, when we only consider the functions on a single subinterval $[t_i, t_{i+1}]$, the above description of 1-D transform becomes simplified. Namely, on each such subinterval, we have only two non-zero membership functions $A_i(t)$ and $A_{i+1}(t)$ (that add up to 1):

- the function $A_i(t)$ that is equal to 1 at one of the endpoints of the subinterval, when $t = t_i$ and to 0 at the other endpoint, when $t = t_{i+1}$, and
- the function $A_{i+1}(t)$ that is equal to 1 at the endpoint $t = t_{i+1}$ and to 0 at the endpoint $t = t_i$.

Since all the functions $A_i(t)$ are obtained from each other by a shift, to describe all the functions $A_i(t)$, it is sufficient to describe two basic functions $a(s)$ and $b(s)$ that transform interval $[0, 1]$ into itself, and that transform 0 into 0 and 1 into 1. For this purpose, we can take $a(s) \stackrel{\text{def}}{=} A_{i+1}(t_i + s \cdot (t_{i+1} - t_i))$ and $b(s) \stackrel{\text{def}}{=} A_i(t_{i+1} - s \cdot (t_{i+1} - t_i))$. In terms of these functions, the functions $A_{i+1}(t)$ and $A_i(t)$ can be obtained by applying a linear transformation from $[0, 1]$ to, correspondingly, interval $[t_i, t_{i+1}]$ and to the same interval with the opposite direction – which we will denote by $[t_{i+1}, t_i]$. Namely we have

$$A_{i+1}(t) = a\left(\frac{t - t_i}{t_{i+1} - t_i}\right)$$

and

$$A_i(t) = b\left(\frac{t_{i+1} - t}{t_{i+1} - t_i}\right).$$

In particular, triangular membership function corresponds to $a(s) = b(s) = s$.

In terms of these basic functions, the condition $A_i(t) + A_{i+1}(t) = 1$ takes the following simplified form:

$$a(s) + b(1 - s) = 1. \qquad (1)$$

This condition is, of course, always satisfied in the case of triangular membership functions, when $a(s) = b(s) = s$.

The above reformulation leads to a natural extension of 1-D F-transform to a triangular domain. The above reformulation uses the fact that an interval has two endpoints. For each of these two endpoints, we formed a membership function that is equal to 1 in the selected endpoint and equal to 0 at the other endpoint. This provides the values of these two membership functions at both endpoints of the interval. To get the values of each of the two membership functions at a point P inside the interval, we use two basic functions $a(s)$ and $b(s)$ defined on the basic interval $[0, 1]$ for which $a(0) = b(0) = 0$ and $a(1) = b(1) = 1$.

For each of the two desired membership functions m and for each point P, we take $m(P) = a(s)$ (or, correspondingly, $m(P) = b(s)$), where $s = L(P)$ for a linear transformation L that:

- maps the point $s = 0$ (where the basic function has value 0) into the point where the desired membership function has the value 0, and
- maps the point $s = 1$ (where the basic function has value 1) into the point where the desired membership function has the value 1.

Of course, a linear transformation L from an interval to an interval is uniquely determined by the values $L(e)$ for both endpoints e, so this transformation – and thus, the resuting membership functions – are uniquely defined.

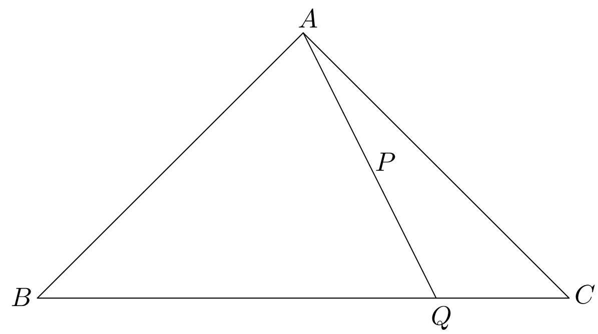
A triangular domain has *three* vertices. We will denote them by A, B, and C. It is therefore reasonable to come up with *three* membership functions $m_a(x, y)$, $m_b(x, y)$, and $m_c(x, y)$, for which:

- the function $m_a(x, y)$ is equal to 1 at the point A and is equal to 0 at the two other vertices B and C – and on the whole segment BC;
- the function $m_b(x, y)$ is equal to 1 at the point B and is equal to 0 at the two other vertices A and C – and on the whole segment AC; and
- the function $m_c(x, y)$ is equal to 1 at the point C and is equal to 0 at the two other vertices A and B – and on the whole segment AB.

To describe the values of these three membership functions at a point P inside the triangular domain, we select three basic functions $a(s)$, $b(s)$, and $c(s)$ from $[0, 1]$ to $[0, 1]$, for which $a(0) = b(0) = c(0) = 0$ and $a(1) = b(1) = c(1) = 1$.

Then, to find the value $m_a(P)$, we take a straight line segment AQ starting with A and going through P until it reaches the segment BC at some point $Q \in BC$. We know that $m_a(Q) = 0$ and that $m_a(A) = 1$. So, to find the value $m_a(P)$, we use the value $a(s)$, where $s = L(P)$ is obtained by a linear transformation L from the interval QA to the interval $[0, 1]$ – a linear transformation L for which:

- the point Q at which $m_a(Q) = 0$ maps into the value $s = 0$ for which $a(s) = 0$, and
- the point A at which $m_a(A) = 1$ maps into the value $s = 1$ for which $a(s) = 1$.



Similarly, to find the value $m_b(P)$, we take a straight line segment BQ starting with B and going through P until it reaches the segment AC at some point $Q \in AC$. We know that $m_b(Q) = 0$ and that $m_b(B) = 1$. So, to find the value $m_b(P)$, we use the value $b(s)$, where $s = L(P)$ is obtained by a linear transformation L from the interval QB to the interval $[0, 1]$ – a linear transformation L for which:

- the point Q at which $m_b(Q) = 0$ maps into the value $s = 0$ for which $b(s) = 0$, and
- the point B at which $m_b(B) = 1$ maps into the value $s = 1$ for which $b(s) = 1$.

Finally, to find the value $m_c(P)$, we take a straight line segment CQ starting with C and going through P until it reaches the segment AB at some point $Q \in AB$. We know that $m_c(Q) = 0$ and that $m_c(C) = 1$. So, to find the value $m_c(P)$, we use the value $c(s)$, where $s = L(P)$ is obtained by a linear transformation L from the interval QC to the interval $[0, 1]$ – a linear transformation L for which:

- the point Q at which $m_c(Q) = 0$ maps into the value $s = 0$ for which $c(s) = 0$, and
- the point C at which $m_c(C) = 1$ maps into the value $s = 1$ for which $c(s) = 1$.

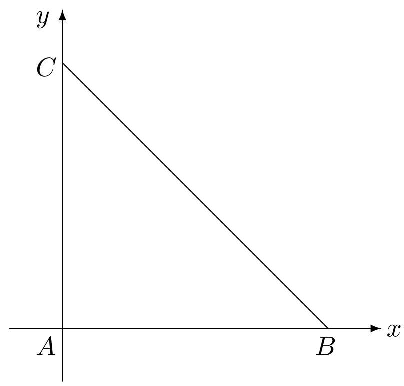
These three functions should add up to 1.

This is how the extension was done. This is how an extension was done in [5] – with $a(s) = b(s) = c(s) = s$.

Remaining challenge. Remaining challenge is to check if we can extend it to more general membership functions – i.e., to more general basic functions. As promised, in the following (last) section, we prove that this is not possible: that only for triangular membership functions, we can have an extension to a triangular domain.

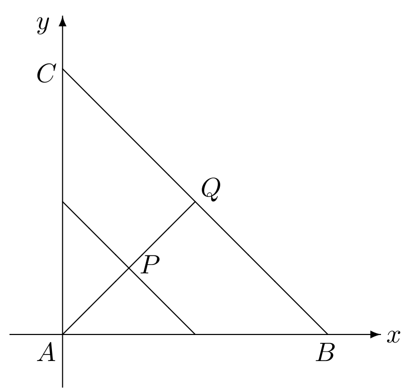
4 Main Result: Extension to Triangular Domains Is Only Possible for Triangular Membership Functions

Specific case. To prove our result, let us consider a specific example of a triangular domain: a right equilateral triangle with sides of length 1:

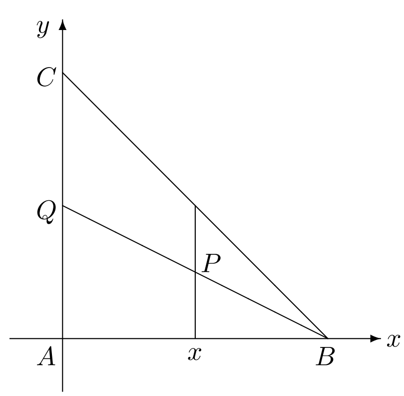


Corresponding membership functions. One can check that for $m_a(P)$, the above procedure leads to

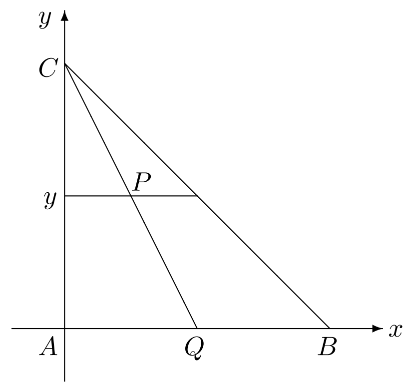
$$m_a(x, y) = a(1 - (x + y)) :$$



For $m_b(P)$, we get $m_b(x, y) = b(x)$:



Finally, for $m_c(P)$, we get $m_c(x, y) = c(y)$:



The functional equation describing the requirement that the membership functions should add to 1. Now that we have explicit expressions for all three membership functions in terms of the basic functions, the requirement $m_a(P) + m_b(P) + m_c(P) = 1$ (that the sum of the three membership functions be equal to 1) takes the following form:

$$a(1 - (x + y)) + b(x) + c(y) = 1. \tag{2}$$

Let us find all possible triples of functions that satisfy this functional equation.

Let us solve the resulting functional equation. For $y = 0$, we have $c(y) = c(0) = 0$ and thus, the equation (2) takes the following simplified form:

$$a(1 - x) + b(x) = 1. \tag{3}$$

Thus, for all x, we have

$$a(1 - x) = 1 - b(x). \tag{4}$$

Similarly, for $x = 0$, we have $b(x) = b(0) = 0$ and thus, the equation (2) takes the following simplified form:

$$a(1 - y) + c(y) = 1. \tag{5}$$

Thus, for all y, we have

$$a(1 - y) = 1 - c(y). \tag{6}$$

For $x = y$, from (4) and (6), we can conclude that $1 - b(y) = 1 - c(y)$, so

$$c(y) = b(y). \tag{7}$$

Now, from (4), we can conclude that

$$a(1 - (x + y)) = 1 - b(x + y). \tag{8}$$

Substituting the expressions (7) and (8) into the equality (2). we conclude that

$$1 - b(x + y) + b(x) + b(y) = 1. \tag{9}$$

Subtracting 1 from both sides and adding $b(x + y)$ to both sides, we conclude that

$$b(x + y) = b(x) + b(y). \tag{10}$$

It is known (see, e.g., [1] that every continuous solution of the equation (10) has the form $b(x) = c \cdot x$ for some c. Since $b(1) = 1$, we get $c = 1$ and $b(x) = x$. From the formulas (7) and (8), we can now conclude that $c(x) = a(x) = x$, i.e., that indeed $a(s) = b(s) = c(s) = s$ for all s.

In other words, the only case when the three above-defined membership functions $m_a(P)$, $m_b(P)$, and $m_c(P)$ add up to 1 is when all these functions are triangular. The result is thus proven.

5 Acknowledgments

This work was supported in part by the National Science Foundation grants 1623190 (A Model of Change for Preparing a New Generation for Professional Practice in Computer Science), HRD-1834620 and HRD-2034030 (CAHSI Includes), EAR-2225395 (Center for Collective Impact in Earthquake Science C-CIES), and by the AT&T Fellowship in Information Technology.

It was also supported by a grant from the Hungarian National Research, Development and Innovation Office (NRDI), and by the Institute for Risk and Reliability, Leibniz Universitaet Hannover, Germany.

References

1. J. Aczél and J. Dhombres, *Functional Equations in Several Variables*, Cambridge University Press, 2008.
2. F. Di Martino et al., "A color image reduction based on fuzzy transforms", *Information Sciences*, 2014, Vol. 266, pp. 101–111.
3. I. Perfilieva, "F-transform", In: *Springer Handbook of Computational Intelligence*, Springer, 2015, pp. 113–130.
4. T. M. T. Pham, J., Janeček, and I. Perfilieva, "Fuzzy transform on 1-D manifolds", In: *Biomedical and Other Applications of Soft Computing*, Springer, Cham, Switzerland, 2022, pp. 13–24.
5. H. Zámečniková and I. Perfilieva, "F-transform on triangulated domain", In: Marie-Jeanne Lesot, Marek Reformat, Susana Vieira, Joao P. Carvalho, Fernando Batista, Bernadette Bouchon-Meunier, and Ronald R. Yager (eds.), *Information Processing and Management of Uncertainty in Knowledge-Based Systems, Abstracts of the 20th International Conference IPMU 2024*, Lisbon, Portugal, July 22–26, 2024.
6. H. Zámečniková, I. Perfilieva, and R. Valášek, "Nonlocal Laplace operator in image processing", In: *Developments of Artificial Intelligence Technologies in Computations and Robotics: Proceedings of the 14th International FLINS Conference*< 2020, pp. 956–963.