

Article

Computation in Algebraic Hyperstructures

Yuming Feng¹ , Enrico Talotti² and Violeta Leoreanu-Fotea^{3,*} 

¹ School of Computer Science and Engineering, Chongqing Three Gorges University, Chongqing 404100, China; yumingfeng25928@163.com

² Center for Information Technologies and Applied Mathematics, University of Nova Gorica, Vipavska 13, SI-5000 Nova Gorica, Slovenia; enrico.talotti@ung.si

³ Faculty of Mathematics, “Al.I. Cuza” University, Bd. Carol I, No. 11, 700506 Iași, Romania

* Correspondence: foteavioleta@gmail.com

Abstract

The concept of the relation β plays a central role in the study of hypercompositional structures. In this paper, we extend the definition of β to the general framework of hypergroupoids and develop an algorithm to compute its elements and its transitive closure β^* . We then apply this algorithm to determine the β -class of partial identities in a hypergroup, a process equivalent to computing the heart of the given algebraic structure. Furthermore, we propose a more general algorithm that is also applicable to the case of H_v -groups. By extracting the quotient set with respect to β^* and endowing it with an appropriate group structure, we obtain the so-called fundamental group. The identity of this fundamental group can then be computed directly, yielding the heart of the structure.

Keywords: fundamental relation; semihypergroup; hypergroup; H_v -semigroup; H_v -group; heart; algorithms



Academic Editors: Volodimir Simulik and Xiaoqiang Hua

Received: 15 August 2025

Revised: 28 October 2025

Accepted: 28 October 2025

Published: 4 November 2025

Citation: Feng, Y.; Talotti, E.; Leoreanu-Fotea, V. Computation in Algebraic Hyperstructures.

Computation **2025**, *13*, 261.

<https://doi.org/10.3390/computation13110261>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The term algebraic hypercompositional structure denotes a generalization of classical algebraic structures such as groups, semigroups, or rings. In a classical algebraic structure, the law of synthesis of two elements yields a single element, whereas in a hypercompositional structure, the law of synthesis of two elements produces a set of elements. This branch of algebra was introduced for the first time by the French mathematician Marty [1] during the 8th International Congress of Scandinavian Mathematicians. Subsequently, other mathematicians, including Krasner [2,3], Wall [4], Ore, and Dresher [5], laid the foundations of what is now known as hypercompositional algebra. Over the years, the law of synthesis of hypercompositional algebras has been enriched with additional axioms, giving rise to structures with diverse and intriguing properties and applications. Among them there are hypergroupoids, semihypergroups, and hypergroups, which generalize the classical notion of groupoids, semigroups, and groups, respectively, and the wider classes of H_v -semigroups and H_v -groups. In [6] a MATLAB algorithm is presented, which determines the strong fuzzy grade of a hypergroup, considering a particular class of hypergroups associated with genetics. Other algorithms for algebraic hyperstructures were presented in [7,8]. Moreover, several books [9–15] and survey papers [16–20] have been devoted to hypercompositional structures, reflecting the growing interest and development in this field. It is not surprising that there is a significant overlap between the tools and problems of (semi-)group theory and those of (semi-)hypergroup theory. In 1970, M. Koskas [21] formalized this connection by introducing the notion of a strongly regular relation. This, in turn, led to the definition

of the relation β and its transitive closure β^* , both of which belong to the class of fundamental relations. Subsequently, Corsini [10] and Sureau [22,23] further investigated these concepts in the context of semihypergroups and hypergroups, and the equality between β and β^* has been proved in the case of hypergroups (see Theorem 16, [10]). It turns out that the fundamental relation β^* is the smallest equivalence relation in a hypergroup H such that the corresponding quotient set forms a group, called the fundamental group (see Corollary 21, [10]). The class of the identity element in the fundamental group is called the heart [24–27]. The heart is a special subhypergroup of H that provides detailed information about the partition of H induced by the fundamental relation. In [28,29], Vougiouklis defines the notion of relation β in the larger class of H_v -groups, while in [30], the authors studied the H_v module of functions on the H_v ring of arithmetics and the corresponding fundamental module. Vougiouklis' approach was completed by proving that the fundamental relation β^* is the transitive closure of the relation β . It was also shown that if H is an H_v -(semi)group, then the quotient H/β^* is a (semi)group. However, whether β and β^* coincide in the case of H_v -groups remains an open question. In [31] another important equivalence relation α^* is studied in the context of topological quotient hyperrings.

The purpose of this paper is to present the notion of the β relation, following the approach proposed by Vougiouklis. In particular, we will extend its definition to the broader class of hypergroupoids. Our definition leads to a deterministic algorithm capable of computing the elements of β . Furthermore, employing the Warshall algorithm, we obtain the transitive closure β^* , which we then use to define the fundamental group structure on the quotient set induced by β^* on a hypergroup or on an H_v -group. This allows us to compute the heart of the structure by identifying the elements in the β^* -class of the identity of the fundamental group. In the case of hypergroups, a shortcut is possible. In fact, in Corollary 1, we show that the heart of a hypergroup coincides with the β -class of partial identities. This result offers computational advantages, as it bypasses the construction of the fundamental group and the determination of its identity.

To make this manuscript self-contained, in Section 2 we present the basic notions and results on hypercompositional structures, with an emphasis on hypergroups and H_v -groups. For a comprehensive and detailed introduction to the theory of hypergroups and H_v -groups, we refer the reader to [9–11,15,17] and [29,32], respectively. In Section 3, we formalize the definition of the set $\mathcal{U}$ containing all finite hyperproducts of elements in a hypergroupoid. This leads to the definition of the relation β in the class of hypergroupoids. Moreover, we recall well-known results regarding the heart of a hypergroup, which will be useful for its computation. In Section 4, we further analyze the set $\mathcal{U}$ by introducing an increasing chain of subsets of a hypergroupoid, and in Proposition 1 we show how this chain can be used to concretely compute $\mathcal{U}$ in a finite number of steps. This work provides the foundation for computing the heart of both hypergroups and H_v -groups, with the corresponding procedures and algorithms presented in Section 5. In Section 7, we provide a concrete analysis of the heart of a hypergroup of order 7 and an H_v -group of order 10. The implementations are written in MATLAB and are publicly available (see [33]). Furthermore, a more efficient and optimized library, written in RUST and offering additional features, is also available (see [34]). Finally, conclusions and directions for future research are presented in Section 9.

2. Overview of Hypercompositional Structures

Let H be a nonempty set and let $\mathcal{P}(H)$ be the power set of H . A hypercomposition (or hyperoperation) in H is a map $\circ: H \times H \rightarrow \mathcal{P}(H)$ that associates, for any $x, y \in H$, a subset $x \circ y$ of H , rather than a single element, as in the case of an ordinary composition. A nonempty set H endowed with at least one hypercomposition $\circ$ forms a hypercomposi-

tional structure. Whenever there is no risk of confusion, we will write xy instead of $x \circ y$. Furthermore, we restricted our study to hypercompositional structures equipped with a single hyperoperation. A hyperoperation in H induces an operation in $\mathcal{P}(H)$, defined by

$$AB = \bigcup_{a \in A, b \in B} a \circ b, \text{ for all } A, B \in \mathcal{P}(H).$$

Throughout this paper, we identify each singleton with its unique element. Therefore, if A is the singleton $\{a\}$, then we will write aB instead of $\{a\}B$ and similarly if B is a singleton.

A hypercompositional structure H is called a *semihypergroup* if its hyperoperation is *associative*; i.e.,

$$a(bc) = (ab)c, \text{ for all } a, b, c \in H.$$

A hypercompositional structure H is called a *quasihypergroup* if its hyperoperation is *reproductive*; i.e.,

$$aH = H = Ha, \text{ for all } a \in H.$$

A hypercompositional structure H is called a *hypergroupoid* if the hypercomposition of any two elements of H is a nonempty set. A hypercompositional structure H that is both associative and reproductive is called a *hypergroup*. One can easily prove that every hypergroup is a hypergroupoid [17]. A hypercompositional structure is called an H_v -*semigroup* if its hypercomposition is *weakly associative*; that is,

$$a(bc) \cap (ab)c \neq \emptyset, \text{ for all } a, b, c \in H.$$

An H_v -semigroup that is also reproductive is called an H_v -group [29].

An element e in a hypercompositional structure H is called a *partial right identity* if there is an element $x \in H$ such that $x \in xe$. Analogously, $e \in H$ is a *partial left identity* if there is $x \in H$ such that $x \in ex$. An element $e \in H$ is a *partial identity* if it is a partial right identity or a partial left identity. The sets of partial right identities and partial left identities are denoted by $I_{p,r}(H)$ and $I_{p,l}(H)$, respectively. It follows that the set of partial identities is

$$I_p(H) = I_{p,r}(H) \cup I_{p,l}(H).$$

An element e in H is called a *right identity* if $x \in xe$ holds for all $x \in H$. Similarly, $e \in H$ is said to be a *left identity* in H if $x \in ex$ for all $x \in H$. Thus, $e \in H$ is called an *identity* if $x \in ex \cap xe$ for all $x \in H$. The set of right identities in H is denoted by $i_r(H)$, while the set of left identities in H is denoted by $i_l(H)$. It follows that the set of identities in H is

$$i(H) = i_r(H) \cap i_l(H).$$

A subset K of H is *multiplicatively closed* if $xy \subseteq K$ holds for all $x, y \in K$. A *subhypergroup* K of a hypergroup H is a nonempty reproductive subset of H ; that is, $xK = K = Kx$ for any x in K . This also implies that, for any $x, y \in K$, we have $xy \subseteq xK = K$; i.e., subhypergroups are multiplicatively closed. The definition of H_v -subgroups follows mutatis mutandis.

3. A General Definition for the β Relation

In this section, we give the definitions of regular and strongly regular relations in a hypergroupoid. These coincide with the definitions given in [15] in the context of semihypergroups.

3.1. The Fundamental Relation on Hypergroupoids

Definition 1. Let H be a hypergroupoid and let ρ be an equivalence on H .

- We say that ρ is regular on the right if for all $a, b \in H$,

$$a \rho b \Rightarrow \forall x \in H, \forall u \in ax, \exists v \in bx, \text{ such that } u \rho v.$$

Similarly, the regularity on the left can be defined. An equivalence relation that is both regular on the right and regular on the left is called regular.

- We say that ρ is strongly regular on the right if for all $a, b \in H$,

$$a \rho b \Rightarrow \forall x \in H, \forall u \in ax, \forall v \in bx, \text{ we have } u \rho v.$$

Similarly, the strong regularity on the left can be defined. An equivalence relation that is both strongly regular on the right and on the left is called strongly regular.

Notation 1. We denote by H/ρ the quotient set of H with respect to an equivalence ρ . For each $x \in H$, we denote the equivalence classes of x by $\bar{x} := \{z \in H : x \rho z\}$.

Theorem 1. Let H be a hypergroupoid and let ρ be an equivalence in H . Then ρ is regular if and only if the hyperoperation on the quotient set H/ρ defined by

$$\bar{x} \otimes \bar{y} := \{\bar{z} \mid z \in xy\} \quad (1)$$

is well defined.

Proof. ($\Rightarrow$) Assume that ρ is regular and let $x\rho x', y\rho y'$. We prove $x \otimes y = x' \otimes y'$. Let $\bar{u} \in \bar{x} \otimes \bar{y}$; then we have $u \in xy$. By the regularity on the right, there exists $v \in x'y'$ such that $u\rho v$. Moreover, the regularity on the left implies that there exists $w \in x'y'$ such that $v\rho w$. By the transitivity of ρ , we have $u\rho w$; i.e., $\bar{u} = \bar{w} \in x' \otimes y'$. This proves $x \otimes y \subseteq x' \otimes y'$. Similarly, the reverse inclusion follows.

($\Leftarrow$) Let $a, b \in H$ be such that $a\rho b$ and let x be an arbitrary element in H . If $u \in ax$, then $\bar{u} \in \bar{a} \otimes \bar{x} = \bar{b} \otimes \bar{x}$. Hence, there exists $v \in bx$ such that $u\rho v$. This proves the left regularity of ρ . Similarly, the right regularity follows. $\square$

Remark 1. Every strongly regular relation is also regular. Obviously, if ρ is a regular relation in a hypergroupoid H , then the canonical projection $\pi: H \rightarrow H/\rho$ verifies the identity $\pi(xy) = \pi(x) \otimes \pi(y)$ for all $x, y \in H$; that is, π is a good homomorphism.

Definition 2. Let H be a hypergroupoid. The fundamental relation β^* on a hypergroupoid H is the smallest equivalence (with respect to inclusion) such that $(H/\beta^*, \otimes)$ is a group. The quotient group H/β^* is called the fundamental group.

Here we give a definition for the relation β in the general framework of hypergroupoids. Let H be a hypergroupoid and let $a_1, \dots, a_n \in H$. We denote by $\prod_{i=1}^k a_i$ the set of all hyperproducts of $a_1, \dots, a_n$, chosen in this order, with all possible parentheses among them. For any $n \in \mathbb{N}$, we define the following sequence of subsets of H :

- $\mathcal{U}_1 = \{\{x\} \mid x \in H\}$;
- $\mathcal{U}_n = \{u \mid u \in \prod_{i=1}^n a_i \text{ for some } a_1, \dots, a_n \in H\}$ for all $n > 1$.

Then, we set $\mathcal{U} = \bigcup_{n \geq 1} \mathcal{U}_n$.

Remark 2. For any $n \geq 1$, $\mathcal{U}_n$ contains all hyperproducts of n factors of H . In particular, singletons in $\mathcal{U}_1$ are considered as hyperproducts of 1 factor. It follows that $\mathcal{U}$ is the set of all finite hyperproducts of H .

Definition 3. Let H be a hypergroupoid. For any $x, y \in H$, we define the relation β as follows:

$$x \beta y \Leftrightarrow \exists u \in \mathcal{U} \text{ such that } \{x, y\} \subseteq u.$$

It is easy to see that β is reflexive and symmetric, but not transitive, in general. The transitive closure of β is denoted by $\hat{\beta}$.

Remark 3. We note that Definition 3 coincides with the definition of the β relation introduced by Vougiouklis for H_v -semigroups [29]. If $(H, \circ)$ is a semihypergroup and $a_1, \dots, a_n \in H$, then the set $\prod_{i=1}^n a_i$ reduces to a singleton $\{a_1 \circ \dots \circ a_n\}$. Thus, $\prod_{i=1}^n a_i$ can be identified with its unique element $u = a_1 \circ \dots \circ a_n$.

If H is a semihypergroup, the definition of the relation β can be rewritten as

$$x \beta y \Leftrightarrow \exists n \in \mathbb{N}, \exists a_1, \dots, a_n \in H \text{ such that } \{x, y\} \subseteq a_1 \circ \dots \circ a_n.$$

This formulation coincides with the definition of the relation β given by Koskas in [21].

Theorem 2. Let H be a hypergroupoid. Then $\hat{\beta}$ is strongly regular in H .

Proof. Let $x, y \in H$ such that $x \hat{\beta} y$ and let a be an arbitrary element in H . It follows that there exist $x_0 = x, x_1, \dots, x_n = y$ such that $x_i \beta x_{i+1}$ for all $i \in \{0, 1, \dots, n-1\}$. Let $r_1 \in xa$ and $r_1 \in ya$. We prove $r_1 \hat{\beta} r_1$. From $x_i \beta x_{i+1}$ it follows that there exists $u_i \in \mathcal{U}$ such that $\{x_i, x_{i+1}\} \subseteq u_i$. Thus, $x_i a \subseteq u_i a$ and $x_{i+1} a \subseteq u_i a$. It follows that for any $s_i \in x_i a$, we have $s_i \beta s_{i+1}$, $i \in \{0, 1, \dots, n-1\}$. Then, if we take $s_0 = r_1$ and $s_n = r_2$ we obtain $r_1 \hat{\beta} r_2$. This shows that $\hat{\beta}$ is strongly regular on the right. The strong regularity on the left follows dually. $\square$

3.2. The Heart of a Hypergroup and of an H_v -Group

By Theorem 1, hyperoperation $\otimes$ is well defined in $H/\hat{\beta}$. Moreover, Vougiouklis [29] proved that if H is an H_v -group, then $\beta^* = \hat{\beta}$; in particular $H/\hat{\beta}$ is a group and coincides with the fundamental group H/β^* . Since hypergroups form a subclass of H_v -groups, this result also holds for hypergroups (see also [21]). Let $1_{H/\beta^*}$ be the identity in H/β^* . The β^* -class of $1_{H/\beta^*}$ is called the heart of H and is denoted by ω_H . Furthermore, the canonical projection $\pi: H \rightarrow H/\beta^*$ is a good homomorphism, and hence ω_H is the kernel of the canonical projection π , namely,

$$\omega_H := \{x \in H : \pi(x) = 1_{H/\beta^*}\}.$$

Freni in [23] proved that for hypergroups, the equality $\beta = \hat{\beta} = \beta^*$ also holds. It is still unknown whether this identity holds for the H_v -groups.

Theorem 3 ([23]). Let H be a reproductive semihypergroup (i.e., a hypergroup). Then $\beta = \beta^*$.

In what follows, we recall the notion of the complete part of a semihypergroup H and present a connection to the heart in the case of hypergroups.

A nonempty subset A of a semihypergroup H is called a complete part if $\forall n \in \mathbb{N}$ and $\forall (x_1, \dots, x_n) \in H^n$,

$$x_1 \circ \dots \circ x_n \cap A \neq \emptyset \Rightarrow x_1 \circ \dots \circ x_n \subseteq A.$$

It is easy to see that the class of complete parts of H is closed under intersection. Therefore, for a nonempty subset A of H , the complete closure of A can be defined as the intersection of all the complete parts of H that contain A . This is indicated by $\mathcal{C}(A)$.

An increasing chain of subsets of H can be defined to describe the complete closure of A . This leads to the definition of a new equivalence relation, which is equivalent to β^* .

Let H be a semihypergroup and let A be a nonempty subset of H . For all $n \in \mathbb{N}$, we define the following chain of subsets $K_n(A)$ recursively:

$$\begin{aligned} K_1(A) &:= A, \\ K_n(A) &:= \left\{ x \in H \mid \exists p \in \mathbb{N}, \exists a_1, \dots, a_p \in H \text{ such that} \right. \\ &\quad \left. x \in \prod_{i=1}^p a_i \text{ and } \prod_{i=1}^p a_i \cap K_{n-1}(A) \neq \emptyset \right\}. \end{aligned}$$

$$\text{Let } K(A) = \bigcup_{n \geq 1} K_n(A).$$

Theorem 4 ([15]). $K(A) = \mathcal{C}(A)$.

Notation 2. If there is no risk of confusion, we will write $\mathcal{C}(x)$ instead of $\mathcal{C}(\{x\})$ and $K(x)$ instead of $K(\{x\})$.

For $x, y \in H$, define the relation K as follows:

$$x K y \Leftrightarrow y \in K(x).$$

It can be shown that K is an equivalence relation (see [15]). In particular, we have the following results.

Theorem 5 ([15]). Let H be a semihypergroup. Then the relations β^* and K coincide. In particular, for any $x \in H$, the β^* -class of x coincides with both $K(x)$ and $\mathcal{C}(x)$; i.e.,

$$\bar{x} = K(x) = \mathcal{C}(x). \quad (2)$$

Theorem 6 ([15]). Let H be a hypergroup and let A be a subset of H . Then

$$\mathcal{C}(A) = A\omega_H = \omega_H A. \quad (3)$$

The following result will be used to compute the heart of a hypergroup when a partial identity is known.

Corollary 1. Let H be a hypergroup and let $e \in H$ be a partial identity. Then we have

$$\omega_H = \bar{e}.$$

Proof. Without loss of generality, we may assume that there exists $x \in H$ such that $x \in xe$. Using the canonical projection $\pi: H \rightarrow H/\beta^*$, we have the identity

$$\pi(x) = \pi(x) \otimes \pi(e)$$

and since we are in a group (the fundamental group), the cancelation law yields $\pi(e) = 1_{H/\beta^*}$; i.e., $e \in \omega_H$. As ω_H is a subhypergroup (see [23]), the reproductivity holds, and we have $e\omega_H = \omega_H = \omega_H e$. By Theorems 5 and 6, it follows that $\omega_H = e\omega_H = \mathcal{C}(e) = \bar{e}$. $\square$

A hypergroup is called complete if every hyperproduct of it is a complete part. The heart of a complete hypergroup consists of the set of its bilateral identities (see Theorem 13, Chapter VI, [9]).

4. Computing the Set of Finite Hyperproducts and the β Relation in a Hypergroupoid

In what follows, we will consider a finite hypercompositional structure with a single hyperoperation. Following the idea in [7], any finite hypercompositional structure H can be explicitly defined by its Cayley table (or synthesis table), that is, a matrix M of type $n \times n$, with rows and columns indexed by the elements of H , where the entry at position (a, b) corresponds to the result of hyperoperation $ab \in \mathcal{P}(H)$. The Cayley table can be constructed in several ways:

- Manually, filling the subset ab into a matrix for each pair $(a, b) \in H \times H$;
- Automatically, using a defining function that specifies the hyperoperation for all such pairs.

Example 1. The hypercompositional structure in Listing 1 is implemented manually by specifying the entries of the matrix M . For a clear visualization of the Cayley table, we provide a dedicated function, `show_cayley_table`, available in the accompanying repository [33]. Since its implementation is not relevant to the mathematical discussion, we omit the source code from the listings.

Listing 1. Example of input data.

```

1 >> H = {'a', 'b', 'c'};
2 >> M = {
3     {'a'}, {'a', 'b'}, {'a', 'c'};
4     {'a'}, {'b'}, {'a', 'c'};
5     {'a'}, {'a', 'b'}, {'c'};
6 };
7 >> show_cayley_table(H,M)
8      | a      b      c
9  -----+-----
10 a      |{a}    {a, b}  {a, c}
11 b      |{a}    {b}    {a, c}
12 c      |{a}    {a, b}  {c}

```

Example 2. In Listing 2 we define a set H and a generating function computing the b -hyperoperation, which is defined as $xy = \{x, y\}$ for all $x, y \in H$ (see [17]). The function `new_from_function` (Listing 3) is then used to automatically fill the corresponding Cayley table.

Listing 2. The b -hypercomposition on $H = \{a, b, c\}$.

```

1 H = {'a', 'b', 'c'};
2 b_hyperoperation = @(a, b) unique(H([a b]));
3 M = new_from_function(H, b_hyperoperation);
4 show_cayley_table(H,M)
5      | a      b      c
6  -----+-----
7 a      |{a}    {a, b}  {a, c}
8 b      |{a, b}  {b}    {b, c}
9 c      |{a, c}  {b, c}  {c}

```

Listing 3. MatLab code for new_from_function.

```

1 % NEW_FROM_FUNCTION Construct a Cayley table from a binary operation
  handle.
2 %
3 % result = NEW_FROM_FUNCTION(H, binary_function)
4 %
5 % Given a finite set H and a binary function defined on its elements,
6 % this function builds the corresponding Cayley table as a cell array.
7 % INPUT:
8 %     H: 1 x n cell array or array representing the elements of the set
  H.
9 %     binary_function: Function handle @(x,y) returning the
  hyperoperation
10 %                                value for elements x, y in H. It must accept any
11 %                                pair of elements from H as input.
12 % OUTPUT:
13 %     result: n x n cell array where result{i,j} = binary_function(H{i},
  H{j}).
14 %                                Each entry represents the (possibly set-valued) product of
15 %                                elements H{i} and H{j}.
16 %
17 function result = new_from_function(H, binary_function)
18     [I, J] = ndgrid(1:length(H), 1:length(H));
19     result = arrayfun(@(i,j) binary_function(H(i), H(j)), I, J, '
  UniformOutput', false);
20 end

```

In this section, we present an algorithm to compute the elements of the relation β for a finite hypergroupoid H , following an approach similar to that in [8]. Employing the Warshall algorithm [35], we can obtain the transitive closure β^* , which we can use to compute the fundamental group of a hypergroup H and of an H_v -group, as well as the heart ω_H . Proposition 1 and Corollary 2 allow for the computation of the set $\mathcal{U}$ of all finite hyperproducts of elements in H , which is an important step for the computation of β .

In fact, if $\mathcal{U}$ is known, then we have

$$\beta = \{(x, y) \in H \times H : \exists u \in \mathcal{U} \text{ with } \{x, y\} \subseteq u\} = \bigcup_{u \in \mathcal{U}} u \times u, \quad (4)$$

where the equality holds because $(x, y) \in u \times u$ if and only if both $x \in u$ and $y \in u$. The algorithms for computing $\mathcal{U}$ and the elements in β are given in Algorithm 1 and Algorithm 2, respectively.

Algorithm 1 Computation of the set $\mathcal{U}$ of all finite hyperproducts.

Input: finite set H and Cayley table M defining a hyperoperation

Output: $\mathcal{U} \subseteq \mathcal{P}(H)$ of all finite hyperproducts

```

1: function HYPERPRODUCTSET(H, M)
2:    $\mathcal{A}_{\text{prev}} \leftarrow \emptyset;$ 
3:    $\mathcal{A}_{\text{curr}} \leftarrow \{\{x\} \mid x \in H\};$ 
4:   while  $\mathcal{A}_{\text{prev}} \neq \mathcal{A}_{\text{curr}}$  do
5:      $\mathcal{A}_{\text{prev}} \leftarrow \mathcal{A}_{\text{curr}};$ 
6:      $\mathcal{A}_{\text{curr}} \leftarrow \mathcal{A}_{\text{curr}} \cup \{q \circ q' \mid q, q' \in \mathcal{A}_{\text{prev}}\};$ 
7:   end while
8:   return  $\mathcal{A}_{\text{curr}}$ 
9: end function

```

$\triangleright H$ finite set, M Cayley table
 $\triangleright$ Initialize previous set to emptyset
 $\triangleright$ Start with singletons $\mathcal{A}_0$
 $\triangleright$ Final set equals $\mathcal{A}_m = \mathcal{U}$

Algorithm 2 Computation of the β relation.**Input:** finite set H and Cayley table M defining a hyperoperation**Output:** β

```

1: function GETBETARELATION( $(H, M)$ )
2:    $\beta \leftarrow \emptyset$ 
3:    $\mathcal{U} \leftarrow \text{HYPERPRODUCTSET}(H, M)$        $\triangleright$  Get all hyperproducts (see Algorithm 1)
4:   for  $u \in \mathcal{U}$  do
5:     for  $x \in u$  do
6:       for  $y \in u$  do
7:          $\beta \leftarrow \beta \cup \{(x, y)\};$ 
8:       end for
9:     end for
10:  end for
11:  return  $\beta$ 
12: end function

```

4.1. An Increasing Chain of Subsets of H to Compute $\mathcal{U}$

In this subsection, we introduce an increasing chain of subsets of H , and in Proposition 1 and Corollary 2 we show how it can be used to compute the set $\mathcal{U}$ of all finite hyperproducts of H in finitely many steps.

Definition 4. Let H be a hypergroupoid. For $n \in \mathbb{N}$, we define the following recursively:

1. $\mathcal{A}_0 := \{\{x\} : x \in H\}$ (the set of singletons);
2. For $n > 0$, $\mathcal{A}_n := \mathcal{A}_{n-1} \cup \{qq' : q, q' \in \mathcal{A}_{n-1}\}$.

Furthermore, we set $\mathcal{A} = \bigcup_{n \geq 0} \mathcal{A}_n$.

In other words, each set $\mathcal{A}_n$ contains all hyperproducts in H with at most 2^n factors. This follows immediately by induction on n . Indeed, for $n = 0$, we see that $\mathcal{A}_0$ is the set of all singletons, and these are precisely the hyperproducts of the 1-factor of H . Moreover, if $p \in \mathcal{A}_n$ then $p \in \mathcal{A}_{n-1}$ or $p = qq'$ for some $q, q' \in \mathcal{A}_{n-1}$. In the first case p has at most 2^{n-1} factors by induction, and the thesis follows. In the second case, p is the hyperproduct of $q, q' \in \mathcal{A}_{n-1}$; thus p has at most $2 \cdot 2^{n-1} = 2^n$ factors. Thus, $\mathcal{A} \subseteq \mathcal{U}$, where $\mathcal{U}$ is the set of all finite hyperproducts of elements in H . Conversely, if $u \in \mathcal{U}$, then $u \in \mathcal{U}_k$ for some $k \in \mathbb{N}$; i.e., u is a hyperproduct of k factors. Then, it follows that $u \in \mathcal{A}_l$ for $l \geq \lceil \log_2 k \rceil$, and hence $u \in \mathcal{A}$. This proves $\mathcal{U} \subseteq \mathcal{A}$. Therefore, we have proved the following theorem.

Theorem 7. Let H be a hypergroupoid and let $\mathcal{U}$ be the set of all finite hyperproducts. Let $(\mathcal{A}_n)_{n \in \mathbb{N}}$ be as in Definition 4. Then $\mathcal{U} = \mathcal{A}$.

Proposition 1. Let H be a finite hypergroupoid and let $(\mathcal{A}_n)_{n \in \mathbb{N}}$ be as in Definition 4. Then the following hold:

1. $\mathcal{A}_n \subseteq \mathcal{A}_{n+1}$ for all $n \in \mathbb{N}$.
2. The sequence $(\mathcal{A}_n)$ must eventually stabilize; that is, there exists $m \in \mathbb{N}$ such that $\mathcal{A}_m = \mathcal{A}_{m+j}$ for all $j \geq 1$. We will call m the stabilization index.
3. Let m be the stabilization index. Then m is the minimum value such that $\mathcal{A}_m = \mathcal{A}_{m+1}$.

Proof.

1. It follows immediately by Definition 4.
2. By Item 1, we have that $\mathcal{A}_{n+1}$ extends to $\mathcal{A}_n$ for all $n \in \mathbb{N}$. But $\mathcal{A}_n \in \mathcal{P}(H)$ for all $n \in \mathbb{N}$, and since H is finite, so is $\mathcal{P}(H)$. Therefore, the sequence must eventually stabilize to some finite number $m \in \mathbb{N}$.

3. It is enough to prove that if $k \in \mathbb{N}$ is such that $\mathcal{A}_k = \mathcal{A}_{k+1}$, then $\mathcal{A}_k = \mathcal{A}_{k+j}$ for all $j \geq 1$, and hence $k = m$. We use induction in j . The base case follows from the hypothesis. Assuming $\mathcal{A}_k = \mathcal{A}_{k+j-1}$, we prove $\mathcal{A}_k = \mathcal{A}_{k+j}$. Since $\mathcal{A}_k = \mathcal{A}_{k+1}$ we also have $\mathcal{A}_{k+1} = \mathcal{A}_{k+j-1}$. It follows that $\mathcal{A}_{k+j} = \mathcal{A}_{k+j-1} \cup \{qq' : q, q' \in \mathcal{A}_{k+j-1}\} = \mathcal{A}_k \cup \{qq' : q, q' \in \mathcal{A}_k\} = \mathcal{A}_{k+1} = \mathcal{A}_k$.

□

Notice that $\mathcal{A}_0 = \mathcal{U}_1, \mathcal{A}_1 = \mathcal{U}_2$.

Corollary 2. Let H be a finite hypergroupoid and $\mathcal{U}$ be the set of all finite hyperproducts of elements of H . Let $(\mathcal{A}_n)_{n \in \mathbb{N}}$ be as in Definition 4 and let m be the stabilization index. Then $\mathcal{U} = \mathcal{A}_m$.

Example 3. Let $H = \{z_1, z_2, z_3, z_4\}$. We consider the following table:

$\circ$	z_1	z_2	z_3	z_4
z_1	$\{z_1\}$	$\{z_2\}$	$\{z_3, z_4\}$	$\{z_4\}$
z_2	$\{z_2\}$	$\{z_1, z_2\}$	$\{z_3, z_4\}$	$\{z_3, z_4\}$
z_3	$\{z_3\}$	$\{z_3, z_4\}$	$\{z_1, z_2\}$	$\{z_1, z_2\}$
z_4	$\{z_3, z_4\}$	$\{z_3, z_4\}$	$\{z_1, z_2\}$	$\{z_1, z_2\}$

Then $(H, \circ)$ is a hypergroup; z_1 is an identity, because $z_i \in z_1 \circ z_i \cap z_i \circ z_1$, for all $i \in \{1, 2, 3, 4\}$.

z_2 is a partial identity, since $z_2 \in z_2 \circ z_2$. Notice that z_2 is not an identity, since $z_1 \notin z_1 \circ z_2$.

We have $(z_1, z_2) \in \beta, (z_3, z_4) \in \beta$, since $z_2 \circ z_2 = \{z_1, z_2\}, z_2 \circ z_3 = \{z_3, z_4\}$.

There are not hyperproducts that contain both z_1, z_3 or both z_2, z_4 ; hence $(z_1, z_3) \notin \beta, (z_1, z_4) \notin \beta$. We have $\mathcal{U}_1 = \{\{z_1\}, \{z_2\}, \{z_3\}, \{z_4\}\}, \mathcal{U}_2 = \{\{z_1\}, \{z_2\}, \{z_3\}, \{z_4\}, \{z_1, z_2\}, \{z_3, z_4\}\}$.

Therefore,

$$\omega_H = I_p(H) = \{z_1, z_2\}.$$

Example 4 ([9]). Let $(A, \cdot)$ be a total hypergroup (i.e., $x \cdot y = A$ for all $x, y \in A$) such that $|A| \geq 2$.

Let T be a set such that $A \cap T \neq \emptyset, T = \{t_i \mid i \in \mathbb{N}^*\}$, with $t_i \neq t_j$ for i, j different elements.

We define the following hyperoperation on $H = A \cup T$:

$$x \cdot y = y \cdot x = A, \text{ if } (x, y) \in A^2;$$

$$x \cdot y = y \cdot x = (A - \{x\}) \cup T, \text{ if } (x, y) \in A \times T;$$

$$x \cdot y = y \cdot x = A \cup \{t_{i+j}\}, \text{ if } (x, y) = (t_i, t_j) \in T^2.$$

Then $(H, \cdot)$ is a hypergroup. Let us calculate its heart ω_H .

We have $I_p(H) = A$, since $\forall x, y \in A, x \in x \cdot y$.

Notice that $t_1 \circ t_n = A \cup \{t_{n+1}\}, a \cdot t_i = (A - \{a\}) \cup T$, for all $a \in A$.

We have $\mathcal{U}_2 = \{A, A \cup \{t_n\}, (A - \{a\}) \cup T \mid n \geq 2, a \in A\}$.

Hence for all $a \in A, t \in T$ we have $a\beta t$, whence $\omega_H = H$.

4.2. Algorithm Computing $\mathcal{U}$ and β

Here we present pseudocodes to compute the set $\mathcal{U}$ of all finite hyperproducts of elements of H . Furthermore, we can compute the elements in β as discussed in the previous section, see (4). These algorithms have been implemented in RUST VERSION 1.91.0 and MATLAB and are available in the following repositories [33,34].

The stabilization index m introduced in Proposition 1 satisfies the inequality

$$m \leq 2^{|H|} - 1 - |H|.$$

Indeed, the sequence starts with $\mathcal{A}_0$, which has cardinality equal to $|H|$, and at each step at least one new element is added until the stabilization is reached. This means that we can compute the set $\mathcal{A}_m$ in a finite number of steps, which is less than or equal to $2^{|H|} - 1 - |H|$. By Corollary 2, this is equivalent to computing the set U .

The pseudocode is presented in Algorithm 1.

Now, using the identity (4), the elements of the relation β are easily computed, as shown in Algorithm 2.

The transitive closure β^* can be calculated using the Warshall algorithm [35], as shown in Algorithm 3. However, it should be noted that this step is unnecessary in the case of a hypergroup, as discussed in Theorem 3.

Algorithm 3 Warshall algorithm for transitive closure.

Input: A finite set H , a relation β on H

Output: The transitive closure β

```

1: function GETTRANSITIVECLOSURE( $H, \beta$ )
2:    $n \leftarrow \text{cardinality}(H)$ 
3:    $\hat{\beta} \leftarrow \emptyset$ 
4:    $T \leftarrow \text{GETZEROONEMATRIX}(H, \beta)$  ▷ Initialize relation matrix of  $\beta$ 
5:   for  $k = 1$  to  $n$  do
6:     for  $i = 1$  to  $n$  do
7:       for  $j = 1$  to  $n$  do
8:         if  $T(i, k)$  and  $T(k, j)$  then
9:            $T(i, j) \leftarrow 1$ ;
10:        end if
11:      end for
12:    end for
13:  end for
14:   $\hat{\beta} \leftarrow \text{FROMMATRIXTORELATION}(T)$  ▷ Convert the matrix into a relation
15:  return  $\hat{\beta}$ 
16: end function

```

5. Computation of the Heart in Hypergroups and H_v -Groups

In what follows, we will present algorithms that compute the heart ω_H of a hypergroup H and of an H_v -group. We will adopt the following procedure for a hypergroup, which is based on Corollary 1.

- Step 1: Find the β^* -equivalence classes of H .
- Step 2: Find at least one partial identity $e \in I_p$. (Note that not all hypergroups have identities, but every hypergroup has partial identities.)
- Step 3: Determine the heart ω_H by finding the β -equivalence class of e .

In the case of an H_v -group, the naive procedure will be applied, i.e., the following:

- Step 1: Find the quotient set H/β^* .
- Step 2: Implement the hyperoperation as in Equation (1), which will turn it into the fundamental group.
- Step 3: Find the identity $1_{H/\beta^*}$ and obtain the elements in its class.

5.1. Computation of the Heart of a Hypergroup

As described in Corollary 1, if H is a hypergroup and $e \in H$ is a partial identity, then

$$\omega_H = \bar{e}.$$

Algorithm 4 describes how to determine the partial identities of a hypercompositional structure. In Algorithm 5 we present an algorithm that computes the β -class of an element $x \in H$.

Algorithm 4 Computation of partial identities in H .

Input: A finite set H , the Cayley table M defining a hyperoperation

Output: The set of partial identities in H

```

1: function COLLECTPARTIALIDENTITIES( $H, M$ )
2:    $n \leftarrow \text{cardinality}(H)$ ;
3:    $\text{partial\_identity} \leftarrow [\text{false}; n]$   $\triangleright$  Initialize logical array of size  $n = |H|$ 
4:   for  $i \in H$  do
5:     for  $j \in H$  do
6:       if  $j \in m_{ij}$  or  $j \in m_{ji}$  then  $\triangleright$  Check left or right partial identity condition
7:          $\text{partial\_identity}[i] \leftarrow \text{true}$ ;
8:         break  $\triangleright$  Proceed to next  $i$ 
9:       end if
10:    end for
11:  end for
12:  return  $\text{partial\_identity}$ 
13: end function
  
```

Algorithm 5 Computation of the class of an element $x \in H$ with respect to an equivalence ρ .

Input: A finite set H , an element x in H , an equivalence ρ

Output: The ρ -class of x

```

1: function GETCLASS( $H, x, \rho$ )
2:    $\text{class} \leftarrow \emptyset$   $\triangleright$  Initialize the  $\beta$ -class to emptyset
3:   for  $z \in H$  do
4:     if  $(x, z) \in \rho$  then  $\triangleright$  If  $x \rho z$ , add  $z$  to the class
5:        $\text{class} \leftarrow \text{class} \cup \{z\}$ 
6:     end if
7:   end for
8:   return  $\text{class}$ 
9: end function
  
```

By combining these two procedures, we obtain an algorithm to compute the heart of a hypergroup, as detailed in Algorithm 6.

Algorithm 6 Compute the heart of a hypergroup.

Input: A finite set H , the Cayley table M

Output: The heart ω_H

```

1: function HEARTHYPERGROUP( $H, M$ )  $\triangleright H$  finite hypergroup with Cayley table  $M$ 
2:   if not ISHYPERGROUP( $H, M$ ) then
3:     error " $H$  is not a hypergroup!"
4:   end if
5:    $\beta = \text{GETBETARELATION}(H, M)$ 
6:    $I_\beta \leftarrow \text{COLLECTPARTIALIDENTITIES}(H, M)$ 
7:   for  $k \in I_\beta$  do
8:     if  $k$  is true then
9:        $\omega_H \leftarrow \text{GETCLASS}(H, k, \beta)$ 
10:    return  $\omega_H$ 
11:   end if
12: end for
13: end function
  
```

5.2. Computation of the Heart of an H_v -Group

The computation of the heart of a hypergroup described in Section 5.1 is based on Theorem 3, Corollary 1, and its proof depends on the results in Theorems 5 and 6, where the associativity of the hypercompositional structure is an essential hypothesis. In the larger context of H_v -groups, the same result cannot be directly applied, as the associative is not true in general. Moreover, as underlined in [23], it is not clear if β and its transitive closure $\hat{\beta}$ coincide for H_v -groups. However, in [29], the author shows that the transitive closure $\hat{\beta}$ and the fundamental relation β^* are equal also in the case of an H_v -group. Then, the following naive procedure can be applied to compute the heart of an H_v -group. We first compute β as described in Section 4.2. Then we compute its transitive closure $\hat{\beta}$ (which coincides with β^*) using the Warshall algorithm Algorithm 3, and we collect the quotient set $H/\hat{\beta}$, as described in Algorithm 7. Then, in Algorithm 8, we compute the Cayley table of the fundamental group H/β^* , where the law of synthesis is introduced in Equation (1). Its identity $\bar{e}$ is easily computed through Algorithm 9 and the elements in this class are the elements of ω_H .

Algorithm 7 Compute the quotient set of H modulo an equivalence ρ .

Input: A finite set H , an equivalence relation ρ

Output: The quotient set H/ρ

```

1: function QUOTIENTSET( $H, \rho$ )
2:    $quotient\_set \leftarrow \emptyset$  ▷ Empty list of classes
3:   for  $x \in H$  do
4:      $new\_class \leftarrow \text{GETCLASS}(H, x, \rho)$ 
5:     if  $new\_class \notin quotient\_set$  then ▷ Check if class is already listed
6:        $quotient\_set \leftarrow quotient\_set \cup \{new\_class\};$ 
7:     end if
8:   end for
9:   return  $quotient\_set$ 
10: end function

```

Algorithm 8 COMPUTATION OF THE FUNDAMENTAL GROUP.

Input: A finite set H , a Cayley table M defining a hyperoperation

Output: The fundamental group $H/\hat{\beta}$

```

1: function FUNDAMENTALGROUP( $H, M$ )
2:    $\beta \leftarrow \text{BETARELATION}(H, M);$ 
3:    $\hat{\beta} \leftarrow \text{TRANSITIVECLOSURE}(H, \beta);$ 
4:    $H/\hat{\beta} \leftarrow \text{QUOTIENTSET}(H, \hat{\beta});$ 
5:    $n \leftarrow \text{cardinality}(H/\hat{\beta});$ 
6:    $I \leftarrow \{1, \dots, n\};$  ▷ The indexes for the Cayley table
7:    $Cayley\_Table \leftarrow \emptyset;$  ▷ Initialize Cayley table to empty
8:   for  $(i, \bar{x}) \in I \times H/\hat{\beta}$  do
9:     for  $(j, \bar{y}) \in I \times H/\hat{\beta}$  do
10:       $\bar{x}\bar{y} \leftarrow \emptyset$ 
11:      for  $a \in \bar{x}$  do
12:        for  $b \in \bar{y}$  do
13:           $\bar{x}\bar{y} \leftarrow \bar{x}\bar{y} \cup \text{HYPEROPERATION}(H, M, a, b);$ 
14:        end for
15:      end for
16:       $Cayley\_Table(i, j) \leftarrow \bar{x}\bar{y};$ 
17:    end for
18:  end for
19:  return  $H/\hat{\beta}, Cayley\_Table$ 
20: end function

```

Algorithm 9 Computation of identities in H .**Input:** A finite set H , a Cayley table M defining the hyperoperation**Output:** The set of identities in H

```

1: function COLLECTIDENTITIES( $H, M$ )                                 $\triangleright H$  finite set,  $M$  Cayley table
2:    $n \leftarrow \text{cardinality}(H)$ ;
3:    $\text{identities} \leftarrow \emptyset$ ;                                   $\triangleright$  Initialize set of identities
4:   for all  $e \in H$  do
5:     for all  $j \in H$  do
6:       if  $j \notin m_{ej}$  or  $j \notin m_{je}$  then                         $\triangleright$  Check identity condition
7:         continue
8:       else
9:          $\text{identities} \leftarrow \text{identities} \cup \{e\}$ 
10:      end if
11:    end for
12:  end for
13:  return  $\text{identities}$ 
14: end function

```

Remark 4. Obviously, the procedure described above is also valid for hypergroups, since every hypergroup is an H_v -group.

6. Experimental Analysis of the Stabilization Index

To study the asymptotic behavior of the stabilization index m with respect to the cardinality of the underlying hypergroupoid H , we performed a random sampling experiment. For each cardinality $|H| \in \{3, 4, 5, 6, 7, 8, 9, 10, 11, 12\}$ we generated 500 random hypergroupoids and computed the stabilization index m of the sequence $(\mathcal{A}_n)$ defined in Definition 4. For each cardinality, we record the empirical mean, standard deviation, and minimum and maximum values of m , as well as the ratio $m/|H|$. Our analysis is presented in Table 1 with the support of Figures 1–3. We can state the following:

- The mean stabilization index m remains small and increases only slightly with $|H|$, staying close to 3 even for $|H| = 12$.
- The ratio $m/|H|$ decreases, supporting the hypothesis that m grows most sublinearly with the cardinality of H .
- Standard deviations decrease slightly as cardinality increases, indicating that larger random hypergroupoids tend to have even more consistent stabilization indices.
- The minimum and maximum values show that, although the mean is stable, some variation still exists among random instances.

Table 1. Empirical behavior of the stabilization index m as a function of $|H|$.

$ H $	Mean m	Std. Dev.	Min	Max	Ratio $m/ H $
3	2.50	0.55	2	4	0.832
4	2.88	0.46	2	4	0.720
5	3.13	0.39	2	6	0.626
6	3.20	0.43	3	6	0.533
7	3.29	0.47	3	5	0.470
8	3.25	0.46	3	5	0.407
9	3.21	0.42	3	5	0.357
10	3.17	0.38	3	5	0.317
11	3.10	0.29	3	4	0.281
12	3.06	0.24	3	4	0.255

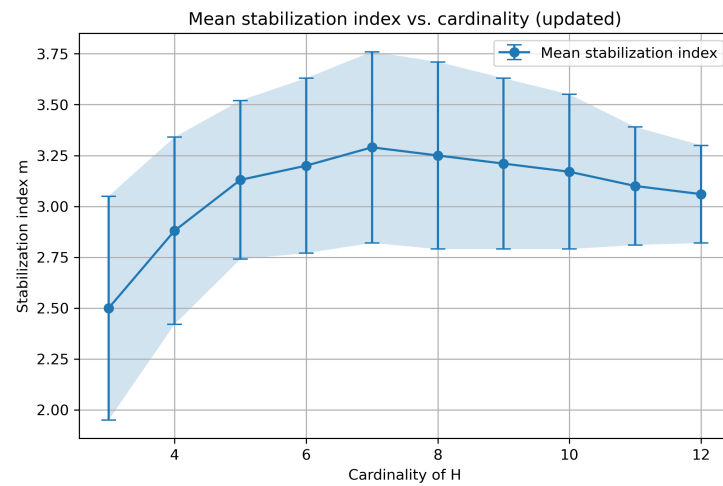


Figure 1. Mean stabilization index m with standard deviations for each cardinality $|H|$.

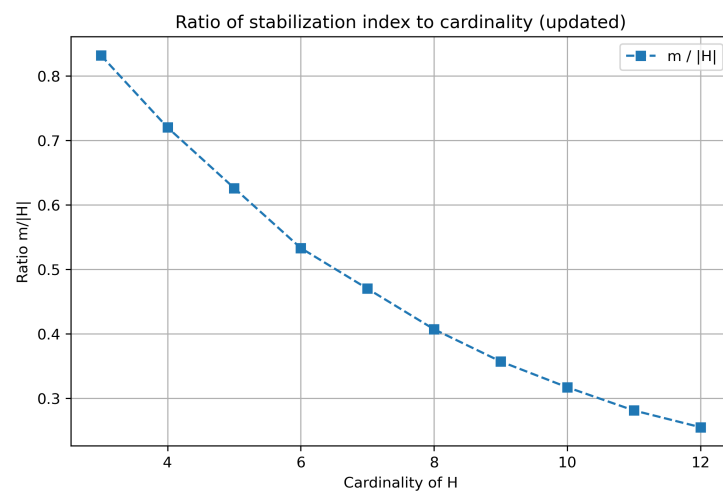


Figure 2. Ratio $m/|H|$ as a function of the cardinality.

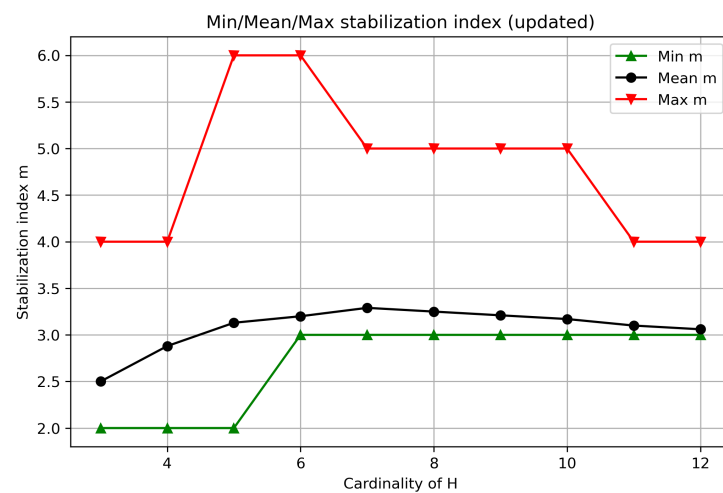


Figure 3. Minimum, mean, and maximum stabilization index m per cardinality.

In general, these experiments support the hypothesis that, for randomly generated hypergroupoids, the stabilization index is typically a small integer essentially independent of $|H|$ for the tested range and that its relative size decreases with increasing cardinality.

Further experiments with larger cardinalities and sample sizes would allow a more refined statistical analysis and perhaps the formulation of an asymptotic limit for m .

7. A MATLAB R2018B Session

The authors have prepared a GITHUB repository that contains complete implementations of the algorithms presented in the preceding sections. In this section, we illustrate the computation of the heart of a hypergroup and of an H_v -group using our MATLAB-based code. Additional utility functions and test cases are also provided (see [33]). For comparison, a more advanced and efficient library implemented in RUST VERSION 1.91.0 with similar functionality is available (see [34]).

In Listing 4, we define a set H of cardinality 7 along with a Cayley table describing a hyperoperation on H . The function `show_cayley_table` produces a neatly formatted visualization of this Cayley table for a hypercompositional structure. It is available in the same repository, in the file `show_cayley_table.m`.

Listing 4. A hypercompositional structure of order 7.

```

1 >> H7 = {'a','b','c','d','e','f','g'};
2 >> M7 = {
3     {'a'},{'b'},{'c'},{'d'},{'e'},{'f','g'},{'f','g'};
4     {'b'},{'a'},{'e'},{'f','g'},{'c'},{'d'},{'d'};
5     {'c'},{'f','g'},{'a'},{'e'},{'d'},{'b'},{'b'};
6     {'d'},{'e'},{'f','g'},{'a'},{'b'},{'c'},{'c'};
7     {'e'},{'d'},{'b'},{'c'},{'f','g'},{'a'},{'a'};
8     {'f','g'},{'c'},{'d'},{'b'},{'a'},{'e'},{'e'};
9     {'f','g'},{'c'},{'d'},{'b'},{'a'},{'e'},{'e'};
10    };
11 >> show_cayley_table(H7,M7)
12
13      | a      b      c      d      e      f      g
14 ----+-----
15 a    | {a}    {b}    {c}    {d}    {e}    {f, g}  {f,
16      | g}
17 b    | {b}    {a}    {e}    {f, g}  {c}    {d}    {d}
18 c    | {c}    {f, g}  {a}    {e}    {d}    {b}    {b}
19 d    | {d}    {e}    {f, g}  {a}    {b}    {c}    {c}
20 e    | {e}    {d}    {b}    {c}    {f, g}  {a}    {a}
      | {f, g}  {c}    {d}    {b}    {a}    {e}    {e}
      | {f, g}  {c}    {d}    {b}    {a}    {e}    {e}

```

In Listing 5, we verify whether the given hypercompositional structure satisfies the axioms of a hypergroup. We remind the reader that, in standard MATLAB notation, logical values are represented by 0 for false and 1 for true. The calculation confirms that the structure H_7 indeed forms a hypergroup.

Listing 5. Verification of hypergroup axioms for H_7 .

```

1 >> is_hypergroup(H7, M7)
2 ans = 1

```

We can show the partial identity analysis by calling the function `show_partial_identities` (see Listing 6). The calculation shows that a is the only partial identity in H_7 . Therefore, by Corollary 1, we expect the heart of H_7 to be the singleton $\{a\}$. The heart computation is performed by the function `heart`, as described in Algorithm 6. Its output is presented in Listing 7, and it matches the expected result.

Listing 6. Partial identities analysis.

```

1 >> show_partial_identities
2 Partial Identity Analysis:
3 Element      | Partial Identity |
4 -----
5 a            | true
6 b            | false
7 c            | false
8 d            | false
9 e            | false
10 f           | false
11 g           | false

```

Listing 7. Computation of the heart of H_7 .

```

1 >> heart(H7,M7)
2 ans =
3 {
4   [1,1] = a
5 }

```

In Listing 8 we define a set H with cardinality 10 with a Cayley table that identifies a hyperoperation. The computation shows that this hypercompositional structure is an H_v -group.

Listing 8. Computation of the heart of a H_v -group.

```

1 >> Hv_10 = {'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j'};
2 >> Mv_10 = {
3 {'a'}, {'b'}, {'c'}, {'d'}, {'e'}, {'a','f'}, {'g'}, {'h'}, {'i'}, {'j'},
4 {'b'}, {'c'}, {'d'}, {'e'}, {'f'}, {'g'}, {'h'}, {'i'}, {'j'}, {'a'},
5 {'c'}, {'d'}, {'e'}, {'f'}, {'g'}, {'h'}, {'i'}, {'j'}, {'a'}, {'b'},
6 {'d'}, {'e'}, {'f'}, {'g'}, {'h'}, {'i'}, {'j'}, {'a'}, {'b'}, {'c'},
7 {'e'}, {'f'}, {'g'}, {'h'}, {'i'}, {'j'}, {'a'}, {'b'}, {'c'}, {'d'},
8 {'a','f'}, {'g'}, {'h'}, {'i'}, {'j'}, {'a'}, {'b'}, {'c'}, {'d'}, {'e'},
9 {'g'}, {'h'}, {'i'}, {'j'}, {'a'}, {'b'}, {'c'}, {'d'}, {'e'}, {'f'},
10 {'h'}, {'i'}, {'j'}, {'a'}, {'b'}, {'c'}, {'d'}, {'e'}, {'f'}, {'g'},
11 {'i'}, {'j'}, {'a'}, {'b'}, {'c'}, {'d'}, {'e'}, {'f'}, {'g'}, {'h'},
12 {'j'}, {'a'}, {'b'}, {'c'}, {'d'}, {'e'}, {'f'}, {'g'}, {'h'}, {'i'},
13 };
14 >> is_hv_group(Hv_10,Mv_10)
15 >> ans = 1

```

The fundamental group is computed in Listing 9. The corresponding Cayley table is printed using the `show_cayley_table` function, from which one can verify that the fundamental group is isomorphic to $\mathbb{Z}_5$. An analysis of the group structure shows that the identity element corresponds to the class $\{a, f\}$, which represents the heart of H .

Listing 9. Computation of the heart of a H_v -group.

```

1 >> quotient_set = quotient_set(Hv_10,Mv_10);
2 >> fundamental_group_table = fundamental_group(Hv_10,Mv_10);
3 >> show_cayley_table(quotient_set,fundamental_group_table)
4
5      | af      bg      ch      di      ej
6 -----+-----
7 af      |{a, f}    {b, g}    {c, h}    {d, i}    {e, j}
8 bg      |{b, g}    {c, h}    {d, i}    {e, j}    {a, f}
9 ch      |{c, h}    {d, i}    {e, j}    {a, f}    {b, g}
10 di     |{d, i}    {e, j}    {a, f}    {b, g}    {c, h}
11 ej     |{e, j}    {a, f}    {b, g}    {c, h}    {d, i}
12
13 >> collect_identities(quotient_set,fundamental_group_table)
14 ans =
15 {
16     [1,1] = 1
17     [2,1] = 0
18     [3,1] = 0
19     [4,1] = 0
20     [5,1] = 0
21 }
22 >> heart_hv(Hv_10,Mv_10)
23 ans =
24 {
25     [1,1] = a
26     [1,2] = f
27 }

```

8. Computational Performance of Heart Computation in Hypergroups

An important aspect of our study is the computational cost of computing the heart of a hypergroup. We compared two methods: the shortcut procedure valid for hypergroups (see Corollary 1 and Algorithm 6) and the naive procedure applicable to the wider class of H_v -groups (see Algorithm 10).

Algorithm 10 COMPUTATION OF THE HEART OF AN H_v -GROUP(H, M).

Input: A finite set H , a Cayley table M defining the hyperoperation

Output: The heart ω_H

```

1: function HEARTHV( $H, M$ )
2:    $\omega_H \leftarrow \emptyset$ ; ▷ Initialize  $\omega_H$  as the emptyset
3:    $\beta \leftarrow \text{GETBETARELATION}(H, M)$ 
4:    $\hat{\beta} \leftarrow \text{GETTRANSITIVECLOSURE}(H, \beta)$ 
5:    $(H/\hat{\beta}, F) \leftarrow \text{FUNDAMENTALGROUP}(H, M)$ 
6:    $\omega_H \leftarrow \text{COLLECTIDENTITIES}(H/\hat{\beta}, F)$  ▷ In a group a unique identity exists
7:   return  $\omega_H$ ;
8: end function

```

8.1. Execution Time Analysis in MATLAB R2018B

Using the hypergroup H_7 from Listing 4, we measured the execution times in MATLAB R2018B. The results are given in Listing 10.

Listing 10. Execution times for the computation of the heart of H_7 in MATLAB.

```

1 >> tic; heart(H7,M7); toc
2 Elapsed time is 0.138428 s.
3 >> tic; heart_hv(H7,M7); toc
4 Elapsed time is 0.539330 s.

```

The shortcut method is therefore roughly four times faster than the naive H_v -group approach. Table 2 summarizes the results.

Table 2. Computational time comparison for computing the heart of H_7 using MATLAB.

Method	Time (s)
Shortcut (Algorithm 6)	0.138
Naive H_v -group method	0.539

8.2. Execution Time Analysis in RUST VERSION 1.91.0

We also implemented both methods in RUST VERSION 1.91.0, taking advantage of its high performance. The same hypergroup H_7 was used. Table 3 summarizes the results.

Table 3. Computational time comparison for computing the heart of H_7 using RUST.

Method	Time
Shortcut (Algorithm 6)	142.956 μ s
Naive H_v -group method	7.55193 ms

The shortcut method in RUST VERSION 1.91.0 runs in approximately 0.143 ms, while the naive H_v -group method requires 7.55 ms. This shows that the shortcut method is more than 50 times faster than the naive one. Thus, for larger hypergroups, the shortcut method provides a practical and scalable approach to compute the heart, whereas the naive H_v -group procedure quickly becomes prohibitive.

8.3. Complexity Analysis

The computation of the heart proceeds through three main algorithmic stages: (1) the generation of the set $\mathcal{U}$ of all finite hyperproducts (Algorithm 1), (2) the construction of the β relation (Algorithm 2), and (3) the computation of its transitive closure via Warshall's algorithm (Algorithm 3). In the first stage, each iteration of Algorithm 1 performs pairwise hyperoperations among subsets of H , requiring $O(|H|^2)$ operations per iteration. Since stabilization occurs after a small number m of iterations (empirically $m \leq 4$ for $|H| \leq 12$), the overall cost of this phase is $O(|H|^3)$ in time and $O(|H|^2)$ in space. The second stage, Algorithm 2, scans all hyperproducts $u \in \mathcal{U}$ and produces every ordered pair (x, y) with $x, y \in u$, resulting in a quadratic complexity $O(|H|^2)$ in both time and memory. Finally, the transitive closure step, implemented in Algorithm 3, corresponds to the classical Warshall algorithm with time complexity $O(|H|^3)$ and space complexity $O(|H|^2)$ (see [35]). Consequently, the overall asymptotic complexity of the complete pipeline is $O(|H|^3)$, which remains computationally tractable for the finite hypergroupoids typically analyzed in this work.

9. Conclusions and Future Works

In this paper, we present a computational approach to study the fundamental relation β^* . In particular, we extend the definition of the fundamental relation β to the general framework of hypergroupoids. Our main contributions are Theorem 7, Proposition 1, and Corollary 2, which form the basis for a deterministic algorithm (Algorithm 1) to compute the set $\mathcal{U}$ of all finite hyperproducts in a hypergroupoid, an essential step in determining the elements of the relation β .

Moreover, determining the heart of a hypergroup can be achieved by considering the class of partial identities (Corollary 1 and Algorithm 6), which are guaranteed to exist in any reproductive hypercompositional structure. The proof of this result relies on

the reproductivity of H and on the fact that, in a semihypergroup, the β^* -class of any x coincides with the complete part of x (Theorem 5).

Our computational and empirical analysis further supports these results. The examples given on the stabilization index of the sequence $(\mathcal{A}_n)$ show that it remains small and essentially bounded, even as the cardinality of the hypergroupoid increases. The ratio $m/|H|$ decreases with the size of H , suggesting sublinear growth of the stabilization index. Additionally, benchmarks comparing the shortcut algorithm for hypergroups and the naive procedure for H_v -groups, both in MATLAB R2018B/GNU OCTAVE, VERSION 8.4.0 and RUST VERSION 1.91.0, demonstrate significant computational savings when exploiting the algebraic structure, highlighting the practical value of our approach.

However, it remains unclear whether these results can be generalized to the broader class of H_v -groups. Future research could focus on extending the notion of a complete part to an H_v -semigroup in a consistent manner, with the aim of generalizing Theorem 7, Proposition 1, and Corollary 2. Such an extension would enable the computation of the heart of an H_v -group using the same principles as in the case of hypergroups, leading to a more efficient algorithm. Nevertheless, the calculation of the heart for H_v -groups remains feasible, as demonstrated by Algorithm 10.

Author Contributions: Conceptualization, E.T.; Methodology, Y.F., E.T. and V.L.-F.; Software, Y.F. and E.T.; Validation, V.L.-F.; Formal analysis, E.T. and V.L.-F.; Writing—original draft, Y.F., E.T. and V.L.-F. All authors have read and agreed to the published version of the manuscript.

Funding: This work is sponsored by the Natural Science Foundation of Chongqing Science and Technology Commission, P.R. China (CSTB2024NSCQ-LZX0083, 2023TIADZXX0017, CSTB2023NSCQ-LZX0135), the Scientific and Technological Research Program of Chongqing Municipal Education Commission, P.R. China (KJZD-K202301023, KJQN202201209, 233356), the Scientific and Technological Research Program of Wanzhou District, P.R. China (WZSTC-20230309), and the National Natural Science Foundation of China (12201086).

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Marty, F. Sur une généralisation de la notion de groupe. In Proceedings of the 8 ème Congrès des Mathématiciens, Scandinaves, Stockholm, Sweden, 14–18 August 1934; pp. 45–49.
2. Krasner, M.; Kuntzmann, J. Remarques sur les hypergroupes. *C. R. Acad. Sci.* **1947**, *224*, 525–527. [\[CrossRef\]](#)
3. Krasner, M. *Sur la Primitivité des Corps B-Adiques*; Mathematica: Cluj, Romania, 1937; Volume 13, pp. 72–191.
4. Wall, H.S. Hypergroups. *Amer. J. Math.* **1937**, *59*, 77–98. [\[CrossRef\]](#)
5. Dresher, M.; Ore, O. Theory of Multigroups. *Amer. J. Math.* **1938**, *60*, 705–733. [\[CrossRef\]](#)
6. Feng, Y.; Leoreanu-Fotea, V. Algorithm for Determining the Strong Fuzzy Grade of a Hypergroup. *Algorithms* **2025**, *18*, 540 [\[CrossRef\]](#)
7. Pourhaghani, A.; Anvariye, S.M.; Davvaz, B. Algorithms for Finding Specific Elements in Algebraic Hyperstructures with One Hyperoperation. *Math. Int. Res.* **2025**, *10*, 49–70.
8. Pourhaghani, A.; Anvariye, S.M.; Davvaz, B. An algorithm to calculate the members of the relation β in hypergroupoids. *J. Algebr. Hyperstruct. Log. Algebr.* **2024**, *5*, 137–152 [\[CrossRef\]](#)
9. Corsini, P. *Prolegomena of Hypergroup Theory*; Aviani: Udine, Italy, 1993.
10. Corsini, P.; Leoreanu, V. *Applications of Hyperstructure Theory*; Advances in Mathematics; Kluwer Academic Publishers: Dordrecht, The Netherlands; Boston, MA, USA; London, UK, 2003.
11. Davvaz, B.; Leoreanu-Fotea, V. *Hypergroup Theory and Applications*; International Academic Press: Palm Harbor, FL, USA, 2008.
12. Davvaz, B.; Leoreanu-Fotea, V. *Krasner Hypergroup Theory*; World Scientific: Singapore, 2024.
13. Davvaz, B. *Semihypergroups Theory*; Academic Press: Cambridge, MA, USA, 2016.

14. Davvaz, B.; Vougiouklis, T. *A Walk Through Weak Hyperstructures*; World Scientific Publishing Company: Singapore, 2018. [CrossRef]
15. Davvaz, B.; Leoreanu-Fotea, V. *Hypergroup Theory*; World Scientific Book: Singapore, 2022; ISBN 978–9811249389.
16. Massouros, G.G. On the hypergroup theory. In *Fuzzy Systems & AI-Reports and Letters*; Romanian Academy: Bucharest, Romania, 1995; Volume 4.
17. Massouros, C.; Massouros, G. An Overview of the Foundations of the Hypergroup Theory. *Mathematics* **2021**, *9*, 1014. [CrossRef]
18. Massouros, G. Hypercompositional Structures in the Theory of the Languages and Automata. *Sci. Ann. Cuza Univ.* **1994**, *3*, 65–74.
19. Massouros, G.; Massouros, C. Hypercompositional Algebra, Computer Science and Geometry. *Mathematics* **2020**, *8*, 1338. [CrossRef]
20. Leoreanu-Fotea, V.; Hoskova-Mayerova, S. Join Spaces and Lattices. *Axioms* **2024**, *13*, 705. [CrossRef]
21. Koskas, M. Groupoides, Demi-hypergroupes et hypergroupes. *J. Math. Pures Appl.* **1970**, *49*, 155–192.
22. Sureau, Y. Hypergroupes opérant transitivement sur un ensemble. In *Annales Scientifiques de l'Université de Clermont; Mathématiques; UER de Sciences Exactes et Naturelles de l'Université de Clermont*: Clermont-Ferrand, France, 1979; Volume 68, pp. 83–96.
23. Freni, D. A note on the core of a hypergroup and on the transitive closure β^* of β . *Riv. Mat. Pura Appl.* **1991**, *8*, 153–156.
24. Leoreanu, V. On the heart of join spaces and of regular hypergroups. *Riv. Mat. Pura Appl.* **1996**, *17*, 133–142.
25. Leoreanu, V. The heart of some important classes of hypergroups. *Pure Math. Appl.* **1998**, *9*, 351–360.
26. Leoreanu, V. The class of hypergroups in which the heart is the set of identities. *Mathematics* **1999**, *54*, 49–54.
27. Hošková-Mayerová, Š.; Antampoufis, N. A Brief Survey on the Two Different Approaches of Fundamental Equivalence Relations on Hyperstructures. *Ratio Math.* **2017**, *33*, 47–60.
28. Vougiouklis, T. Fundamental relations in hyperstructures. *Bull. Greek Math. Soc.* **1999**, *42*, 113–118.
29. Vougiouklis, T. *Hyperstructures and Their Representations*; Hadronic Press: Palm Harbor, FL, USA, 1994.
30. Tahan, M.A.; Davvaz, B. H_v module of functions over H_v ring of arithmetics and its fundamental modules. *An. St. Univ. Ovidius Constanța* **2024**, *32*, 5–18.
31. Zare, A.; Davvaz, B. On topological quotient hyperrings and α^* -relation. *An. St. Univ. Ovidius Constanța* **2025**, *33*, 179–193.
32. Dramalidis, A. Geometrical H_v -Structures. Structure Elements of Hyperstructures. In *Proceedings of the 13-AHA Conference-Algebraic Hyperstructures and Their Applications, Samothrace, Greece, 1–9 September 2002*; Lygeros, N., Vougiouklis, T., Eds.; Spanidis Press: Alexandroupolis, Greece, 2005; pp. 41–52.
33. Talotti, E. Heart_Hypergroups, GitHub Repository. 2025. Available online: https://github.com/enh11/heart_hypergroups (accessed on 30 September 2025).
34. Talotti, E. Hypercompositional_Structures, GitHub Repository. 2025. Available online: https://github.com/enh11/hypercompositional_structures (accessed on 15 October 2025).
35. Rosen, K. *Discrete Mathematics and Its Applications*, 8th ed.; McGraw Hill: New York, NY, USA, 2018.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.