Short thesis for the degree of doctor of philosophy (PhD)

COMPUTER-ASSISTED INVESTIGATIONS ON FUNCTIONAL EQUATIONS AND CONVEXITY

written by: LAN NHI TO
supervised by: DR. ATTILA GILÁNYI

UNIVERSITY OF DEBRECEN
Doctoral School of Mathematical and Computational Sciences

Debrecen, 2025

This short thesis includes the main results of the doctoral dissertation. Specifically, we developed new programs using the computer algebra system Maple to investigate certain functional equations and convexity problems. All results have been thoroughly covered in the dissertation and are comprehensively elaborated in the papers [24], [25], [26], and [27].

Alienness of linear functional equations

The concepts of alienness and strong alienness of functional equations were introduced by J. Dhombres in his paper [13] in the following form. Let $E_1(f) = 0$ and $E_2(f) = 0$ be two functional equations for a function f defined on a nonempty set X and mapping to a groupoid Y with an identity element 0. The equations E_1 and E_2 are called alien with respect to X and Y, if each solution $f : X \to Y$ of

$$E_1(f) + E_2(f) = 0$$

is a solution of the system

$$\begin{aligned} E_1(f) &= 0 \\ E_2(f) &= 0. \end{aligned}$$

The equations E_1 and E_2 are said to be strongly alien with respect to X and Y, if all solutions $f, g : X \to Y$ of

$$E_1(f) + E_2(g) = 0$$

are solutions of the system

$$\begin{aligned} E_1(f) &= 0 \\ E_2(g) &= 0. \end{aligned}$$

These properties of functional equations have been investigated by several authors during the last decades (cf., e.g., [4], [5], [17], [18], [20], [32], [37], [38], and [41]).

We considered the alienness phenomenon for linear functional equations of the type

$$\sum_{i=0}^{n+1} f_i(p_i x + q_i y) = 0 \qquad (x, y \in X), \tag{1}$$

where n is a positive integer, $p_0, \dots, p_{n+1}$ and $q_0, \dots, q_{n+1}$ are rational numbers, X, Y are linear spaces and $f_0, \dots, f_{n+1} : X \to Y$ are unknown functions.

Classical results on the class of functional equations above and on its important sub-classes were published by M. Fréchet ([15], [16]) and W. H. Wilson ([49]). Their solutions, under general conditions, were determined by L. Székelyhidi ([44], [47]; cf., also, [46] and [43]). Recent studies of (1) and its generalizations, were performed, among others, in [21], [30], [39] and [42]. The alienness property, connected to linear functional equations, was also considered in several papers (e.g. in [3], [14], [19] and [42]).

Basic properties of linear functional equations

Fundamental results on solutions of functional equations of type (1) determined by L. Székelyhidi ([44], [47]; cf., also, [49]). In these theorems and in the following parts of the section, we use the convention $0^0 = 1$.

Theorem 1. *Let X and Y be linear spaces over the field of the rationals, let $p_0, p_1, \dots, p_{n+1}$ and $q_0, q_1, \dots, q_{n+1}$ be rational numbers and assume that*

$$M_k^{(i)} : X \to Y$$

are monomial functions of degree k for $i = 0, \dots, n+1$ and $k = 0, \dots, n$. The functions $f_0, \dots, f_{n+1} : X \to Y$,

$$f_i(x) = \sum_{k=0}^{n} M_k^{(i)}(x) \quad (x \in X,\ i = 0, \dots, n+1)$$

solve functional equation (1) if and only if the monomial functions $M_k^{(i)} : X \to Y$ given above fulfil

$$\sum_{i=0}^{n+1} p_i^j q_i^{k-j} M_k^{(i)}(x) = 0 \qquad (x \in X,\ k = 0, \dots, n,\ j = 0, \dots, k).$$

Theorem 2. *Let X and Y be linear spaces over the field of the rationals, let $p_0, p_1, \dots, p_{n+1}$ and $q_0, q_1, \dots, q_{n+1}$ be rational numbers that satisfy*

$$p_i q_j \neq p_j q_i \quad (i, j = 0, \dots, n+1,\ i \neq j). \tag{2}$$

The functions $f_0, \dots, f_{n+1} : X \to Y$ solve functional equation (1) if and only if they are of the form

$$f_i(x) = \sum_{k=0}^{n} M_k^{(i)}(x) \qquad (x \in X,\ i = 0, \dots, n+1),$$

where

$$M_k^{(i)} : X \to Y \quad (i = 0, \dots, n+1,\ k = 0, \dots, n)$$

are monomial functions of degree k satisfying the equations

$$\sum_{i=0}^{n+1} p_i^j q_i^{k-j} M_k^{(i)}(x) = 0 \qquad (x \in X,\ k = 0, \dots, n,\ j = 0, \dots, k).$$

Extensions of the concepts of alienness and strong alienness

In order to consider alienness and strong alienness properties for more than two functional equations and more than one (or two) functions, we extend the corresponding definitions of J. Dhombres ([13]) in the following way.

Let m, n be positive integers ($m \geq 2$) and let

$$\begin{aligned} E_1(f_1, \dots, f_n) &= 0 \\ &\vdots \\ E_m(f_1, \dots, f_n) &= 0 \end{aligned}$$

be functional equations for the unknown functions $f_1, \dots, f_n$ defined on a nonempty set X and mapping to a groupoid Y with an identity element 0.

Definition 1. The equations $E_1, \dots, E_m$ are called alien with respect to X and Y, if all solutions $f_i : X \to Y$, ($i = 1, \dots, n$), of

$$E_1(f_1, \dots, f_n) + \dots + E_m(f_1, \dots, f_n) = 0 \tag{3}$$

are solutions of the system

$$\begin{aligned} E_1(f_1, \dots, f_n) &= 0 \\ &\vdots \\ E_m(f_1, \dots, f_n) &= 0, \end{aligned} \tag{4}$$

too.

Definition 2. The equations $E_1, E_2, \ldots, E_m$ are said to be strongly alien with respect to X and Y, if all solutions $f_{ij} : X \to Y$, $(i = 1, \ldots, m,\ j = 1, \ldots, n)$ of

$$E_1(f_{11}, \ldots, f_{1n}) + \cdots + E_m(f_{m1}, \ldots, f_{mn}) = 0 \tag{5}$$

are solutions of the system

$$\begin{aligned} E_1(f_{11}, \ldots, f_{1n}) &= 0 \\ &\vdots \\ E_m(f_{m1}, \ldots, f_{mn}) &= 0 \end{aligned} \tag{6}$$

as well.

The program isalien

The program function **`isalien`** was developed in the computer algebra system Maple, which investigates the alienness and strong alienness of linear functional equations of type (1). It uses another user-defined Maple program **`lfesolve`**, which determines solutions of functional equations belonging to class (1). (Cf. [11], furthermore, [9], [10], [28] and [29].).

The function **`isalien`** is contained in the Maple package **`FunctionalEquations`**, sub-package **`AlienationCheck`**. The input of the program should be of the form:

```
> with(FunctionalEquations:-AlienationCheck);
> isalien(e,f,options);
```

where

- `e`: the list of linear functional equations,
- `f`: the list of the unknown functions included in the given equations,
- The program has three types of options:
  . CHECK = {‘`alien`’, ‘`strongalien`’}
  . OUTPUT = {‘`brief`’, ‘`verbose`’, ‘`full`’}
  . ErrorWarn_STYLE = {‘`ewYes`’, ‘`ewNo`’}

  The default options are: [‘`alien`’, ‘`brief`’, ‘`ewYes`’].

Some examples of running the program `isalien`:

```
> E1 := f(x + y) − f(x) − f(y) = 0:
  E2 := g(2x + 3y)−2 g(x − y) = 0:
  isalien([E1, E2], [f, g], strongalien, full);
```

gives

```
    The functional equations
    E1:
        f(x + y) − f(x) − f(y) = 0
    E2:
        g(2x + 3y) − 2g(x − y) = 0
    are not strongly alien
    In details,
    The solution of the function equation
    E1 + E2 = 0 are
        {f = − M(1, 0) + M(0, 1), g = M(1, 0)}
    while the solution of the system
    [E1 = 0, E2 = 0] are
        {f = M(0, 1), g = 0}.
```

```
> E1 := f_1(2x − y) − 2f_1(x) + f_1(y) + f_2(x + 5y)
        + 2f_2(2x − 7y) − 2f_2(x) − f_2(y)
        + f_3(2x − 3y) − f_3(y) = 0:
  E2 := g_1(x + y) − g_1(x) − g_1(y) + g_2(x + y)
        + 2g_2(x − y) − 3g_2(x) − g_2(y)
        + g_3(2x − y) − 2g_3(y) = 0:
  isalien([E1, E2], [[f_1, f_2, f_3], [g_1, g_2, g_3]],
                full);
```

gives

```
    The functional equations
    E1:
        f_1(2x − y) − 2f_1(x) + f_1(y) + f_2(x + 5y)
            + 2f_2(2x − 7y) − 2f_2(x) − f_2(y)
            + f_3(2x − 3y) − f_3(y) = 0
    E2:
        f_1(x + y) − f_1(x) − f_1(y) + f_2(x + y)
            + 2f_2(x − y) − 3f_2(x) − f_2(y)
            + f_3(2x − y) − 2f_3(y) = 0
    are alien if polynomial solutions are considered
    only.
    In details,
```

```
The polynomial solution of function equation
E1 + E2 = 0 and the polynomial solution of the
system[E1 = 0, E2 = 0] are
      {f_1 = - M(1, 0) - M(2, 0) + M(0, 1),
          f_2 = M(1, 0), f_3 = M(2, 0)}.
```

Levi-Civita type functional equations

We consider Levi-Civita type functional equations, i.e. the class

$$f(x+y)=\sum_{i=1}^{n} g_i(x)h_i(y), \tag{7}$$

where n is a positive integer, G is an Abelian group and $f, g_i, h_i : G \to \mathbb{C}$ $(i = 1, 2, \dots, n)$ are unknown functions.

Basic results about solutions of Levi-Civita type functional equations

Fundamental results on solutions of functional equations of type (7) determined by L. Székelyhidi ([47], [45]).

According to L. Székelyhidi, any function $f : G \to \mathbb{C}$, satisfying the Levi-Cevita functional equation (7), is a normal exponential polynomial of order at most n. And the converse remains true, that is, any normal exponential polynomial f fulfills the equation (7).

If the functions $h_1, h_2, \dots, h_n$ are linearly independent, then $g_1, g_2, \dots, g_n$ are linear combinations of translates of f. Hence, they are also normal exponential polynomial of order at most n. Moreover, they are built up from the same additive and exponential functions, as f.

We call the case when the functions $g_1, g_2, \dots, g_n$ and the functions $h_1, h_2, \dots, h_n$ are linearly independent as the non-degenerate case.

In the non-degenerate case, the functions f, g_i, h_i, $(i = 1, \dots, n)$

have the form:

$$f(x) = \sum_{j=1}^{k} P_j(a_{j,1}(x), \dots, a_{j,n_j-1}(x))m_j(x)$$

$$g_i(x) = \sum_{j=1}^{k} Q_{i,j}(a_{j,1}(x), \dots, a_{j,n_j-1}(x))m_j(x) \tag{8}$$

$$h_i(x) = \sum_{j=1}^{k} R_{i,j}(a_{j,1}(x), \dots, a_{j,n_j-1}(x))m_j(x)$$

where

- $k, n_1, \dots, n_k$ are positive integers such that $n_1 + \cdots + n_k = n$,
- $m_1, \dots, m_k$ are different non-zero complex exponentials,
- $\{a_{j,1}(x), \dots, a_{j,n_j-1}(x)\}$ are sets of linearly independent real additive functions for $j = 1, \dots, k$,
- $P_j, Q_{i,j}, R_{i,j}$ are complex polynomials of degree at most $n_j - 1$ in $n_j - 1$ variables for $i = 1, \dots, n$, $j = 1, \dots, k$

With the same assumptions, for any $j = 1, \dots, k$ and for any multi-indices $A_j = (\alpha_1, \alpha_2, \dots, \alpha_{n_j-1})$ and $B_j = (\beta_1, \beta_2, \dots, \beta_{n_j-1})$ in $\mathbb{N}^{n_j-1}$, we introduce the matrix $M_j(P; A_j, B_j)$ and the matrix $N_j(Q; A_j)$ as follows:

- for any choice of $p, q = 0, \dots, n_j - 1$,

$$M_j(P; A_j, B_j)_{n_j-p, n_j-q} =$$

$$\begin{cases} \frac{1}{p!q!}\partial_1^{\alpha_1} \dots \partial_p^{\alpha_p} \partial_1^{\beta_1} \dots \partial_q^{\beta_q} P_j(0, \dots, 0) & \text{for } p + q < n_j \\ 0, & \text{for } p + q \geq n_j, \end{cases} \tag{9}$$

- for any choice of $p = 1, \dots, n_j$, and $q = 1, \dots, n$,

$$N_j(Q; A_j)_{p,q} = \frac{1}{(n_j - p)!}\partial_1^{\alpha_1} \dots \partial_{n_j-p}^{\alpha_{n_j-p}} Q_{q,p}(0, \dots, 0). \tag{10}$$

Then, we define two block matrices

$$M(P;A_1,\ldots,A_k,B_1,\ldots,B_k) =$$

$$\begin{bmatrix} M_1(P;A_1,B_1) & 0 & \ldots & 0 \\ 0 & M_2(P;A_2,B_2) & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & M_k(P;A_k,B_k) \end{bmatrix}; \quad (11)$$

and

$$N(Q;A_1,\ldots,A_k) = \begin{bmatrix} N_1(Q;A_1) \\ \vdots \\ N_k(Q;A_k) \end{bmatrix}. \quad (12)$$

Then we have a necessary and sufficient condition in terms of the coefficients of the complex polynomials $P_j, Q_{i,j}, R_{i,j}$ such that f, g_i, h_i is a non-degenerate solution of the Levi-Civita functional equation (**??**).

$$M(P;A_1,\ldots,A_k,B_1,\ldots,B_k) =$$
$$N(Q;A_1,\ldots,A_k)N(R;B_1,\ldots,B_k)^T \quad (13)$$

holds for any choice of the multi-indices A_j, B_j in $\mathbb{N}^{n_j-1}$ ($j = 1,\ldots,k$).

The program LCfesolve

The program function `LCfesolve` was developed in the computer algebra system Maple, which investigates solutions to Levi-Civita-type functional equations (7).

The program `LCfesolve` is included in the Maple package `FunctionalEquations:-LeviCivitafesolve`. Accessing function `LCfesolve` follows the same conventions as other Maple procedures:

```
> with(FunctionalEquations:-LeviCivitafesolve);
> LCfesolve(fe, output_option);
```

where

- `fe`: the Levi-Civita functional equations,
- `output_option`: {‘`brief`’, ‘`full`’}

  The default option is: ‘`brief`’.

In the following, example of running the program with the sine equation is presented.

```
> fe := f(x + y) = f(x)g(y) + g(x)f(y);
> LCfesolve(fe, full);
```

yields

Case: [1, 1]

f $= \alpha_1 * m_1(x) + \alpha_2 * m_2(x)$

g $= \gamma_1 * m_1(x) + \gamma_2 * m_2(x)$

Moreover, the coefficients fulfill the equation Q.R = P, with the matrices Q, R, P as follows, respectively

Q =

$$\begin{bmatrix} \alpha_1 & \gamma_1 \\ \alpha_2 & \gamma_2 \end{bmatrix}$$

R =

$$\begin{bmatrix} \gamma_1 & \gamma_2 \\ \alpha_1 & \alpha_2 \end{bmatrix}$$

P =

$$\begin{bmatrix} \alpha_1 & 0 \\ 0 & \alpha_2 \end{bmatrix}$$

The coefficients fulfill the system of equations:

$\{2\alpha_1\gamma_1 - \alpha_1, 2\alpha_2\gamma_2 - \alpha_2, \alpha_1\gamma_2 + \alpha_2\gamma_1\}$.

The solutions of the above system of equations:

$\{\alpha_1 = 0, \alpha_2 = 0, \gamma_1 = \gamma_1, \gamma_2 = \gamma_2\}$,

$\left\{\alpha_1 = 0, \alpha_2 = \alpha_2, \gamma_1 = 0, \gamma_2 = \frac{1}{2}\right\}$,

$\left\{\alpha_1 = \alpha_1, \alpha_2 = 0, \gamma_1 = \frac{1}{2}, \gamma_2 = 0\right\}$,

$\left\{\alpha_1 = \alpha_1, \alpha_2 = -\alpha_1, \gamma_1 = \frac{1}{2}, \gamma_2 = \frac{1}{2}\right\}$.

Case: [2]

$f = (\alpha_1 * A_1(x) + \alpha_2) * m_1(x)$

$g = (\gamma_1 * A_1(x) + \gamma_2) * m_1(x)$

Moreover, the coefficients fulfill the

equation Q.R = P, with the matrices Q, R, P
as follows , respectively

$$Q = \begin{bmatrix} \alpha_1 & \gamma_1 \\ \alpha_2 & \gamma_2 \end{bmatrix}$$

$$R = \begin{bmatrix} \gamma_1 & \gamma_2 \\ \alpha_1 & \alpha_2 \end{bmatrix}$$

$$P = \begin{bmatrix} 0 & \alpha_1 \\ \alpha_1 & \alpha_2 \end{bmatrix}$$

The coefficients fulfill the system of
equations:

$$\{2\alpha_1\gamma_1, 2\alpha_2\gamma_2 - \alpha_2, \alpha_1\gamma_2 + \alpha_2\gamma_1 - \alpha_1\}$$

The solutions of the above system of
equations:

$$\{\alpha_1 = \alpha_1, \alpha_2 = 0, \gamma_1 = 0, \gamma_2 = 1\},$$
$$\{\alpha_1 = 0, \alpha_2 = 0, \gamma_1 = \gamma_1, \gamma_2 = \gamma_2\},$$
$$\left\{\alpha_1 = 0, \alpha_2 = \alpha_2, \gamma_1 = 0, \gamma_2 = \frac{1}{2}\right\}.$$

Types of functional equations

The field of functional equations has grown significantly as a result of the works of key mathematicians from the late eighteenth century to the nineteenth century. In the 1960s, János Aczél's studies ([1], [2]) provided an in-depth exploration of functional equations and their wide-ranging applications in various mathematical and scientific fields.

In the 1980s, János Aczél posed the question if it was possible to create a computer program to solve functional equations, specifically those in the class of Linear Functional Equations, using Székelyhidi's method ([44], [47]). The first programs to address these equations were created over 20 years ago and were described in a thesis ([28]) and the newest version of it in papers ([10], [11]). Over the last few decades, many researchers have used computers to study functional equations ([6], [7], [8], [12], [31], [36], [40] and [25]).

Today, symbolic calculations combined with general computer programming are strong methods for exploring functional equations. We are currently developing a software package that includes programs related to functional equations. A natural idea that comes up is to create a program capable of determining the types or classes of functional equations. We introduced these programs and their applications.

Types of functional equations considered

- The **Cauchy's functional equations**

$$\begin{aligned} f(x+y) &= f(x)+f(y) \\ f(x+y) &= f(x)f(y) \\ f(xy) &= f(x)+f(y) \\ f(xy) &= f(x)f(y). \end{aligned}$$

- The **Pexider functional equations**

$$\begin{aligned} f(x+y) &= g(x)+h(y) \\ f(x+y) &= g(x)h(y) \\ f(xy) &= g(x)+h(y) \\ f(xy) &= g(x)h(y). \end{aligned}$$

- The **Jensen functional equation**

$$f\left(\frac{x+y}{2}\right) = \frac{f(x)+f(y)}{2}.$$

- The **Hosszú functional equation**

$$f(x+y-xy)+f(xy) = f(x)+f(y).$$

- The **Mikusiński functional equation**

$$f(x+y)[f(x+y)-f(y)-f(x)] = 0.$$

- The **alternative functional equation**

$$[f(x+y)]^2 = [f(x)+f(y)]^2.$$

– The **quadratic functional equation**

$$f(x+y) + f(x-y) = 2f(x) + 2f(y).$$

– The **D'Alembert functional equation**

$$f(x+y) + f(x-y) = 2f(x)f(y)$$

– The **inhomogeneous linear functional equation of order 1**

$$f(ax+by+c) = Af(x) + Bf(y) + C, \quad abAB \neq 0.$$

– The **trigonometric functional equations**

$$\begin{aligned} f(x+y) &= f(x)f(y) - g(x)g(y) \\ f(x-y) &= f(x)f(y) + g(x)g(y) \\ g(x+y) &= g(x)f(y) + f(x)g(y) \\ g(x-y) &= g(x)f(y) - g(y)f(x) \end{aligned}$$

– The class of **linear functional equations**

$$\sum_{i=0}^{n+1} f_i(p_i x + q_i y) = 0 \qquad (x, y \in X),$$

where n is a positive integer, $p_0, \ldots, p_{n+1}$ and $q_0, \ldots, q_{n+1}$ are rational numbers, X, Y are linear spaces and $f_0, \ldots, f_{n+1} : X \to Y$ are unknown functions.

– The class of **Levi-Civita functional equations**

$$f(x+y) = \sum_{i=1}^{n} g_i(x)h_i(y),$$

where n is a positive integer, G is an Abelian group and $f, g_i, h_i : G \to \mathbb{C}$ $(i = 1, 2, \ldots, n)$ are unknown functions.

In the next part, we present detailed descriptions of three program functions `fewhattype`, `fetype`, `feinfo`. They are developed in the computer algebra system Maple, and contained in the package `FunctionalEquations`, subpackage `FETypes`.

The function `fewhattype`

The input of the program should be of the form

```
> fewhattype(fe, [funcs, vars]);
```

where,

- `fe`: the functional equations,
- `funcs`: the list of unknown functions,
- `vars`: the list of unknown function's variables.

Two arguments `funcs` and `vars` are optional and in default,

```
funcs := [f,g,h,k,l];
vars := [x,y,z,u,v];
```

If the input is given correctly, the program will list all types and classes to which the given functional equation `fe` belongs. In the following part, some examples will be presented.

```
> fewhattype(f(x + y) - f(x) - f(y));
```

yields

```
Cauchy equation
General linear type equation
Linear type equation
Levi-Civita type equation
```

The program is able to simplify the given functional equation before starting the main process to determine its types or classes. For example:

```
> fe:= √3f(x + y) + √3f(x - y) = √12f(x) + √12f(y);
> fewhattype(fe);
```

gives

```
Linear type equation
Quadratic equation
```

When the unknown functions and their variables are not within the default case, users can add extra parameters to use the program, as follows:

```
> fe := s(a + b) = s(a).c(b) + c(a).s(b);
> fewhattype(fe, funcs = [s, c], vars = [a, b]);
```

gives

```
Levi−Civita type equation
Functional equation for trigonometric functions,
 the addition formula for sine.
```

The function `fetype`

If users input a functional equation `fe` and specify a type or class of functional equation `type`, this function will simply return value `True` or `False` to indicate whether `fe` is of `type`. Similarly to the previous program, extra parameters `funcs` and `vars` can be added as optional arguments to specify the list of unknown functions and their variables. Therefore, the input of the program should be as follows

```
> fetype(fe, type, [funcs, vars]);.
```

For example,

```
> fetype(f(xy) − f(x) − f(y) = 0, Cauchy);
```

shows

```
true
```

The program is able to simplify the considered functional equation before evaluating, as follows

```
> fetype(2.f((x + y)/2) = f(x) + f(y), Jensen);
```

yields

```
true
```

The extra parameter `funcs` is used in the example below for using the program in case the unknown function is not contained in the default set.

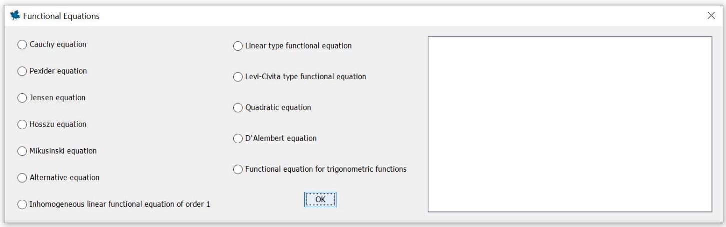
```
> fe := m(x + y − xy) + m(xy) = m(x) + m(y);
> fetype(fe, Hosszu, funcs = [m]);
```

gives

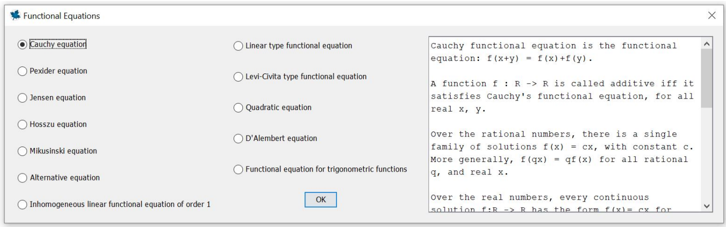
```
true
```

The function `feinfo`

Function *feinfo()* is used to get information about all considered types and classes of functional equations. By calling *feinfo()*, the following window will pop up



For example, when *Cauchy equation* is chosen



m-convex hull of sets of points

The concept of m-convexity in the sense of G. Toader is a mathematical problem that was introduced about thirty-five years ago and investigated by many authors (cf., eg., [48], [35], [34], [33]). However, the geometrical properties of m-convex sets have not been researched deeply. With the assistance of computers in general or computer programming in particular, we can get a geometric visualization of m-convexity for further investigation on the concept of convexity as well as its applications in functional equations and inequalities problems (cf., eg., [22], [23]).

We presented computer programs developed in the computer algebra system Maple, which visualizes the animation of the m-convex hull of sets of points on the Cartesian coordinate systems for a two-dimensional plane and a three-dimensional space; and determines the relative position of a given point and the m-convex hull of a set of points in both two and three-dimensional cases.

The concept of m-convexity

It is well known that, the set $C \subseteq \mathbb{R}^n$ is called convex if

$$tx + (1-t)y \in C, \quad \text{for all } x, y \in C \text{ and } t \in [0,1]$$

The convex hull of a non-empty set $A \subseteq \mathbb{R}^n$ is defined as the intersection of all convex subsets of $\mathbb{R}^n$ containing A.

According to G. Toader [48], let $m \in [0,1]$ be a fixed real number, a set $C \subseteq \mathbb{R}^n$ is called m-convex if for each $t \in [0,1]$

$$tx + m(1-t)y \in C, \quad \text{for all } x, y \in C$$

The m-convex hull of the non-empty set $A \subseteq \mathbb{R}^n$ is defined as the intersection of all m-convex subsets of $\mathbb{R}^n$ containing A.

From a geometric point of view, the set C is m-convex if for each two points $x, y \in C$, the segment $[x, my]$ connecting x and my is also included in C.

The animation of m-convex hull of sets on the Cartesian two and three-dimensional spaces

It has been shown in [33] that the m-convex hull generated by two different points $x, y \in \mathbb{R}^2$ can be obtained by dropping the triangle $T_m = \triangle(x, y, z)$ from triangle $T = \triangle(O, x, y)$, where O is the origin and z is a point defined by

$$z = \frac{m}{m+1}(x+y).$$

Thus,

$$T - T_m = \triangle(O, x, y) \setminus \triangle(x, y, z) = \triangle(O, x, z) \cup \triangle(O, z, y).$$

Algorithm - Generating an animation of the m-convex hull of a finite set of points in $\mathbb{R}^2$

- Input: Finite set of points P, $m \in [0, 1)$.
- Output: m-convex hull of P.
- Method: The m-convex hull C_P of P can be generated by the following

  $$\text{For all } A, B \in P \quad (A \neq B), \quad C_P = C_P \cup \triangle(OAz) \cup \triangle(OzB),$$

  $$\text{where } z = \frac{m}{m+1}(A+B)$$

Similarly, the m-convex hull generated by three different points $x, y, z \in \mathbb{R}^3$ can be obtained by dropping the polyhedron, which is composed by six vertex corners: x, y, z, t, u, v from the tetrahedron generated by four points O, x, y, z, where O is the origin and u, v, t are points defined by

$$t = \frac{m}{m+1}(x+y), u = \frac{m}{m+1}(x+z), v = \frac{m}{m+1}(y+z),$$

Accordingly to the two-dimensional plane result, it can be considered as the union of four tetrahedrons which are generated by $[O, x, t, u]$, $[O, y, u, v]$, $[O, z, u, v]$, $[O, t, u, v]$ respectively.

Algorithm - Generating an animation of the m-convex hull of a finite set of points in $\mathbb{R}^3$

- Input: Finite set of points P, $m \in [0, 1)$.
- Output: m-convex hull of P.
- Method: the m-convex hull C_P of P can be generated by the following

$$\begin{aligned} &\text{For all } A, B, C \in P \quad (A \neq B \neq C), \\ C_P = C_P &\cup tetrahedron[O, A, T, U] \cup tetrahedron[O, B, T, V] \\ &\cup tetrahedron[O, C, U, V] \cup tetrahedron[O, T, U, V], \end{aligned}$$

where

$$\begin{aligned} T &= \frac{m}{m+1}(A+B), \quad U = \frac{m}{m+1}(A+C), \\ V &= \frac{m}{m+1}(B+C), \quad O(0,0,0). \end{aligned}$$

We presented computer programs, which visualizes the animation of the m-convex hull of sets of points on the Cartesian coordinate systems and determines the relative position of a given point and the m-convex hull of the set of points in two- and three-dimensional cases.

The program functions `mconvex2d` and `mconvex3d`

Our programs `mcvhull2d` and `mcvhull3d` are contained in the Maple package `MConvexHull`. The input of the program should be of the forms

```
> mcvhull2d(P);
```

result will be six figures of m-convex hull of `P` with $m \in [0, 1)$.

```
> mcvhull2d(P, m);
```

where `m` is the fixed real number, such that $m \in [0, 1)$. Then the result will be the figure of the m-convex hull of `P` with given `m`.

```
> mcvhull2d(P, m1, m2);
```

here `m1` and `m2` are fixed real numbers, such that $0 \leq m1 < m2 < 1$. In this case, the program will show six figures of the m-convex hull of `P` with m, where $m1 \leq m \leq m2$.

```
> mcvhull2d(P, m1, m2, dfr);
```

where `m1` and `m2` are fixed real numbers, such that $0 \leq m1 < m2 < 1$ and `dfr` is the number of frames (or figures) the user wants to be shown. Therefore, the result will be `dfr` figures of the m-convex hull of `P` with m in $m1 \leq m \leq m2$.
All options of the program `mcvhull3d` are similar. In the following part, examples of running the programs will be given.

```
> P := [[14, 16], [13, 12], [5, 20], [25, 13],
        [7, 10], [10, 25], [1.5, 15], [16, 6],
        [26, 5], [20, 5], [17, 13], [15, 9.5]];
> mcvhull2d(P, 0.18);
```

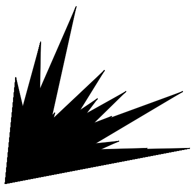


Figure 1: m = 0.18

When the input set contains duplicate points, the program will show a warning message on the screen and continue to run normally after removing the duplication.

```
> mcvhull3d([2, 2, 1], [2, 2, 3], [3, 0, 1],
        [2, 2, 1], 0.4);
```

yields

```
Warning, There are duplicate points in the set
of points.
```

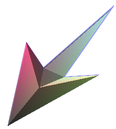


Figure 2: m = 0.4

Barycentric coordinate system

In geometry, a barycentric coordinate system is used for specifying the location of a point within a simplex.

Barycentric coordinates on triangles

Given the points specified as $A = (a_x, a_y), B = (b_x, b_y), C = (c_x, c_y)$ and a point $Q = (q_x, q_y)$, the barycentric coordinates of the point Q can be found by solving the following system of linear equations:

$$\begin{aligned} ax \cdot \lambda_1 + b_x \cdot \lambda_2 + c_x \cdot \lambda_3 &= q_x \\ a_y \cdot \lambda_1 + b_y \cdot \lambda_2 + c_y \cdot \lambda_3 &= q_y \\ \lambda_1 + \quad \lambda_2 + \quad \lambda_3 &= 1. \end{aligned}$$

If $\lambda_1, \lambda_2, \lambda_3 \geq 0$ then Q is inside the triangle.

Barycentric coordinates on tetrahedrons

Given the vertices of the tetrahedron as $A = (a_x, a_y, a_z)$, $B = (b_x, b_y, b_z)$, $C = (c_x, cy, c_z)$, $D = (d_x, d_y, d_z)$ and a point $Q = (q_x, q_y, q_z)$, the barycentric coordinates of the point Q can be found by solving the following a system of linear equations:

$$\begin{aligned} a_x \cdot \lambda_1 + b_x \cdot \lambda_2 + c_x \cdot \lambda_3 + d_x \cdot \lambda_4 &= q_x \\ a_y \cdot \lambda_1 + b_y \cdot \lambda_2 + c_y \cdot \lambda_3 + d_y \cdot \lambda_4 &= q_y \\ a_z \cdot \lambda_1 + b_z \cdot \lambda_2 + c_z \cdot \lambda_3 + d_z \cdot \lambda_4 &= q_z \\ \lambda_1 + \quad \lambda_2 + \quad \lambda_3 + \quad \lambda_4 &= 1. \end{aligned}$$

If $\lambda_1, \lambda_2, \lambda_3, \lambda_4 \geq 0$ then Q is inside the tetrahedron.

As mentioned before, in $\mathbb{R}^2$ the m-convex hulls of sets are constructed by the union of finitely many triangles with known vertices. Similarly, in $\mathbb{R}^3$ the m-convex hulls of sets are built up by the union of finitely many tetrahedrons with known vertices. With that structure, by applying the barycentric coordinate system to determine the position of a given point with respect to a simplex (triangle or tetrahedron), we developed a program to decide whether a given point in $\mathbb{R}^2$ or in $\mathbb{R}^3$ is an element of a m-convex hull of a set or not.

The program functions `PointInMCVHull2d` and `PointInMCVHull3d`

Our programs `PointInMCVHull2d` and `PointInMCVHull3d` are also contained in the Maple package `MConvexHull`. The input of the program should be of the form

```
> PointInMCVHull2d(Q, P, m, true/false);
> PointInMCVHull3d(Q, P, m, true/false);
```

where

- the given point called `Q`,
- the list of points `P` in $\mathbb{R}^2$ or in $\mathbb{R}^3$,
- the fixed real value `m`, $0 \leq m < 1$.

The program will decide whether the point `Q` is an element of the `m`-convex hull of the set of points `P` or not. Moreover, if the last parameter is `true`, the program will also show the animation for more information.

For example,

```
> P := [[14, 16], [13, 12], [5, 20], [25, 13],
        [7, 10], [10, 25], [1.5, 15], [16, 6],
        [26, 5], [20, 5], [17, 13], [15, 9.5]];
  PointInMCVHull2d([15, 16], P, 0.3, true);
```

shows

```
        False.
```

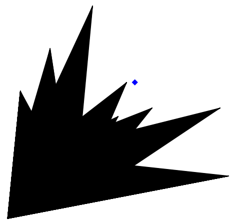


Figure 3

```
> P := [[1/2, 1/2, 1], [1, 0, 0], [0, 1, 0]];
  PointInMCVHull3d([1/4, 1/4, 1/4], P, 0.5, true);
```

returns

```
        True.
```

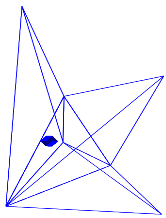


Figure 4

References

[1] J. Aczél. *Vorlesungen über Funktionalgleichungen und ihre Anwendungen.* Birkhäuser, Basel, 1961. Lehrbücher und Monographien aus dem Gebiete der exakten Wissenschaften, Mathematische Reihe, Bd. 25.

[2] J. Aczél. *Lectures on Functional Equations and Their Applications*, volume 19 of *Mathematics in Science and Engineering.* Academic Press, New York–London, 1966.

[3] M. Adam. Alienation of the quadratic and additive functional equations. *Anal. Math.*, 45:449–460, 04 2019.

[4] Youssef Aissi, Driss Zeglami, and Brahim Fadli. Alienation of drygas' and cauchy's functional equations. *Ann. Math. Sil.*, 35, 04 2021.

[5] A. Bahyrycz and J. Sikorska. On a general bilinear functional equation. *Aequationes Math.*, 95(6):1257–1279, 2021.

[6] Sz. Baják and Zs. Páles. Computer aided solution of the invariance equation for two-variable Gini means. *Comput. Math. Appl.*, 58:334–340, 2009.

[7] Sz. Baják and Zs. Páles. Computer aided solution of the invariance equation for two-variable Stolarsky means. *Appl. Math. Comput.*, 216(11):3219–3227, 2010.

[8] Sz. Baják and Zs. Páles. Solving invariance equations involving homogeneous means with the help of computer. *Appl. Math. Comput.*, 219(11):6297–6315, 2013.

[9] G. Gy. Borus and A. Gilányi. On a computer program for solving systems of functional equations. In 4^{th} *IEEE International Conference on Cognitive Infocommunications (CogInfoCom)*, page 939. IEEE, 2013.

[10] G. Gy. Borus and A. Gilányi. Solving systems of linear functional equations with computer. In 4^{th} *IEEE International Conference on Cognitive Infocommunications (CogInfoCom)*, pages 559–562. IEEE, 2013.

[11] G. Gy. Borus and A. Gilányi. Computer assisted solution of systems of two variable linear functional equations. *Aequationes Math.*, 94:723–736, 2020.

[12] S. Czirbusz. Testing regularity of functional equations with computer. *Aequationes Math.*, 84(3):271–283, 2012.

[13] J. Dhombres. Relations de dépendance entre les équations fonctionnelles de Cauchy. *Aequationes Math.*, 35(2-3):186–212, 1988.

[14] B. Fadli. Alienation of Cauchy's and the quadratic functional equations on semigroups. *Aequationes Math.*, 95(3):527–534, 2021.

[15] M. Fréchet. Une définition fonctionnelle des polynômes. *Nouv. Ann.*, 49:145–162, 1909.

[16] M. Fréchet. Les polynômes abstraits. *Journal de Math. Pures et Appl. Sér. IX*, 8:71–92, 1929.

[17] R. Ger. Additivity and exponentiality are alien to each other. *Aequationes Math.*, 80(1-2):111–118, 2010.

[18] R. Ger. Alienation of additive and logarithmic equations. *Ann. Univ. Sci. Budap. Rolando Eötvös, Sect. Comput.*, 40:269–274, 01 2013.

[19] R. Ger. On alienation of two functional equations of quadratic type. *Aequationes Math.*, 95:1169–1180, 2021.

[20] R. Ger and M. Sablik. Alien functional equations: a selective survey of results. In *Developments in functional equations and related topics*, volume 124 of *Springer Optim. Appl.*, pages 107–147. Springer, Cham, 2017.

[21] A. Gilányi and A. Lewicka. On linear functional equations modulo $\mathbb{Z}$. *Aequationes Math.*, 95(6):1301–1311, 2021.

[22] A. Gilányi, N. Merentes, and R. Quintero. Mathability and an animation related to a convex-like property. In 7^{th} *IEEE Conference on Cognitive Infocommunications (CogInfoCom)*, pages 227–232. IEEE, 2016.

[23] A. Gilányi, N. Merentes, and R. Quintero. Presentation of an animation of the m-convex hull of sets. In 7^{th} *IEEE Conference on Cognitive Infocommunications (CogInfoCom)*, pages 307–308. IEEE, 2016.

[24] A. Gilányi, R. Quintero, and L. N. To. Computer-assisted investigations of m-convex hulls of sets. *Manuscript*, 2025.

[25] A. Gilányi and L. N. To. Computer assisted investigation of alienness of linear functional equations. *Aequationes Math.*, 97:1185–1199, 2023.

[26] A. Gilányi and L. N. To. Computer-assisted investigation of Levi-Civita type functional equations. *Manuscript*, 2025.

[27] A. Gilányi and L. N. To. On a computer program determining types of functional equations. *Aequationes Math. (Published online first)*, 2025.

[28] A. Gilányi. *Charakterisierung von monomialen Funktionen und Lösung von Funktionalgleichungen mit Computern.* Diss., Universität Karlsruhe, Karlsruhe, Germany, 1995.

[29] A. Gilányi. Solving linear functional equations with computer. *Math. Pannon.*, 9(1):57–70, 1998.

[30] E. Gselmann, G. Kiss, and Cs. Vincze. On a class of linear functional equations without range condition. *Aequationes Math.*, 94(3):473–509, 2020.

[31] A. Házy. Solving linear two variable functional equations with computer. *Aequationes Math.*, 67(1-2):47–62, 2004.

[32] Z. Kominek and J. Sikorska. Alienation of the logarithmic and exponential functional equations. *Aequationes Math.*, 90, 2016.

[33] T. Lara, N. Merentes, Zs. Páles, R. Quintero, and E. Rosales. On m-convexity on real linear spaces. *UPI-JMB*, 1(2), 2018.

[34] T. Lara, R. Quintero, E. Rosales, and J. L. Sánchez. On a generalization of the class of Jensen convex functions. *Aequationes Math.*, 90(3):569–580, 2016.

[35] T. Lara, E. Rosales, and J. L. Sánchez. New properties of m-convex functions. *Int. J. Math. Anal.*, 9(15):735–742, 2015.

[36] A. Lundberg. Some methods for a Sutô–Aczél project. *Aequationes Math.*, 89:957–980, 2015.

[37] Gy. Maksa. On the alienation of the logarithmic and exponential cauchy equations. *Aequationes Math.*, 92(3):543–547, 2018.

[38] Gy. Maksa and M. Sablik. On the alienation of the exponential Cauchy equation and the Hosszú equation. *Aequationes Math.*, 90(1):57–66, 2016.

[39] T. Nadhomi, C. P. Okeke, M. Sablik, and T. Szostok. On a new class of functional equations satisfied by polynomial functions. *Aequationes Math.*, 95(6):1095–1117, 2021.

[40] C. P. Okeke and M. Sablik. Functional equation characterizing polynomial functions and an algorithm. *Results Math.*, 77(3), 2022.

[41] B. Sobek. Alienation of the jensen, cauchy and d'alembert equations. *Ann. Math. Sil.*, 30, 01 2016.

[42] T. Szostok. Alienation of two general linear functional equations. *Aequationes Math.*, 94(2):287–301, 2020.

[43] L. Székelyhidi. The stability of linear functional equations. *C. R. Math. Rep. Acad. Sci. Canada*, 3(2):63–67, 1981.

[44] L. Székelyhidi. On a class of linear functional equations. *Publ. Math. Debrecen*, 29(1-2):19–28, 1982.

[45] L. Székelyhidi. *On the Levi-Civita Functional Equation.* Berichte der mathematisch-statistischen Sektion in der Forschungsgesellschaft Joanneum Graz, Graz, 1988.

[46] L. Székelyhidi. On a linear functional equation. *Aequationes Math.*, 38(2-3):113–122, 1989.

[47] L. Székelyhidi. *Convolution Type Functional Equations on Topological Abelian Groups.* World Scientific Publishing Co. Inc., Teaneck, NJ, 1991.

[48] G. Toader. Some generalizations of the convexity. In *Proc. Coll. Approx. and Opt. (Cluj-Napoca)*, pages 329–338. Univ. Cluj-Napoca, Cluj-Napoca, 1984.

[49] W. H. Wilson. On a certain general class of functional equations. *Amer. J. Math.*, 40(3):263–282, 1918.

Appendix
List of author's publications

List of publications related to the dissertation

[**P1**] A. Gilányi and L. N. To. On a computer program determining types of functional equations.

Aequationes. Math. (Published online first), 2025.

ISSN: 0001-9054.

DOI: http://dx.doi.org/10.1007/s00010-025-01184-3

SJR: Q3 (2024)

IF: 0.7 (2024)

[**P2**] A. Gilányi and L. N. To. Computer assisted investigation of alienness of linear functional equations.

Aequationes. Math. 97, 1185-1199, 2023.

ISSN: 0001-9054.

DOI: http://dx.doi.org/10.1007/s00010-023-00978-7

SJR: Q2 (2023)

IF: 0.9 (2023)

[**P3**] A. Gilányi and L. N. To. Computer-assisted investigation of Levi-Civita type functional equations.

Manuscript, 2025.

[**P4**] A. Gilányi, R. Quintero, and L. N. To. Computer-assisted investigations of m-convex hulls of sets.

Manuscript, 2025.

List of other abstracts

[**A1**] L. N. To and A. Gilányi. Computer assisted investigation of Levi-Civita type functional equations. In report of Meeting: The Twenty-third Katowice-Debrecen Winter Seminar on Functional Equations and Inequalities Brenna (Poland), January 31 - February 3, 2024.

Ann. Math. Silesianae. 38 (2), 394, 2024.

ISSN: 0860-2107.

DOI: http://dx.doi.org/10.2478/amsil-2024-0010

[**A2**] L. N. To and A. Gilányi. On computer assisted investigations of m-convex hulls of sets. In: Report of Meeting: The 60th International Symposium on Functional Equations, Hotel Rewita, Kościelisko (Poland), June 9-15, 2024.

Aequationes. Math. 98 (6), 1703, 2024.

ISSN: 0001-9054.

DOI: http://dx.doi.org/10.1007/s00010-024-01126-5

[**A3**] L. N. To and A. Gilányi. On computer assisted visualization of m-convex hulls of sets. In: Report of meeting: The 59th International Symposium on Functional Equations Hotel Aurum, Hajdúszoboszló (Hungary), June 18-25, 2023.

Aequationes. Math. 97, 1274, 2023.

ISSN: 0001-9054.

DOI: http://dx.doi.org/10.1007/s00010-023-01007-3

[**A4**] L. N. To and A. Gilányi. Computer assisted investigation of the alienness of linear functional equations. In: Report of Meeting: 19th International Conference on Functional Equations and Inequalities, Bedlewo, Poland, September 11-18, 2021.

Ann. Univ. Paedag. Cracoviensis. Studia Math. 20 (1), 166-167, 2021.

ISSN: 2081-545X.

DOI: http://dx.doi.org/10.2478/aupcsm-2021-0009

List of talks related to the topic of the dissertation

[**T1**] *PhD Qualification at the End of the First Year*, Institute of Mathematics, University of Debrecen, June 18, 2021.

[**T2**] *Computer assisted investigation of the alienness of linear functional equations*, The 19th International Conference on Functional Equations and Inequalities, Bedlewo, Poland, September 11–18, 2021.

[**T3**] *Complex Exam*, Institute of Mathematics, University of Debrecen, June 29, 2022.

[**T4**] *PhD Qualification at the End of the Third Year*, Institute of Mathematics, University of Debrecen, June 6, 2023.

[**T5**] *On computer assisted visualization of m-convex hulls of sets*, The 59th International Symposium on Functional Equations, Hajdúszoboszló, Hungary, June 18–25, 2023.

[**T6**] *Computer assisted investigation of Levi-Civita type functional equations*, The 23rd Katowice–Debrecen Winter Seminar on Functional Equations and Inequalities, Brenna, Poland, January 31–February 3, 2024.

[**T7**] *On computer assisted investigations of m-convex hulls of sets*, The 60th International Symposium on Functional Equations, Kościelisko, Poland, June 9–15, 2024.

[**T8**] *Computer-assisted investigation of alienness of linear functional equations*, Dr. Varecza Árpád emlékkonferencia, University of Nyíregyháza, November 14–15, 2024.

[**T9**] *Computer-assisted investigation of types of functional equations*, The 61st International Symposium on Functional Equations, Cluj-Napoca, Romania, June, 15–21, 2025.

UNIVERSITY of DEBRECEN

UNIVERSITY AND NATIONAL LIBRARY
UNIVERSITY OF DEBRECEN
H-4002 Egyetem tér 1, Debrecen
Phone: +3652/410-443, email: publikaciok@lib.unideb.hu

Registry number: DEENK/487/2025.PL
Subject: PhD Publication List

Candidate: Lan Nhi To
Doctoral School: Doctoral School of Mathematical and Computational Sciences
MTMT ID: 10091055

List of publications related to the dissertation

Foreign language scientific articles in international journals (2)

1. Gilányi, A., **To, L. N.**: On a computer program determining types of functional equations.
   Aequ. Math. [Epub ahead of print], 1-15, 2025. ISSN: 0001-9054.
   DOI: http://dx.doi.org/10.1007/s00010-025-01184-3
   IF: 0.7 (2024)

2. Gilányi, A., **To, L. N.**: Computer assisted investigation of alienness of linear functional equations.
   Aequ. Math. 97, 1185-1199, 2023. ISSN: 0001-9054.
   DOI: http://dx.doi.org/10.1007/s00010-023-00978-7
   IF: 0.9

List of other publications

Foreign language abstracts (4)

3. **To, L. N.**, Gilányi, A.: Computer assisted investigation of Levi-Civita type functional equations. In report of Meeting: The Twenty-third Katowice-Debrecen Winter Seminar on Functional Equations and Inequalities Brenna (Poland), January 31 - February 3, 2024.
   Ann. Math. Silesianae. 38 (2), 394, 2024. ISSN: 0860-2107.
   DOI: http://dx.doi.org/10.2478/amsil-2024-0010

4. **To, L. N.**: On computer assisted investigations of m-convex hulls of sets. In: Report of Meeting: The 60th International Symposium on Functional Equations, Hotel Rewita, Kościelisko (Poland), June 9-15, 2024.
   Aequ. Math. 98 (6), 1703, 2024. ISSN: 0001-9054.
   DOI: http://dx.doi.org/10.1007/s00010-024-01126-5

UNIVERSITY of DEBRECEN

UNIVERSITY AND NATIONAL LIBRARY
UNIVERSITY OF DEBRECEN
H-4002 Egyetem tér 1, Debrecen
Phone: +3652/410-443, email: publikaciok@lib.unideb.hu

5. **To, L. N.**, Gilányi, A.: On computer assisted visualization of m-convex hulls of sets. In: Report of meeting: The 59th International Symposium on Functional Equations Hotel Aurum, Hajdúszoboszló (Hungary), June 18-25, 2023.
   Aequ. Math. 97, 1274, 2023. ISSN: 0001-9054.
   DOI: http://dx.doi.org/10.1007/s00010-023-01007-3
6. **To, L. N.**, Gilányi, A.: Computer assisted investigation of the alienness of linear functional equations. In: Report of Meeting: 19th International Conference on Functional Equations and Inequalities, Będlewo, Poland, September 11-18, 2021.
   Ann. Univ. Paedag. Cracoviensis. Studia Math. 20 (1), 166-167, 2021. ISSN: 2081-545X.
   DOI: http://dx.doi.org/10.2478/aupcsm-2021-0009

Total IF of journals (all publications): 1,6
Total IF of journals (publications related to the dissertation): 1,6

The Candidate's publication data submitted to the Tudóstér have been validated by DEENK on the basis of the Journal Citation Report (Impact Factor) database.

22 August, 2025