Expanding Prodsum sets with Recursion

Matthew S. Lozier-Davis

Summer Creek High School; matthewlozier580@gmail.com

Abstract

This paper delves into the fascinating properties of recursive product-sum sets, where the sum of the elements equals their product. Building upon previous work establishing the foundation for rational finite prodsums, we introduce a recursive method for generating these unique sets, starting from a single rational number. We derive a general formula for the nth element of such a set based on its preceding elements, highlighting the inherent relationship between the additive and multiplicative structures. Through the analysis of dual and tri-element recursive prodsum sets, we uncover fundamental rules and patterns governing their construction, including the crucial exclusion of the initial element being equal to one. This exploration provides a deeper understanding of the intricate balance between sum and product within these recursively generated sets.

Introduction

The intersection of addition and multiplication gives rise to intriguing mathematical structures, one of which is the product-sum set – a collection of numbers whose sum is equal to their product. This paper investigates the generation and characteristics of such sets through a recursive approach. While prior work has explored the existence and properties of finite product-sum sets within the realm of rational numbers, this study focuses specifically on how these sets can be constructed iteratively, starting with a single rational number. We will begin by establishing the notation for recursive product-sum sets and then analyze the simplest case: the dual-element set. By deriving a formula for the second element based on the first, we will uncover a critical constraint on the initial element. Subsequently, we will extend this analysis to tri-element sets, demonstrating two methods for determining the third element: direct substitution and expressing it solely in terms of the initial element. Finally, we will generalize this recursive process to derive a formula for the n
The nth element of any recursive product-sum set, revealing the underlying pattern that governs their structure. This exploration aims to provide a comprehensive understanding of how recursive relationships can be employed to generate and analyze these unique mathematical objects.

Keywords: Prodsums, Recursive Sets, Sum Equals Product, Recursion

Recursive Product-Sum Sets

This paper presumes a basic understanding of prodsums provided in the previous "Establishing the Basis for Rational Finite Prodsums" paper.
In that paper, we looked at product-sum sets with a pre-existing set of numbers, however, there is a way to generate these gets recursively, given just one rational number.

Establishing Notation

Let us establish the notation of $rpd(a_1, a_2, \cdots, a_n) = o$, where rpd is used to signify the following sequence of numbers is the basis for a recursive prodsum set. O will still represent the o-value of the prod-sum.

The Dual Element Product-Sum

The simplest Recursive Product-Sum Set is one with only two elements.

The base rule states $\sum_{i=1}^{n} a_i = \prod_{i=1}^{n} a_i$, which, in this case, equates to $a_1 + a_2 = a_1 \times a_2$.

1. Rearrange the terms to get a_2 on one side

$a_1 = a_1 a_2 - a_2$

2. Factor out a_2 from the right side of the equation

$a_1 = a_2(a_1 - 1)$

3. To solve for a_2, divide both sides of the equation by $(a_1 - 1)$. This gives us the following:

$a_2 = \frac{a_1}{(a_1 - 1)}$ (1)

The result of this is the formula that, when provided with a rational a_1, will give you an a_2 which will make the set $\{a_1, a_2\}$ a Product-Sum set.

The Dual Element Recursive Product-Sum Set Function

Equation 1 can also be described as an algebraic rational function consisting of a single variable relationship

$f(x) = \frac{x}{(x-1)}$

Let us now consider the graph of the function

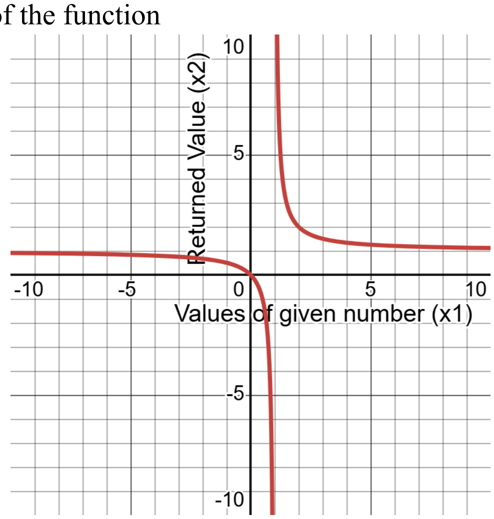

Figure 1. The Dual Element Recursive Prodsum's Graph

This graph is continuous except for a vertical asymptote at $x = 1$; this behavior is expected because of the $(x - 1)$ in the denominator and reveals a rule common to all recursive products.

Rule: There are no recursive Product-Sum sets where $x_1 = 1$

Explanation: By its definition, all elements of a recursive Product-Sum set are generated based on an initial element x_1 through a recursive relationship. Allowing $x_1 = 1$ has distinct effects on the product and the sum of the set. Multiplication by 1 acts as an identity operation, meaning any element multiplied by 1 remains unchanged. Conversely, the addition of 1 increases the value of an element. If the recursive definition involves both multiplication and addition of x_1, then when$x_1 = 1$, the product is less affected by the recursion (potentially remaining constant if the operation is multiplication by 1), while the sum will consistently increase with each additional element generated through the addition of 1. This inherent difference in the behavior of multiplication and addition with the number 1 makes it impossible for the sum and product of a recursive product-sum set with more than one element to be equal if the recursion is fundamentally based on $x_1 = 1$

The Tri-Element Recursive Product-Sum Set

The goal of this subsection find the value of $x_3$in terms of a given $x_1$where the set $S = \{x_1 , x_2 \; x_3\}$ is a recursive Product-Sum set.

I wish to express here one idea that is very fundamental to recursive product sum sets, and recursion in general, substitution.

We know that for a Product-Sum set with three elements, the base rule must hold. That is:

$a_1 + a_2 + a_3 = a_1 \times a_2 \times a_3$

Assuming we do not know the value of a_2, there are two ways for us to solve recursive product sum expressions, depending on your particular use case.

Method One: Finding And Substituting a_2 To Solve For a_3

This method is the quickest, and is best at quickly finding a a_3. It works by recursively finding the value of a_2 and substituting to solve for a_3

Take for example, a Tri-Element Recursive Product-Sum set where $a_1 = 1.16$

We can compute $a_2 = 7.25$ by using the deal element recursive function .

We know from the base rule that all Tri-element Recursive Product-Sum sets must satisfy this equality:

$a_1 + a_2 + a_3 = a_1 \times a_2 \times a_3$

Since we know $a_1$and a_2, we can substitute their values into the equality.

$1.16 + 7.25 + a_3 = 1.16 \times 7.25 \times a_3$

Which simplifies down to

$8.41 + a_3 = 8.41a_3$

We can then solve this for a_3 by

1. Subtract a_3 from both sides of the equation: $8.41 = (8.41a3) - a_3$
2. Factor out a_3 from the right side of the equation: $8.41 = a_3(8.41 - 1)$
3. Simplify the term in the parentheses: $8.41 = a_3(8.41 - 1)$
4. You can then solve this dive both sides by $(8.41 - 1)$

$\frac{8.41}{(8.41-1)} = a_3$

5. Solving the resulting rational expression tells us that $a_3 = 1.13495276653$

Method Two: Directly Expressing a_3 In Terms Of a_1 Via Substitution

This method is slower, but is good for cases where you have to use the formula multiple times because it returns the formula for which you can simply plug in values of a_1

Since we know a_1, we can compute a_2 using the dual element formula, and then use the above to compute a_3. However, a far more interesting result occurs when you try to express a_3 in terms of a_1.

We know from the base rule that all Tri-element Recursive Product-Sum sets must satisfy this equality: $a_1 + a_2 + a_3 = a_1 \times a_2 \times a_3$

We also know that in a Dual Element Recursive Product-Sum Set, $a_2 = \frac{a_1}{(a_1 - 1)}$,

Thus, we can use substitution to assist us in expressing a_3 in terms of a_1

$a_1 + a_2 + a_3 = a_1 \times a_2 \times a_3$, Given from the base rule

Rearrange the terms to gather all terms containing a_3 on one side $a_1 + a_2 = a_1 \times a_2 \times a_3 - a_3$

Factor out a_3 from the right side of the equation: $a_1 + a_2 = a_3(a_1 \times a_2 - 1)$

Isolate a_3 by dividing both sides by $(a_1 a_2 - 1)$,: $a_3 = \frac{a_1 + a_2}{(a_1 a_2) - 1}$

Deriving The Given Element General Product Sum Expression

The recursive product sum is called recursive because new numbers can be generated recursively, that will allow the complete set to fulfill the base rule. There is a pattern at play that will allow us to calculate the nth element of the product sum at any particular value, if all the elements up to that value are calculated.

Notice: $a_2 = \frac{a_1}{(a_1 - 1)}$ and $a_3 = \frac{a_1 + a_2}{(a_1 a_2) - 1}$ The observation we've made accurately reflects a fundamental property arising from the definition of product-sum sets when considering a recursive construction. If we imagine building a product-sum set element by element, where the nth element is added to a set of n−1 elements that already satisfy a modified product-sum relationship, the nth element must indeed take this form. Let's consider a set {a1,a2,…,an} where the sum equals the product. If we assume that the first n−1 elements have a sum Sn−1 and a product Pn−1, then when we add the nth element an, the new sum is Sn−1+an, and the new product is Pn−1 ⋅ an. For the entire set to be a product-sum set, these must be equal: Sn−1+an=Pn−1 ⋅ an. By rearranging this equation to solve for an, we get an=Pn−1−1Sn−1. This formula clearly shows that the nth element is a rational expression where the numerator is the sum of all preceding terms (a1 through an−1), and the denominator is the product of all preceding terms minus one. This pattern is indeed common to all recursive product-sum sets that are constructed by sequentially adding elements and maintaining the product-sum property at each step. However, it's important to note that this formula is undefined if the product of the previous terms is equal to one, which might occur in specific cases, such as when the set includes elements that are reciprocals of each other or contains only the number one. This means that we can compute the nth element of a product-sum set where all the previous elements are known using this formula, which we will call the Given Element General Product Sum Expression. In the formula, n is a given integer.

$$a_n = \frac{\sum_{i=1}^{n} a_i}{-1 + \prod_{i=1}^{n} a_i}$$